PRML 3. Linear models for Regression

Regression: Given $\{(\mathbf{x}_n, t_n)\}_{n=1 \text{ to } N}$ ($\mathbf{x}$: D-dim.), estimate t for new $\mathbf{x}$.

$\cdots t = y(\mathbf{x})$ or $p(t|\mathbf{x})$

- Linear regression ··· too simple

$$y(\mathbf{x}, \mathbf{w}) = w_0 + x_1 w_1 + \cdots + x_D w_D$$

- Extended version (bias; basis function $\mathbb{R}^D \to \mathbb{R}$)

$$y(\mathbf{x}, \mathbf{w}) = w_0 + \sum_{i=1}^{M-1} w_i \phi_i(\mathbf{x})$$

$$\equiv \sum_{i=0}^{M-1} w_i \phi_i(\mathbf{x})$$

$$= \mathbf{w}^T \boldsymbol{\phi}$$

w. $\mathbf{x} \in \mathbb{R}^D$, $\mathbf{w} \in \mathbb{R}^M$, $\phi_0(\mathbf{x}) = 1$.

linear in $\mathbf{w}$

Polynomial $\phi_i(x) = x^i$ ··· global : $\mathbf{w}$ has long-range interaction

Gaussian $\phi_i(x) = \exp\left[-\frac{(x-\mu_i)^2}{2s^2}\right]$ (s: common)

logistic sigmoid $\phi_i(x) = \sigma\left(\frac{x-\mu_i}{s}\right)$ [or $\tanh \frac{x-\mu_i}{s}$] (s: common)

$$\sigma(a) = \frac{1}{1+e^{-a}}$$

$[\tanh a = 2\sigma(2a) - 1]$

Fourier

⋮

Let's assume: t is given by $t = y(x, w) + \varepsilon$ ← Gaussian noise

i.e. $p(t|x, w, \beta) = \mathcal{N}(t|y(x, w), \beta^{-1})$ ($\beta = 1/\sigma$ (precision))

$\Rightarrow$ • Unimodality is assumed.

• $\mathbb{E}_t[t|x] = y(x, w)$

Review: §1.5

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

for $L = [t - y(x)]^2$,

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

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

With data $(\mathbf{t}, \mathbb{X}) = (t_1, x_1), \ldots, (t_N, x_N)$,

likelihood

$\ln p(\mathbf{t}|\mathbb{X}, w, \beta) = \sum_{i=1}^{N} \ln \mathcal{N}(t_i | y(x_i, w), \beta^{-1})$

hereafter, omitted for simplicity

$= \frac{N}{2} \ln \frac{\beta}{2\pi} - \beta \left[\frac{1}{2} \sum \left(t_n - w^T \phi(x_n) \right)^2 \right]$

$=: E_D(w)$ sum-of-square error function

maximizing likelihood w.r.t. w,

$\nabla_w \ln p(\mathbf{t}|w, \beta) = \beta \sum \left(t_n - w^T \phi(x_n) \right) \phi(x_n) = 0$

$\therefore \sum_n t_n \phi_i(x_n) = \sum_n \sum_k w_k \phi_k(x_n) \phi_i(x_n)$

$\therefore w_{ML} = \Phi^\dagger \mathbf{t}$ where $\Phi_{ij} = \phi_j(x_i)$ design matrix ($N \times M$)

$A^\dagger := (A^T A)^{-1} A^T$ Moore-Penrose pseudo-inverse

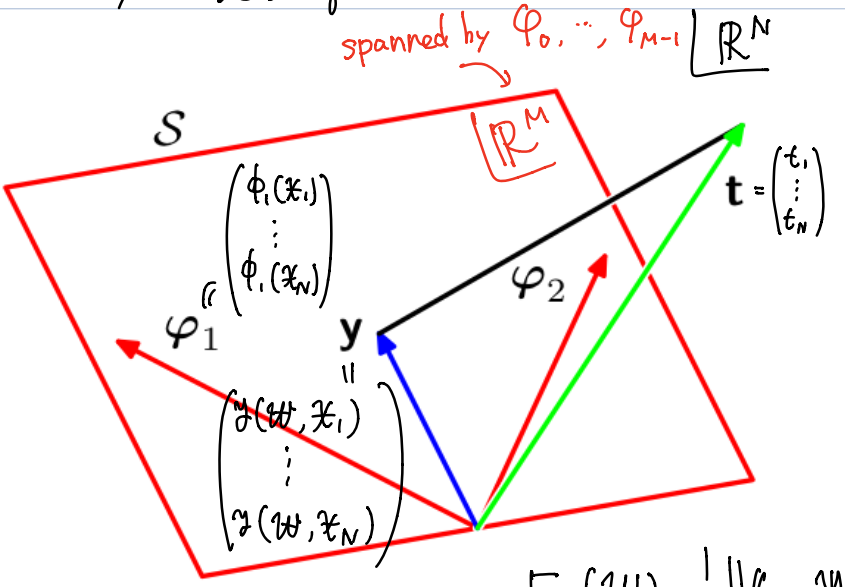
especially $w_0^{ML} = \overline{t} - \sum_{m=1}^{M-1} w_m \overline{\phi_m}$

avg. of t_n ; avg. of $\phi_m(x_i)$

As w_{ML} is indep. of β, we can calculate (w_{ML}, β_{ML}) simply by $\frac{\partial}{\partial \beta} \mathcal{L} = 0$:

$\beta_{ML}^{-1} = \frac{1}{N} \sum \left[t_i - w_{ML}^T \phi(x_i) \right]^2$.

§3.1.2 Geometry of least squares



$$E_D(\mathbf{w}) = \frac{1}{2}\|\mathbf{t} - \mathbf{y}\|^2$$

For $M < N$, $\{\varphi_0, \cdots, \varphi_{M-1}\}$ spans a vector space $\mathcal{S} \subseteq \mathbb{R}^M \subsetneq \mathbb{R}^N$

where $\mathbf{y} \in \mathcal{S}$, $\mathbf{t} \in \mathbb{R}^N$. so, $\mathbf{y}_{ML}$ is the projection of $\mathbf{t}$ onto $\mathcal{S}$.

this is $\boldsymbol{\Phi}^\dagger \mathbf{t}_{ML}$

PROOF

$$\begin{aligned} &\varphi_i \cdot (\mathbf{t} - \boldsymbol{\Phi}\mathbf{w}_{ML}) \\ &= \varphi_i \cdot (\mathbf{t} - \boldsymbol{\Phi}(\boldsymbol{\Phi}^T\boldsymbol{\Phi})^{-1}\boldsymbol{\Phi}^T\mathbf{t}) \\ &= [\boldsymbol{\Phi}^T(\mathbf{t} - \boldsymbol{\Phi}(\boldsymbol{\Phi}^T\boldsymbol{\Phi})^{-1}\boldsymbol{\Phi}^T\mathbf{t}]_i = 0_i \end{aligned}$$

$\therefore \mathbf{y}_{ML} = \boldsymbol{\Phi}^\dagger \mathbf{t}_{ML}$.

§3.1.3 Sequential learning

"stochastic gradient descent"

if $E(\mathbf{w})$ is given by $\sum_n E_n(\mathbf{w})$,

$$\mathbf{w}^{(\tau+1)} = \mathbf{w}^{(\tau)} - \eta \nabla_{\mathbf{w}} E_n\big|_{\mathbf{w}=\mathbf{w}^{(\tau)}}$$ minimizes $E(\mathbf{w})$.

↖ chosen so that $\mathbf{w}^{(\tau)}$ converges.

(for sum-of-square,

$$\mathbf{w}^{(\tau+1)} = \mathbf{w}^{(\tau)} + \eta(t_n - \mathbf{w}^{(\tau)T}\boldsymbol{\phi}_n)\boldsymbol{\phi}_n$$

... least-mean-square algorithm.

§3.1.4 Regularized least squares

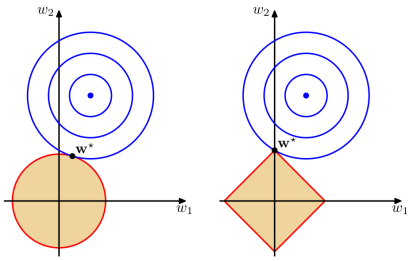
"weight decay" $E_W(\mathbf{w}) = \frac{1}{2}\mathbf{w}^T\mathbf{w}$.

e.g. $E(\mathbf{w}) = \frac{1}{2}\sum_{n=1}^{N}\left[t_n - \mathbf{w}^T\boldsymbol{\phi}(\mathbf{x}_n)\right]^2 + \frac{\lambda}{2}\mathbf{w}^T\mathbf{w}$

$\Rightarrow \mathbf{w} = \left(\lambda\mathbf{I} + \boldsymbol{\Phi}^T\boldsymbol{\Phi}\right)^{-1}\boldsymbol{\Phi}^T\mathbf{t}$.

other regularizers: $E_W(\mathbf{w}) = \frac{\lambda}{2}\sum_{m=0}^{M-1}|w_m|^q$

(or 1)



- Regularizers w. $q \le 1$
  drive some parameters to zero
  "sparse models"

§3.1.5 Multiple outputs ... $t \to \mathbf{t}$.

- introduce extra basis functions for respective t_k.

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

↑ common for k

OR

- extend parameters $\mathbf{w}$ to $\mathbf{W}$.

