PRML Study: Chapter 3

Linear models for regression

(발표자 : 신진수

Machine Learning 목표 : Function Approximation

Parametric

Non-Parametric

Generalisation !!

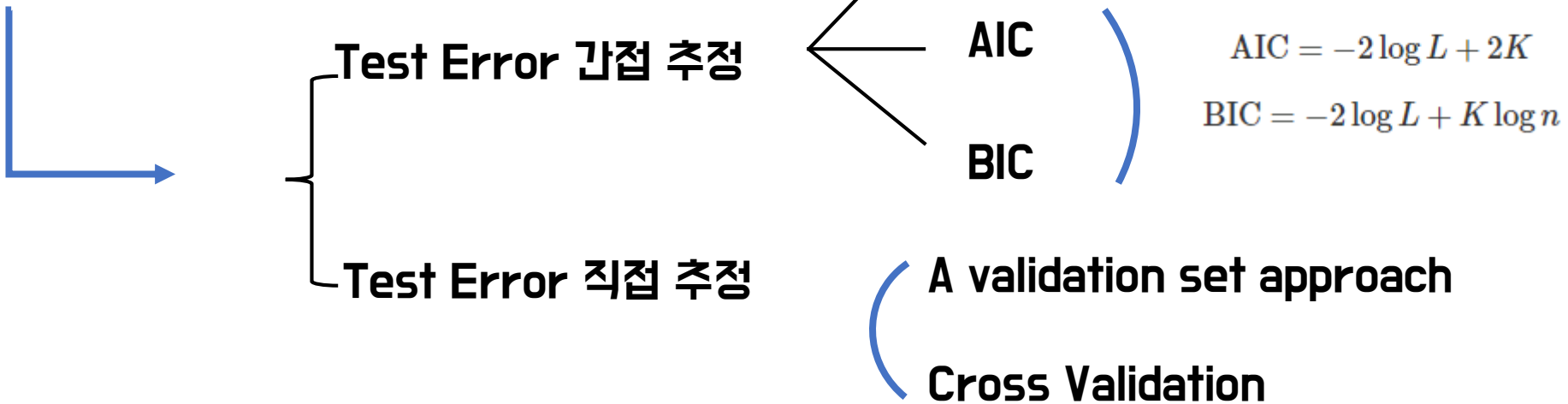
좋은 추정량 선택을 위한 조건

1. Sufficiency — 베이지안(우도원리) / Likelihood Ratio → Factorisation Theorem
2. Unbiasedness — Bias 작으면 variance 커짐
3. Consistency — Convergence in Probability
4. Efficiency — 최소 분산성

Choosing the optimal model

Test error 가 가장 작게 나오게 만들기

1. Subset Selection



2. Shrinkage Method Overfitting?

Decision Theory 1.5.5 내용인데 3.1 Intro에 다시 등장

1. Loss function 도입한 이유 ? To evaluate how good an estimator is

1. Squared loss: $L(\widehat{\theta}, \theta) = (\widehat{\theta} - \theta)^2$.
2. Absolute loss: $L(\widehat{\theta}, \theta) = |\theta - \theta|$.
3. Kullback-Leibler loss: $L(\widehat{\theta}, \theta) = \mathrm{KL}(\widehat{\theta}, \theta) \equiv \int p_\theta(u) \log \left(\frac{p_\theta(u)}{p_{\widehat{\theta}}(u)} \right) du$.

2. Risk Function Expectation of Loss function

→ Risk Function을 최소화 해주는 Theta hat찾기

각각 1, 2의 결과?

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

Basis Function

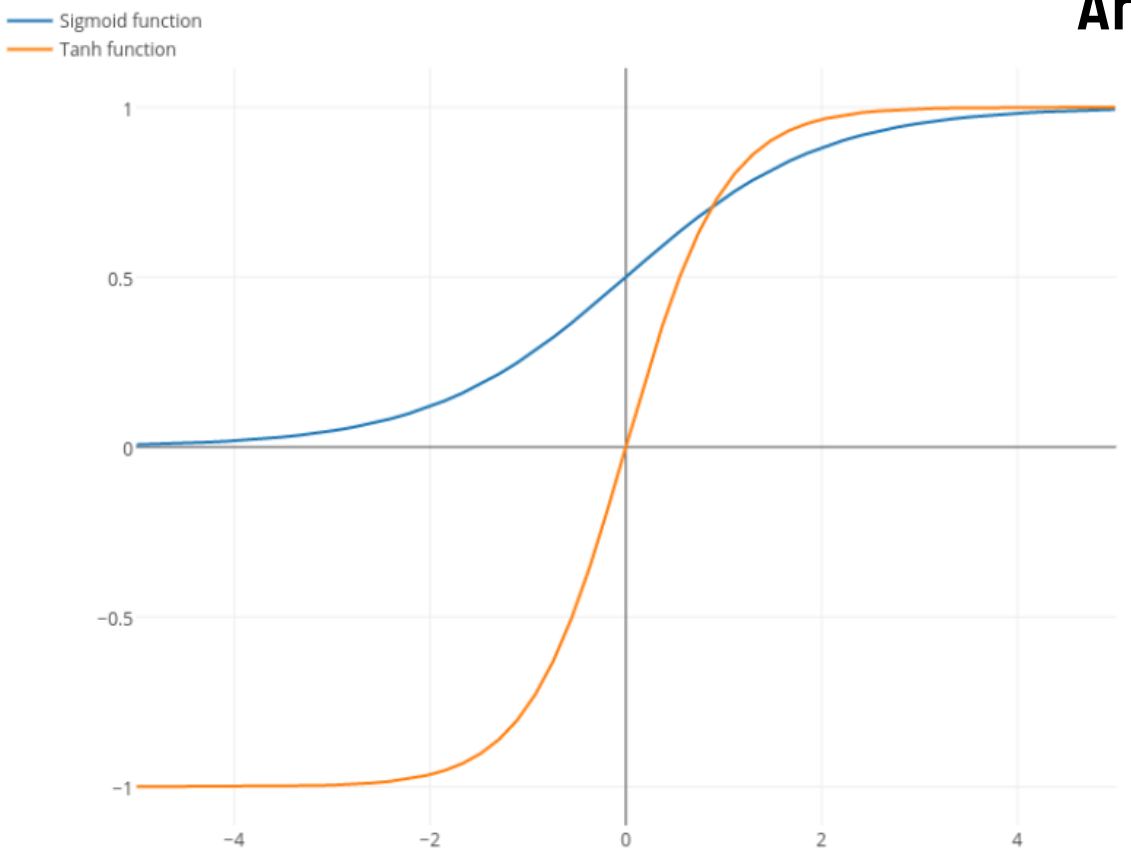
Output을 새로운 범위로 Mapping

Generalised Linear Model

Exponential Family 가정

→ Likelihood Function을 이용해 회귀계수 추정

Analytical Method X -> Numerical Method



Sigmoid와 거의 똑같은 함수?

Sign이 중요할 때는 ?

