
Probabilistic Deep Learning:

1. Curve Fitting, Decision Theory, & Information Theory with Probability

Contents

1. Probability on ML
2. Probabilistic & Non-probabilistic Curve Fitting
3. Decision Theory
4. Information Theory

1

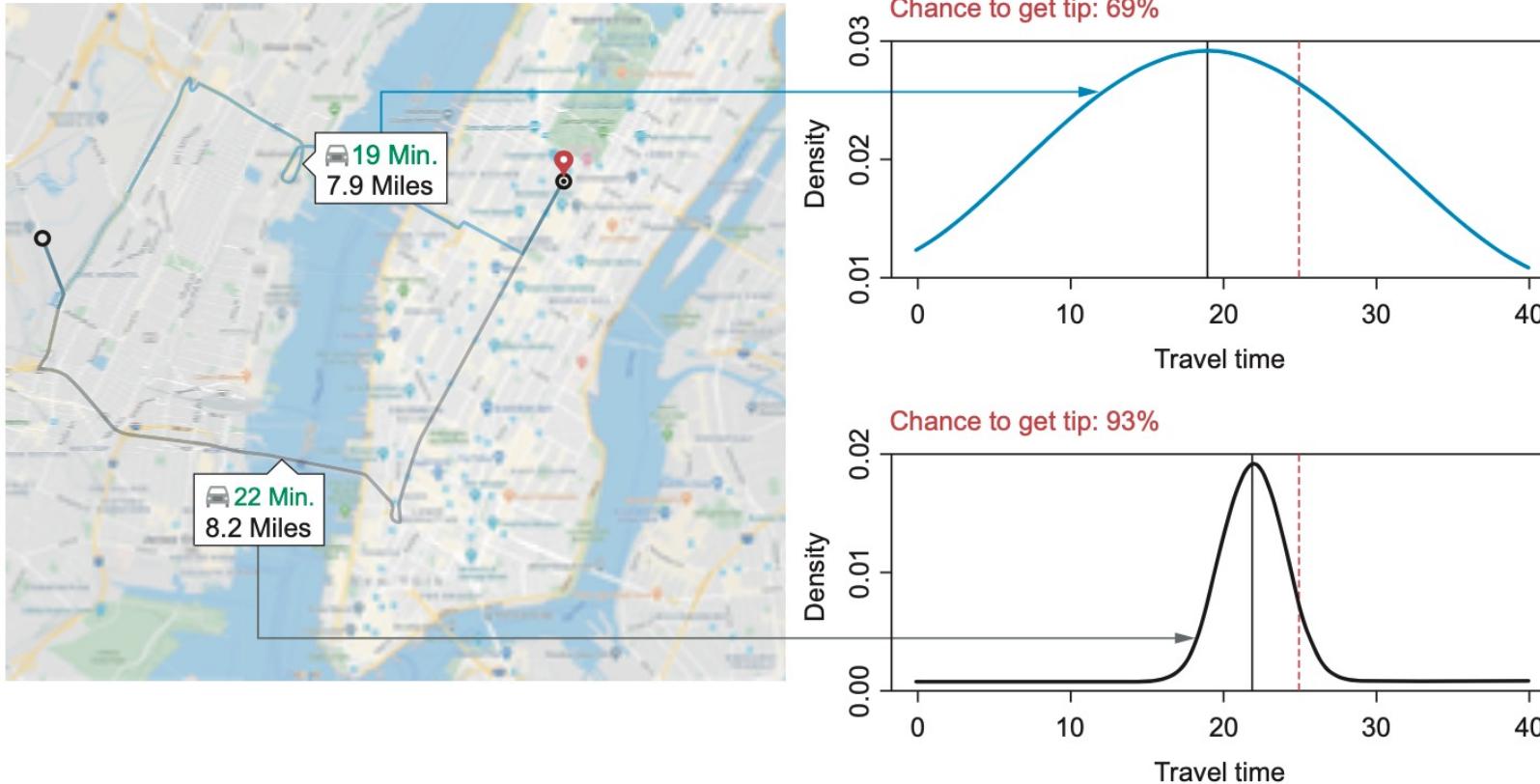
Probability on ML

Introduction to Probabilistic Deep Learning

Probability on Machine Learning

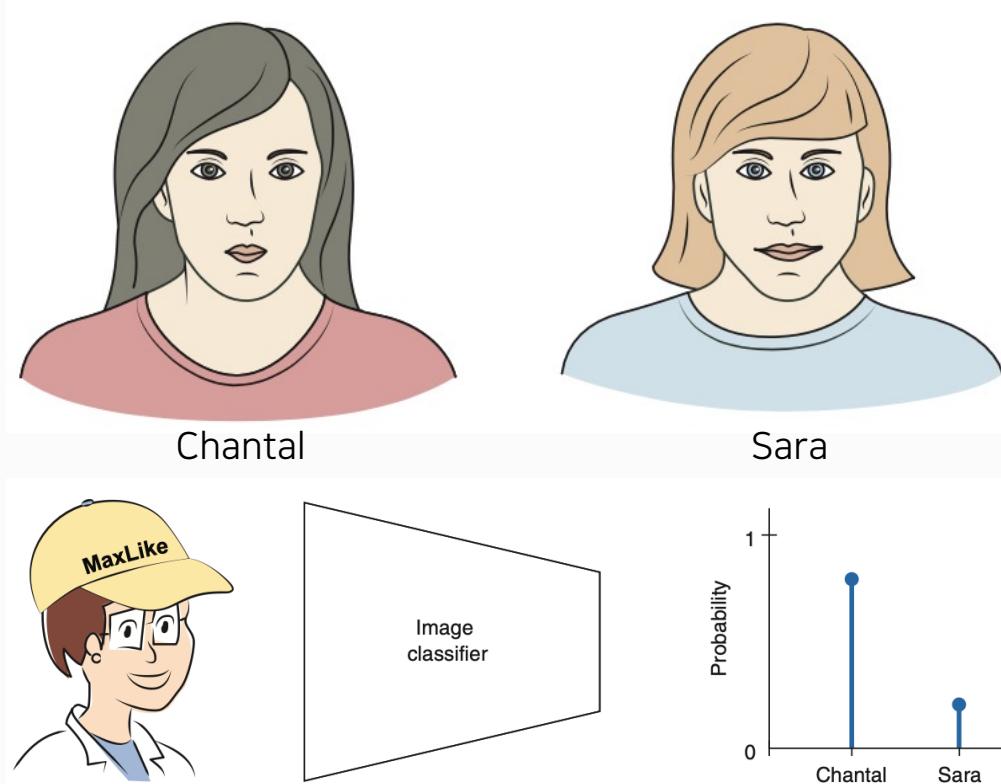


Probability on Machine Learning



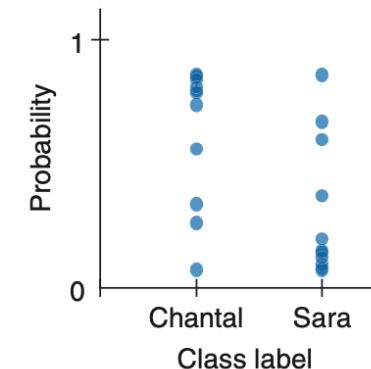
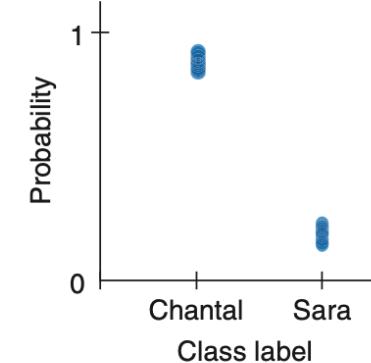
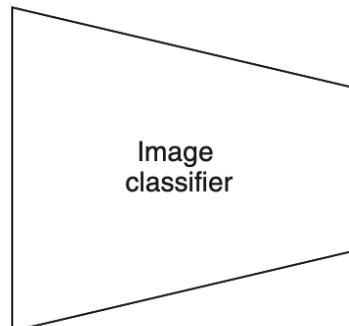
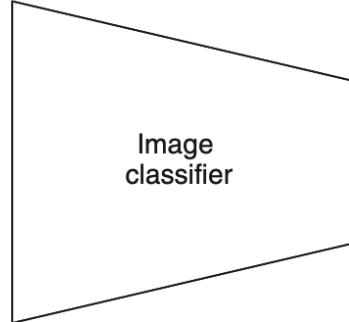
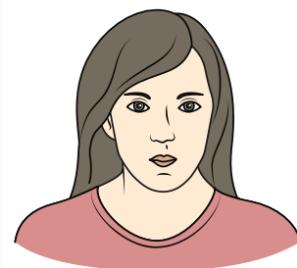
- 좌측 Figure는 deterministic view로 결과를 예측한 것으로서, 단일한 예측값밖에 보여주지 않는다.
- 하지만 교통 상황 등 다른 요인에 따라 도착 시간은 바뀔 수 있다. 우리가 예상 도착시간의 분포를 알 수 있다면 네비게이션 상에서 22분이 걸리는 경로를 선택하는 것이 팁을 받을 확률을 높이는 방법임을 쉽게 파악할 수 있다.