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

likelihood $p(\mathbf{T} \mid \mathbf{X}, \mathbf{W}, \beta)$

$$p(\mathbf{t} \mid \mathbf{x}, \mathbf{W}, \beta) = \mathcal{N}(\mathbf{t} \mid \mathbf{W}^T\boldsymbol{\phi}(\mathbf{x}), \beta^{-1}\mathbf{I})$$

returns $\mathbf{W}_{ML} = \left(\boldsymbol{\Phi}^T\boldsymbol{\Phi}\right)^{-1}\boldsymbol{\Phi}^T\mathbf{T}$ as before.

§3.2 Bias-Variance trade off

Review §1.5

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

$L = [t - y(x)]^2$ leads to $\tilde{y}(x) = \mathbb{E}_t[t|x]$ ← optimal solution y

and $\mathbb{E}[L] = \underbrace{\int dx\, [y(x) - \tilde{y}(x)]^2 p(x)}_{①} + \underbrace{\int dx\, dt\, [t - \tilde{y}(x)]^2 p(x,t)}_{②}$

Here we choose squared loss function.

[$\neq$ sum-of-square error function (consequence of assuming a Gaussian noise)]

② remains even if we obtained the best solution $y = \tilde{y}$.

Of course, based on limited data $\mathcal{D}$,

our estimated $y = y(x; \mathcal{D}) \neq h(x)$.

Note: $\mathbb{E}_{\mathcal{D}}[(y(x;\mathcal{D}) - h(x))^2]$ depends on learning algorithms.

$$\Rightarrow \mathbb{E}_{\mathcal{D}}[①] = \mathbb{E}_{\mathcal{D}}\left[\{y(x;\mathcal{D}) - \tilde{y}(x)\}^2\right]$$

$$= \underbrace{\{\tilde{y}(x) - \mathbb{E}_{\mathcal{D}}[y(x;\mathcal{D})]\}^2}_{\text{"bias"}} + \underbrace{\mathbb{E}_{\mathcal{D}}\left[\{y(x;\mathcal{D}) - \mathbb{E}_{\mathcal{D}}[y(x;\mathcal{D})]\}^2\right]}_{\text{"variance"}}$$

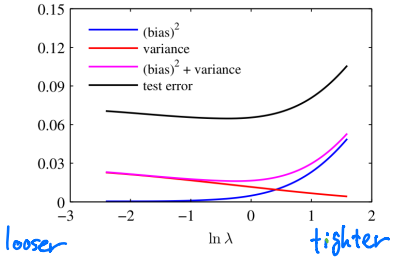
of our learning algorithm

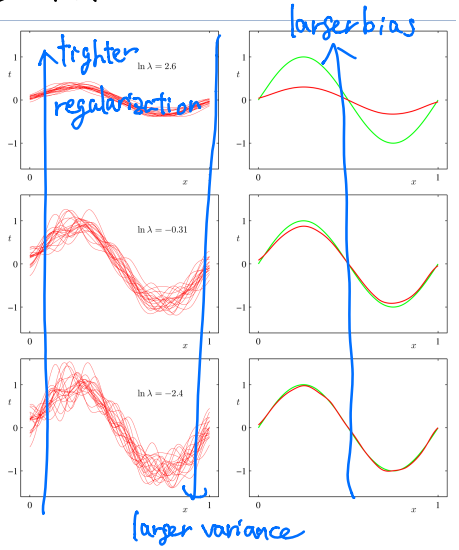
$\therefore$ Expected loss $\mathbb{E}_{\mathcal{D},x}[L] = (\text{bias})^2 + \text{variance} + \text{noise}$

$$(\text{bias})^2 = \int \{\mathbb{E}_{\mathcal{D}}[y(x;\mathcal{D})] - \tilde{y}(x)\}^2 p(x)\, dx$$

$$\text{variance} = \int \mathbb{E}_{\mathcal{D}}\left[\{y(x;\mathcal{D}) - \mathbb{E}_{\mathcal{D}}[y(x;\mathcal{D})]\}^2\right] p(x)\, dx$$

$$\text{noise} = \int [\tilde{y}(x) - t]^2 p(x,t)\, dx\, dt.$$





§3.3 Bayesian linear regression

- ML method may lead to overfitting
  - limit #basis (M) ··· less flexibility.
  - use regulators ··· which value of λ ?
  - use Bayesian method ··· needs proper prior

We use Gaussian likelihood ··· conjugate prior is Gaussian.

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

posterior: $p(\mathbf{w} \mid \mathbf{t}) = \mathcal{N}(\mathbf{w} \mid \mathbf{m}_N, \mathbf{S}_N)$

$$\begin{cases} \mathbf{m}_N = \mathbf{S}_N\left(\mathbf{S}_0^{-1}\mathbf{m}_0 + \beta \boldsymbol{\Phi}^T \mathbf{t}\right) \\ \mathbf{S}_N^{-1} = \mathbf{S}_0^{-1} + \beta \boldsymbol{\Phi}^T \boldsymbol{\Phi} \end{cases}$$

- non-informative prior $\mathbf{S}_0 \to \infty \mathbf{I}$ gives $\mathbf{m}_N \to \boldsymbol{\Phi}^\dagger \mathbf{t}$, $\mathbf{S}_N \to \beta \boldsymbol{\Phi}^T \boldsymbol{\Phi}$

A simple prior $p(\mathbf{w}|\alpha) = \mathcal{N}(\mathbf{w} \mid 0, \alpha^{-1}\mathbf{I}) \propto \exp\left[-\frac{\alpha}{2}\mathbf{w}^T\mathbf{w}\right]$

...equivalent to least-square error + squared regulator

··· $\mathbf{m}_N = \beta \mathbf{S}_N \boldsymbol{\Phi}^T \mathbf{t}$,
$\mathbf{S}_N = \alpha \mathbf{I} + \beta \boldsymbol{\Phi}^T \boldsymbol{\Phi}$,

$$\ln p(\mathbf{w}|\mathbf{t}) = -\frac{\beta}{2}\sum_{n=1}^{N}\left[t_n - \mathbf{w}^T\boldsymbol{\phi}(\mathbf{x}_n)\right]^2 - \frac{\alpha}{2}\mathbf{w}^T\mathbf{w} + \text{const.}$$

Another prior

$$p(\mathbf{w}|\alpha) = \left[\frac{q}{2\Gamma(1/q)}\left(\frac{\alpha}{2}\right)^{1/q}\right]^M \exp\left(-\frac{\alpha}{2}\sum_{m=0}^{M-1}|w_m|^q\right)$$

$$\Rightarrow \ln p(\mathbf{w}|\mathbf{t}) = -\frac{\beta}{2}\sum\left[t_n - \mathbf{w}^T\boldsymbol{\phi}(\mathbf{x}_n)\right]^2 - \frac{\alpha}{2}\sum_{m=0}^{M-1}|w_m|^q$$

· Remember that our goal is to predict t' from x'.

$$p(t|x, \mathbf{t}, \mathbf{X}, \alpha, \beta) = \int \underbrace{p(t|x, w, \beta)}_{①} \underbrace{p(w|\mathbf{t}, \mathbf{X}, \alpha, \beta)}_{②} dw \quad \cdots (1)$$

(noise: assumed known; prior)

$$① = \mathcal{N}(t|y(x,w), \beta^{-1}) \qquad ② = \mathcal{N}(w|m_N, S_N)$$

(2.115) $A \equiv \phi^T$, $b = 0$

$$= \mathcal{N}(t|m_N^T \phi(x), \beta^{-1} + \phi^T(x) S_N \phi(x))$$

β^{-1}: noise from ①; $\phi^T(x) S_N \phi(x)$: uncertainty of w; $\to 0$ for infinite data

WARNING | If we use localized basis functions, (e.g. gaussian)

regions away from the locations will have too small $\phi^T S_N \phi$, and the model becomes too confident on its prediction.

This can be patched by "Gaussian process".
= constructing the equivalent kernel directly w/o using basis functions.

§3.3.3 Equivalent kernel

$$y(x, w)|_{w=m_N} = m_N^T \phi(x)$$ ← mean of the predictive distribution $p(t|x, \mathbf{t}, \mathbf{X}, \alpha, \beta)$

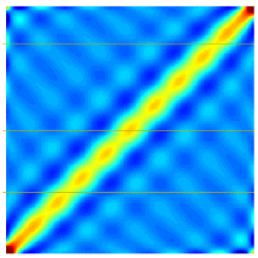
$$= \beta \phi(x) S_N \Phi^T \mathbf{t} = \sum_{n=1}^{N} \underbrace{\left[\beta \phi(x)^T S_N \phi(x_n)\right]}_{=: k(x, x_n)} t_n$$

"smoother matrix" "equivalent kernel"

i.e. the mean of output t for an input x is given by

$$\sum_{n=1}^{N} k(x, x_n) t_n$$

$k(x, x_n)$: modelled by $\phi_i(x_n)$, etc...; t_n: learned input/output



Also:

$$\operatorname{Cov}_{w}[y(x), y(x')] = \mathbb{E}\left[\left(w^T\phi - m_N^T\phi\right)\left(w^T\phi' - m_N^T\phi'\right)\right]$$

$$= \phi(x)^T \mathbb{E}[ww^T]\phi(x') + \phi(x)^T m_N m_N^T \phi(x')$$

$$- \phi(x)^T\left(\underbrace{\mathbb{E}[w]^T}_{=m_N} m_N + m_N^T \mathbb{E}[w]\right)\phi(x')$$

$p(w) = \mathcal{N}(w \mid m_N, S_N)$

$$= \phi(x)^T\left(\mathbb{E}[ww^T] - m_N m_N^T\right)\phi(x')$$

$$= \phi(x)^T \mathbb{V}[w]\,\phi(x')$$

$$= \phi(x)^T S_N \phi(x')$$

$$= \beta^{-1} k(x, x')$$

"predictive means of nearby / distant points are highly / less correlated"

- $\sum_{n=1}^{N} k(x, x_n) = 1$ (?) PROOF? or do we need $\begin{cases} \alpha = 0 \\ \phi_i(x) : \text{L. indep.} \\ N \geq M \end{cases}$ etc.?
- $k(x, x') = \psi^T(x)\,\psi(x')$ with $\psi(x) = \sqrt{\beta}\, S_N^{1/2}\phi(x)$.