회귀분석(Regression Analysis)

왜 배우는지?

$$y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_k x_{ik} + \varepsilon_i, \qquad (i = 1, 2, \ldots, n)$$

$$\boldsymbol{y} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon},$$

회귀계수 추정(Regression Coefficients Estimation)

- 최소제곱법(OLS) 이용한 추정 : 오차제곱합 최소화
  - 꼭 Squared 로 해야 하나? Popular Choice
  - 오차항이 정규분포를 따른다는 가정이 없을 때에도 적용가능
- 오차항의 정규성 가정 : Likelihood Function 이용
  - 보통 항상 값이 딱 떨어지게 나오지 않는 경우가 많음
  - Closed form expression X

회귀분석(Regression Analysis)

핵심 "Orthogonal Projection"

$$\boldsymbol{y} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\epsilon},$$

Design Matrix

$$\boldsymbol{y} = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix}, \qquad \boldsymbol{X} = \begin{pmatrix} 1 & x_{11} & x_{12} & \dots & x_{1k} \\ 1 & x_{21} & x_{22} & \dots & x_{2k} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & x_{n1} & x_{n2} & \dots & x_{nk} \end{pmatrix},$$

$$\boldsymbol{\beta} = \begin{pmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_k \end{pmatrix}, \qquad \text{and} \qquad \boldsymbol{\epsilon} = \begin{pmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_n \end{pmatrix}$$

Basis Function으로
Mapping 해준 거

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

마할라노비스 거리 / 3.5 유사 아이디어

$$(\mathbf{y} - \mathbf{X}\boldsymbol{\beta})^{\mathrm{T}}(\mathbf{y} - \mathbf{X}\boldsymbol{\beta}) = (\mathbf{y} - \mathbf{X}\hat{\boldsymbol{\beta}})^{\mathrm{T}}(\mathbf{y} - \mathbf{X}\hat{\boldsymbol{\beta}}) + (\boldsymbol{\beta} - \hat{\boldsymbol{\beta}})^{\mathrm{T}}(\mathbf{X}^{\mathrm{T}}\mathbf{X})(\boldsymbol{\beta} - \hat{\boldsymbol{\beta}})$$

회귀분석(Regression Analysis)

Closed Form Solution = **Trial and error**

$$\sum_{i=1}^{n} \varepsilon_i^2 = (\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta})'(\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta})$$

$$= \boldsymbol{y}'\boldsymbol{y} - \boldsymbol{\beta}'\boldsymbol{X}'\boldsymbol{y} - \boldsymbol{y}'\boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\beta}'\boldsymbol{X}'\boldsymbol{X}\boldsymbol{\beta}$$

$$= \boldsymbol{y}'\boldsymbol{y} - 2\boldsymbol{y}'\boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\beta}'\boldsymbol{X}'\boldsymbol{X}\boldsymbol{\beta}$$

$$\left.\frac{\partial S}{\partial \boldsymbol{\beta}}\right|_{\hat{\boldsymbol{\beta}}} = -2\boldsymbol{X}'\boldsymbol{y} + 2\boldsymbol{X}'\boldsymbol{X}\hat{\boldsymbol{\beta}} = 0$$

오차제곱합은 Convex 함수?

$$\boldsymbol{X}'\boldsymbol{X}\hat{\boldsymbol{\beta}} = \boldsymbol{X}'\boldsymbol{y}$$ **Normal Equation**

$$\hat{\boldsymbol{\beta}} = (\boldsymbol{X}'\boldsymbol{X})^{-1}\boldsymbol{X}'\boldsymbol{y}$$

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

Cost Function

$$J(\Theta_0, \Theta_1) = \frac{1}{2m}\sum_{i=1}^{m}[h_\Theta(x_i) - y_i]^2$$

$h_\Theta(x_i)$: Predicted Value, y_i: True Value

Gradient Descent

$$\Theta_j = \Theta_j - \alpha\frac{\partial}{\partial\Theta_j}J(\Theta_0, \Theta_1)$$

α: Learning Rate

Now,

$$\frac{\partial}{\partial\Theta}J_\Theta = \frac{\partial}{\partial\Theta}\frac{1}{2m}\sum_{i=1}^{m}[h_\Theta(x_i) - y]^2$$

$$= \frac{1}{m}\sum_{i=1}^{m}(h_\Theta(x_i) - y)\frac{\partial}{\partial\Theta_j}(\Theta x_i - y)$$

$$= \frac{1}{m}(h_\Theta(x_i) - y)x_i$$

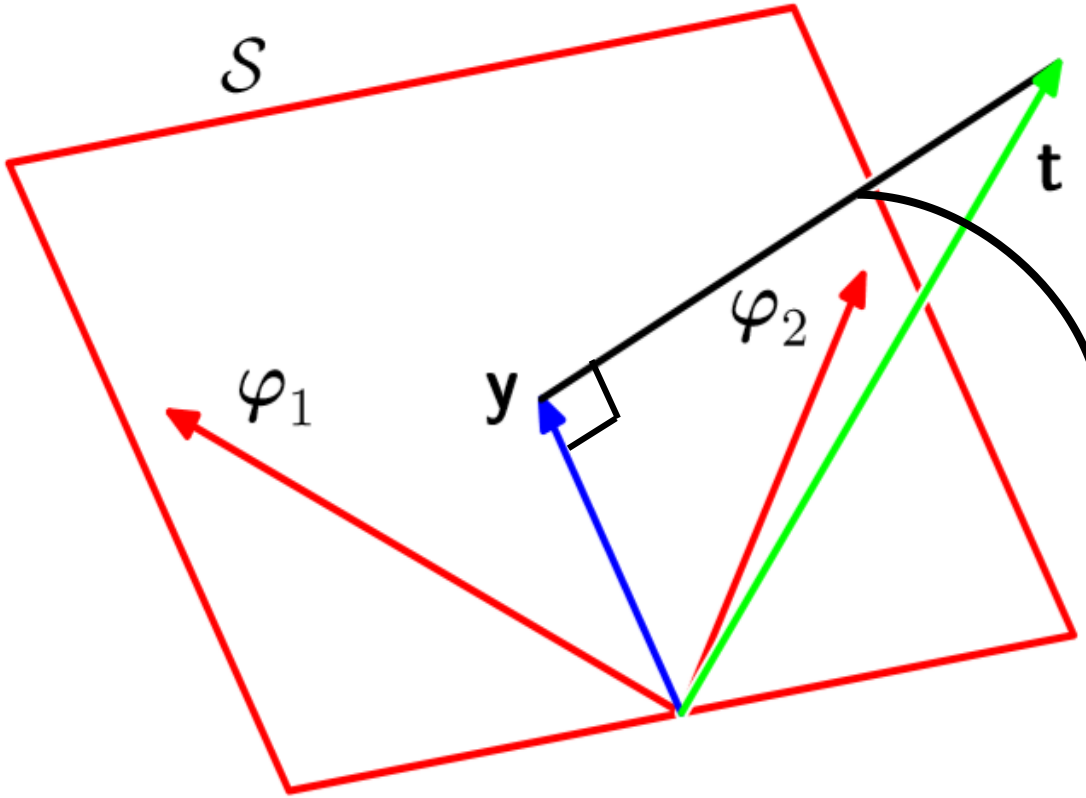
Therefore,

$$\Theta_j := \Theta_j - \frac{\alpha}{m}\sum_{i=1}^{m}[(h_\Theta(x_i) - y)x_i]$$

Geometry of Least Squares

Orthogonal Projection

Basis Function $\phi_j(\mathbf{x})$이 spanned 하는 공간 S로의 Orthogonal Projection



여기서의 Projection matrix(hat matrix)는 ?

$$\mathbf{\Phi}\left(\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi}\right)^{-1}\mathbf{\Phi}^{\mathrm{T}}$$

t - y

회귀에서는 잔차(residual)를 최소화

= Orthogonal Projection

Regularised Least Squares

3.4에 유사한 식 등장

Minimise $\frac{1}{2}\sum_{n=1}^{N}\{t_n - \mathbf{w}^{\mathrm{T}}\boldsymbol{\phi}(\mathbf{x}_n)\}^2 + \frac{\lambda}{2}\sum_{j=1}^{M}|w_j|^q$

1부터 시작하는 이유?

$$\hat{\beta} = (X^T X + \lambda \begin{bmatrix} 0 & & \\ & 1 & \\ & & \ddots \\ & & & 1 \end{bmatrix})^{-1} X^T \vec{y}$$

추가해주는 이유?
Rank?

