[Spring, 2017]

Introduction to Pattern Recognition

PRML (BRI623)

Heung-Il Suk

hisuk@korea.ac.kr

http://www.ku-milab.org

Department of Brain and Cognitive Engineering,
Korea University

Pattern Recognition

[from Wikipedia]

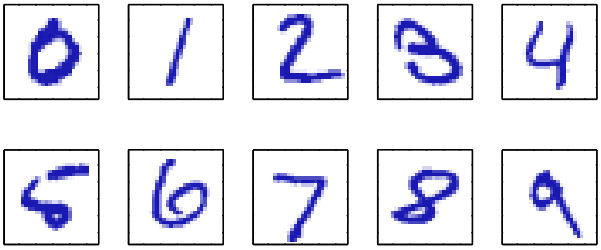
- A branch of machine learning that focuses on the recognition of patterns and regularities in data, although is in some cases considered to be nearly synonymous with machine learning
- Pattern recognition algorithms generally aim to provide a reasonable answer for all possible inputs and to perform "most likely" matching of the inputs, taking into account their statistical variation.

Contents

General Framework in PRML

Example: Handwritten Digits

Goal: to build a machine that will produce the identity of the digit as the output



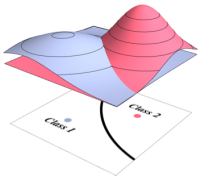
28×28 pixel image: 784 real numbers

- Handcrafted rules or heuristics: shapes of the strokes
- Leads to a proliferation of rules, exceptions
- Nontrivial due to wide variability of handwriting

- Pattern recognition approach
  - Collect a large set of N digits, *training set*, $\{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$
  - Express the category of a digit using a *target vector* $\mathbf{t}$
  - Determine a function $f(\mathbf{x})$, *training* or *learning*
    - Generates an output vector $\mathbf{y}$, encoded in the same way as the target vector $\mathbf{t}$

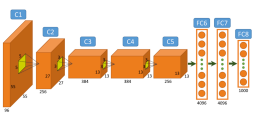
$$\mathbf{x} \quad \Rightarrow \quad f(\mathbf{x};\theta) \quad \Rightarrow \quad \mathbf{y}$$

$p(y|\mathbf{x};\mu,\Sigma)$



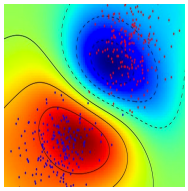
Bayes classifier

$S_j^{\downarrow}\left[F_j\left[\sum_j w_{ij} * X_j\right]\right]$



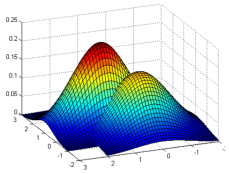
Convolutional neural network

$\sum_{m\in S} a_m k(\mathbf{x},\mathbf{x}_m)+b$



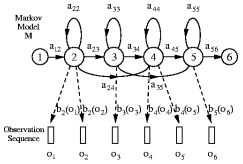
Support vector machine

$\sum_{k=1}^{K} \pi_k N(\mathbf{x}|\mu_k,\Sigma_k)$



Gaussian mixture model

$p(S_t|S_{t-1})p(\mathbf{x}_t|S_t)$



Hidden Markov model

Generalization: the ability to categorize correctly new examples that differ from those used for training

Feature Extraction/Representation

- To transform the original input variables into some new space of variables
- Hope to be easier to solve the problem in the new space
- To lessen computational burden (in real-time applications)
- Careful not to discard the useful discriminator information
- New test data must be preprocessed using the same steps as the training data

PRML Overview

Training session

1. Collecting training samples
2. Preprocessing
3. Feature extraction/representation
4. Feature selection
5. Classifier/regressor learning

Testing session

1. Given testing samples
2. Preprocessing
3. Feature extraction/representation
4. Feature selection
5. Outputs from classifier/regressor

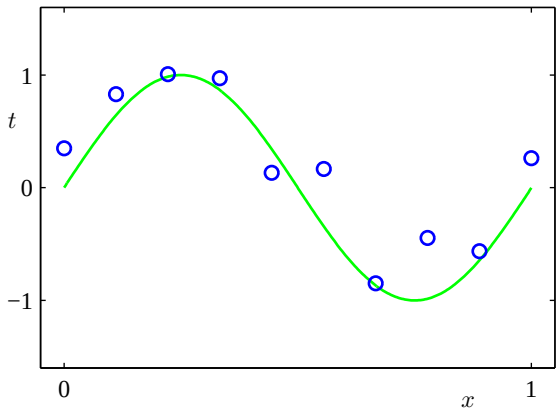
Basic Concepts in PRML

Terminology

- Supervised learning
  - Regression: continuous outputs
  - Classification: discrete or category outputs
- Unsupervised learning
  - Clustering
  - Density estimation
  - Visualization
- Reinforcement learning
  - Finding suitable actions to take in a given situation in order to maximize a reward
  - No optimal outputs are given, but must discover them by a process of trial and error

Example: Polynomial Curve Fitting

- N observations of $x \in \mathbb{R}$: $\mathbf{x} \equiv (x_1, \ldots, x_N)^\top$
- Corresponding target values of $t \in \mathbb{R}$: $\mathbf{t} \equiv (t_1, \ldots, t_N)^\top$
- Goal: to exploit the training set to predict value of $\hat{t}$ from x



10 samples generated from $\sin(2\pi x)$ by adding Gaussian noise

- Polynomial function

$$y(x, \mathbf{w}) = w_0 + w_1 x + w_2 x^2 + \cdots + w_M x^M = \sum_{j=0}^{M} w_j x^j$$

  - M: order of the polynomial

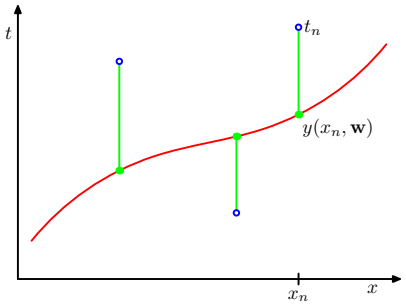
- Non-linear function of the input x
- Linear function of the coefficients $\mathbf{w} = [w_0, w_1, \ldots, w_M]^\top$

Linear model

[Error Function]

- Sum of squares of the errors between the predictions $y(x_n, \mathbf{w})$ for each data point x_n and target value t_n
  - Motivation for this choice of error function: discussed later

$$E(\mathbf{w}) = \frac{1}{2}\sum_{n=1}^{N}\{y(x_n, \mathbf{w}) - t_n\}^2$$



- Solve the problem by choosing the value of $\mathbf{w}$ for which $E(\mathbf{w})$ is as small as possible

$$\min_{\mathbf{w}} E(\mathbf{w})$$

[Minimization of Error Function]

- Quadratic in coefficients $\mathbf{w}$
- Derivative w.r.t. coefficients will be linear in elements of $\mathbf{w}$
- Unique solution!!! $\mathbf{w}^\star$

$$E(\mathbf{w}) = \frac{1}{2}\sum_{n=1}^{N}\{y(x_n, \mathbf{w}) - t_n\}^2$$

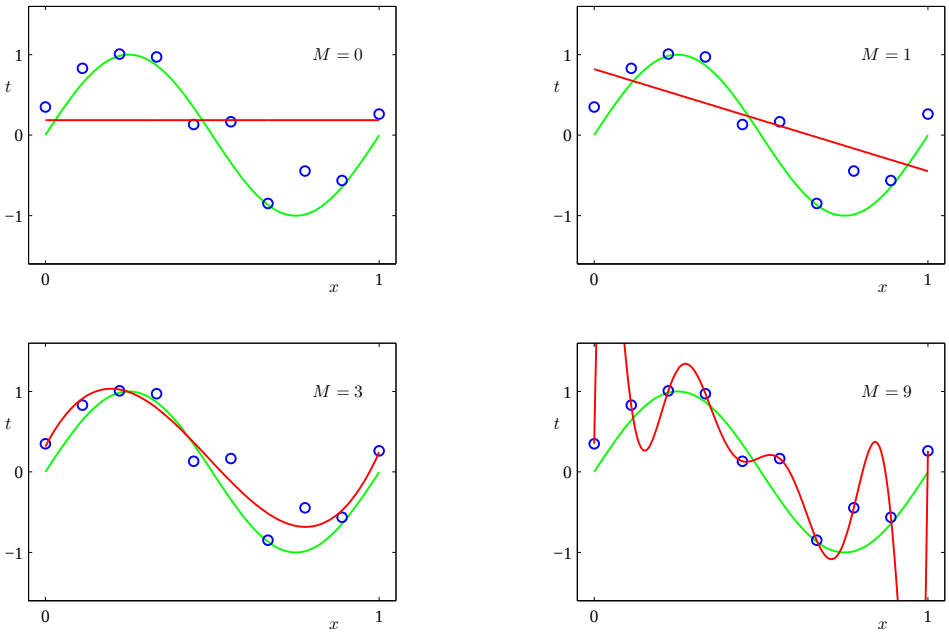
$$y(x, \mathbf{w}) = \sum_{j=0}^{M} w_j x^j$$

$$\frac{\partial E(\mathbf{w})}{\partial w_i} = \sum_{n=1}^{N}\left\{\sum_{j=0}^{M} w_j x_n^j - t_n\right\} x_n^i = 0$$

$$\sum_{n=1}^{N}\sum_{j=0}^{M} w_j x_n^{i+j} = \sum_{n=1}^{N} t_n x_n^i$$

[Choosing the Order of M]

- Model comparison or model selection



[Generalization Performance]

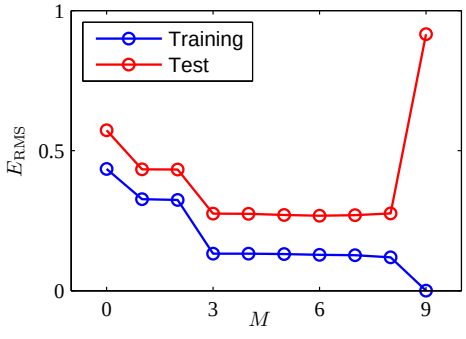
- Consider a separate test set of 100 points
- For each value of M, evaluate the error function for training data and test data

$$E\left(\mathbf{w}^{\star}\right)=\frac{1}{2}\sum_{n=1}^{N}\left\{y\left(x_n,\mathbf{w}^{\star}\right)-t_n\right\}^2$$

- Use Root-Mean-Square (RMS) error

$$E_{\mathrm{RMS}}=\sqrt{2E\left(\mathbf{w}^{\star}\right)/N}$$

  - Division by N allows different sizes of N to be compared on equal footing
  - Square root ensures E_{RMS} is measured in same units as t



Paradoxical !?