§ 3.4 Bayesian model comparison

Suppose we have L models $\{\mathcal{M}_i\}$ and data $\mathcal{D}$ is generated from one of them.

"a model refers to a prob. dist. over the observed data"

… a model $\mathcal{M}_i$ generates a data set $\mathcal{D}$ with a probability $\int dw\, p(\mathcal{D}|\mathcal{M}_i, w)\, p(w|\mathcal{M}_i)$

prior probability distribution $p(\mathcal{M}_i)$

↓

posterior $p(\mathcal{M}_i|\mathcal{D}) \propto p(\mathcal{D}|\mathcal{M}_i)\, p(\mathcal{M}_i)$

which model is more likely?

"model evidence" "marginal likelihood"

$$\cdots \int \underbrace{p(\mathcal{D}|\mathcal{M}_i, \text{params})}_{\text{(usual) likelihood}} p(\text{params})\, d(\text{params})$$

* "Bayes factor" $\dfrac{p(\mathcal{D}|\mathcal{M}_i)}{p(\mathcal{D}|\mathcal{M}_i)}$

↓

predictive distribution

$$p(t|x, \mathcal{D}) = \sum_i p(t|x, \mathcal{M}_i, \mathcal{D})\, p(\mathcal{M}_i|\mathcal{D})$$ "model mixture"

or we can choose a model by hand: $p(t|x, \mathcal{D}) := p(t|x, \tilde{\mathcal{M}}, \mathcal{D})$ "model selection"

Here,

$$\mathbb{E}_{\mathcal{D}}\left[\ln \frac{p(\mathcal{D}|\mathcal{M}_{\text{true}})}{p(\mathcal{D}|\mathcal{M}_i)}\right] = \int p(\mathcal{D}|\mathcal{M}_{\text{true}}) \ln \frac{p(\mathcal{D}|\mathcal{M}_{\text{true}})}{p(\mathcal{D}|\mathcal{M}_i)}\, d\mathcal{D}$$

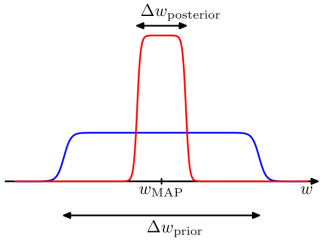
$$= KL(p_{\text{true}} \| p_i) > 0,$$

which proves the Bayes factor chooses the true model on average as far as $\{\mathcal{M}_i\}$ includes the true model $\mathcal{M}_{\text{true}}$.

• Interpretation of $p(\mathcal{D})$

Consider a model $\mathcal{M}$ that has one parameter w:

$$p(\mathcal{D}) = \int dw\ p(\mathcal{D}|\mathcal{M}, w)\, p(w).$$



if we assume the posterior $p(w|\mathcal{D})$ is centered at w_{MAP} and flat and the prior $p(w)$ is also flat,

$$p(\mathcal{D}) \approx \Delta w_{\text{post}}\ p(\mathcal{D}|\mathcal{M}, w_{MAP})\ \frac{1}{\Delta w_{\text{prior}}}$$

$$\therefore \ln p(\mathcal{D}) \approx \ln p(\mathcal{D}|\mathcal{M}, w_{MAP}) - \ln \frac{\Delta w_{\text{prior}}}{\Delta w_{\text{posterior}}}$$

fit result

$\ln \frac{\Delta w_{\text{prior}}}{\Delta w_{\text{posterior}}} > 0$: larger penalty for tuned w (smaller Δw_{post})

tuning penalty

Note that the penalty could be larger for larger-parameter models.

Here we can see the prior must be proper (and thus informative).
Therefore, Bayesian approach also needs some validation data
to validate the prior assumption
(instead of the model comparison.)

§4.4.1 Bayesian Information Criterion

Using Laplace approx. to $f = p(\mathcal{D}|\Theta)p(\Theta)$,

$$\ln p(\mathcal{D}) \approx \ln p(\mathcal{D}|\Theta_{MAP}) + \underbrace{\ln p(\Theta_{MAP}) - \frac{1}{2}\ln|A| + \frac{M}{2}\ln 2\pi}_{\text{occam factor}}$$

with Θ_{MAP} is the mode and

$$A = -\nabla\nabla \ln p(\mathcal{D}|\Theta)p(\Theta)\big|_{MAP} = -\nabla\nabla p(\Theta_{MAP}|\mathcal{D}).$$

If prior is broad and A is full rank,

$$\ln p(\mathcal{D}) \approx \ln p(\mathcal{D}|\Theta_{MAP}) - \frac{1}{2} M \ln N \quad : \text{BIC}.$$

… BIC gives bad result when A is not full rank !!

§3.5 The Evidence Approximation

(empirical Bayes / type-2 maximal likelihood / generalized maximal likelihood)

Approximation for marginalizing α and β (hyperparameters) as well as $\mathbf{w}$ (parameters).

$$p(\mathbf{t}|\mathbf{x},\mathbf{t},\mathbf{X}) = \int d\mathbf{w}\, d\alpha\, d\beta \; \underbrace{p(\mathbf{t}|\mathbf{w},\beta)}_{\mathcal{N}(\mathbf{t}|y(\mathbf{w},\mathbf{x}),\beta^{-1})} \underbrace{p(\mathbf{w}|\mathbf{t},\alpha,\beta)}_{\mathcal{N}(\mathbf{w}|\mathbf{m}_N,\mathbf{S}_N)} p(\alpha,\beta|\mathbf{t})$$

intrinsic noise; parameters from hyperparameter; hyperparameter estimation

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

likelihood (Evidence function)

- Analitic method

$$p(\alpha,\beta) \equiv \mathrm{Gam}(\alpha|a,b)\,\mathrm{Gam}(\beta|a',b')$$

$$\Rightarrow p(\mathbf{t}|\mathbf{t}) = \int d\mathbf{w}\, d\alpha\, d\beta \; \mathcal{N}(\mathbf{t}|y,\beta^{-1})\,\mathcal{N}(\mathbf{w}|\mathbf{m}_N,\mathbf{S}_N)$$

($\mathbf{m}_N = \beta \mathbf{S}_N \boldsymbol{\Phi}^T \mathbf{t}$, $\mathbf{S}_N = [\alpha \mathbf{I} + \beta \boldsymbol{\Phi}^T \boldsymbol{\Phi}]^{-1}$)

(?)

$$\times \frac{p(\mathbf{t}|\alpha,\beta)}{p(\mathbf{t})}\,\mathrm{Gam}(\alpha|a,b)\,\mathrm{Gam}(\beta|a',b')$$

$p(\mathbf{t})$: irrelevant constant

I guess we need these approximations...

$$\approx \int d\mathbf{w}\, d\alpha\, d\beta \; \mathcal{N}(\mathbf{t}|y,\beta^{-1})\, \frac{p(\mathbf{t}|\hat{\alpha},\hat{\beta})}{p(\mathbf{t})} \times \mathcal{N}(\mathbf{w}|\hat{\beta}\mathbf{S}_N \boldsymbol{\Phi}^T \mathbf{t}, [\alpha \mathbf{I} + \beta \boldsymbol{\Phi}^T \boldsymbol{\Phi}]^{-1})\,\mathrm{Gam}(\alpha)\,\mathrm{Gam}(\beta)$$

omit

Student's t ; Student's t

⊕ "Laplace approximation" for $\int d\mathbf{w}$

(but this approx is known as poorer than the following ones.)

- maximize $p(\mathbf{t}|\alpha,\beta)$ [instead of $p(\alpha,\beta|\mathbf{t})$] ← assuming $p(\alpha,\beta)$ is less informative. → §3.5.1
- expectation-maximization algorithm → §9.3.4

§3.5.1 - 2

$$p(\alpha,\beta\mid \mathbf{t}) \propto p(\mathbf{t}\mid\alpha,\beta)\,p(\alpha,\beta)$$

$$p(\mathbf{t}\mid\alpha,\beta) = \int d\mathbf{w}\; p(\mathbf{t}\mid\mathbf{w},\beta)\,p(\mathbf{w}\mid\alpha)$$

$$= \int d\mathbf{w}\left[\prod_n \mathcal{N}(t_n\mid \mathbf{w}^T\boldsymbol{\phi}(\mathbf{x}_n),\beta^{-1})\right]\mathcal{N}(\mathbf{w}\mid \mathbf{0},\alpha^{-1}\mathbf{I})$$

Gaussian noise assumption — simple prior assumption

$$= \left(\frac{\beta}{2\pi}\right)^{N/2}\left(\frac{\alpha}{2\pi}\right)^{M/2}\int \exp\left[-E(\mathbf{w})\right]d\mathbf{w}$$

Here

$$E(\mathbf{w}) = \frac{\beta}{2}\|\mathbf{t}-\boldsymbol{\Phi}\mathbf{w}\|^2 + \frac{\alpha}{2}\mathbf{w}^T\mathbf{w}$$

$$= E(\mathbf{m}_N) + \frac{1}{2}(\mathbf{w}-\mathbf{m}_N)^T\mathbf{A}(\mathbf{w}-\mathbf{m}_N)$$

∴ showing that $\mathbf{w}$ tends to be $\mathbf{m}_N$ with an uncertainty $\mathbf{S}_N$.