Probability on Machine Learning



- Chantal과 Sara를 구분하는 Binary Classifier가 있다.
- 고전적인 ML 분류 모델들 – Logistic Regression, SVD, DL 등 – 은 단일한 결과물을 반환하므로 훈련되지 않았던 데이터가 입력되었을 때 훈련 데이터를 벗어난 결과물을 내놓지 못한다.

Probability on Machine Learning



- 위 figure는 Bayesian probabilistic classifier의 classification 과정을 보여준다.
- 통계학에서의 Bayesian view는 uncertainty를 표현할 수 있게 해준다.
(아래 패널에서 모델이 알지 못하는 데이터가 들어오자 확률값이 요동치는 모습을 보여준다.)
- 즉, 우리는 각종 모델들에 Bayesian view를 도입함으로써 불확실성을 처리할 수 있는 것이다.

2

Probabilistic & Non-probabilistic Curve Fitting

Curve Fitting (Regression), Regularization, Probability

Non-probabilistic Curve Fitting

- $\mathbf{x} = (x_1, \dots, x_N)^T$ 를 input dataset, $\mathbf{t} = (t_1, \dots, t_N)^T$ 를 이에 대응되는 target dataset이라 하자.

- 우리는 \mathbf{x} 를 잘 표현해줄 수 있는 curve 를 찾고, 그 모델을 바탕으로 새로운 input variable \hat{x} 에 대응되는 target variable \hat{t} 를 예측하고자 한다.