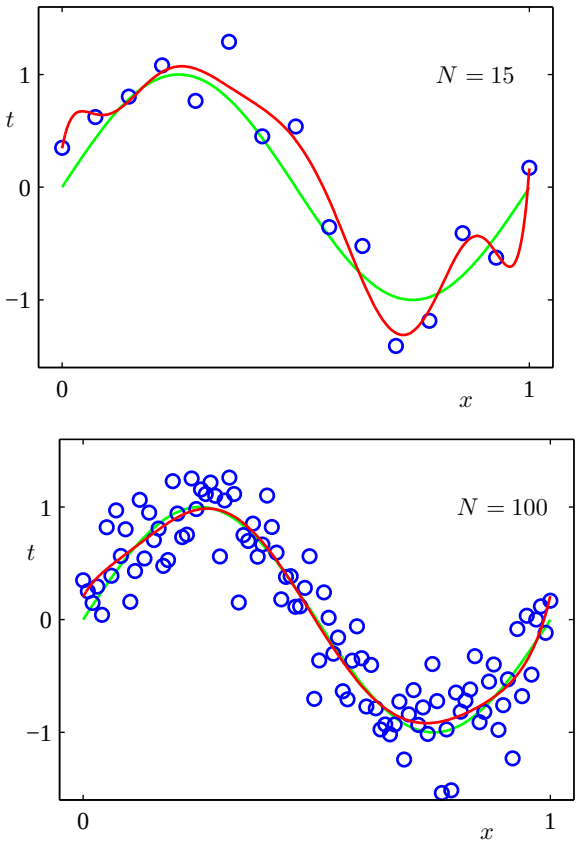
Table of the coefficients $\mathbf{w}^\star$ for polynomials of various order. Observe how the typical magnitude of the coefficients increases dramatically as the order of the polynomial increases.

	$M=0$	$M=1$	$M=6$	$M=9$
$w_0^\star$	0.19	0.82	0.31	0.35
$w_1^\star$		-1.27	7.99	232.37
$w_2^\star$			-25.43	-5321.83
$w_3^\star$			17.37	48568.31
$w_4^\star$				-231639.30
$w_5^\star$				640042.26
$w_6^\star$				-1061800.52
$w_7^\star$				1042400.18
$w_8^\star$				-557682.99
$w_9^\star$				125201.43

- As M increases, magnitude of coefficients increases
- At $M = 9$ finely tuned to random noise in target values

More flexible polynomials with large values of M are becoming increasingly tuned to the random noise on the target values.

[Increasing Data Set Size]



- For a given model complexity, overfitting problem is less severe as the size of a data set increases

- Unsatisfying to limit the number of parameters to size of the training set
- More reasonable to choose model complexity according to the complexity of the problem itself
- The least squares is a specific case of maximum likelihood. (Discussed later)
  - Overfitting: a general property of maximum likelihood

- Bayesian approach avoids overfitting problem
  - Allows for the number of parameters to exceed the number of data points
  - Effective number of parameters adapts automatically to the size of data set

[Regularization of Least Squares]

- Using relatively complex models with data sets of limited size

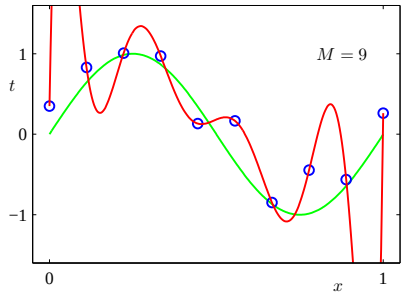
Table of the coefficients $\mathbf{w}^\star$ for polynomials of various order. Observe how the typical magnitude of the coefficients increases dramatically as the order of the polynomial increases.

	$M=0$	$M=1$	$M=6$	$M=9$
$w_0^\star$	0.19	0.82	0.31	0.35
$w_1^\star$		-1.27	7.99	232.37
$w_2^\star$			-25.43	-5321.83
$w_3^\star$			17.37	48568.31
$w_4^\star$				-231639.30
$w_5^\star$				640042.26
$w_6^\star$				-1061800.52
$w_7^\star$				1042400.18
$w_8^\star$				-557682.99
$w_9^\star$				125201.43

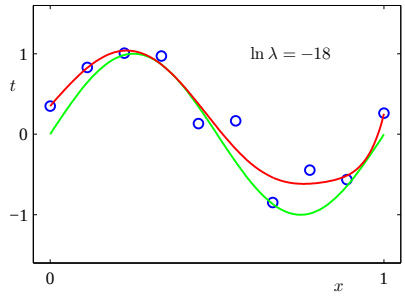
- Add a penalty term to error function to discourage coefficients from reaching large values

$$\tilde{E}(\mathbf{w}) = \frac{1}{2}\sum_{n=1}^{N}\{y(x_n, \mathbf{w}) - t_n\}^2 + \frac{\lambda}{2}\|\mathbf{w}\|^2$$

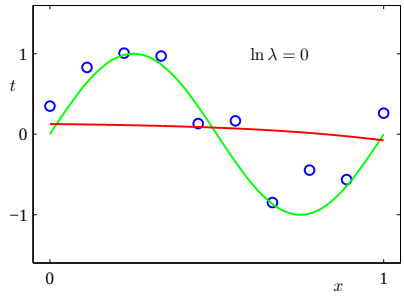
  - λ: governing the relative importance of the regularization term
  - Can be minimized exactly in a closed form
  - a.k.a. 'shrinkage' in statistics, 'weight decay' in neural networks



No regularization ($\lambda = 0$)



Optimal



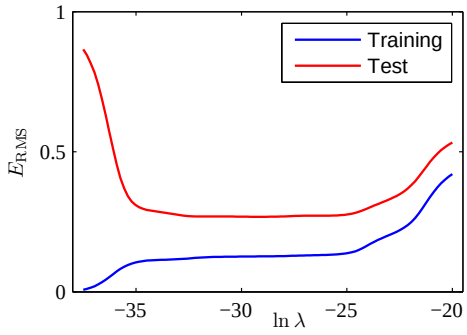
Too large

Table of the coefficients $\mathbf{w}^\star$ for $M = 9$ polynomials with various values for the regularization parameter λ. Note that $\ln\lambda = -\infty$ corresponds to a model with no regularization, i.e., to the graph at the bottom right in Figure 1.4. We see that, as the value of λ increases, the typical magnitude of the coefficients gets smaller.

	$\ln\lambda = -\infty$	$\ln\lambda = -18$	$\ln\lambda = 0$
$w_0^\star$	0.35	0.35	0.13
$w_1^\star$	232.37	4.74	-0.05
$w_2^\star$	-5321.83	-0.77	-0.06
$w_3^\star$	48568.31	-31.97	-0.05
$w_4^\star$	-231639.30	-3.89	-0.03
$w_5^\star$	640042.26	55.28	-0.02
$w_6^\star$	-1061800.52	41.32	-0.01
$w_7^\star$	1042400.18	-45.95	-0.00
$w_8^\star$	-557682.99	-91.53	0.00
$w_9^\star$	125201.43	72.68	0.01

Impact of Regularization

- λ: controls the complexity of the model and hence degree of overfitting
  - Analogous to the choice of M
- Suggestion: partition data into two sets
  - Training set: to determine coefficients $\mathbf{w}$
  - Validation set: to optimize model complexity (M or λ)
  - BUT, too wasteful of valuable training data; need to seek more sophisticated approaches



General Learning Scheme in PRML

Given *i.i.d.* samples $X = \{\mathbf{x}_n, y_n\}_{n=1}^{N}$,
the aim is to build a good and useful approximation to y_n

$$\mathbf{x} \quad \Rightarrow \quad f\left(\mathbf{x}|\theta\right) \quad \Rightarrow \quad y$$

1. Model: $f\left(\mathbf{x}_n|\theta\right) \rightarrow$ enough capacity, tractable inference
2. Loss function: $J(\theta|X) = \sum_n L\left(y_n, f(\mathbf{x}_n|\theta)\right) \rightarrow$ sufficient training data
3. Learning: $\theta^* = \underset{\theta}{\operatorname{argmin}}\, J(\theta|X) \rightarrow$ good optimization method

Probabilistic Approach in PRML: Overview

[Frequentist vs. Bayesian]

- Classical or Frequentist view of probabilities
  - Probability is frequency of a random, repeatable event
  - Frequency of a tossed coin coming up heads is 1/2
- Bayesian view
  - Probability is a quantification of uncertainty
  - Examples of uncertain events as probabilities
    - Whether the moon was once in its own orbit around the sun
    - Whether the Arctic ice cap will have disappeared by the end of the century

Bayesian Representation of Uncertainty

- Use of probability to represent uncertainty is not an ad-hoc choice
- (Cox, 1946) If numerical values are used to represent degrees of belief, then simple set of axioms for manipulating degrees of belief leads to the sum and product rules of probability
- (Jaynes, 2003) Probability theory can be regarded as an extension of Boolean logic to situations involving uncertainty