… $E(\mathbf{m}_N) = \frac{\beta}{2}\|\mathbf{t}\|^2 - \frac{1}{2}\mathbf{m}_N^T\mathbf{A}\mathbf{m}_N$

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

$\mathbf{m}_N = \beta\mathbf{A}^{-1}\boldsymbol{\Phi}^T\mathbf{t} = \mathbf{m}_N$

} of $p(\mathbf{w}\mid\mathbf{t})$

[There α and β were "known" (assumed) parameters]

$A_{ij} = \frac{\partial}{\partial w_i}\frac{\partial}{\partial w_j}E(\mathbf{w})$ (Hessian of E)

$$\therefore \int \exp[-E(\mathbf{w})]d\mathbf{w} = e^{-E(\mathbf{m}_N)}(2\pi)^{M/2}|\mathbf{A}|^{-1/2} \quad \text{and}$$

$$\ln p(\mathbf{t}\mid\alpha,\beta) = \frac{N}{2}\ln\frac{\beta}{2\pi} + \frac{M}{2}\ln\alpha - E(\mathbf{m}_N) - \frac{1}{2}\ln|\mathbf{A}|.$$

We maximize $p(\mathbf{t}\mid\alpha,\beta)$ w.r.t. α and β.

With the eigenvalues $\{\lambda_i\}$ of $\beta\boldsymbol{\Phi}^T\boldsymbol{\Phi}$ and $\gamma = \sum_{i=1}^{M}\frac{\lambda_i}{\alpha+\lambda_i}$,

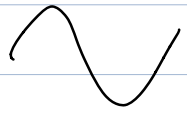
$$\frac{\partial p}{\partial \alpha} = 0 \iff \|\mathbf{m}_N\|^2 = \gamma/\alpha \qquad (\lambda_i > 0)$$

$$\frac{\partial p}{\partial \beta} = 0 \iff \sum_{n=1}^{N}\left(t_n - \mathbf{m}_N^T\boldsymbol{\phi}(\mathbf{x}_n)\right)^2 = \frac{N-\gamma}{\beta}$$

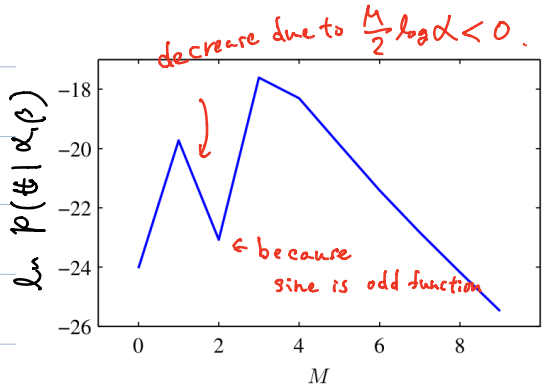
A is positive definite (semi) $\iff \forall \mathbf{x}\neq 0,\ (A\mathbf{x},\mathbf{x}) > 0\ (\geq) \iff$ all eigenvalues > 0 $(\geq)$.

∴ $\beta\boldsymbol{\Phi}^T\boldsymbol{\Phi}$ is positive semi definite.

Example:

to fit 

with M-th polynomial



Maximization

$\Phi^T\Phi = V \Lambda V^\dagger$ ($\Phi^T\Phi$ symmetric, V unitary; Λ: eigenvalues λ_i ($= \lambda_i/\beta$ in text))

$[A = V(\alpha I + \beta\Lambda)V^\dagger,\ A^{-1} = V(\alpha I + \beta\Lambda)^{-1}V^\dagger]$, $= Z$

$$E(m_N) = \frac{\beta}{2}\|\beta\Phi A^{-1}\Phi^T t - t\|^2 + \frac{\alpha\beta^2}{2}\|A^{-1}\Phi^T t\|^2$$

$$= t^T\left[\frac{\beta}{2}(\beta\Phi A^{-1}\Phi^T - I)(\beta\Phi A^{-1}\Phi^T - I) + \frac{\alpha\beta^2}{2}\Phi A^{-1}A^{-1}\Phi^T\right]t$$

$$[\ ] = \frac{\beta}{2}I + \frac{\beta^3}{2}\Phi V Z \Lambda Z V^\dagger \Phi^T - \beta^2 \Phi V Z V^\dagger \Phi^T + \frac{\alpha\beta^2}{2}\Phi V Z^2 \Phi^T$$

$$\to \frac{\beta^2}{2}\frac{\beta\lambda_i}{(\alpha+\beta\lambda_i)^2} - \frac{\beta^2}{\alpha+\beta\lambda_i} + \frac{\beta^2}{2}\frac{\alpha}{(\alpha+\beta\lambda_i)^2} = -\frac{\beta^2}{2}Z$$

$$= \frac{\beta}{2}I - \frac{\beta^2}{2}\Phi V(\alpha I + \beta\Lambda)^{-1}V^\dagger\Phi^T$$

$$\therefore \frac{\partial}{\partial\alpha}E(m_N) = \frac{\beta^2}{2}t^T\Phi V(\alpha I+\beta\Lambda)^{-2}V^\dagger\Phi^T t = \frac{1}{2}\|m_N\|^2$$

$$\frac{\partial}{\partial\beta}E(m_N) = \frac{1}{2}\|t\|^2 + \frac{\beta^2}{2}t^T\Phi V Z^{-2}\Lambda V^\dagger\Phi^T t - \beta t^T\Phi V Z V^\dagger\Phi^T$$

$$= \frac{1}{2}\|t\|^2 + \frac{1}{2}\|\Phi m_N\|^2 - t^T\Phi m_N$$

$$\ln|A| = \sum\ln(\alpha+\beta\lambda_i) \quad \therefore \frac{\partial\ln|A|}{\partial\alpha} = \sum\frac{1}{\alpha+\lambda_i},\ \frac{\partial\ln|A|}{\partial\beta} = \sum\frac{\lambda_i/\beta}{\alpha+\lambda_i}$$

$$\therefore \frac{\partial p}{\partial\alpha} = 0 \Leftrightarrow \alpha\|m_N\|^2 = M - \sum\frac{\alpha}{\alpha+\lambda_i} = \sum\frac{\lambda_i}{\alpha+\lambda_i}$$

$$\frac{\partial p}{\partial\beta} = 0 \Leftrightarrow \beta\sum_{n=1}^{N}\left(t_n - m_N^T\phi(x_n)\right)^2 = N - \sum\frac{\lambda_i}{\alpha+\lambda_i}$$

§ 3.5.3

maximum of the evidence function

$$p(\mathbf{t}|\alpha,\beta) = \int d\mathbf{w}\; p(\mathbf{t}|\mathbf{w},\beta)\, p(\mathbf{w}|\alpha)$$

$$= \left(\frac{\beta}{2\pi}\right)^{N/2} \left(\frac{\alpha}{2\pi}\right)^{M/2} \int \exp\left[-E(\mathbf{w})\right] d\mathbf{w}$$

$$\begin{cases} \alpha = \dfrac{\gamma}{\|\mathbf{m}_N\|^2} \\ \beta^{-1} = \dfrac{1}{N-\gamma} \displaystyle\sum_{n=1}^{N} \left(t_n - \mathbf{m}_N^T \boldsymbol{\phi}(\mathbf{x}_n)\right)^2 \end{cases}$$

where $\{\alpha+\lambda_i, \mathbf{u}_i\}$ is the eigensystem of the Hessian $\mathbf{A}$ of $E(\mathbf{w})$.

$$\gamma = \sum_{i=1}^{M} \frac{\lambda_i}{\alpha+\lambda_i} \qquad \left(0 < \frac{\lambda_i}{\alpha+\lambda_i} < 1\right)$$

effective number of well-determined parameters

$$\mathbf{w}_{ML} = \left(\boldsymbol{\Phi}^T\boldsymbol{\Phi}\right)^{-1}\boldsymbol{\Phi}^T\mathbf{t} \Rightarrow \beta_{ML}^{-1} = \frac{1}{N}\sum\left(t_n - \mathbf{w}_{ML}^T\boldsymbol{\phi}(\mathbf{x}_n)\right)^2$$

$$\mathbf{m}_N = \mathbf{w}_{MAP} = \left(\frac{\alpha}{\beta} + \boldsymbol{\Phi}^T\boldsymbol{\Phi}\right)^{-1}\boldsymbol{\Phi}^T\mathbf{t} \Rightarrow \beta_{MAP}^{-1} = \frac{1}{N-\gamma}\sum\left(t_n - \mathbf{w}_{MAP}^T\boldsymbol{\phi}(\mathbf{x}_n)\right)^2$$

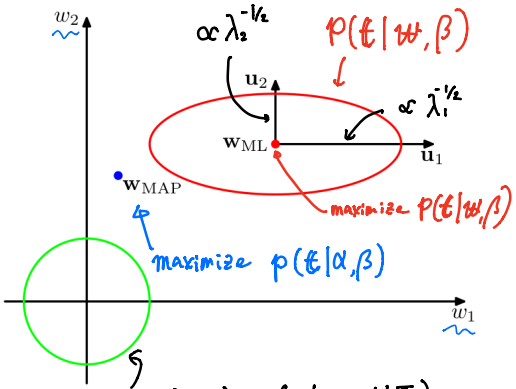


Figure 3.15 Contours of the likelihood function (red) and the prior (green) in which the axes in parameter space have been rotated to align with the eigenvectors $\mathbf{u}_i$ of the Hessian. For $\alpha = 0$, the mode of the posterior is given by the maximum likelihood solution $\mathbf{w}_{\mathrm{ML}}$, whereas for nonzero α the mode is at $\mathbf{w}_{\mathrm{MAP}} = \mathbf{m}_N$. In the direction w_1 the eigenvalue λ_1, defined by (3.87), is small compared with α and so the quantity $\lambda_1/(\lambda_1+\alpha)$ is close to zero, and the corresponding MAP value of w_1 is also close to zero. By contrast, in the direction w_2 the eigenvalue λ_2 is large compared with α and so the quantity $\lambda_2/(\lambda_2+\alpha)$ is close to unity, and the MAP value of w_2 is close to its maximum likelihood value.