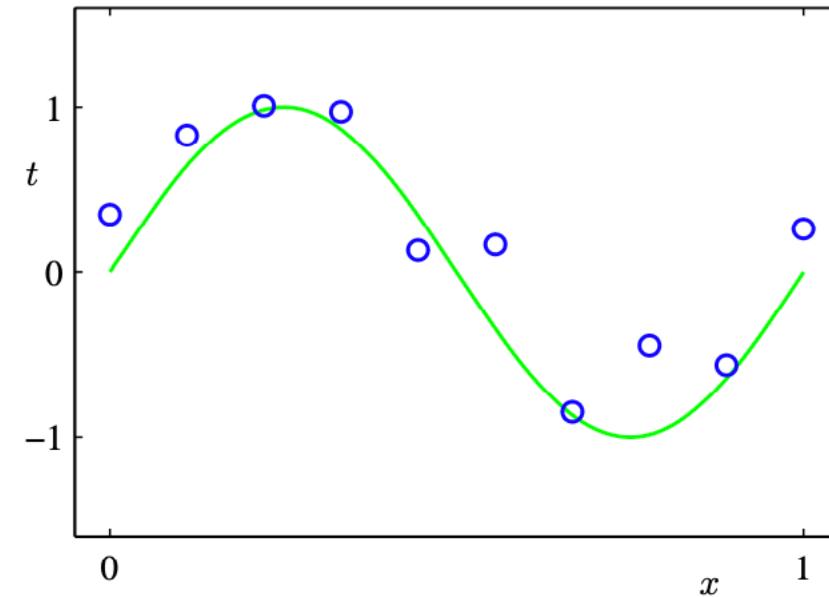
- 다음과 같은 M 차 다항회귀식을 모델로 결정하였다 가정하자:

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

- 이제 loss function 을 설정하고 이를 최소화하는 \mathbf{w} 를 구해보자. 다양한 함수가 있지만 가장 널리 쓰이는 것은 SSE 이다:

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

($1/2$ 은 계산의 편의성을 위해 붙여졌다.)

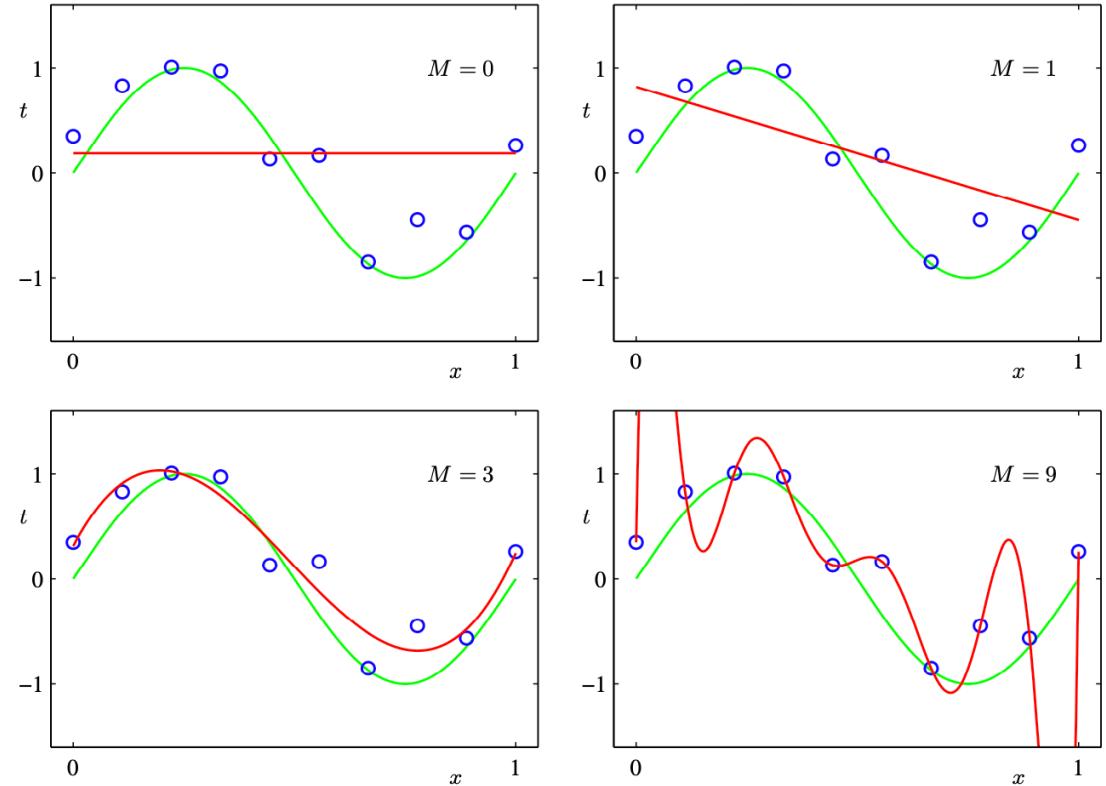


Non-probabilistic Curve Fitting

- 그렇다면 M 은 어떻게 결정할까?
- $\mathbf{w}^* = \operatorname{argmin}_{\mathbf{w}} E(\mathbf{w})$ 로 설정하자. 이제 $E(\mathbf{w}^*)$ 의 residual value를 평가하기 위해 새로운 statistic, root-mean-square (RMS) error를 도입하자.