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

Shrinkage Method 언제사용??

$$\underset{\beta}{\text{minimize}} \left\{ \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 \right\} \quad \text{subject to} \quad \sum_{j=1}^{p} |\beta_j| \le s \tag{6.8}$$

and

$$\underset{\beta}{\text{minimize}} \left\{ \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 \right\} \quad \text{subject to} \quad \sum_{j=1}^{p} \beta_j^2 \le s, \tag{6.9}$$

제약조건 있는 최적화?

Largrange Mutlplier 이용? 너무 수식이 복잡하다ㅜㅜ

Lasso

$$\sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 + \lambda \sum_{j=1}^{p} |\beta_j|$$

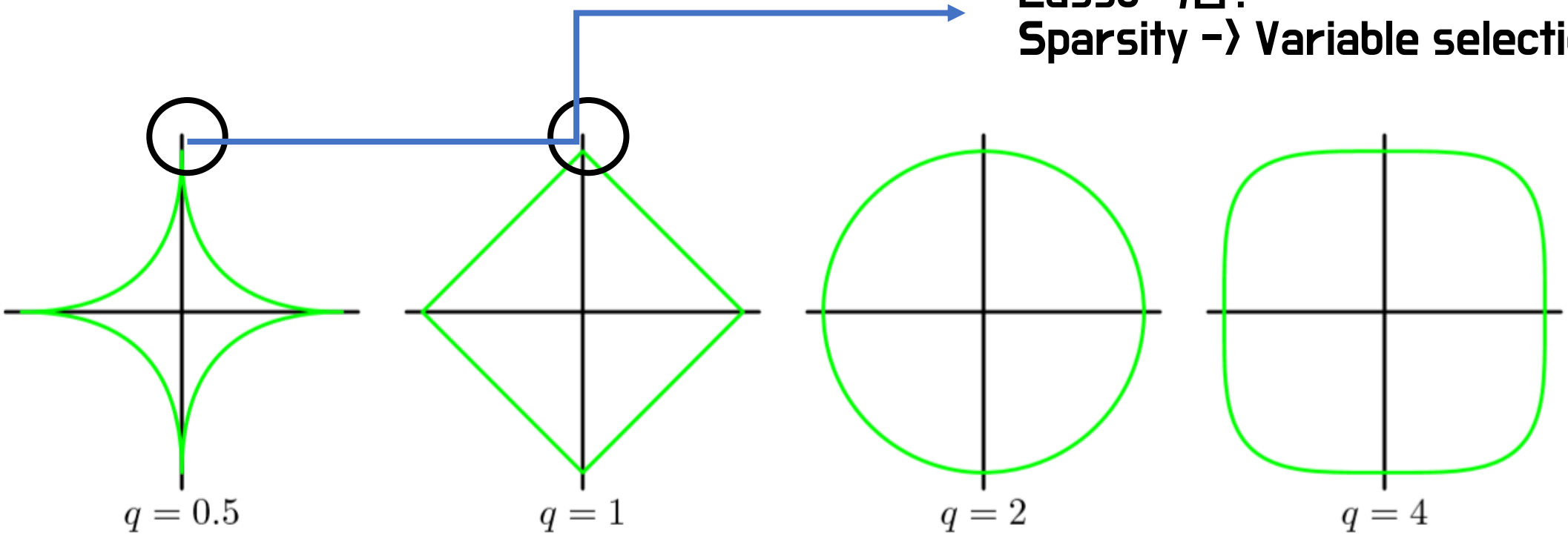
Ridge

$$\sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 + \lambda \sum_{j=1}^{p} \beta_j^2$$

적절한 Lambda(turning parameter)를 찾는 방법?

Shrinkage Method

Lasso 핵심?
Sparsity -> Variable selection



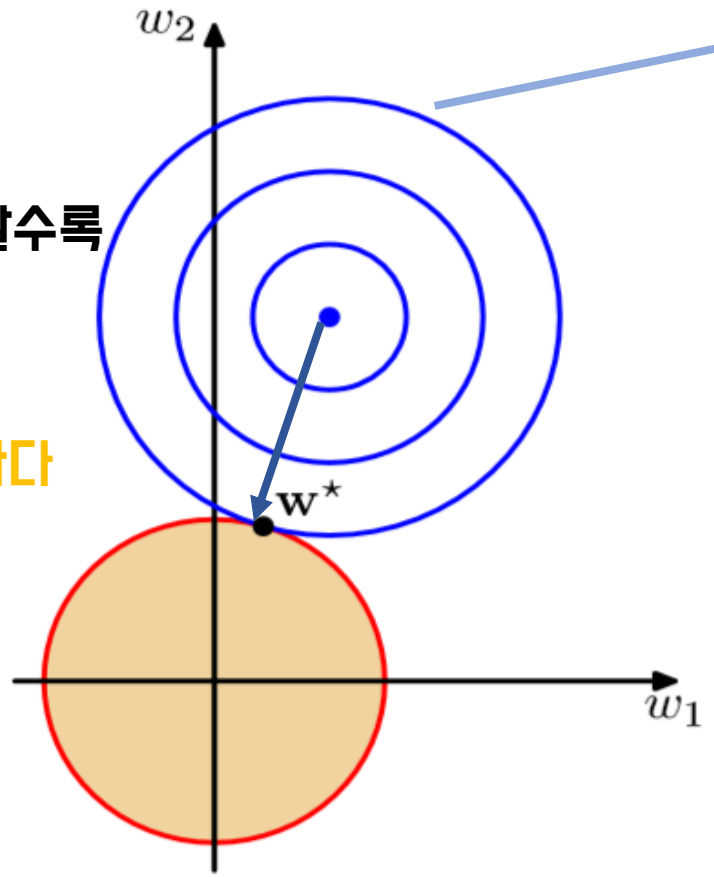
Shrinkage Method 그림으로 만나보기

이 그림은 단순히 2차원으로 볼 수 있지만
w3 축이 PPT를 뚫고 나온다고 생각해 본다면?