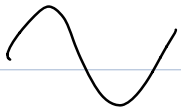
prior $p(\mathbf{w}|\alpha) = \mathcal{N}(\mathbf{w}|\mathbf{0}, \alpha^{-1}\mathbf{I})$

$$E(\mathbf{w}) = E(\mathbf{m}_N) + \tfrac{1}{2}(\mathbf{w}-\mathbf{m}_N)^T\mathbf{V}\,(\alpha\mathbf{I}+\boldsymbol{\Lambda})\mathbf{V}^T(\mathbf{w}-\mathbf{m}_N)$$
$$= E(\mathbf{m}_N) + \tfrac{1}{2}(\mathbf{w}'-\mathbf{m}_N')^T(\alpha\mathbf{I}+\boldsymbol{\Lambda})(\mathbf{w}'-\mathbf{m}_N')$$

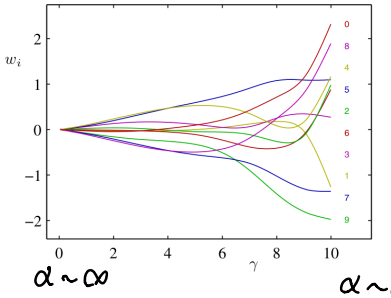
$$p(\mathbf{t}|\mathbf{w},\beta) = E(\mathbf{w}_{ML}) + \tfrac{1}{2}(\mathbf{w}-\mathbf{w}_{ML})^T[\beta\boldsymbol{\Phi}^T\boldsymbol{\Phi}](\mathbf{w}-\mathbf{w}_{ML})$$
$$= E(\mathbf{w}_{ML}) + \tfrac{1}{2}(\mathbf{w}'-\mathbf{w}_{ML}')^T\boldsymbol{\Lambda}(\mathbf{w}'-\mathbf{w}_{ML}')$$

- Matrix A is diagonalizable $\Leftrightarrow A\bar{A}^T = \bar{A}^T A$
  $\Leftrightarrow$ a basis $\{\mathbf{e}_i\}$ exists, where each $\mathbf{e}_i$ is an eigenvector of A

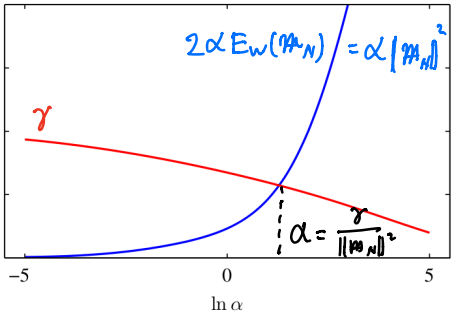
Since Hessian is real & symmetric,
its eigenvectors $\{\mathbf{u}_i\}$ is a basis of $\mathbb{R}^M$.

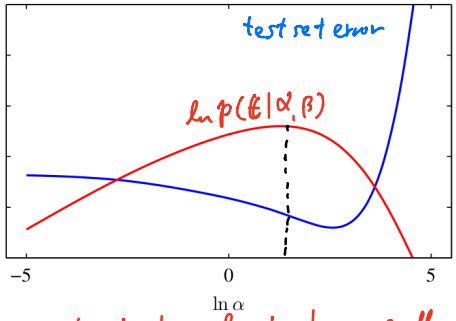
Example: to fit 

with 9-th polynomial
$(w_0, \cdots, w_9)$ (with true β)



$\alpha \sim \infty$ $\alpha \sim 0$





larger α ⇒ tighter regularization ⇒ parameter is drawn less by data: smaller γ

§3.6

Linear models = linear in w

⊕ non-linear $\phi(x)$ fixed before reading the data.

⇓

difficulty, especially in large dimension.

two properties to alleviate this problem:

① data tend to live in a lower-dimensional manifold

⇒ arrange ϕ so that they are scattered only in the manifold

- support vector machine
- relevance vector machine

or adaptive ϕ

- neural network

② t tends to have strong dependency on specific direction of the manifold

… neural network can exploit this property.

PRML 4. LINEAR MODELS FOR CLASSIFICATION

Classification: to separate the input space into K disjoint regions ("decision regions")
(possibly disconnected)
i.e. to find $(D-1)$-dimensional hyperplane.
"decision boundary" "decision surface"

Three approaches

(a) calculate $p(C_k, \mathbf{x})$, $P(\mathbf{x})$, etc. and then decide "generative model"
... heavy computation

(b) calculate $p(C_k|\mathbf{x})$ and then decide. "discriminative model"

(c) find a function $f : \mathbf{x} \mapsto k$ "discriminant function"
... no access to posterior probabilities.

- a target variable notation for probabilistic models

$$\mathbf{t} = (P_1, \cdots, P_K)^T \text{ where } P_k = P(\mathbf{x} \in C_k)$$

or we can transform to $\phi(\mathbf{x})$.

$\Rightarrow$ "Generalized" "linear" model" $y(\mathbf{x}, \mathbf{w}) = f(\mathbf{x}^T\mathbf{w} + w_0)$

f: activation function

§4.1.1 Discriminant functions for 2-class case ($\mathbf{x} \mapsto 0$ or 1)

- linear discriminant $y(\mathbf{x}) = w_0 + \mathbf{w}^T\mathbf{x} = \tilde{\mathbf{w}}^T\tilde{\mathbf{x}}$.
(w_0: bias, $\mathbf{w}$: weight vector)

$\mathbf{x} \in C_1$ if $y(\mathbf{x}) \geq 0$ ($\mathbf{w}^T\mathbf{x} \geq -w_0$)
$\mathbf{x} \in C_2$ if $y(\mathbf{x}) < 0$ ($-w_0$: threshold)

$\rightarrow$ decision plane P is $D-1$-dim. hyperplane

characterized by $P \perp \mathbf{w}$. (and $d(0, P) = -\frac{w_0}{\|\mathbf{w}\|}$)

§4.1.2 Discriminant functions for multiple-class case

- one-versus-the-rest classifier
- one-versus-one classifier

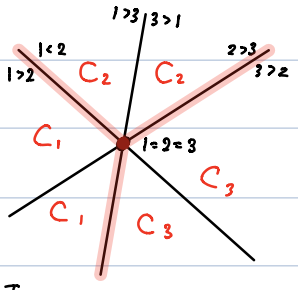
) do not work.

$y_k(x) = w_k \cdot x + w_{k0}$; $x \in C_k$ where $\forall j \neq k,\ y_k(x) > y_j(x)$

decision boundary : $y_k(x) = y_j(x)$

$\cdots (w_k - w_j)^T x + (w_{k0} - w_{j0}) = 0.$

always singly connected & convex.



$$\begin{pmatrix} y_1 \\ \vdots \\ y_k \end{pmatrix} = \begin{pmatrix} w_1 & \cdots & w_k \end{pmatrix}^T x + \begin{pmatrix} w_{01} \\ \vdots \\ w_{0k} \end{pmatrix}$$

$$= \begin{pmatrix} w_{01} & & w_{0k} \\ w_1 & \cdots & w_k \end{pmatrix}^T \begin{pmatrix} 1 \\ x \end{pmatrix}$$

$\therefore y(x) = W^T x + w_0$

$= \tilde{W}^T \tilde{x}$

$\tilde{W} = \begin{pmatrix} w_{01} & w_{02} & \cdots & w_{0k} \\ w_1 & w_2 & & w_k \end{pmatrix}$

§4.1.3 Least square to determine $\tilde{W}$ does not work

because least square method corresponds to ML under Gaussian assumption.