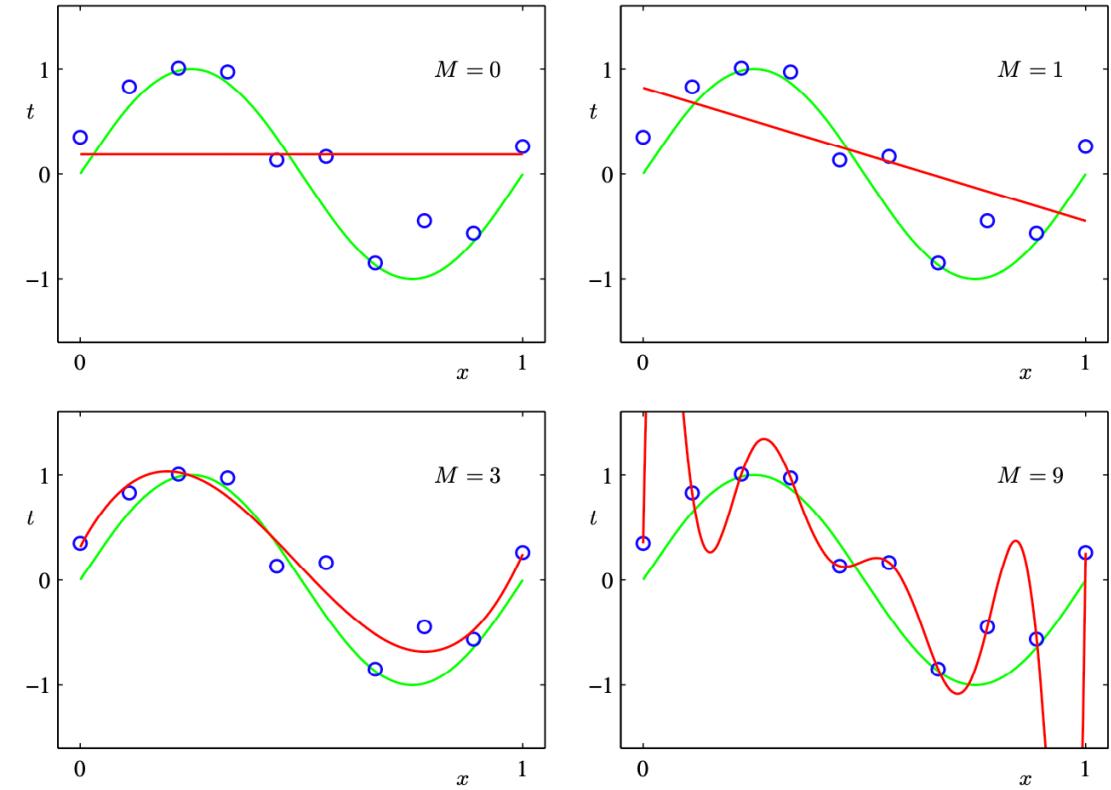
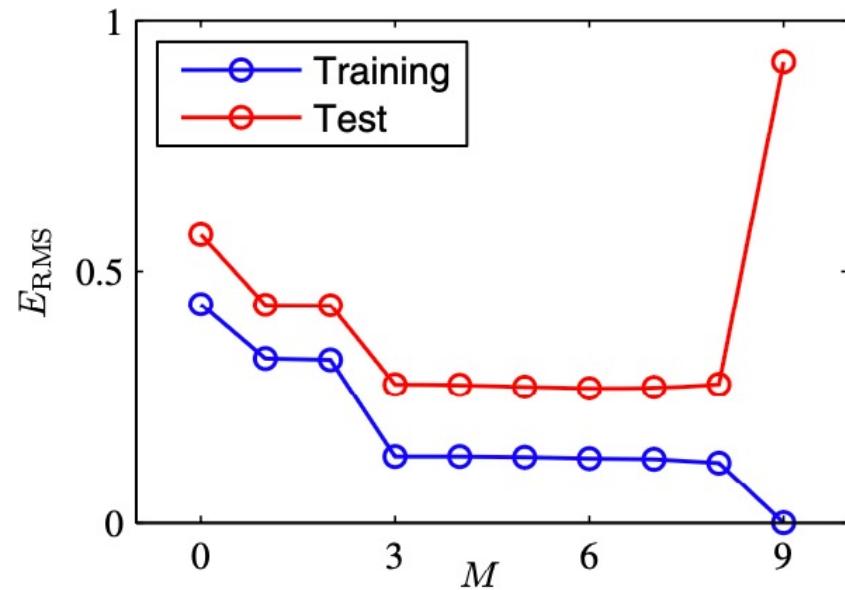
$$E_{RMS} = \sqrt{E(\mathbf{w}^*)/N}$$

(데이터의 개수 N 으로 나누는 것은 서로 다른 개수를 가진 데이터도 비교할 수 있게 해주며, $\sqrt{}$ 는 error가 타겟 변수 t 와 동일한 스케일에서 측정될 수 있게 해준다.)



Non-probabilistic Curve Fitting

- 이제 특정한 M 에서 RMS값이 폭증하는 것을 확인할 수 있는데, 이는 곧 모델이 데이터에 overfitting되었다는 의미이다.



Non-probabilistic Curve Fitting

- 오버피팅을 막는 방법에는 무엇이 있을까?