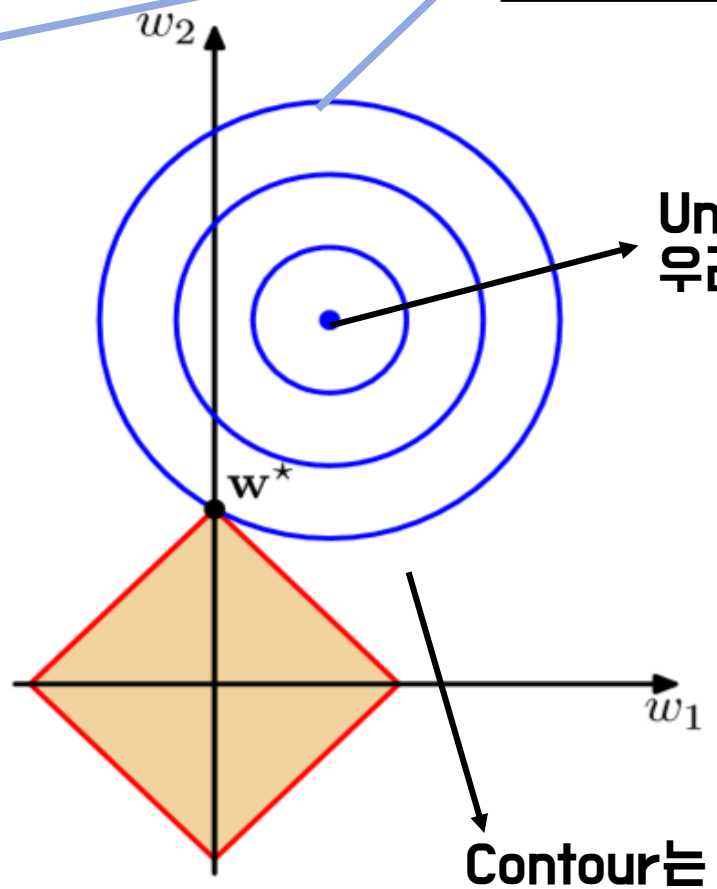
타원의 special case
w1, w2가 scaling 된 경우

화살표 방향으로 갈수록
Bias 커진다

다시 말해,
Variance 감소한다



Ridge



Lasso

Unbiased estimator ?
우리가 구한 OLS solution?

Contour는 뭘까? Likelihood function

Bias-Variance Decomposition

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

$(\text{bias})^2 + \text{variance}$

noise **(irreducible)**

Proof

$$E\{(t_i - y_i)^2\} = E\{(t_i - f_i + f_i - y_i)^2\}$$

$$= E\{(t_i - f_i)^2\} + E\{(f_i - y_i)^2\} + 2E\{(f_i - y_i)(t_i - f_i)\}$$

$$= E\{\varepsilon^2\} + E\{(f_i - y_i)^2\} + 2\left(E\{f_i t_i\} - E\{f_i^2\} - E\{y_i t_i\} + E\{y_i f_i\}\right)$$

Note : $E\{f_i t_i\} = f_i^2$ since f is deterministic and $E\{t_i\} = f_i$

: $E\{f_i^2\} = f_i^2$ since f is deterministic

: $E\{y_i t_i\} = E\{y_i(f_i + \varepsilon)\} = E\{y_i f_i + y_i \varepsilon\} = E\{y_i f_i\} + 0$

: (the last term is zero because the noise in the infinite test set over which

: we take the expectation is probabilistically independent of the NN

: prediction). Thus the last term in the expectation above cancels to zero.

$$E\{(t_i - y_i)^2\} = E\{\varepsilon^2\} + E\{(f_i - y_i)^2\}$$

Thus the MSE can be decomposed in expectation into the variance of the noise and the MSE between the true function and the predicted values. This term can be further composed with the same augmentation trick as above.

$$E\{(f_i - y_i)^2\} = E\{(f_i - E\{y_i\} + E\{y_i\}_i - y_i)^2\}$$

$$= E\{(f_i - E\{y_i\})^2\} + E\{(E\{y_i\} - y_i)^2\} + 2E\{(E\{y_i\} - y_i)(f_i - E\{y_i\})\}$$

$$= bias^2 + Var\{y_i\} + 2\left(E\{f_i E\{y_i\}\} - E\{E\{y_i\}^2\} - E\{y_i f\}_i + E\{y_i E\{y_i\}\}\right)$$

Note : $E\{f_i E\{y_i\}\} = f_i E\{y_i\}$ since f is deterministic and $E\{E\{z\}\} = z$

: $E\{E\{y_i\}^2\} = E\{y_i\}^2$ since $E\{E\{z\}\} = z$

: $E\{y_i f_i\} = f_i E\{y_i\}$

: $E\{y_i E\{y_i\}\} = E\{y_i\}^2$

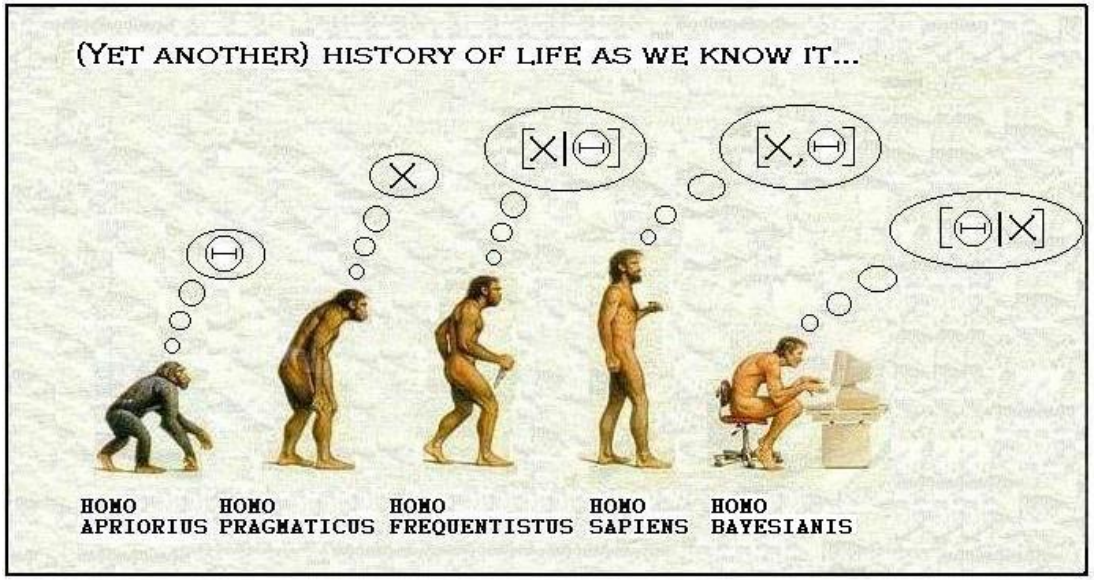
: Thus the last term in the expectation above cancels to zero.

$$E\{(f_i - y_i)^2\} = bias^2 + Var\{y_i\}$$

베이지안을 쓰는 이유? / Weighted Averaging (P.151)

1. 사람의 인지체계 모르는 게 나타났을 때 사람의 인지체계 변화하는 방식을 통계관점으로 적립

2. Sequential Learning



Bayesian Optimization in AlphaGo

Yutian Chen, Aja Huang, Ziyu Wang, Ioannis Antonoglou, Julian Schrittwieser,
David Silver & Nando de Freitas

DeepMind, London, UK
yutianc@google.com

Bayesian Linear Regression

항상 Overfitting 방지? Still in dispute

https://stats.stackexchange.com/questions/265094/is-it-true-that-bayesian-methods-dont-overfit

Conjugate Prior Normal distribution breeds 'normal distribution'

Prior $p(\mathbf{w}) = \mathcal{N}(\mathbf{w}|\mathbf{m}_0, \mathbf{S}_0)$ → Prior가 달라지면?

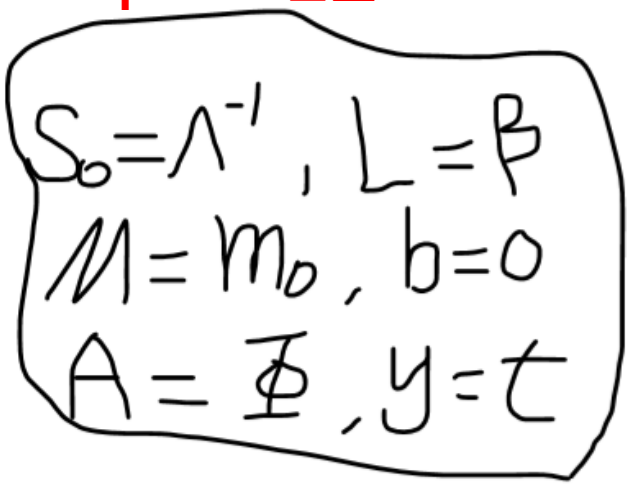
Likelihood $p(\mathbf{t}|\mathbf{X}, \mathbf{w}, \beta) = \prod_{n=1}^{N} \mathcal{N}(t_n|\mathbf{w}^{\mathrm{T}}\boldsymbol{\phi}(\mathbf{x}_n), \beta^{-1})$

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

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

설명이 빠져있는거 같은데?