(and weak against outliers)

least square : for input $\{x_n, t_n\}_{n=1,\dots,N}$ $\left[X = \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix},\ T = \begin{pmatrix} t_1 \\ \vdots \\ t_n \end{pmatrix} \right]$

minimize $E_D(\tilde{W}) = \frac{1}{2} Tr\left[(\tilde{X}\tilde{W} - T)^T (\tilde{X}\tilde{W} - T) \right]$

automatically satisfies $\sum_{k=1}^{k} y_k(x) = 1$

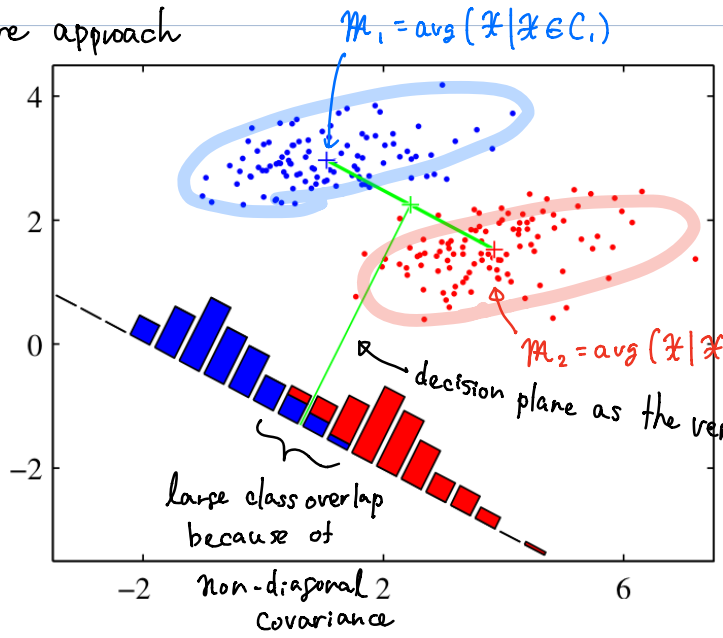
but not restricted to $[0,1]$

$\cdots \tilde{W}_{ML} = (\tilde{X}^T \tilde{X})^{-1} \tilde{X}^T T = \tilde{X}^\dagger T.$ $\quad y_{ML}(x) = (\tilde{X}^\dagger T)^T \tilde{x}$

closed form, but very poor because t_n is far from Gaussian.

§4.1.4 Fisher's linear discriminant

naive approach



Fisher's method

within-class variance $s_k^2(w) = \sum_{n \in C_k} \left(w^T (x_n - m_k) \right)^2$

⇒ Fisher criterion:

maximize $J(w) = \dfrac{(m_1 - m_2)^2}{s_1^2 + s_2^2} = \dfrac{\left[w^T (m_2 - m_1) \right]^2}{\sum_n \left[w^T (x_n - m_k) \right]^2}$

larger $w \cdot \Delta m$ … $w \parallel (m_1 - m_2)$

smaller $w \cdot (m_1 - x)$ … $w \perp (m_1 - x)$

k: for the class x_n belongs to

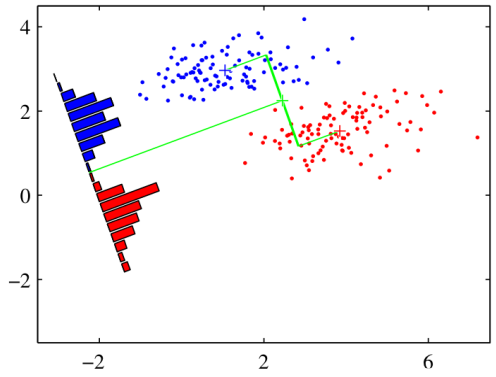
OR in matrix form.

$$J(w) = \frac{w^T S_B w}{w^T S_w w}$$

where $S_B = (m_2 - m_1)(m_2 - m_1)^T$

$S_w = \sum_n (x_n - m_k)(x_n - m_k)^T$

k: for k to which x belongs.



… J is maximized when $(w^T S_w w) S_B w = (w^T S_B w) S_w w$ Fisher's linear discriminant

$S_B w \propto m_2 - m_1$

$\therefore w \propto S_w^{-1} (m_2 - m_1)$

§4.1.5 Fisher's solution is equivalent to least square under a certain target coding scheme

$$t = \begin{cases} N/N_1 & \text{for } C_1 \\ -N/N_2 & \text{for } C_2 \end{cases}$$

N/N_1 ↙ $-N/N_2$ ↙

$$\Rightarrow E(w) = \frac{1}{2}\left[\sum_{n\in C_1}(w^T x_n + w_0 - t_1)^2 + \sum_{n\in C_2}(w^T x_n + w_0 - t_2)^2\right]$$

least square: $\sum_n (w^T x_n) x_n + w_0 \sum_n x_n = N(m_1 - m_2)$

$$w_0 = -w^T m \qquad \text{where } m = \text{avg}(x) = \frac{N_1}{N} m_1 + \frac{N_2}{N} m_2$$

$$\Rightarrow \left(S_w + \frac{N_1 N_2}{N} S_B\right) w = N(m_1 - m_2)$$

$$\therefore \begin{cases} w \propto S_w^{-1}(m_2 - m_1) \\ w_0 = -w^T m \end{cases}$$

$S_w + \frac{N_1 N_2}{N} S_B = \sum x_n x_n^T - N_1 m_1 m_1^T - N_2 m_2 m_2^T + \frac{N_1 N_2}{N}(m_1 - m_2)(m_1 - m_2)^T$
$= \sum x_n x_n^T - N m m^T.$

§4.1.6 Fisher's discriminant for (K > 2)-classes [with (D > K)-input dimension]

Consider mapping x_n to D' "features" $y_n = W^T x_n$ ($y_n \in \mathbb{R}^{D'}$, $x_n \in \mathbb{R}^D$)

↖ no "bias" included

$$S_W = \sum_n (y_n - m_k)(y_n - m_k)^T$$

$D' \times D'$

for C_k to which x_n belongs

$$S_B = \sum_n (m - m_k)(m - m_k)^T = \sum_{k=1}^{K} N_k (m_k - m)(m_k - m)^T \quad ①$$

$D' \times D'$

where $m_k = \text{avg}(y_n \mid x_n \in C_k)$, $m = \text{avg}(y_n) = \sum_{k=1}^{K} \frac{N_k}{N} m_k$.

⇒ We are to find W that maximizes $J(W)$ ($= \text{Tr}[S_W^{-1} S_B]$, for example.)

※ Note that rank(①) = 1 and thus rank(S_B) ≤ K − 1.
So we cannot find more than K − 1 features by this method.
… D′ should be at most K (or K − 1).

§ 4.1.7 Perceptron

bias $\phi_0(x) = 1$ is often included

$$y(x) = f(w^T \phi(x)) \qquad [2\text{-class}]$$

where $f(a) = \begin{cases} +1 & (a \geq 0) \\ -1 & (a < 0) \end{cases}$ (also $t = 1$ for C_1, $t = -1$ for C_2)

Error function — "perceptron criterion"

$$E_p(w) = -\sum_{n \in M} w^T \phi_n t_n \qquad \begin{cases} \phi_n := \phi(x_n) \\ M = \{ n \in N \mid \text{misclassified} \} \end{cases}$$

negative for $n \in M$

positive for $n \in$ correctly classified

"stochastic gradient descent"

if $E(w)$ is given by $\sum_n E_n(w)$,

$w^{(\tau+1)} = w^{(\tau)} - \eta \nabla_w E_n|_{w = w^{(\tau)}}$ minimizes $E(w)$.

↖ chosen so that $w^{(\tau)}$ converges.

$$\Rightarrow w^{(\tau+1)} = w^{(\tau)} + \phi_n t_n \quad \text{for a misclassified } x_n. \qquad [\eta \equiv 1 \text{ since } f(\eta a) = f(a)]$$

- perceptron converges $\Leftrightarrow$ data is linearly-separable

Cons)
- We cannot determine the separability until it converges.
- Even if separable, the result depends on $w^{(0)}$ and the data order
- It won't converge for linearly-non-separable.
- No probabilistic result.
- Not ready to extend for multiple classes.

§4.2 Probabilistic Generative Models

For 2-class case,

$$p(C_1|\mathbf{x}) = \frac{p(\mathbf{x}|C_1)p(C_1)}{p(\mathbf{x})} = \frac{p(\mathbf{x}|C_1)p(C_1)}{p(\mathbf{x}|C_1)p(C_1)+p(\mathbf{x}|C_2)p(C_2)} = \frac{1}{1+e^{-a}} \equiv \sigma(a)$$

(logistic sigmoid)

where $a = \ln \frac{p(\mathbf{x}|C_1)p(C_1)}{p(\mathbf{x}|C_2)p(C_2)}$ $\quad (\sigma(a)+\sigma(-a)=1)$

For K-classes: $p(C_k|\mathbf{x}) = \frac{\exp(a_k)}{\sum \exp(a_i)}$ where $a_k = \ln\left[p(\mathbf{x}|C_k)p(C_k)\right]$

§4.2.1 Assume that $\mathbf{x} \in C_k$ comes from $N(\mathbf{x}|\mu_k, \Sigma)$.

(Σ: assumed universal)

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

For $K=2$,

$$p(C_1|\mathbf{x}) = \sigma(\mathbf{x}^T w + w_0)$$

$$a = \ln\frac{p(C_1)}{p(C_2)} - \frac{1}{2}\left[(\mathbf{x}-\mu_1)^T\Sigma^{-1}(\mathbf{x}-\mu_1) - (\mathbf{x}-\mu_2)^T\Sigma^{-1}(\mathbf{x}-\mu_2)\right]$$

$$= \mathbf{x}^T \underbrace{\Sigma^{-1}(\mu_1-\mu_2)}_{=:w} \underbrace{-\frac{1}{2}\left(\mu_1^T\Sigma^{-1}\mu_1 - \mu_2^T\Sigma^{-1}\mu_2\right) + \ln\frac{p(C_1)}{p(C_2)}}_{=:w_0}$$

For general K,

$$a_k = -\frac{1}{2}(\mathbf{x}-\mu_k)^T\Sigma^{-1}(\mathbf{x}-\mu_k) + \ln p(C_k) + \text{universal const.}$$

$$\equiv \mathbf{x}^T\Sigma^{-1}\mu_k - \frac{1}{2}\mu_k^T\Sigma^{-1}\mu_k + \ln p(C_k)$$

$$=: \mathbf{x}^T w_k + w_{0k}$$

$\therefore$ Σ is universal $\Rightarrow$ boundary is linear in $\mathbf{x}$ (generalized linear model)

non-universal $\Rightarrow$ quadratic decision boundary

§4.2.2 ML solution

*: Weak against outliers.