Maximum Likelihood Estimation

In frequentist setting, $\mathbf{w}$ is considered to be a fixed parameter

- $\mathbf{w}$ is set to a value that maximizes the likelihood function $p(\mathcal{D}|\mathbf{w})$
- Correspond to choosing the value of $\mathbf{w}$ for which the probability of the observations is maximized
- In PRML literature, the negative log of the likelihood function is widely used as an '*error function*'
  - Negative logarithm: a monotonically decreasing function
  - Maximizing the likelihood $\equiv$ minimizing the error

Bayesian Approach

- Quantify uncertainty around the choice of parameters $\mathbf{w}$
  - e.g., $\mathbf{w}$ is a vector of parameters in curve fitting
- Uncertainty before observing data: by a prior probability distribution $p(\mathbf{w})$
- Given an observed data set $\mathcal{D} = \{\mathbf{x}, \mathbf{t}\}$, $\mathbf{x} \equiv (x_1, \ldots, x_N)^\top$, $\mathbf{t} \equiv (t_1, \ldots, t_N)^\top$
  - Uncertainty in $\mathbf{w}$ after observing $\mathcal{D}$ by Bayes rule:

$$p(\mathbf{w}|\mathcal{D}) = \frac{p(\mathcal{D}|\mathbf{w})\, p(\mathbf{w})}{p(\mathcal{D})}$$

$$\underbrace{p(\mathbf{w}|\mathcal{D})}_{\text{posterior}} = \frac{\overbrace{p(\mathcal{D}|\mathbf{w})}^{\text{likelihood}}\overbrace{p(\mathbf{w})}^{\text{prior}}}{\underbrace{p(\mathcal{D})}_{\text{evidence}}} = \frac{p(\mathcal{D}|\mathbf{w})p(\mathbf{w})}{\int p(\mathcal{D}|\mathbf{w}')p(\mathbf{w}')d\mathbf{w}'}$$

- $p(\mathcal{D}|\mathbf{w})$ can be viewed as a function of $\mathbf{w}$
  - Represents how probable the data set is for different settings of the parameter vector $\mathbf{w}$
  - Called '*likelihood function*'
  - Not a probability distribution over $\mathbf{w}$
  - Its integral w.r.t. $\mathbf{w}$ does not (necessarily) equal one
- $p(\mathcal{D})$: a normalization factor, involves marginalization over $\mathbf{w}$
- Allows us to modify our prior probability into a posterior probability by taking information provided by $\mathcal{D}$ into account

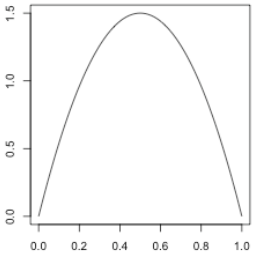
- Frequentist perspective
  - $\mathbf{w}$ is considered to be a fixed parameter determined by some form of 'estimator'
  - Error bars on this estimate are obtained by considering the distribution of possible data sets $\mathcal{D}$ (bootstrap)
- Bayesian perspective
  - There is only a single data set $\mathcal{D}$ (the one that is actually observed)
  - The uncertainty in the parameters is expressed as a probability distribution over $\mathbf{w}$

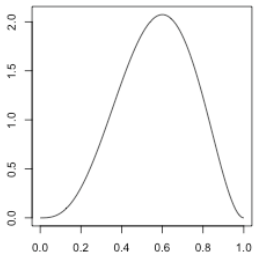
In both paradigms, the likelihood function plays a central role

Go to Bootstrap

Bayesian vs. Frequentist

- Bayesian: inclusion of prior knowledge arises naturally
- Coin tossing exaxmple
  - Fair-looking coin tossing three times: $\{H, H, H\}$
  - Frequentist: $P(H) = 1$; all future tosses will hand heads!!!
  - Bayesian: with any reasonable prior, will lead to less extreme conclusion





- Controversy and debate with the relative merits
- One common criticism of the Bayesian approach
  - The prior distribution is often selected on the basis of mathematical convenience rather than as a reflection of any prior belief

[Conjugate Priors]

[From Wikipedia] In Bayesian probability theory, if the posterior distributions $p(\theta|x)$ are in the same family as the prior probability distribution $p(\theta)$, the prior and posterior are then called conjugate distributions, and the prior is called a conjugate prior for the likelihood function.

Likelihood	Model parameters	Conjugate prior distribution	Prior hyperparameters	Posterior hyperparameters	Interpretation of hyperparameters	Posterior predictive[note 4]
Normal with known variance σ^2	μ (mean)	Normal	$\mu_0,\ \sigma_0^2$	$\left(\frac{\mu_0}{\sigma_0^2}+\frac{\sum_{i=1}^n x_i}{\sigma^2}\right)\Big/\left(\frac{1}{\sigma_0^2}+\frac{n}{\sigma^2}\right),\ \left(\frac{1}{\sigma_0^2}+\frac{n}{\sigma^2}\right)^{-1}$	mean was estimated from observations with total precision (sum of all individual precisions) $1/\sigma_0^2$ and with sample mean μ_0	$\mathcal{N}(\tilde{x}\|\mu_0', \sigma_0^{2\prime}+\sigma^2)$[5]
Normal with known precision τ	μ (mean)	Normal	$\mu_0,\ \tau_0$	$\left(\tau_0\mu_0+\tau\sum_{i=1}^n x_i\right)\Big/(\tau_0+n\tau),\ \tau_0+n\tau$	mean was estimated from observations with total precision (sum of all individual precisions) τ_0 and with sample mean μ_0	$\mathcal{N}\left(\tilde{x}\|\mu_0', \frac{1}{\tau_0'}+\frac{1}{\tau}\right)$[5]
Normal with known mean μ	σ^2 (variance)	Inverse gamma	$\alpha,\ \beta$ [note 5]	$\alpha+\frac{n}{2},\ \beta+\frac{\sum_{i=1}^n (x_i-\mu)^2}{2}$	variance was estimated from 2α observations with sample variance β/α (i.e. with sum of squared deviations 2β, where deviations are from known mean μ)	$t_{2\alpha'}(\tilde{x}\|\mu, \sigma^2=\beta'/\alpha')$[5]
Normal with known mean μ	σ^2 (variance)	Scaled inverse chi-squared	$\nu,\ \sigma_0^2$	$\nu+n,\ \frac{\nu\sigma_0^2+\sum_{i=1}^n (x_i-\mu)^2}{\nu+n}$	variance was estimated from ν observations with sample variance σ_0^2	$t_{\nu'}(\tilde{x}\|\mu, \sigma_0^{2\prime})$[5]
Normal with known mean μ	τ (precision)	Gamma	$\alpha,\ \beta$[note 3]	$\alpha+\frac{n}{2},\ \beta+\frac{\sum_{i=1}^n (x_i-\mu)^2}{2}$	precision was estimated from 2α observations with sample variance β/α (i.e. with sum of squared deviations 2β, where deviations are from known mean μ)	$t_{2\alpha'}(\tilde{x}\|\mu, \sigma^2=\beta'/\alpha')$[5]
Normal[note 6]	μ and σ^2 Assuming exchangeability	Normal-inverse gamma	$\mu_0,\ \nu,\ \alpha,\ \beta$	$\frac{\nu\mu_0+n\bar{x}}{\nu+n},\ \nu+n,\ \alpha+\frac{n}{2},$ $\beta+\frac{1}{2}\sum_{i=1}^{n}(x_i-\bar{x})^2+\frac{n\nu}{\nu+n}\frac{(\bar{x}-\mu_0)^2}{2}$ ▪ $\bar{x}$ is the sample mean	mean was estimated from ν observations with sample mean μ_0; variance was estimated from 2α observations with sample mean μ_0 and sum of squared deviations 2β	$t_{2\alpha'}\left(\tilde{x}\|\mu', \frac{\beta'(\nu'+1)}{\nu'\alpha'}\right)$[5]
Normal	μ and τ Assuming exchangeability	Normal-gamma	$\mu_0,\ \nu,\ \alpha,\ \beta$	$\frac{\nu\mu_0+n\bar{x}}{\nu+n},\ \nu+n,\ \alpha+\frac{n}{2},$ $\beta+\frac{1}{2}\sum_{i=1}^{n}(x_i-\bar{x})^2+\frac{n\nu}{\nu+n}\frac{(\bar{x}-\mu_0)^2}{2}$ ▪ $\bar{x}$ is the sample mean	mean was estimated from ν observations with sample mean μ_0, and precision was estimated from 2α observations with sample mean μ_0 and sum of squared deviations 2β	$t_{2\alpha'}\left(\tilde{x}\|\mu', \frac{\beta'(\nu'+1)}{\alpha'\nu'}\right)$[5]