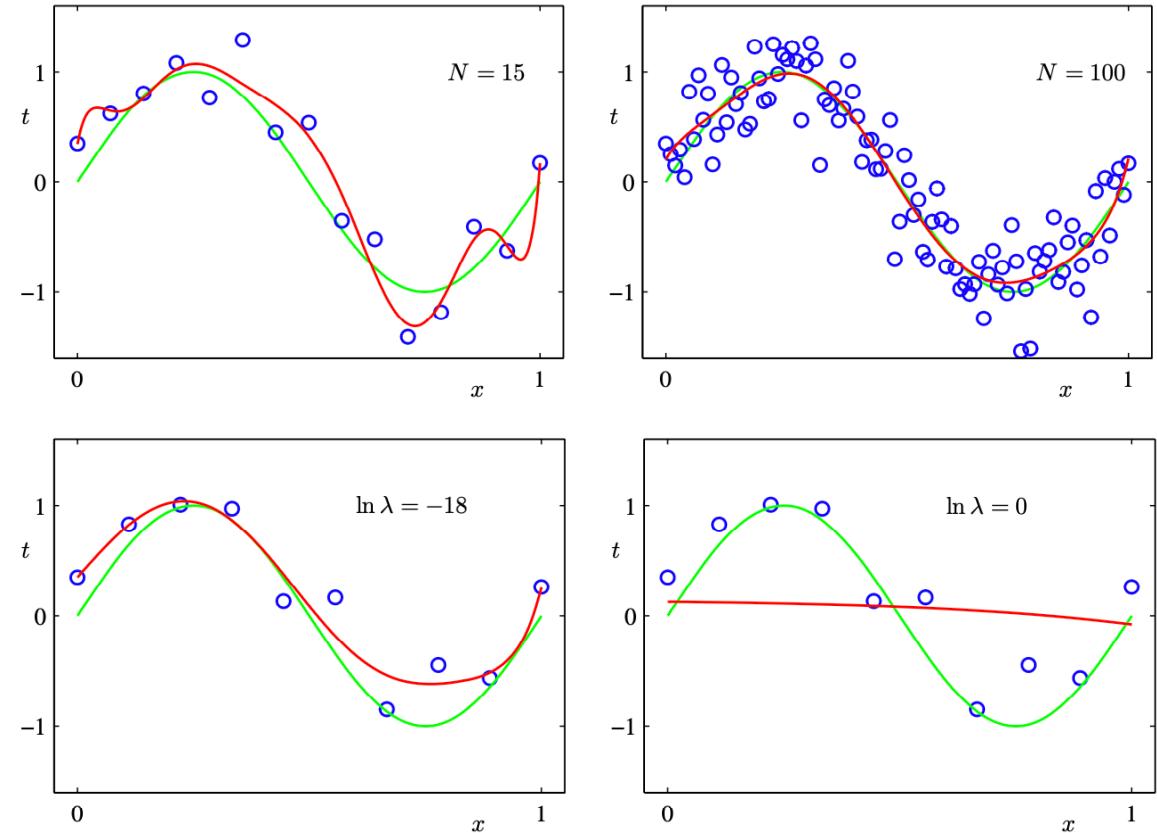
모델의 오버피팅을 막기 위해 Training set 데이터의 개수를 늘리거나 Regularization을 이용할 수 있다.

- Regularization은 penalty term을 추가함으로써 계수가 너무 커지는 것을 방지해준다.

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

where $\|\mathbf{w}\|^2 = \mathbf{w}^T \mathbf{w} = w_0^2 + \dots + w_M^2$.

- 계수 λ 는 SSE term에 대비한 regularization term의 영향을 결정하며, w_0 은 생략 가능. (Since its inclusion causes the results to depend on the choice of origin for the target variable.)



Non-probabilistic Curve Fitting

- Ridge Regression (L2 Regression)

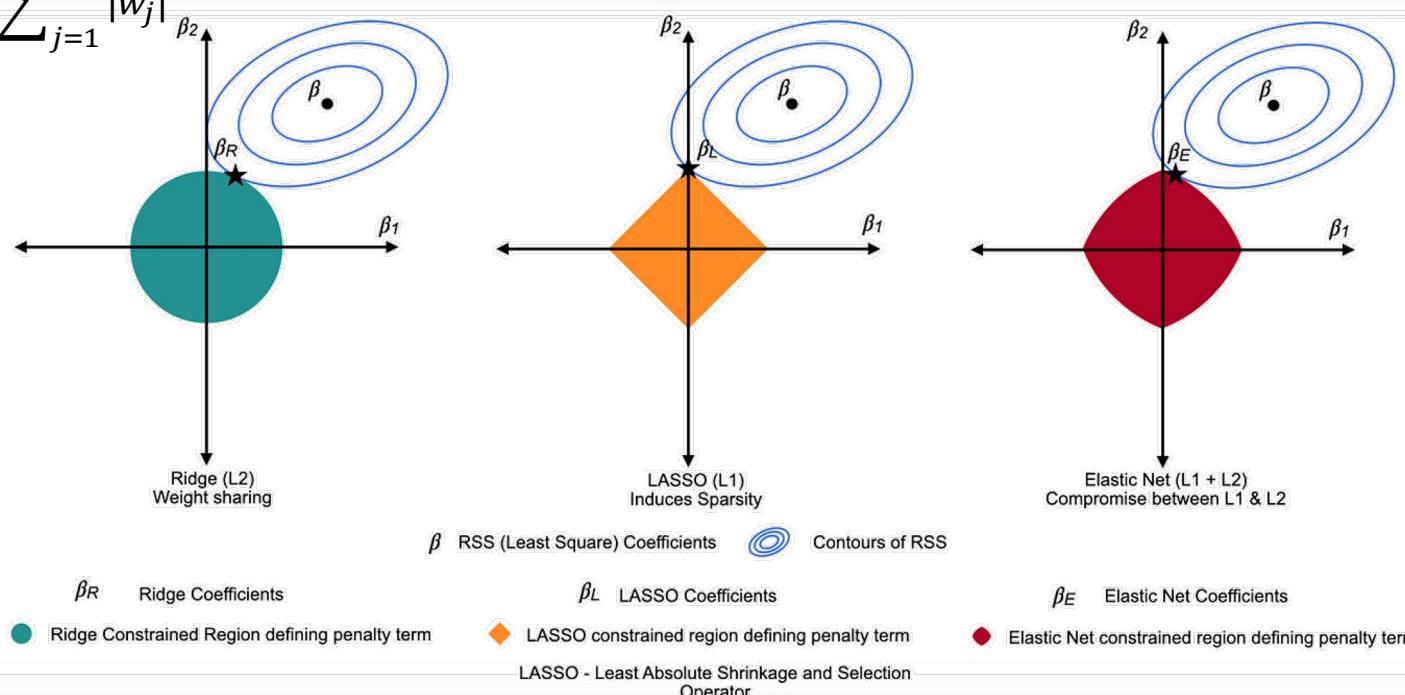
$$\sum_{i=1}^n \left(y_i - w_0 - \sum_{j=1}^p w_j x_{ij} \right)^2 + \lambda \sum_{j=1}^p w_j^2 = SSE + \lambda \sum_{j=1}^p w_j^2$$