For $K=2$, we let $t = \begin{matrix}1\\0\end{matrix}$ for $\begin{matrix}C_1\\C_2\end{matrix}$ and prior $p(C_1) = \pi$, $p(C_2) = 1-\pi$.

Likelihood $p(\mathbf{t}, \mathbf{X} \mid \pi, \mu_1, \mu_2, \Sigma)$ (Σ: assumed universal; Gaussian assumed)

$$= \prod_{n=1}^{N} \left[\pi \mathcal{N}(x_n \mid \mu_1, \Sigma)\right]^{t_n} \left[(1-\pi)\mathcal{N}(x_n \mid \mu_2, \Sigma)\right]^{1-t_n}$$

$$\left(\log p = N_1 \log \pi + N_2 \log(1-\pi) - \frac{N}{2}\log|\Sigma| - \sum_{n=1}^{N} \frac{1}{2}(x_n-\mu_k)^T \Sigma^{-1}(x_n-\mu_k) + \text{Const.}\right)$$

$$\Rightarrow \quad \pi_{ML} = \frac{N_1}{N_1+N_2}, \qquad (\mu_k)_{ML} = \text{avg}\,(x_n \mid x_n \in C_k)$$

$$\Sigma_{ML} = \frac{1}{N}\sum_{n=1}^{N} (x_n-\mu_k)(x_n-\mu_k)^T$$

$$\frac{\partial |A|}{\partial A} = |A|(A^T)^{-1}$$

$$\frac{\partial A^{-1}}{\partial x} = -A^{-1}\frac{\partial A}{\partial x}A^{-1}$$

§4.2.3

For discrete input $x \in \{0,1\}^D$, general $p(x \mid C_k)$ has $2^D - 1$ parameter θ_k for each class.

If we simply assume $p(x \mid C_k) = \prod_{d=1}^{D} \mu_{kd}^{x_d}(1-\mu_{kd})^{1-x_d}$

each x_d is independent with Bernoulli dist. $\Rightarrow$ D parameter for each class

Then $p(C_k \mid x) = \frac{\exp(a_k)}{\sum \exp(a_j)}$ with $a_k = \ln\left[p(x \mid C_k)p(C_k)\right]$

$$= \ln p(C_k) + \sum_{d=1}^{D}\left[x_d \ln \frac{\mu_{kd}}{1-\mu_{kd}} + \ln(1-\mu_{kd})\right]$$

(linear in x_d !)

§4.2.4

If class-conditional density $p(x \mid C_k)$ is in exponential family with $u(x) = x$, a_k is linear in x.

$$p(x \mid C_k) = h(x)\, g(\theta_k) \exp\left(\theta_k^T u(x)\right)$$

$$\Rightarrow a_k = \underbrace{\theta_k^T x}_{\text{Common and irrelevant.}} + \ln h(x) + \ln g(\theta_k) + \ln p(C_k)$$

§4.3 Probabilistic Discriminative model

determine $p(C_k|x)$ without $p(x|C_k)$ etc.

... directly by ML. "iterative reweighted least square"

Here we introduce fixed basis functions $\phi(x)$

§4.3.2

$\mathbb{E}[t|\phi] = p(t=1|\phi) = y(\phi)$

Logistic regression (for 2 classes) — logistic sigmoid $\sigma = \frac{1}{1+e^{-a}}$, $\frac{d\sigma}{da} = \sigma(1-\sigma)$

$$p(C_1|\phi) = y(\phi) = \sigma(w^T\phi)$$

(includes "bias" $\phi_0(x) = 1$)

... M parameters for $\phi(x) \in \mathbb{R}^M$

[Generative model needs $\mu_k, \Sigma, p(C_1) = 2M + \frac{M(M+1)}{2} + 1$ parameters]

$$p(t|X, w) = \prod y_n^{t_n} (1-y_n)^{1-t_n}$$

$$E(w) = -\ln p(t|X, w) = -\sum \left[t_n \log y_n + (1-t_n) \ln(1-y_n) \right]$$

$$\Rightarrow \nabla_w E(w) = \sum \frac{t_n - y_n}{y_n(y_n-1)} \nabla y_n = \sum (y_n - t_n)\phi(x_n)$$

(→ we can employ some sequential algorithm)

※ This ML method can cause Overfitting for linearly-separable data.

ML solution $\tilde{w}$ satisfies $\sum (\tilde{y}_n - t_n)\phi(x_n) = 0$.

If the data is separable,

$t_n = 1 \Leftrightarrow \tilde{y}_n > 0.5$, i.e. the hyperplane $w^T\phi = 0$ is the ML result.

$t_n = 0 \Leftrightarrow \tilde{y}_n < 0.5$

Then ML prefers larger $\|w\|$ and $\sigma(w^T\phi) \approx \Theta(w^T\phi)$,

which is very weak for new data.

⇒ we need prior or regularization.

§4.3.3

Newton-Raphson method

To find $f(\mathbf{x}) = \mathbf{0}$ for $f: \mathbb{R}^m \to \mathbb{R}^m$,

using $f_k(\mathbf{x}) \simeq f_k(\mathbf{x}_0) + \frac{\partial f_k}{\partial x_k}(x_k - x_{0k}) \rightsquigarrow 0$,

$x_k := x_k - (J^{-1})_{kl} f_l(\mathbf{x})$, where $J_{ij} = \frac{\partial f_i}{\partial x_j}(\mathbf{x})$.

$\therefore \mathbf{x} := \mathbf{x} - J^{-1} f(\mathbf{x})$.

→ Maximization of $F(\mathbf{x})$, where $F: \mathbb{R}^m \to \mathbb{R}$, is $\nabla F = 0$, thus

$$x_k := x_k - H^{-1}_{kl} \frac{\partial}{\partial x_l} F(\mathbf{x}) \quad \text{where} \quad H_{ij} = \frac{\partial^2 F}{\partial x_i \partial x_j}.$$

$$\therefore \mathbf{x} := \mathbf{x} - H^{-1} \nabla F.$$

$$\nabla_{\mathbf{w}} E(\mathbf{w}) = \sum (y_n - t_n)\boldsymbol{\phi}_n, \quad H_{ij} = \sum_n y_n(1-y_n)\phi_i(\mathbf{x}_n)\phi_j(\mathbf{x}_n)$$

or $H = \boldsymbol{\Phi}^T R \boldsymbol{\Phi}$ with $\boldsymbol{\Phi}_{ij} = \phi_j(\mathbf{x}_i)$, $R = \mathrm{diag}(y_n(1-y_n))$.

positive definite because E is convex. ∴ unique minimum. "Variance"

$$\Rightarrow \text{Newton-Raphson} \quad \mathbf{w} := \mathbf{w} - (\boldsymbol{\Phi}^T R \boldsymbol{\Phi})^{-1} \boldsymbol{\Phi}^T (\mathbf{y} - \mathbf{t})$$

$$= (\boldsymbol{\Phi}^T R \boldsymbol{\Phi})^{-1} \boldsymbol{\Phi}^T R \mathbf{z}$$

iterative reweighting

where $\mathbf{z} = \boldsymbol{\Phi}\mathbf{w} - R^{-1}(\mathbf{y} - \mathbf{t})$

Mean and variance for logistic regression model $P(C_1|\boldsymbol{\phi}) = \sigma(\mathbf{w}^T\boldsymbol{\phi})$

$\mathbb{E}[t] = P(C_1|\boldsymbol{\phi}) + 0 \times P(C_2|\boldsymbol{\phi}) = y$

$\mathbb{V}[t] = \mathbb{E}[t^2] - \mathbb{E}[t]^2 = y(1-y)$

We can use this z_n directly.

Considering $a_n = \mathbf{w}^T\boldsymbol{\phi}_n$ and $E(\mathbf{a}) = -\sum [t_n \log y_n + (1-t_n)\log(1-y_n)]$,

$$a_n := a_n - \frac{y_n - t_n}{y_n(1-y_n)} = z_n.$$

$\nabla_{\mathbf{a}} E = \mathbf{y} - \mathbf{t}$

$H_{nn} = y_n(1-y_n)$

§4.3.4 logistic regression for K classes

$$p(C_k|\phi) = y_k(\phi) = \frac{\exp(a_k)}{\sum \exp(a_j)} \quad \text{with} \quad a_k = w_k^T \phi$$

t_n : 1-of-K coding scheme

$$p(\mathbf{T}|w_1,\cdots,w_K) = \prod_{n=1}^{N}\prod_{k=1}^{K} y_k(\phi_n)^{t_{nk}} \quad (\to y_{nk})$$

$$E(\mathbf{T}|w_1,\cdots,w_K) = -\sum_{n=1}^{N}\sum_{k=1}^{K} t_{nk} \ln y_{nk}$$ cross entropy error function

$$\nabla_{w_j} E = -\sum_{n=1}^{N}\left(t_{nj} - \sum_{k=1}^{K} t_{nk} y_j\right)\phi(x_n)$$

$$= -\sum_{n=1}^{N}\left(t_{nj} - y_j(\phi_n)\right)\phi_n$$

$$\frac{\partial y_k}{\partial a_j} = (\delta_{jk} - y_j)y_k = \delta_{jk}y_k - y_j y_k$$

§4.3.5 Other activation functions?