Chapter 2 참고



Bayesian Linear Regression

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

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

quadratic regularization term, corresponding to (3.27) with $\lambda = \alpha/\beta$.

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

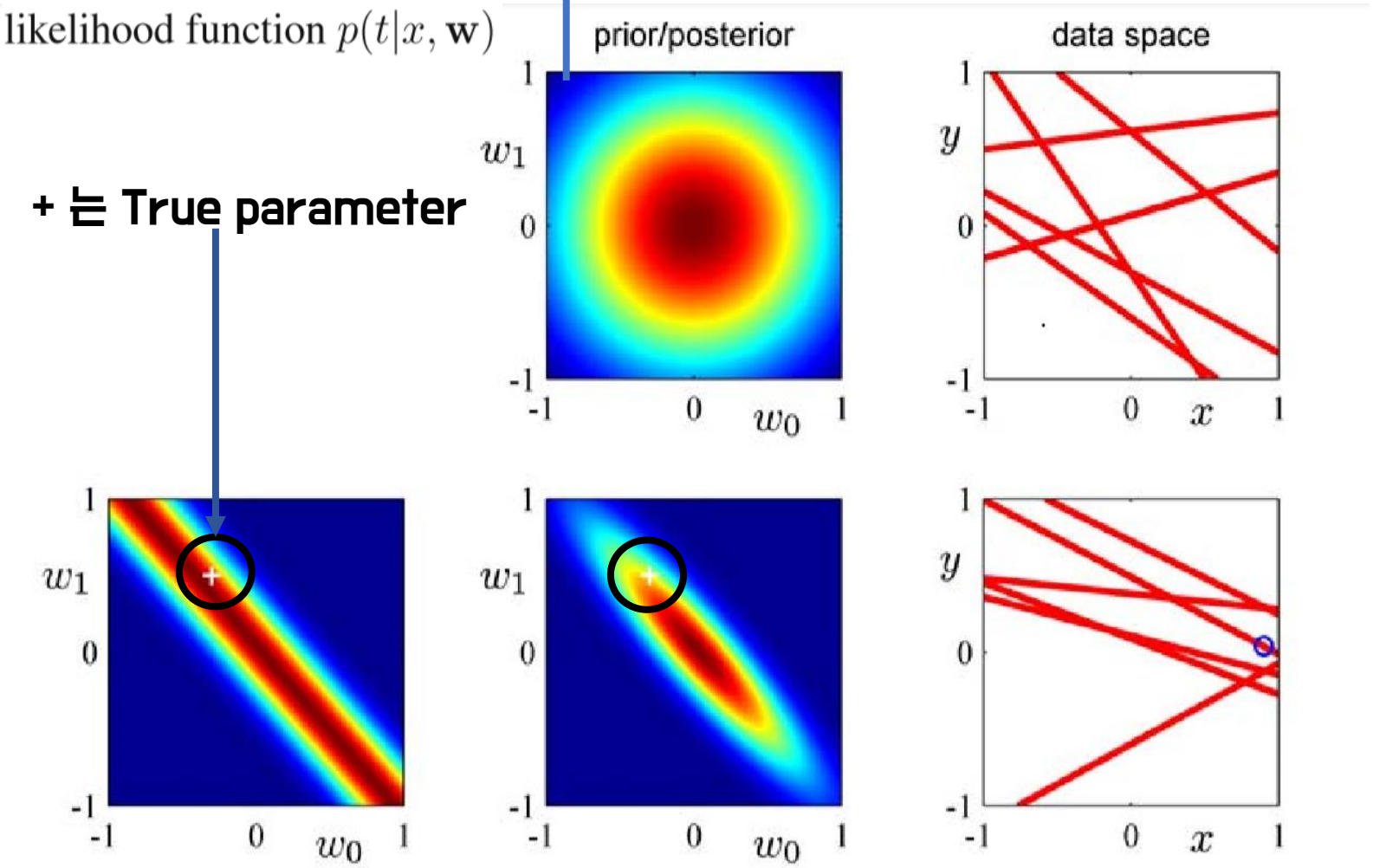
CS231n course note에 다음과 같은 설명이 나오는데 box로 친 부분을 통해 이해하자

A nice feature of this view is that we can now also interpret the regularization term $R(W)$ in the full loss function as coming from a Gaussian prior over the weight matrix W, where instead of MLE we are performing the *Maximum a posteriori* (MAP) estimation. We mention these interpretations to help your intuitions, but the full details of this derivation are beyond the scope of this class.