- Lasso Regression (L1 Regression):

$$\sum_{i=1}^n \left(y_i - w_0 - \sum_{j=1}^p w_j x_{ij} \right)^2 + \lambda \sum_{j=1}^p |w_j| = SSE + \lambda \sum_{j=1}^p |w_j|$$

- Elastic Net (Lasso + Ridge)

$$\hat{\beta} = \operatorname{argmin}_{\beta} (\|y - Xw\|^2 + \lambda_2 \|w\|^2 + \lambda_1 \|w\|_1)$$



Probabilistic Curve Fitting

- 우리가 데이터에 적합한 Curve 모형을 만드는 이유는 새로운 input variable이 주어졌을 때 이에 대응되는 target variable을 예측하기 위해서이다. 이때 우리는 확률 분포를 사용하여 target variable 값에 대한 불확실성을 표현할 수 있다.

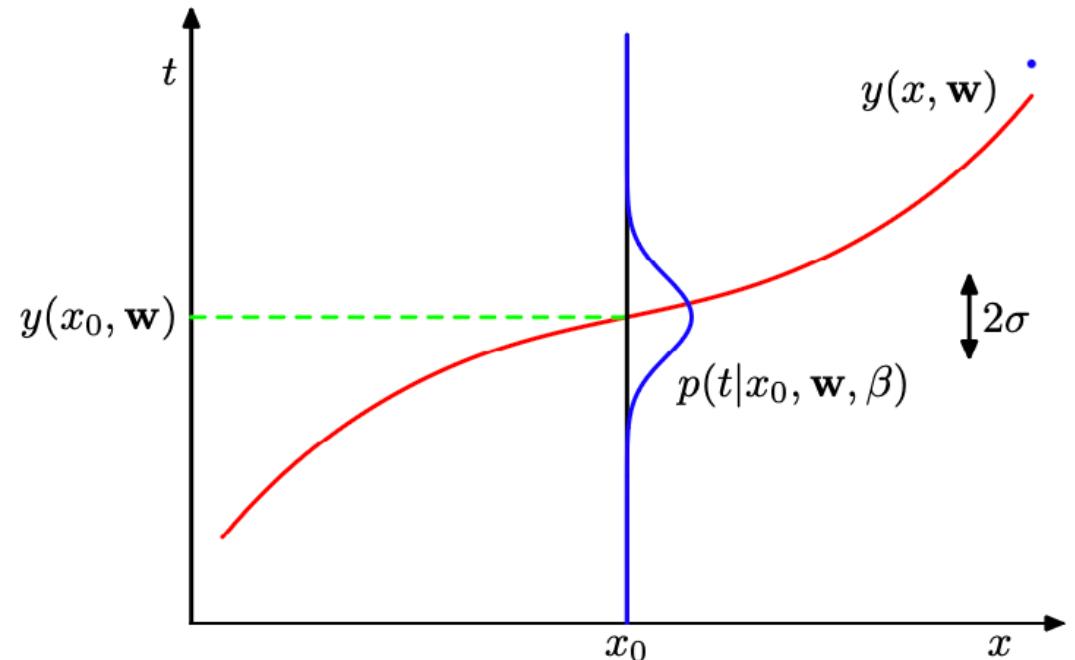
- Target variable t 가 $y(x, \mathbf{w})$ 와 동일한 평균값을 가진 정규분포를 따른다고 하자. 이때 우리는 다음과 같은 확률 분포를 얻을 수 있다:

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

- 이제 training data $\{\mathbf{x}, \mathbf{t}\}$ 로부터 모수 \mathbf{w}, β 를 찾아보자. 각 데이터가 위 정규분포를 따르는 iid라고 가정하면, likelihood function은 다음과 같다:

$$p(\mathbf{t}|\mathbf{x}, \mathbf{w}, \beta) = \prod_{n=1}^N N(t_n|y(x_n, \mathbf{w}), \beta^{-1}).$$

- $\beta = 1/\sigma^2$ 은 *precision*이라 불린다. 따라서 위 정규분포는 분산이 σ^2 인 일반적인 가정과 동일하다.