Multivariate normal with known covariance matrix $\boldsymbol{\Sigma}$	$\boldsymbol{\mu}$ (mean vector)	Multivariate normal	$\boldsymbol{\mu}_0,\ \boldsymbol{\Sigma}_0$	$\left(\boldsymbol{\Sigma}_0^{-1}+n\boldsymbol{\Sigma}^{-1}\right)^{-1}\left(\boldsymbol{\Sigma}_0^{-1}\boldsymbol{\mu}_0+n\boldsymbol{\Sigma}^{-1}\bar{\mathbf{x}}\right),$ $\left(\boldsymbol{\Sigma}_0^{-1}+n\boldsymbol{\Sigma}^{-1}\right)^{-1}$ ▪ $\bar{\mathbf{x}}$ is the sample mean	mean was estimated from observations with total precision (sum of all individual precisions) $\boldsymbol{\Sigma}_0^{-1}$ and with sample mean $\boldsymbol{\mu}_0$	$\mathcal{N}(\tilde{\mathbf{x}}\|\boldsymbol{\mu}_0', \boldsymbol{\Sigma}_0'+\boldsymbol{\Sigma})$[5]
Multivariate normal with known precision matrix $\boldsymbol{\Lambda}$	$\boldsymbol{\mu}$ (mean vector)	Multivariate normal	$\boldsymbol{\mu}_0,\ \boldsymbol{\Lambda}_0$	$\left(\boldsymbol{\Lambda}_0+n\boldsymbol{\Lambda}\right)^{-1}\left(\boldsymbol{\Lambda}_0\boldsymbol{\mu}_0+n\boldsymbol{\Lambda}\bar{\mathbf{x}}\right),\ \left(\boldsymbol{\Lambda}_0+n\boldsymbol{\Lambda}\right)$ ▪ $\bar{\mathbf{x}}$ is the sample mean	mean was estimated from observations with total precision (sum of all individual precisions) $\boldsymbol{\Lambda}$ and with sample mean $\boldsymbol{\mu}_0$	$\mathcal{N}\left(\tilde{\mathbf{x}}\|\boldsymbol{\mu}_0', (\boldsymbol{\Lambda}_0'^{-1}+\boldsymbol{\Lambda}^{-1})^{-1}\right)$[5]
Multivariate normal with known mean $\boldsymbol{\mu}$	$\boldsymbol{\Sigma}$ (covariance matrix)	Inverse-Wishart	$\nu,\ \boldsymbol{\Psi}$	$n+\nu,\ \boldsymbol{\Psi}+\sum_{i=1}^{n}(\mathbf{x_i}-\boldsymbol{\mu})(\mathbf{x_i}-\boldsymbol{\mu})^T$	covariance matrix was estimated from ν observations with sum of pairwise deviation products $\boldsymbol{\Psi}$	$t_{\nu'-p+1}\left(\tilde{\mathbf{x}}\|\boldsymbol{\mu}, \frac{1}{\nu'-p+1}\boldsymbol{\Psi}'\right)$[5]
Multivariate normal with known mean $\boldsymbol{\mu}$	$\boldsymbol{\Lambda}$ (precision matrix)	Wishart	$\nu,\ \mathbf{V}$	$n+\nu,\ \left(\mathbf{V}^{-1}+\sum_{i=1}^{n}(\mathbf{x_i}-\boldsymbol{\mu})(\mathbf{x_i}-\boldsymbol{\mu})^T\right)^{-1}$	covariance matrix was estimated from ν observations with sum of pairwise deviation products $\mathbf{V}^{-1}$	$t_{\nu'-p+1}\left(\tilde{\mathbf{x}}\|\boldsymbol{\mu}, \frac{1}{\nu'-p+1}\mathbf{V}'^{-1}\right)$[5]
Multivariate normal	$\boldsymbol{\mu}$ (mean vector) and $\boldsymbol{\Sigma}$ (covariance matrix)	normal-inverse-Wishart	$\boldsymbol{\mu}_0,\ \kappa_0,\ \nu_0,\ \boldsymbol{\Psi}$	$\frac{\kappa_0\boldsymbol{\mu}_0+n\bar{\mathbf{x}}}{\kappa_0+n},\ \kappa_0+n,\ \nu_0+n,$ $\boldsymbol{\Psi}+\mathbf{C}+\frac{\kappa_0 n}{\kappa_0+n}(\bar{\mathbf{x}}-\boldsymbol{\mu}_0)(\bar{\mathbf{x}}-\boldsymbol{\mu}_0)^T$ ▪ $\bar{\mathbf{x}}$ is the sample mean ▪ $\mathbf{C}=\sum_{i=1}^{n}(\mathbf{x_i}-\bar{\mathbf{x}})(\mathbf{x_i}-\bar{\mathbf{x}})^T$	mean was estimated from κ_0 observations with sample mean $\boldsymbol{\mu}_0$; covariance matrix was estimated from ν_0 observations with sample mean $\boldsymbol{\mu}_0$ and with sum of pairwise deviation products $\boldsymbol{\Psi}=\nu_0\boldsymbol{\Sigma}_0$	$t_{\nu_0'-p+1}\left(\tilde{\mathbf{x}}\|\boldsymbol{\mu}_0', \frac{\kappa_0'+1}{\kappa_0'(\nu_0'-p+1)}\boldsymbol{\Psi}'\right)$[5]
Multivariate normal	$\boldsymbol{\mu}$ (mean vector) and $\boldsymbol{\Lambda}$ (precision matrix)	normal-Wishart	$\boldsymbol{\mu}_0,\ \kappa_0,\ \nu_0,\ \mathbf{V}$	$\frac{\kappa_0\boldsymbol{\mu}_0+n\bar{\mathbf{x}}}{\kappa_0+n},\ \kappa_0+n,\ \nu_0+n,$ $\left(\mathbf{V}^{-1}+\mathbf{C}+\frac{\kappa_0 n}{\kappa_0+n}(\bar{\mathbf{x}}-\boldsymbol{\mu}_0)(\bar{\mathbf{x}}-\boldsymbol{\mu}_0)^T\right)^{-1}$ ▪ $\bar{\mathbf{x}}$ is the sample mean ▪ $\mathbf{C}=\sum_{i=1}^{n}(\mathbf{x_i}-\bar{\mathbf{x}})(\mathbf{x_i}-\bar{\mathbf{x}})^T$	mean was estimated from κ_0 observations with sample mean $\boldsymbol{\mu}_0$; covariance matrix was estimated from ν_0 observations with sample mean $\boldsymbol{\mu}_0$ and with sum of pairwise deviation products $\mathbf{V}^{-1}$	$t_{\nu_0'-p+1}\left(\tilde{\mathbf{x}}\|\boldsymbol{\mu}_0', \frac{\kappa_0'+1}{\kappa_0'(\nu_0'-p+1)}\mathbf{V}'^{-1}\right)$ [5]
Uniform	$U(0,\theta)$	Pareto	$x_m,\ k$	$\max\{x_1,\ldots,x_n,x_m\},\ k+n$	k observations with maximum value x_m	

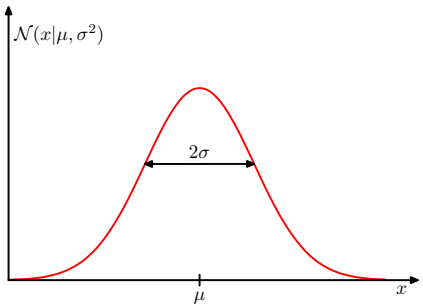
Practicality of Bayesian Approach

- Bayesian framework: its origins in the 18th century
- For a long time, severely limited by the difficulties in carrying through the full Bayesian procedure, particularly the need to marginalize (sum or integrate) over the whole of parameter space
- Factors making it practical
  - Dramatic improvements in the speed and memory capacity of computers
  - Sampling methods such as Markov Chain Monte Carlo (MCMC) methods (Chapter 11)
    - Used for small-scale problems
  - Deterministic approximation schemes such as variational Bayes and expectation propagation (Chapter 10)
    - Used for large-scale problems

Curve Fitting: Probabilistic Perspective

Normal or Gaussian Distribution

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

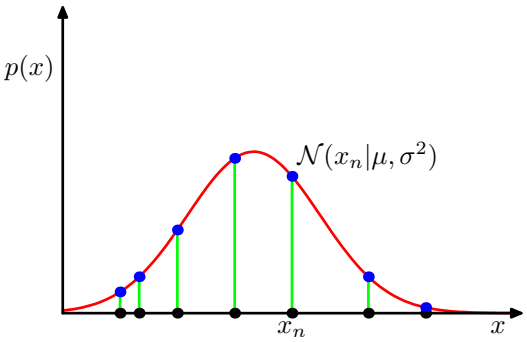


- Parameters: mean μ, variance σ^2
- Standard deviation σ, precision $\beta = 1/\sigma^2$
- Moments: $\mathbb{E}\left[x\right] = \mu$, $\mathrm{Var}\left[x\right] = \sigma^2$
- The maximum of a distribution is its mode that coincides with the mean

Likelihood Function for Gaussian

- Given i.i,d. N observations $\mathbf{x} = (x_1,\ldots,x_N)^\top$, drawn from $\mathcal{N}\left(x|\mu,\sigma^2\right)$
- Unknown parameters: μ, σ^2
- Probability of the data set

$$p\left(\mathbf{x}|\mu,\sigma^2\right) = \prod_{n=1}^{N}\mathcal{N}\left(x_n|\mu,\sigma^2\right)$$



- In MLE, we maximize $p\left(\mathbf{x}|\mu,\sigma^2\right)$, rather than $p\left(\mu,\sigma^2|\mathbf{x}\right)$ (Discussed later)

- Log-likelihood

$$\ln p\left(\mathbf{x}|\mu,\sigma^2\right) = -\frac{1}{2\sigma^2}\sum_{n=1}^{N}(x_n-\mu)^2 - \frac{N}{2}\ln(\sigma^2) - \frac{N}{2}\ln(2\pi)$$

  - Convex w.r.t μ or w.r.t. σ^2 $\rightarrow$ unique solution for each

$$\begin{aligned}\mu_{\mathsf{ML}} &= \frac{1}{N}\sum_{n=1}^{N}x_n \\ \sigma^2_{\mathsf{ML}} &= \frac{1}{N}\sum_{n=1}^{N}(x_n-\mu_{\mathsf{ML}})^2\end{aligned}$$

- The solution for μ decouples from that for σ^2
- μ_{ML}, σ^2_{ML}: functions of the data set values $x_1,\ldots,x_N$