Bayesian Linear Regression

Sequential Learning1

원인 이유? w1. w2 scaled 된 정규분포



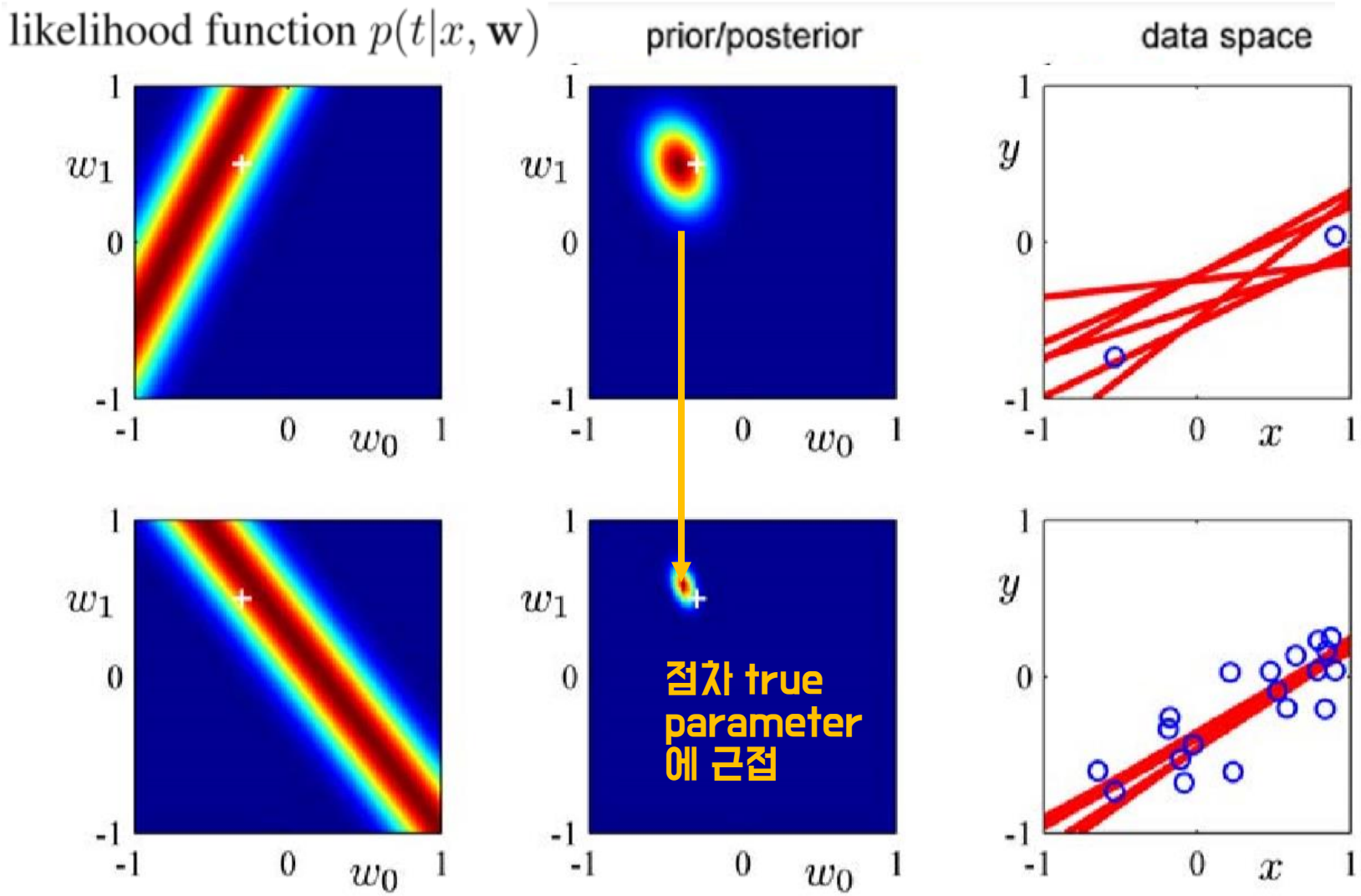
+ 는 True parameter

Data point들 관측되기 전임

한 개의 Data point 관측 후

Bayesian Linear Regression

Sequential Learning2



2개의 Data point들 관측

이전 Posterior는 다음에 Prior로 들어가서 순차적 학습이 진행됨

Sequential learning (20 개 Data point 관측 후 우리가 원하는 회귀 직선에 근접

Precision은 점차 증가 -> Variance 감소

Predictive Distribution

많은 경우 예측분포는 복잡한 형태를 띠고 있어 직접 구하기 어려움

Monte Carlo (Rao-Blackwellisation)

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

$$p(t|\mathbf{x}, \mathbf{t}, \alpha, \beta) = \mathcal{N}(t|\mathbf{m}_N^{\mathrm{T}} \boldsymbol{\phi}(\mathbf{x}), \sigma_N^2(\mathbf{x}))$$

$$\sigma_N^2(\mathbf{x}) = \frac{1}{\beta} + \boldsymbol{\phi}(\mathbf{x})^{\mathrm{T}} \mathbf{S}_N \boldsymbol{\phi}(\mathbf{x}).$$

노이즈 분산

Parameter W에 대한 분산

데이터를 추가할 수록 계속 분산이 줄어든다
-> 예측이 정확해짐

〈Figure 3.8〉

Bayes Factor

Hypothesis Testing → Bayesian Model Comparison

고전적 검정 : 관측되지 않는 값을 포함한 검정통계량에 의존 → 우도원리에 벗어남

여러 개의 다중검정 -> 유의수준 설정의 어려움 (Bonferroni correction)

베이지안 검정 : H0와 H1 구분 X. 검정하고자 하는 가설 수 3개 이상이여도 가능

회귀분석 Model Comparison

Reduced Model VS Full Model

$$H_0: \beta_6 = \beta_7 = 0$$

$$Full: \quad Y_i = \beta_0 + \sum_{j=1}^{7} X_{ij}\beta_j + \varepsilon_i$$

$$Reduced: Y_i = \beta_0 + \sum_{j=1}^{5} X_{ij}\beta_j + \varepsilon_i$$

➡ 설명한 이유? 우리가 한 가설 검정이 결국 Model selection 과정이다!

Bayes Factor

Hypothesis Testing → Bayesian Model Comparison

$$p(\mathcal{D}|\mathcal{M}_i)$$ Model Evidence

Posterior Odds = Bayes Factor X Prior Odds

$$\underbrace{\frac{P(H_0|x)}{P(H_1|x)}}_{\text{Posterior odds}} = \underbrace{\frac{p(x|H_0)}{p(x|H_1)}}_{\text{Bayes factor}} \cdot \underbrace{\frac{P(H_0)}{P(H_1)}}_{\text{Prior odds}}$$

Posterior Odds에서 Prior Odds의 영향을 어느정도 배제한 값

Data가 주는 H1대비 H0에 대한 지지도

The ratio of model evidences

$$p(\mathcal{D}|\mathcal{M}_i) = \int p(\mathcal{D}|\mathbf{w}, \mathcal{M}_i)\boxed{p(\mathbf{w}|\mathcal{M}_i)}\,\mathrm{d}\mathbf{w}$$

Prior에서 랜덤하게 뽑힌 값(W)을 구해 Dataset D를 생성

모델이 복잡할 수록 얻어지는 발생되는 데이터의 변동성이 큼

Bayesian Model Comparison

$$p(\mathcal{D}) = \int p(\mathcal{D}|w)p(w)\,\mathrm{d}w \simeq p(\mathcal{D}|w_{\mathrm{MAP}})\frac{\Delta w_{\mathrm{posterior}}}{\Delta w_{\mathrm{prior}}}$$

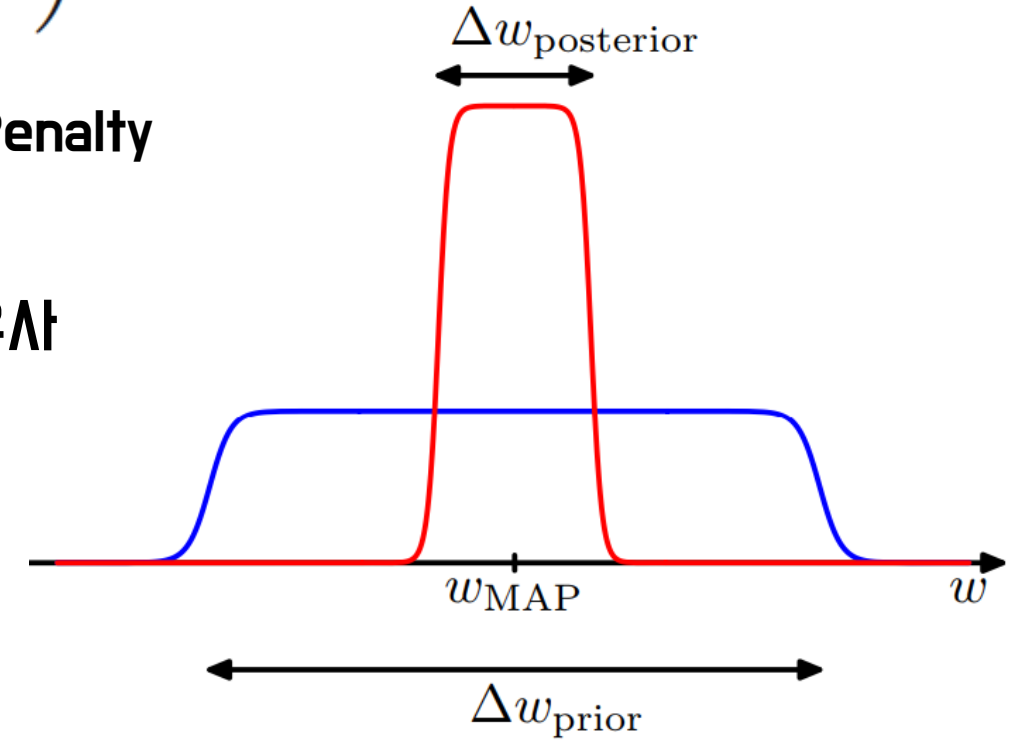
Uniform

$$\ln p(\mathcal{D}) \simeq \ln p(\mathcal{D}|\mathbf{w}_{\mathrm{MAP}}) + M \ln\left(\frac{\Delta w_{\mathrm{posterior}}}{\Delta w_{\mathrm{prior}}}\right)$$