Probabilistic Curve Fitting

- c.f. 정규분포의 pdf는 다음과 같다:

$$N(x|\mu, \sigma^2) \sim \frac{1}{(2\pi\sigma^2)^{1/2}} \exp\left\{-\frac{(x-\mu)^2}{2\sigma^2}\right\}.$$

- 이제 $p(\mathbf{t}|\mathbf{x}, \mathbf{w}, \beta) = \prod_{n=1}^N N(t_n|y(x_n, \mathbf{w}), \beta^{-1})$ 는

$$\prod_{n=1}^N \frac{1}{(2\pi)^{1/2} \beta^{1/2}} \exp\left\{-\frac{\beta}{2} (y(x_n, \mathbf{w}) - t_n)^2\right\}$$

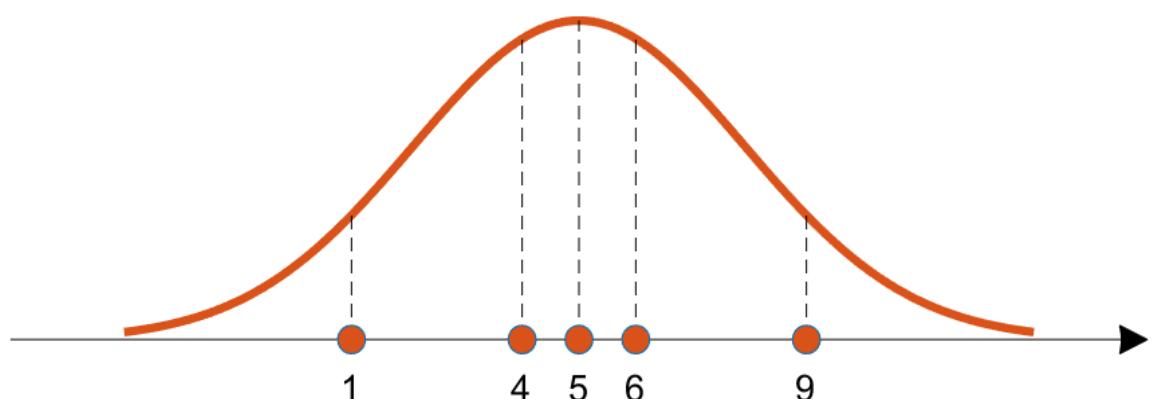
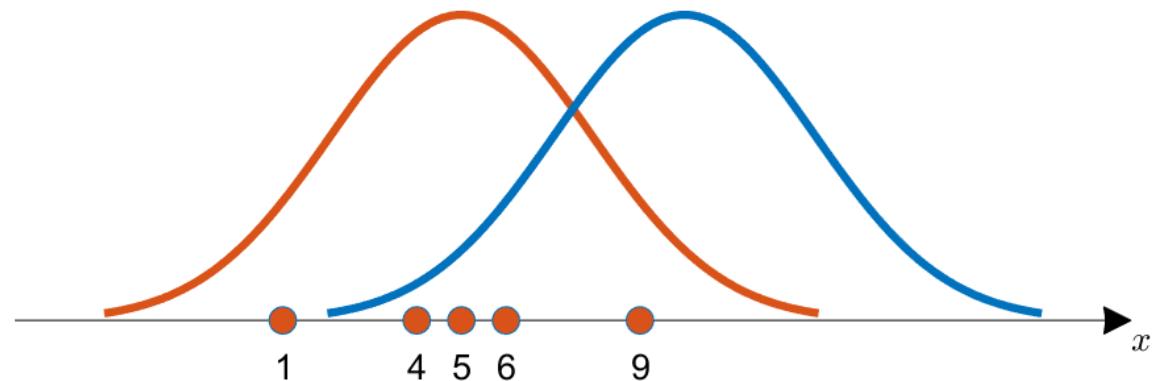
로 표현될 수 있으므로, log-likelihood는 아래와 같다:

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

- 위 식에서 마지막 두 항은 \mathbf{w} 에 영향을 주지 않으므로 생략할 수 있다.
또 β 가 fixed value라고 가정하자. 결국 Frequentist View로 찾아낸 \mathbf{w} 의
MLE는 ML View에서 squared error function을 최소화하는 것과 같다.

$$\text{maximizing } \ln p(\mathbf{t}|\mathbf{x}, \mathbf{w}, \beta) \Leftrightarrow \text{minimizing } \frac{1}{2} \sum_{n=1}^N \{y(x_n, \mathbf{w}) - t_n\}^2$$