Bias Problem in MLE

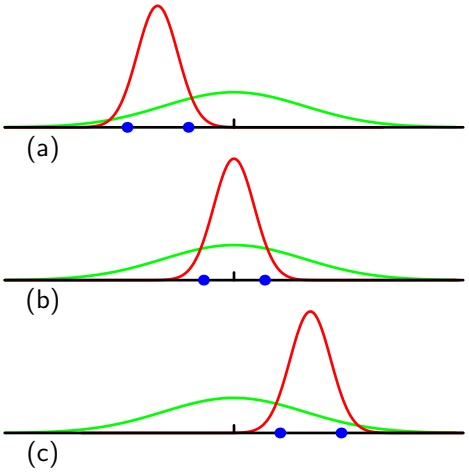
- MLE systemically underestimates variance
  - μ_{ML} (vs. μ)

$$\mathbb{E}\left[\mu_{\mathsf{ML}}\right] = \mathbb{E}\left[\frac{\sum_n x_n}{N}\right] = \frac{1}{N}\mathbb{E}\left[\sum_n x_n\right] = \frac{N\mu}{N} = \mu$$

$$\mathsf{Var}(\mu_{\mathsf{ML}}) = \mathsf{Var}\left(\frac{\sum_n x_n}{N}\right) = \frac{1}{N^2}\sum_n \mathsf{Var}\left(x_n\right) = \frac{N\sigma^2}{N^2} = \frac{\sigma^2}{N}$$

  - σ^2_{ML} (vs. σ^2)

$$\begin{aligned}\mathbb{E}\left[\sigma^2_{\mathsf{ML}}\right] &= \mathbb{E}\left[\frac{\sum_n (x_n-\mu_{\mathsf{ML}})^2}{N}\right] = \mathbb{E}\left[\frac{\sum_n (x_n)^2 - N\mu^2_{\mathsf{ML}}}{N}\right] \\ &= \frac{\mathbb{E}\left[\sum_n (x_n)^2\right] - N\mathbb{E}\left[\mu^2_{\mathsf{ML}}\right]}{N} \\ &= \frac{N\left(\sigma^2+\mu^2\right) - N\left(\sigma^2/N+\mu^2\right)}{N} = \left(\frac{N-1}{N}\right)\sigma^2 \neq \sigma^2\end{aligned}$$

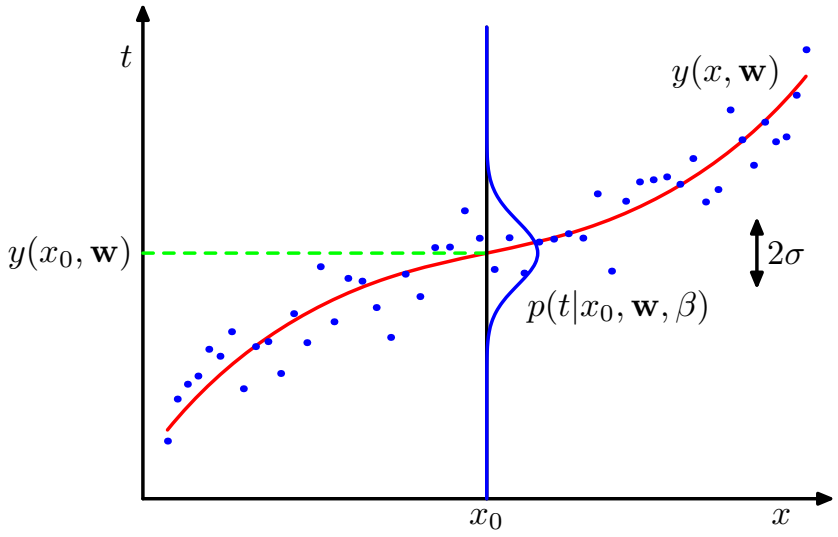


- Green curve: true Gaussian distribution from which data is generated
- Three red curves: Gaussian distributions obtained by fitting to three data sets, each consisting of two data points shown in blue, using the maximum likelihood results
- Averaged across the three data sets, the mean is correct, but the variance is systematically underestimated because it is measured relative to the sample mean and not relative to the true mean

Curve Fitting from a Probabilistic Perspective

- Goal: to predict for the target variable t given some new value of the input variable x
- Given a set of training data: $\mathbf{x}= (x_1, \ldots, x_N)^\top$ and $\mathbf{t}= (t_1 \ldots, t_N)^\top$
- Express our uncertainty over the value of the target variable using a probability distribution
- Assumption: given the value of x, the corresponding value of t has a Gaussian distribution with $\mu = y(x, \mathbf{w})$ of the polynomial curve

$$p(t|x, \mathbf{w}, \beta) = \mathcal{N}\left(t|y(x, \mathbf{w}), \beta^{-1}\right)$$



- Unknown parameters: $\mathbf{w}$, β
- Likelihood function (i.i.d.):
  $p\left(\mathbf{t}|\mathbf{x}, \mathbf{w}, \beta\right) = \prod_{n=1}^{N} \mathcal{N}\left(t_n|y\left(x_n, \mathbf{w}\right), \beta^{-1}\right)$

- Log likelihood:

$$\ln p\left(\mathbf{t}|\mathbf{x}, \mathbf{w}, \beta\right) = -\frac{\beta}{2}\sum_{n=1}^{N}\left\{y\left(x_n, \mathbf{w}\right) - t_n\right\}^2 + \frac{N}{2}\ln(\beta) - \frac{N}{2}\ln(2\pi)$$

- To find maximum likelihood solution for polynomial coefficients $\mathbf{w}$
  - $\frac{N}{2}\ln(\beta) - \frac{N}{2}\ln(2\pi)$: not depend on $\mathbf{w}$
  - Scailing by a positive constant coefficient does not alter the location of the maximum w.r.t. $\mathbf{w}$: $\frac{\beta}{2} \rightarrow \frac{1}{2}$
  - Maximization of the log likelihood $\equiv$ minimization of the the negative log likelihood w.r.t. $\mathbf{w}$
  - End up to minimizing the *sum-of-squares error function*

$$\frac{1}{2}\sum_{n=1}^{N}\left\{y\left(x_n, \mathbf{w}\right) - t_n\right\}^2$$

A sum-of-squares error function has arisen as a consequence of maximizing likelihood under the assumption of a Gaussian noise distribution

- To determine the precision parameter β
  - Maximizing w.r.t. β

$$\frac{1}{\beta_{\mathsf{ML}}} = \frac{1}{N}\sum_{n=1}^{N}\left\{y\left(x_n, \mathbf{w}_{\mathsf{ML}}\right) - t_n\right\}^2$$

  - First determine the parameter vector $\mathbf{w}_{\mathsf{ML}}$, governing the mean and subsequently use this to find the precision β

- Making predictions for new values of x
  - Having determined the parameters $\mathbf{w}$ and β
  - Expressed in terms of the predictive distribution that gives the probability distribution over t, rather than simply a point estimate

$$p\left(t|x, \mathbf{w}_{\mathsf{ML}}, \beta_{\mathsf{ML}}\right) = \mathcal{N}\left(t|y\left(x, \mathbf{w}_{\mathsf{ML}}\right), \beta_{\mathsf{ML}}^{-1}\right)$$

Towards a Bayesian Approach

- Introduce a prior distribution over the polynomial coefficients $\mathbf{w}$
- For simplicity, let us consider a Gaussian distribution

$$p\left(\mathbf{w}|\alpha\right) = \mathcal{N}\left(\mathbf{w}|0, \alpha^{-1}I\right) = \left(\frac{\alpha}{2\pi}\right)^{(M+1)/2}\exp\left[-\frac{\alpha}{2}\mathbf{w}^{\top}\mathbf{w}\right]$$

  - α: precision of the distribution
    - '*Hyperparameter*': controls distribution of model parameters
  - $M+1$: total number of elements in the vector $\mathbf{w}$ for an M^{th} order polynomial

- Posterior distribution
  - Using Bayes theorem, posterior distribution for $\mathbf{w}$ is proportional to production of prior distribution and likelihood function

$$p(\mathbf{w}|\mathbf{x}, \mathbf{t}, \alpha, \beta) \propto p(\mathbf{t}|\mathbf{x}, \mathbf{w}, \beta)\, p(\mathbf{w}|\alpha)$$

- Maximum A Posteriori (MAP)
  - $\mathbf{w}$ can be determined by finding the most probable value of $\mathbf{w}$ given the data, i.e., maximizing the posterior distribution
  - Taking negative logarithm of $p(\mathbf{w}|\mathbf{x}, \mathbf{t}, \alpha, \beta)$ and combining with $\ln p(\mathbf{t}|\mathbf{x}, \mathbf{w}, \beta)$ and $\ln p(\mathbf{w}|\alpha)$

$$\min_{\mathbf{w}} \frac{\beta}{2} \sum_{n=1}^{N} \{y(x_n, \mathbf{w}) - t_n\}^2 + \frac{\alpha}{2} \mathbf{w}^\top \mathbf{w}$$

Maximizing the posterior distribution is equivalent to minimizing the regularized sum-of-squares error function, with a regularization parameter given by $\lambda = \alpha/\beta$