Log - Likelihood

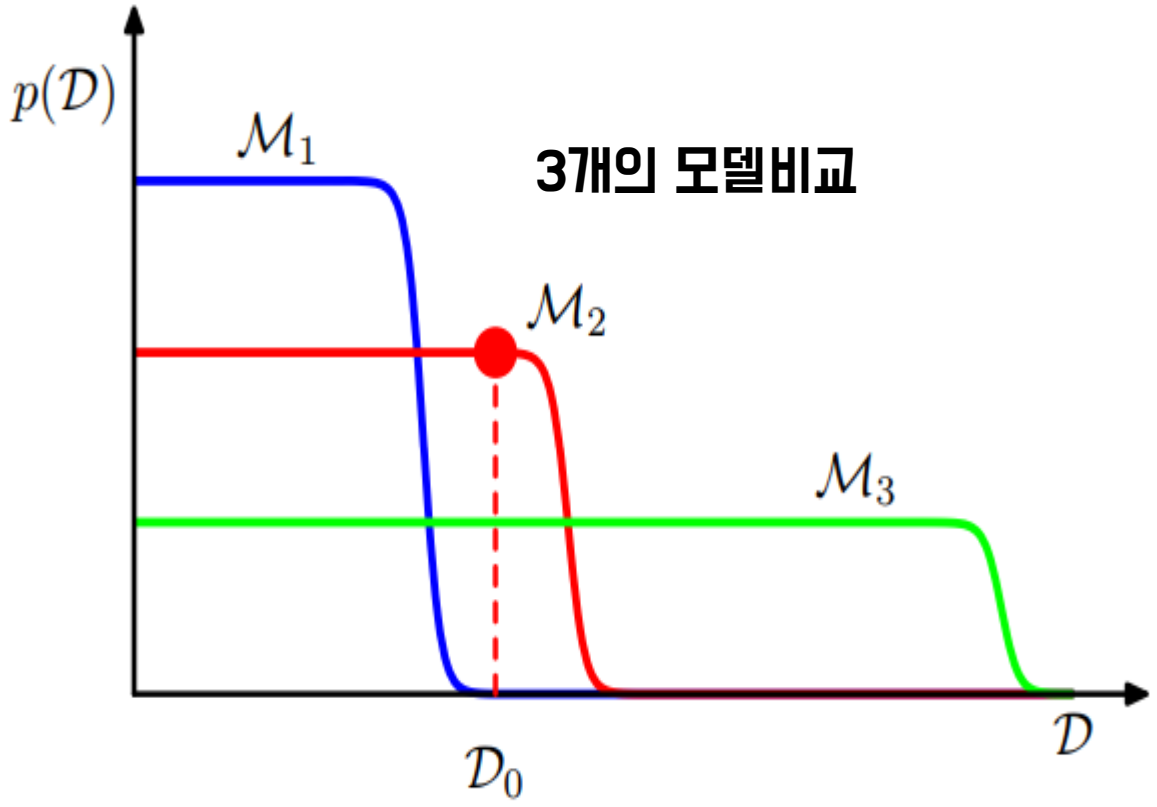
Complexity Penalty

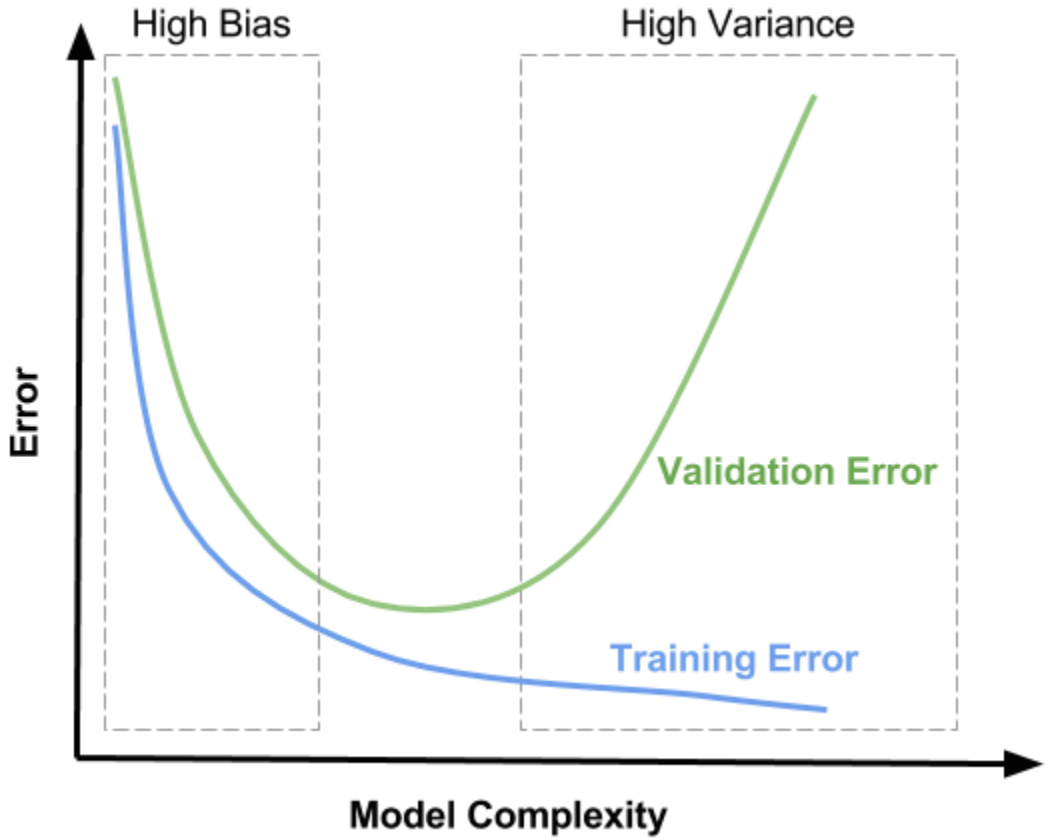
2번째 장의 Model Selection Criteria AIC, BIC, Cp와 유사



Bayesian Model Comparison

Evidence





Evidence Approximation

Marginal likelihood function을 최대화

모수의 모수(Hyperparameters) 추정

$$p(t|\mathbf{t}) = \iiint p(t|\mathbf{w},\beta)p(\mathbf{w}|\mathbf{t},\alpha,\beta)\boxed{p(\alpha,\beta|\mathbf{t})}\,\mathrm{d}\mathbf{w}\,\mathrm{d}\alpha\,\mathrm{d}\beta$$

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

앞의 marginal likelihood 계산하는 과정과 비슷하게 진행

- Derivative
- EM algorithm

Evidence Approximation

Evidence Function

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

앞의 marginal likelihood 계산하는 과정과 비슷하게 진행

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

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

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

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

$$\mathbf{A} = \nabla\nabla E(\mathbf{w}) = \alpha\mathbf{I} + \beta\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi}$$ Likelihood Second Derivate? Fisher Information

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

Evidence Approximation

Eigenvalue Decomposition을 이용한 Hyperparameter 추정

조건? Square Matrix

$$\left(\beta\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{\Phi}\right)\mathbf{u}_i = \lambda_i\mathbf{u}_i$$

Positive definite

λ_i 의 역할? Fisher information measures curvature of the log likelihood.

$$\frac{d}{d\alpha}\ln|\mathbf{A}| = \frac{d}{d\alpha}\ln\prod_i(\lambda_i+\alpha) = \frac{d}{d\alpha}\sum_i\ln(\lambda_i+\alpha) = \sum_i\frac{1}{\lambda_i+\alpha}$$

Linear Algebra!