- noisy threshold model : assign $t_n = 1$ iff $a_n = w^T\phi_n > \theta$

  and θ is "noisy", i.e. given by some PDF $g(\theta)$.

this results in $p(t_n = 1|a) = \int_{-\infty}^{a} g(\theta)\, d\theta$.

if we use $g(\theta) = \mathcal{N}(\theta|0,1)$, ← equivalent to $\mathcal{N}(\theta|\mu,\sigma)$

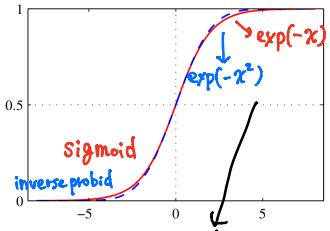
CDF gives the inverse probit function

$$\Phi(a) = \int_{-\infty}^{a} \mathcal{N}(\theta|0,1)\, d\theta$$

$$= \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{a} \exp\left(-\frac{1}{2}\theta^2\right) d\theta$$

$$= \frac{1}{2}\left[1 + \text{erf}\left(\frac{w^T\phi}{\sqrt{2}}\right)\right]$$

$$\text{erf}(a) = \frac{2}{\sqrt{\pi}}\int_0^a e^{-x^2} dx.$$



inverse probit is weak against outliers.

and we can determine w by ML method.

✡ We can introduce a hyperparameter ε : the probability that t has been mislabeled.

Then $p(t|x) = (1-\varepsilon)\sigma(x) + \varepsilon(1-\sigma(x))$

(?) §4.3.6 Canonical link functions

§4.2.4
If class-conditional density $p(x|C_k)$ is in exponential family with $u(x)=x$, a_k is linear in x.

Consider "generalized linear model":

$$p(t|w,\phi) = \frac{1}{s} h\left(\frac{t}{s}\right) g(\eta) \exp\left(\frac{\eta t}{s}\right), \quad \text{where } \eta = \xi(w^T\phi).$$

Then $y \equiv E[t|w,\phi] = -s \frac{d}{d\eta} \ln g(\eta)\Big|_{\eta=\xi(w^T\phi)}$

$0 = \frac{\partial}{\partial\eta}\int p(t)\,dt = \int \left(\frac{g'(\eta)}{g(\eta)} + \frac{t}{s}\right) p(t)\,dt$

$=: \psi^{-1}(\eta)\Big|_{\eta=\xi(w^T\phi)}$ $\quad \therefore y = f(w^T\phi) = (\psi^{-1}\circ\xi)(w^T\phi)$.

ψ^{-1}: Canonical link function; f: activation function; $\cdots \xi = \psi \circ f$

For GLM,

$$-\nabla_w \ln p(t|w,\Phi) = -\sum_{n=1}^{N} \left(\frac{t_n}{s} + \frac{d}{d\eta}\ln g(\eta_n)\right)\frac{\partial\eta}{\partial w}$$

$$= \frac{1}{s}\sum_{n=1}^{N}(y_n - t_n)\nabla_w \xi(w^T\phi_n)$$

if we choose f so that $\xi(a) = a$,

We can't choose f because $f = \psi^{-1}\xi$.

$$\nabla_w E(w) = \frac{1}{s}\sum_{n=1}^{N}(y_n - t_n)\phi_n.$$

I don't understand the importance / meaning of this section

If $p(t|w,\phi) = k^t(1-k)^{1-t}$ with $k = k(w^T\phi)$,

$$p(t|w,\phi) = (1-k)\exp\left[t \ln\frac{k}{1-k}\right] = (1+e^{\eta})^{-1}\exp\left(\frac{\eta t}{s}\right)$$

with $\eta = \xi(w^T\phi) = s \ln\frac{k}{1-k}$, $g(\eta) = s(1+e^{\eta})^{-1}$

$y \equiv E[t|w,\phi] = k(a) = (1+e^{-\eta/s})^{-1}$ $\quad \cdots \psi^{-1}(\eta) = (1+\exp(-\eta/s))^{-1}$.

$\cdots f(a) = k(a)$.

For Gaussian $p(t|\mu) = \frac{1}{\sqrt{2\pi}\sigma}\exp\left(-\frac{1}{2\sigma^2}(t-\mu(x))^2\right)$ with known σ,

$$p(t|\mu) = \frac{1}{\sqrt{2\pi}\sigma}\exp\left(-\frac{t^2}{2\sigma^2}\right)\exp\left(-\frac{\mu^2}{2\sigma^2}\right)\exp\left(\frac{1}{\sigma^2}\mu t\right)$$

$$= \frac{1}{\sqrt{2\pi}\sigma}\exp\left(-\frac{s^2}{2\sigma^2}\frac{t^2}{s^2}\right)\exp\left(-\frac{\sigma^2\eta^2}{2s^2}\right)\exp\left(\frac{\eta t}{s}\right)$$

$\eta = \xi(a) = \frac{s}{\sigma^2}\mu(a)$

$E[t|\mu] = \mu = \frac{\sigma^2}{s}\eta$. $\quad \cdots \psi^{-1}(\eta) = \frac{\sigma^2}{s}\eta$. $\quad f(a) = \mu(a)$.

$f(a) = \mu(a)$. So what?

§4.4 Laplace approximation

... to give $q(z) = \mathcal{N}(z|\mu, \sigma) \approx p(z)$ with μ being the mode of p.

$$p(z) = \frac{f(z)}{Z}, \quad \text{where } Z = \int f(z)\,dz.$$

$$\ln f(z) \approx \ln f(z_0) - \frac{1}{2}\left[-\frac{d^2}{dz^2}\ln f(z_0)\right](z - z_0)^2, \quad \text{where } \begin{cases} f'(z_0) = 0 \\ f''(z_0) < 0. \end{cases}$$

$$\therefore\ p(z) = \frac{f(z)}{Z} \approx \sqrt{\frac{A}{2\pi}} \exp\left(-\frac{A}{2}(z - z_0)^2\right),$$

$$A = -\frac{d^2}{dz^2}\ln f(z_0).$$

For multidimensional $p(\mathbf{z})$,

$$p(\mathbf{z}) \approx \frac{|A|^{1/2}}{(2\pi)^{M/2}} \exp\left[-\frac{1}{2}(\mathbf{z} - \mathbf{z}_0)^T A (\mathbf{z} - \mathbf{z}_0)\right]$$

where $\nabla p = \mathbf{0}$, $\quad A = -\nabla\nabla \ln f(\mathbf{z}_0)$ [positive definite].

[also we can obtain

$$Z \equiv \int d\mathbf{z} f(\mathbf{z}) \approx f(\mathbf{z}_0)\sqrt{\frac{(2\pi)^M}{|A|}} \quad \left(= \frac{f(z_0)}{\sqrt{A/2\pi}}\right)$$

].

§4.5 Bayesian logistic regression

$$p(w|t) \propto p(t|w)\,p(w)$$

$p(t|w)$: Bernoulli, $\prod y_n^{t_n}(1-y_n)^{1-t_n}$ with $y_n = \sigma(w^T \phi_n)$

$p(w)$: $\mathcal{N}(w|m_0, S_0)$

posterior

$$\Rightarrow \ln p(w|t) = -\frac{1}{2}(w-m_0)^T S_0^{-1}(w-m_0) + \sum_{n=1}^{N}\left[t_n \ln y_n + (1-t_n)\ln(1-y_n)\right] + C.$$

With Laplace approx,

$$p(w|t) \approx q(w|t) = \mathcal{N}(w|w_{MAP}, S_N)$$

where $m_N = w_{MAP}$ is given by $\nabla_w \ln p = 0$, and

$$S_N^{-1} = -\nabla\nabla \ln p(w|t) = S_0^{-1} + \sum_{n=1}^{N} y_n(1-y_n)\phi_n \phi_n^T.$$

Predictive distribution is

$$p(C_1|\phi, t) = \int dw\, p(C_1|w,\phi)\,p(w|t)$$

$$\approx \int da\, \sigma(a)\,\mathcal{N}(a|\mu_a, \sigma_a^2)$$

with $\mu_a = w_{MAP}^T \phi$, $\sigma_a^2 = \phi^T S_N \phi$.

Also, $\sigma(a) \approx \Phi(\sqrt{\frac{\pi}{8}}a)$ together with

$$\int_{-\infty}^{\infty} da\, \Phi(\lambda a)\,\mathcal{N}(a|\mu, \sigma^2) = \Phi\left(\frac{\mu}{\sqrt{\sigma^2 + \lambda^{-2}}}\right)$$

gives $p(C_1|\phi, t) \approx \sigma\left(\frac{\mu_a}{\sqrt{1+\pi\sigma_a^2/8}}\right)$.

* If we use the simplest decision boundary $p(C_1|\phi, t) = 0.5$,

it is given by $\mu_a = 0$ ... $\phi \perp w_{MAP}$.

$\Rightarrow$ equivalent to ML approach.