Fully Bayesian

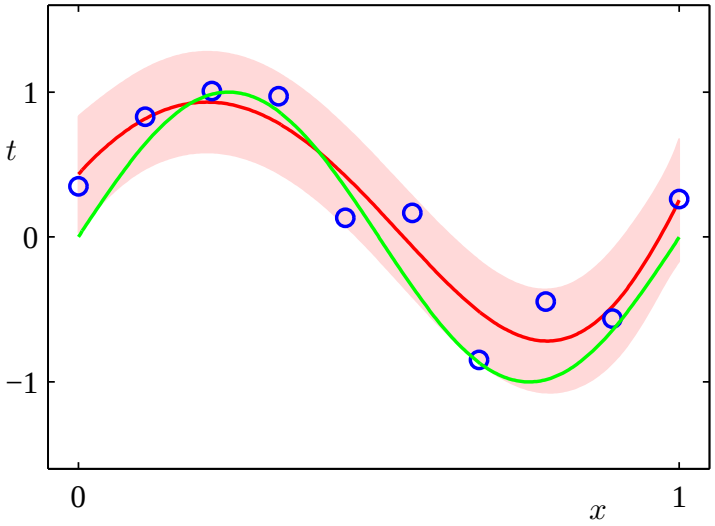
- Although included a prior distribution $p(\mathbf{w}|\alpha)$, so far still making a point estimate of $\mathbf{w}$; not yet amount to a Bayesian treatment
- In a full Bayesian approach, we should consistently apply the sum and product rules of probability
  - Integrate over all values of $\mathbf{w}$
  - Marginalizations lie at the heart of Bayesian methods for PRML
    - For simplicity, assume that α and β are known in advance (in practice should be inferred from data in a Bayesian setting; discussed later)

$$\begin{aligned} p(t|x, \mathbf{x}, \mathbf{t}) &= \int p(t, \mathbf{w}|x, \mathbf{x}, \mathbf{t})\, d\mathbf{w} \\ &= \int p(t|x, \mathbf{w}, \mathbf{x}, \mathbf{t})\, p(\mathbf{w}|x, \mathbf{x}, \mathbf{t})\, d\mathbf{w} \\ &= \int \underbrace{p(t|x, \mathbf{w})}_{=\mathcal{N}(t|y(x,\mathbf{w}),\beta^{-1})} \underbrace{p(\mathbf{w}|\mathbf{x}, \mathbf{t})}_{\text{posterior}}\, d\mathbf{w} \end{aligned}$$

$$p(t|x,\mathbf{x},\mathbf{t}) = \int p(t|x,\mathbf{w})\, p(\mathbf{w}|\mathbf{x},\mathbf{t})\, d\mathbf{w}$$

$$\begin{aligned}
p(t|x,\mathbf{w},\beta) &= \mathcal{N}\left(t|m(x), s^2(x)\right) \\
m(x) &= \beta\phi(x)^\top \mathbf{S} \sum_{n=1}^{N} \phi(x_n)\, t_n \\
s^2(x) &= \beta^{-1} + \phi(x)^\top \mathbf{S}\phi(x) \\
\mathbf{S}^{-1} &= \alpha\mathbf{I} + \beta \sum_{n=1}^{N} \phi(x_n)\,\phi(x_n)^\top \\
\phi(x) &= \left[\phi_i(x) = x^i\right]^\top, i = 0, \ldots, M
\end{aligned}$$

- Mean $m(x)$ and variance $s^2(x)$: dependent on a test point x
- β^{-1}: uncertainty in the predicted value of t due to noise on the target variables; expressed already in the maximum likelihood predictive distribution through β_{ML}^{-1}
- $\phi(x)^\top \mathbf{S}\phi(x)$: arises from the uncertainty in the parameter $\mathbf{w}$, a consequence of the Bayesian treatment



- Predictive distribution resulting from a Bayesian treatment of polynomial curve fitting using an $M = 9$ polynomial
- Fixed parameters $\alpha = 5 \times 10^{-3}$ and $\beta = 11.1$ (corresponding to the known noise variance)
- Red curve: the mean of the predictive distribution
- Red region corresponds to ± 1 standard deviation around the mean

Models in Polynomial Curve Fitting

Model Selection

- Optimal order of polynomial gives the best generalization
- Order of the polynomial controls the number of free parameters, i.e., $M + 1$, in the model and thereby model complexity
- With regularized least squares, λ also controls model complexity
- It may be beneficial to optimize both the order of the polynomial and the regularization control parameter λ jointly

Validation Set to Select Model

- Performance on training set is not a good indicator of predictive performance
- If there is a plenty of observations
  - Use some of them to train a range of models or a given model with a range of values for its parameters
  - Compare them on an independent set, called 'validation set'
  - Select one that achieves the best performance
- If data set is small then some over-fitting can occur and it is necessary to keep aside a test set

- Wish to use as much of the available data as possible for training, which causes to reduce the size of a validation set
- If the validation set is small, it will give a relatively noisy estimate of the predictive performance

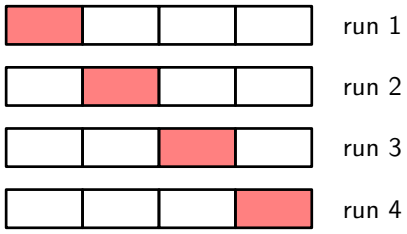
K-fold Cross-Validation

(1) Partition all the available data into K groups

(2) Use $K-1$ groups for training and evaluate the trained model on the remains group

(3) Repeat the steps of (1) and (2) for all K choices of held-out group

(4) Averaged performance from K runs



Equally sized 4 folds, i.e., $K = 4$

What if you have a scare data set?

- 'Leave-one-out' technique: $K = N$
  - N is the total number of observations

- Major drawback: number of training runs increases by a factor of K
  - When the training is computationally expensive
  - Multiple complexity parameters for a single model
    - e.g., several regularization parameters

- Information criteria
  - Need to find a measure of performance that depends only on the training data, not suffering from bias due to over-fitting
  - Akaike Information Criterion (AIC) (Akaike, 1974)

$$\ln p\left(\mathcal{D}|\mathbf{w}_{\mathrm{ML}}\right) - M$$

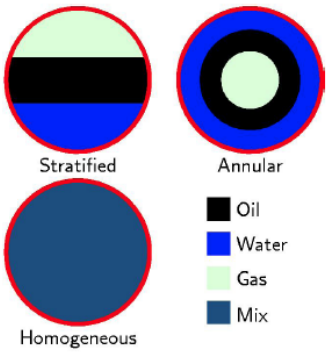
    - $p\left(\mathcal{D}|\mathbf{w}_{\mathrm{ML}}\right)$: best-fit log likelihood
    - M: number of adjustable parameters
    - The higher, the better
  - Bayesian Information Criterion (BIC) (Section 4)
    - Do not take account of the uncertainty in the model parameters, but tend to favour overly simple models

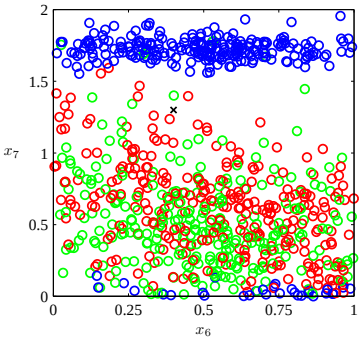
Curse of Dimensionality

Generally work in high-dimensional spaces with many variables in PRML

Example: Oil Flow Data

- Measurements takes from a pipeline containing a mixture of oil/water/gas
  - fractions of the three materials can vary
  - 12-dimensional input vector
- Three classes: stratified, annular, homogeneous

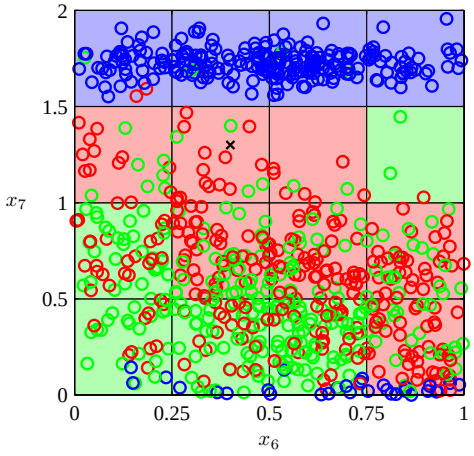


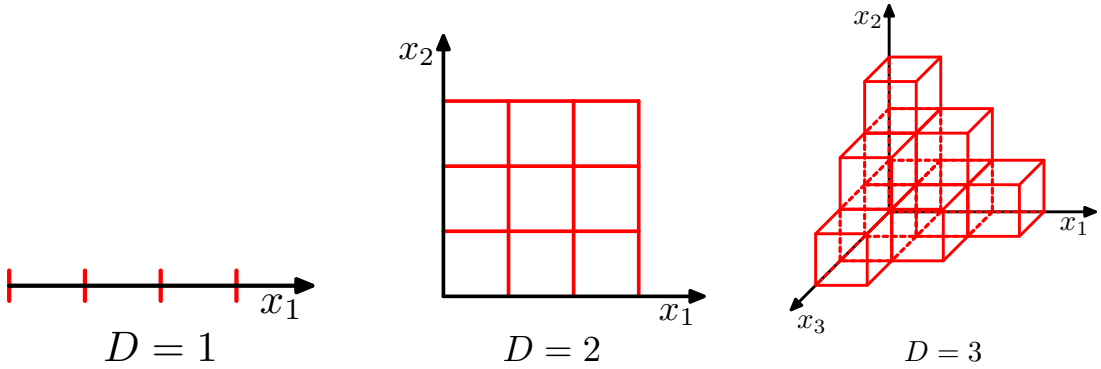


Which class does the cross belong to?

- Intuition: the identify of the cross should be determined strongly by nearby points from the training set and less strongly by more distant points (turns out to be reasonable and discussed more fully later)
- Simple learning strategy: to divide the input space into regular cells

Cell-based classification





- Naïve approach of cell-based voting will fail in a high-dimensional space
  - Exponential growth of cells with dimensionality
  - Hardly any points in each cell

Back to polynomial curve fitting with D input variables
a general polynomial with coefficients up to order 3

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

- As D increases, the number of independent coefficients in $\mathbf{w}$ grows proportionally to D^3
- For M-th order polynomial, grows proportionally to D^M

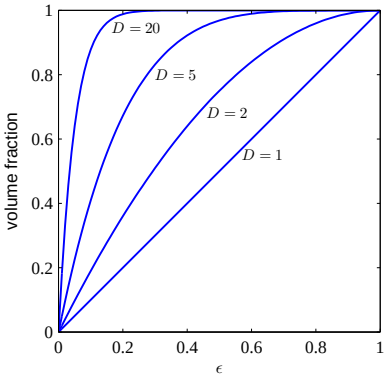
Curse of dimensionality (Bellman, 1961)
the severe difficulty that can arise in spaces of many dimensions...