Eigen values are positive!

Proof

Show that the determinant equals the product of the eigenvalues by imagining that the characteristic polynomial is factored into

$$\det(A-\lambda I) = (\lambda_1-\lambda)(\lambda_2-\lambda)\cdots(\lambda_n-\lambda),$$

and making a clever choice of λ.

Proof. Suppose that $\lambda_1,\ldots,\lambda_n$ are the eigenvalues of the matrix A. Then the λ_i are the roots of the polynomial

$$\det(A-\lambda I),$$

meaning that this polynomial factors as

$$\det(A-\lambda I) = (\lambda_1-\lambda)(\lambda_2-\lambda)\cdots(\lambda_n-\lambda).$$

Letting λ (which we're thinking of purely as an abstract variable) equal zero and simplifying both sides of the above equation, we see that

$$\det A = \lambda_1\lambda_2\cdots\lambda_n,$$

so the determinant of A is equal to the product of the eigenvalues of A. □

Evidence Approximation

Eigenvalue Decomposition을 이용한 Hyperparameter 추정

$$\alpha \mathbf{m}_N^{\mathrm{T}} \mathbf{m}_N = M - \alpha \sum_i \frac{1}{\lambda_i + \alpha} = \gamma.$$

$$0 \leqslant \sum_i \frac{1}{\lambda_i + \alpha} \leqslant 1$$

$$0 \leqslant \gamma = \sum_i \frac{\lambda_i}{\alpha + \lambda_i} \leqslant M$$

유효한 파라미터수 측정

$$\alpha = \frac{\gamma}{\mathbf{m}_N^{\mathrm{T}} \mathbf{m}_N}$$

Alpha와 Gamma를 Iterative하게 수렴할 때 까지 추정 EM Algorithm 처럼

Beta도 이와 유사하게 고유값 분해 후 식 정리

$$\frac{d}{d\beta} \ln |\mathbf{A}| = \frac{d}{d\beta} \sum_i \ln(\lambda_i + \alpha) = \frac{1}{\beta} \sum_i \frac{\lambda_i}{\lambda_i + \alpha} = \frac{\gamma}{\beta}$$

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

Evidence Approximation

Eigenvalue Decomposition을 이용한 Hyperparameter 추정

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

Degree of Freedom

우리가 구한 회귀직선의 Parameter가 γ 개

Q) How come we substitutes γ ?

Maximum likelihood result에 Bias correction을 해주기 위함

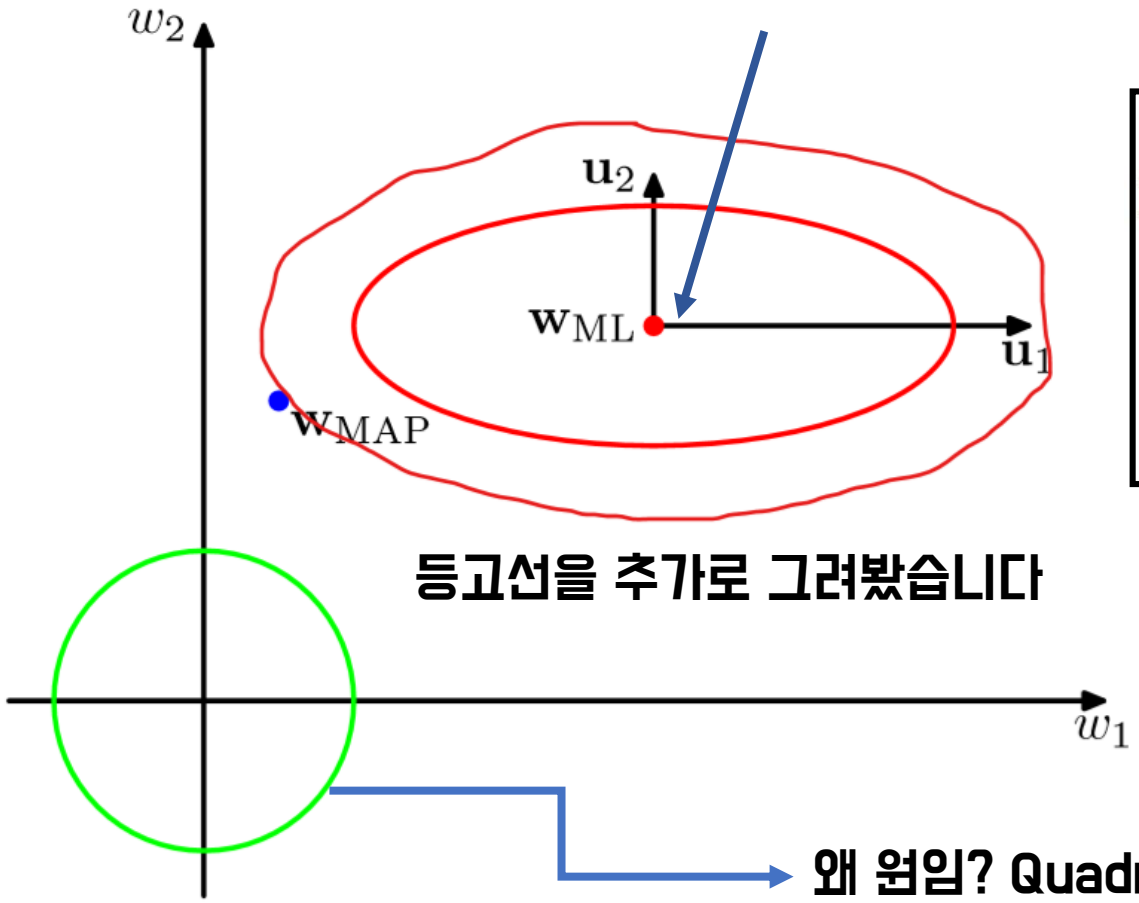
N개 sample의 표준편차에서 자유도가 N-1인 경우처럼

Unbiased estimator로 만들어주려고!

Evidence Approximation

아까 shrinkage method에서 봤던 그림

이 경우는 Prior. 즉 α = 0 일 때의 maximum likelihood solution



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

Posterior를 최대화 하는 게(MAP)
결국 선형회귀에 penalty를 준 효과

등고선을 추가로 그려봤습니다

왜 원임? Quadratic penalised term 이므로 (Ridge)

Reference

참고한 책들

1) Pattern Recognition and Machine Learning(Christopher Bishop 저)
2) Introduction to Statistical Learning (Trevor Hastie외 3명 저)
3) Probability and Statistical Inference(Hogg외 2명 저)
4) Machine Learning (Kevin Murphy 저)
5) Regression Analysis by Example (Samprit Chatterjee 외 1명 저)
6) 베이지안 통계추론 (오만숙 저)
7) 통계수학 강의 (김진경 외 2명 저)

참고링크

https://www.math.colostate.edu/~clayton/teaching/m215s10/homework/hw9solutions.pdf

http://www.stat.cmu.edu/~larry/

https://blog.naver.com/sw4r/221010519300