Volume of a sphere in high dimensions

- Consider a sphere of radius $r = 1$ in a D-dimensional space
- What fraction of the volume lies between radius $r = 1 - \epsilon$ and $r = 1$?

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

∵ Volume of a sphere of radius r in D dimensions: $V_D(r) = K_D r^D$
- K_D: a constant depending on D



For large D, the fraction tends to 1 even for small values of ϵ; Thus, in spaces of high dimensionality, most of the volume of a sphere is concentrated in a thin shell near the surface!

Prevent us from finding effective techniques applicable to high-dimensional space?

1. Real data will often be confined to a region of the space having lower effective dimensionality
   - In particular, the directions over which important variations in the target variables occur may be so confined.
2. Real data will typically exhibit some smoothness properties (at least locally)
   - so that for the most part small changes in the input variables will produce small changes in the target variables
   - we can exploit local interpolation-like techniques to allow use to make predictions of the target variables for new values of the input variables.

Decision Theory

When combined with probability theory, it allows us to make optimal decisions in situations involving uncertainty such as those encountered in PRML.

Decision Theory

- Input vector $\mathbf{x}$, target vector $\mathbf{t}$
  - Regression: $\mathbf{t}$ continuous values
  - Classification: $\mathbf{t}$ discrete class labels

- Summary of uncertainty associated is given by $p(\mathbf{x}, \mathbf{t})$
  - Inference: to obtain $p(\mathbf{x}, \mathbf{t})$ from data (a very difficult problem)
  - Decision: make a specific prediction for the value of $\mathbf{t}$, based on which we take a specific action

[Example: Medical Diagnosis Problem]

- Problem definition: given an X-ray image of a patient, whether the patient has cancer or not
  - Input vector $\mathbf{x}$: a set of pixel intensities
  - Output variable t: $\mathcal{C}_1 = 0$ or $\mathcal{C}_2 = 1$ (convenience for probabilistic models)

- Inference: to determine $p(\mathbf{x}, \mathcal{C}_k)$
  - gives most complete description of the situation
- Decision: (in the end) to decide whether to give treatment or not

- Decision theory tells how to make optimal decisions given the appropriate probabilities.

How do probabilities play a role in making a decision?

$$p\left(\mathcal{C}_k|\mathbf{x}\right) = \frac{p\left(\mathbf{x}|\mathcal{C}_k\right)p\left(\mathcal{C}_k\right)}{p\left(\mathbf{x}\right)} \quad \text{[Bayes decision]}$$

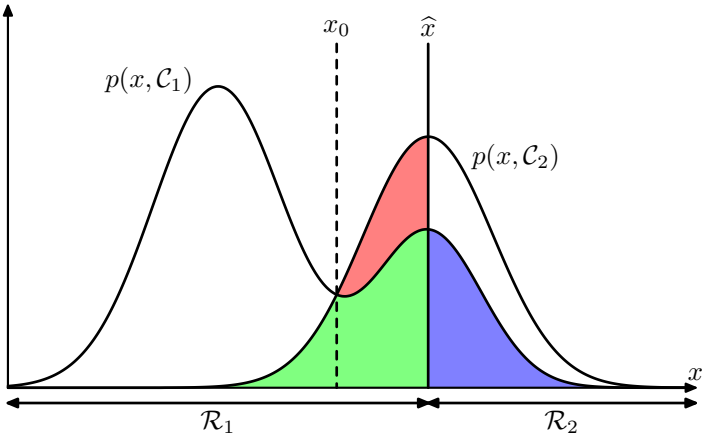
- Quantities in Bayes theorem can be obtained from $p\left(\mathbf{x}, \mathcal{C}_k\right)$ either by marginalizing or conditioning w.r.t. appropriate variable(s)
- To minimize the chance of mis-assignment, choose the class having the higher posterior probability (intuitive!!!)

Minimizing Misclassification Error

Goal: to make as few misclassifications as possible

- Rule to assign each value of $\mathbf{x}$ to one of the available classes
  - Dividing regions (*decision regions*) $\mathcal{R}_k \equiv \mathcal{C}_k$
  - *Decision boundaries* or *decision surfaces*: boundaries between decision regions
  - Each decision region does not need to be contiguous but could comprise some number of disjoint regions

- Probability of making a mistake (2-class)



$$\begin{aligned} p\,(\text{mistake}) &= p\,(\mathbf{x} \in \mathcal{R}_1, \mathcal{C}_2) + p\,(\mathbf{x} \in \mathcal{R}_2, \mathcal{C}_1) \\ &= \int_{\mathcal{R}_1} p\,(\mathbf{x}, \mathcal{C}_2)\, d\mathbf{x} + \int_{\mathcal{R}_2} p\,(\mathbf{x}, \mathcal{C}_1)\, d\mathbf{x} \end{aligned}$$

Clearly to minimize $p\,(\text{mistake})$, we should arrange that each $\mathbf{x}$ is assigned to whichever class has the smaller value of the integrand

- Since $p\,(\mathbf{x}, \mathcal{C}_k) = p\,(\mathcal{C}_k|\mathbf{x})\,p\,(\mathbf{x})$, chose the class for which a posteriori probability $p\,(\mathcal{C}_k|\mathbf{x})$ is the largest.

For the more general case of K classes, it is slightly easier to maximize the probability of being correct:

$$\begin{aligned} p\,(\text{correct}) &= \sum_{k=1}^{K} p\,(\mathbf{x} \in \mathcal{R}_k, \mathcal{C}_k) \\ &= \sum_{k=1}^{K} \int_{\mathcal{R}_k} p\,(\mathbf{x}, \mathcal{C}_k)\, d\mathbf{x} \end{aligned}$$

- Each $\mathbf{x}$ should be assigned to the class for which $p\,(\mathcal{C}_k|\mathbf{x})$ is the largest.

Minimizing Expected Loss

- Unequal importance of mistakes
  - e.g., medical diagnosis: cancer→normal vs. normal→cancer
- Loss/cost function
  - a single, overall measure of loss incurred in taking any of the available decisions or actions
  - given by a loss matrix $\mathbf{L} = [L_{kj}]$
  - e.g., medical diagnosis

$$\begin{array}{c} \\ \text{cancer} \\ \text{normal} \end{array} \begin{array}{c} \begin{array}{cc} \text{cancer} & \text{normal} \end{array} \\ \begin{pmatrix} 0 & 1000 \\ 1 & 0 \end{pmatrix} \end{array}$$

- The optimal solution is the one that minimizes the loss function. The loss function depends on the true class, which is unknown.
- Instead, we minimize the average loss by utilizing the joint probability distribution $p(\mathbf{x}, \mathcal{C}_k)$.

$$\mathbb{E}[L] = \sum_k \sum_j \int_{\mathcal{R}_j} L_{kj} p(\mathbf{x}, \mathcal{C}_k) \, d\mathbf{x}$$

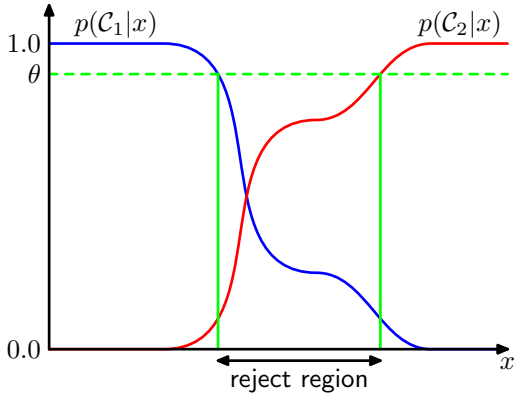
- Minimum loss decision rule
  - Choose class for which $\sum_k L_{kj} p(\mathcal{C}_k, \mathbf{x})$ (or equivalently $\sum_k L_{kj} p(\mathcal{C}_k|\mathbf{x})$) is minimum
  - Assign each new $\mathbf{x}$ to the class j for which the following quantity is minimum:

$$\sum_k L_{kj} p(\mathcal{C}_k|\mathbf{x})$$

  - Trivial once we know the posterior class probabilities $p(\mathcal{C}_k|\mathbf{x})$

Rejection

- Decisions can be made when a posteriori probabilities are significantly less than the unity or joint probabilities have comparable values
- In some applications, it will be appropriate to avoid making decisions on difficult cases
  - e.g., medical diagnosis: for a computer-aided diagnosis system, when there is little doubt as to the correct class, leave it for a human expert to classify the more ambiguous cases



Inference and Decision

- Classification problem broken into two separate stages
  1. Inference: learn a model of $p(\mathcal{C}_k|\mathbf{x})$ from training data
  2. Decision: use posterior probabilities to make optimal class assignments

- Alternative: learn a function that maps inputs directly into labels (or decisions) (called **'discriminant function'**)

[Three Distinct Approaches to Decision Problems]

Generative

Discriminative

Discriminant function

Generative Models

- First solve the inference problem of determining the class-conditional densities $p(\mathbf{x}|\mathcal{C}_k)$ for each class $\mathcal{C}_k$
- Also separately infer the prior class probability $p(\mathcal{C}_k)$
- Then use Bayes' theorem to find posterior class probabilities $p(\mathcal{C}_k|\mathbf{x})$
- Apply the decision theory to determine class membership

Approaches that explicitly or implicitly model the distribution of inputs as well as outputs are known as *generative models*, because by sampling from them it is possible to generate synthetic data points in the input space.

Discriminative Models

- First solve the inference problem to determine posterior probabilities $p(\mathcal{C}_k|\mathbf{x})$
- Apply the decision theory to determine class membership

Approaches that model the posterior probabilities directly are called *discriminative models*.

Discriminant Functions

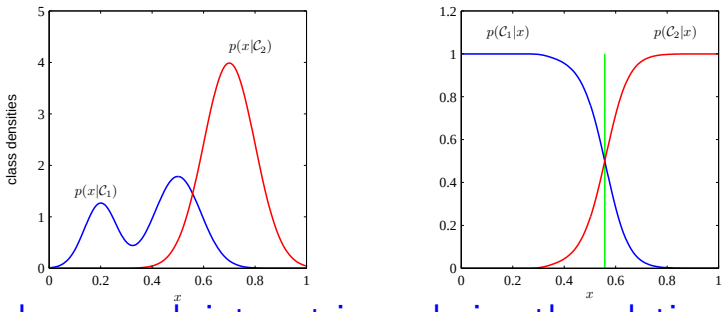
- Find a function $f(\mathbf{x})$ that maps each input $\mathbf{x}$ directly onto a class label
  - In binary classification, $f(\cdot)$ might be binary valued such that 0 and 1 represent $\mathcal{C}_1$ and $\mathcal{C}_2$, respectively.
- Probabilities play no role.
  - No access to posterior probabilities $p(\mathcal{C}_k|\mathbf{x})$

[Comparison of Three Distinct Approaches]

- Generative
  - (B) The most demanding due to involvement of finding the joint distribution over both $\mathbf{x}$ and $\mathcal{C}_k$
  - (B) High-dimension: a large training set is required for reasonable accuracy of class-conditional densities
  - (B) Class prior $p(\mathcal{C}_k)$: often estimated simply from the fractions of the training set data points in each of the classes
  - (G) Marginal density of data $p(\mathbf{x})$
    - Useful for detecting new data points that have low probability under the model (e.g., outlier/novelty detection)
- Discriminative
- Discriminant function

[Comparison of Three Distinct Approaches]

- Generative
- Discriminative
  - As for classification, finding $p(\mathbf{x}, \mathcal{C}_k)$ is wasteful of computational resources and excessively demanding of data
  - Indeed, the class-conditional densities may contain a lot of structure that has little effect on the posterior probabilities



There has been much interest in exploring the relative merits of generative and discriminative approaches to PRML, and in finding ways to combine them

(Jebra, 2004; Lasserre et al., 2006)

- Discriminant function

[Comparison of Three Distinct Approaches]

- Generative
- Discriminative
- Discriminant function
  - Mapping each $\mathbf{x}$ directly into a class label
  - Combining the inference and decision stages into a single learning problem
  - No longer have access to posterior probabilities $p(\mathcal{C}_k|\mathbf{x})$

[Need for Posterior Probabilities]

- Minimizing risk
  - Periodic changes in a loss matrix (e.g., financial application)
  - In a discriminant function, any change to the loss matrix would require that we return to the training data and solve the classification problem fares
- Reject option
  - Minimize the misclassification rate, or more generally the expected loss, for a given fraction of rejected data points
- Compensating for class priors
  - When far more samples from one class compared to another, we use a balanced data set (otherwise we may have 99.9% accuracy always classifying into one class)
  - Take posterior probabilities from balanced data set, divide by class fractions in the data set and multiply by class fractions in population to which the model is applied

- Combining models
  - Medical diagnosis: X-ray images ($\mathbf{x}_I$) and blood test ($\mathbf{x}_B$)
  - Combining all of the heterogeneous information into one huge input space (?)
  - When posterior probabilities are available they can be combined systematically using rules of probability
  - Assuming conditional independence
    $p(\mathbf{x}_I, \mathbf{x}_B|\mathcal{C}_k) = p(\mathbf{x}_I|\mathcal{C}_k)\, p(\mathbf{x}_B|\mathcal{C}_k)$

called 'Naïve Bayes model'

$$\begin{aligned} p(\mathcal{C}_k|\mathbf{x}_I, \mathbf{x}_B) &\propto p(\mathbf{x}_I, \mathbf{x}_B|\mathcal{C}_k)\, p(\mathcal{C}_k) \\ &\propto p(\mathbf{x}_I|\mathcal{C}_k)\, p(\mathbf{x}_B|\mathcal{C}_k)\, p(\mathcal{C}_k) \\ &\propto \frac{p(\mathcal{C}_k|\mathbf{x}_I)\, p(\mathcal{C}_k|\mathbf{x}_B)}{p(\mathcal{C}_k)} \end{aligned}$$

  - Need $p(\mathcal{C}_k)$, estimated from the fractions of data points in each class, and then normalize such that the resulting posterior probabilities sum to one.

Loss Functions for Regression

- Decision stage
  - Choosing a specific estimate $y(\mathbf{x})$ of the value of t for each input $\mathbf{x}$
- Loss: $L(t, y(\mathbf{x}))$
- Average or expected loss

$$\mathbb{E}[L] = \int\int L(t, y(\mathbf{x}))\, p(\mathbf{x}, t)\, d\mathbf{x} dt$$

- Squared loss: a common choice of loss function

$$\mathbb{E}[L] = \int\int \{y(\mathbf{x}) - t\}^2\, p(\mathbf{x}, t)\, d\mathbf{x} dt$$

$$\mathbb{E}[L] = \int \int \{y(\mathbf{x}) - t\}^2 p(\mathbf{x}, t)\, d\mathbf{x} dt$$

- Goal: to choose $y(\mathbf{x})$ so as to minimize $\mathbb{E}[L]$
- Assuming a completely flexible function $y(\mathbf{x})$, we can find $y(\mathbf{x})$ so as to minimize $\mathbb{E}[L]$ using the calculus of variations

$$\frac{\delta \mathbb{E}[L]}{\delta y(\mathbf{x})} = 2 \int \{y(\mathbf{x}) - t\} p(\mathbf{x}, t)\, dt = 0$$

- Solving for $y(\mathbf{x})$

$$y(\mathbf{x}) = \frac{\int t p(\mathbf{x}, t)\, dt}{p(\mathbf{x})} = \int t p(t|\mathbf{x})\, dt = \mathbb{E}_t[t|\mathbf{x}]$$

  - conditional average t conditioned on $\mathbf{x}$, known as '*regression function*'

Deriving in a slightly different way

- Armed with the knowledge that the optimal solution is the conditional expectation

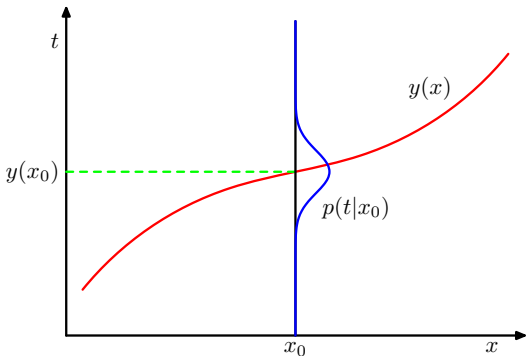
$$\begin{aligned} \{y(\mathbf{x}) - t\}^2 &= \{y(\mathbf{x}) - \mathbb{E}_t[t|\mathbf{x}] + \mathbb{E}_t[t|\mathbf{x}] - t\}^2 \\ &= \{y(\mathbf{x}) - \mathbb{E}_t[t|\mathbf{x}]\}^2 + \{\mathbb{E}_t[t|\mathbf{x}] - t\}^2 \\ &\quad 2\{y(\mathbf{x}) - \mathbb{E}_t[t|\mathbf{x}]\}\{\mathbb{E}_t[t|\mathbf{x}] - t\} \end{aligned}$$

- Substituting into the loss function and performing the integral over t

$$\mathbb{E}[L] = \int \{y(\mathbf{x}) - \mathbb{E}_t[t|\mathbf{x}]\}^2 p(\mathbf{x}) d\mathbf{x} + \int \{\mathbb{E}_t[t|\mathbf{x}] - t\}^2 p(\mathbf{x}, t) d\mathbf{x} dt$$

  - $y(\mathbf{x}) = \mathbb{E}_t[t|\mathbf{x}]$ to minimize $\mathbb{E}[L]$
  - $\int \{\mathbb{E}_t[t|\mathbf{x}] - t\}^2 p(\mathbf{x}, t) d\mathbf{x} dt$: variance of distribution of t, averaged over $\mathbf{x}$
    - intrinsic variability of the target data, i.e., noise
    - independent of $y(\mathbf{x})$, irreducible minimum value of the loss function

The regression function $y(x)$, which minimizes the expected squared loss, is given by the mean of the conditional distribution $p(t|x)$.



- Easily extended to multiple target variables represented by the vector $\mathbf{t}$

$$\mathbf{y}(\mathbf{x}) = \mathbb{E}_{\mathbf{t}}[\mathbf{t}|\mathbf{x}]$$

[Three Distinct Approaches to Regression Problems]

- Generative
  - Determine the joint density $p(\mathbf{x}, t)$
  - Normalize to find conditional density $p(t|\mathbf{x})$
  - Marginalize to find the conditional mean $\mathbb{E}_t[t|\mathbf{x}]$
- Discriminative
  - Determine the conditional density $p(t|\mathbf{x})$
  - Marginalize to find the conditional mean $\mathbb{E}_t[t|\mathbf{x}]$
- Discriminant function
  - Find a regression function $y(\mathbf{x})$ directly from the training data

The relative merits of these three approaches follow the same lines as for classification problems.

Loss Functions

- Squared loss: very poor results when $p(t|\mathbf{x})$ is multimodal
- Need to develop more sophisticated approaches
- Minkowski loss, whose expectation is given by

$$\mathbb{E}[L_q] = \int \int |y(\mathbf{x}) - t|^q \, p(\mathbf{x}, t) \, d\mathbf{x} dt$$

  - Conditional mean for $q = 2$
  - Conditional median for $q = 1$
  - Conditional mode for $q \to 0$