\Leftrightarrow minimizing the *sum-of-squares error function*



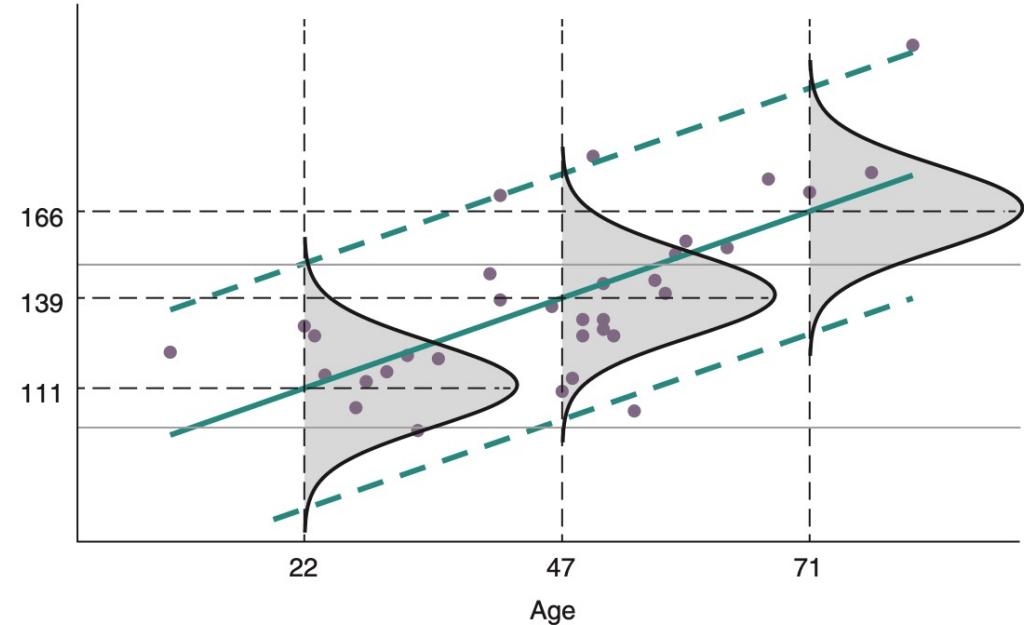
Probabilistic Curve Fitting

- 마찬가지로 β 의 MLE를 찾아 주기 위해 log-likelihood function을 β 에 대해 미분해주자.

$$\begin{aligned} \text{maximizing } \ln p(\mathbf{t}|\mathbf{x}, \mathbf{w}, \beta) &\Leftrightarrow \frac{d}{d\beta} \left(\beta \sum_{n=1}^N \{y(x_n, \mathbf{w}) - t_n\}^2 - N \ln \beta \right) = 0 \\ &\Leftrightarrow \frac{1}{\beta_{ML}} = \frac{1}{N} \sum_{n=1}^N \{y(x_n, \mathbf{w}_{ML}) - t_n\}^2 \end{aligned}$$

- 이제 우리는 MLE를 바탕으로 모수 \mathbf{w} 와 β 를 찾아주었다. 다음과 같은 t 에 대한 predictive distribution을 통해 새로운 input variable x_0 에 대응되는 target variable t_0 의 point estimator가 아니라, 분포까지 찾을 수 있게 되었다:

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



Bayesian Curve Fitting

- $D = \{t_1, \dots, t_N\}$ 이 관측된 데이터, \mathbf{w} 가 다항회귀모델에서 계수들이라 하자. 우리는 데이터를 관측하기 이전에 \mathbf{w} 에 대한 어떠한 가정을 prior probability distribution $p(\mathbf{w})$ 의 형태로 가지고 있다.
- 관측된 데이터 D 의 영향은 조건부 확률인 $p(D|\mathbf{w})$ 의 형태로 나타낼 수 있으며, 아래와 같은 Bayes Theorem에 기반한 식을 만족시킨다:

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

- 위와 같은 식은 우리가 $p(\mathbf{w}|D)$ 의 형태로 D 를 관측된 이후의 \mathbf{w} 에 대한 uncertainty를 측정할 수 있게 해준다.
- 위의 Bayes theorem으로부터 우리는

$$\text{posterior} \propto \text{likelihood} \times \text{prior}$$

와 같은 비례식을 얻을 수 있다. 여기서 주목해야 할 것은 likelihood와 prior 모두 \mathbf{w} 에 대한 함수이며, 분모 $p(D)$ 는 아래와 같이 likelihood와 prior를 통해 구해질 수 있는 normalization constant로 해석될 수 있다.

1. Posterior Probability Distribution ($p(\mathbf{w}|\mathcal{D})$):

- **Definition:** This is the updated probability distribution representing our beliefs about the model parameters (\mathbf{w}) after taking into account both prior information and the observed data.
- **In the Given Text:** The posterior probability $p(\mathbf{w}|\mathcal{D})$ is calculated using Bayes' theorem and allows us to evaluate the uncertainty in \mathbf{w} after observing the data.

2. Likelihood ($p(\mathcal{D}|\mathbf{w})$):

- **Definition:** This is the probability of observing the data (\mathcal{D}) given a specific set of parameter values (\mathbf{w}).
- **In the Given Text:** The effect of the observed data $\mathcal{D} = \{t_1, \dots, t_N\}$ is expressed through the conditional probability $p(\mathcal{D}|\mathbf{w})$, which describes how well the model, with parameters \mathbf{w} , explains or predicts the observed data.

3. Prior Probability Distribution ($p(\mathbf{w})$):

- **Definition:** This is the probability distribution representing our beliefs or assumptions about the model parameters (\mathbf{w}) before observing any data.
- **In the Given Text:** The assumptions about \mathbf{w} are captured in the form of a prior probability distribution $p(\mathbf{w})$.

Bayesian Curve Fitting

- 앞선 Bayes theorem으로부터 우리는
$$\text{posterior} \propto \text{likelihood} \times \text{prior}$$

와 같은 비례식을 얻을 수 있다. 여기서 주목해야 할 것은 likelihood와 prior 모두 \mathbf{w} 에 대한 함수이며, 분모 $p(D)$ 는 아래와 같이 likelihood와 prior를 통해 구해질 수 있는 normalization constant로 해석될 수 있다:

$$p(D) = \int p(D|\mathbf{w})p(\mathbf{w}) d\mathbf{w}.$$

1. Frequentist

- \mathbf{w} 는 고정된 parameter로서, 그 값은 'estimator'라는 형태로 결정된다.
- 추정과 관련된 uncertainty는 \mathbf{w} 의 확률분포를 통해 표현될 수 있음.

2. Bayesian

- 고정된 것은 실제로 관측된 데이터 셋인 D 뿐임.
- parameter들의 uncertainty는 \mathbf{w} 의 확률분포를 통해 표현될 수 있음.

* Frequentist vs. Bayesian approach

- 동전을 3번 던졌더니 모두 앞면이 나왔다고 하자.

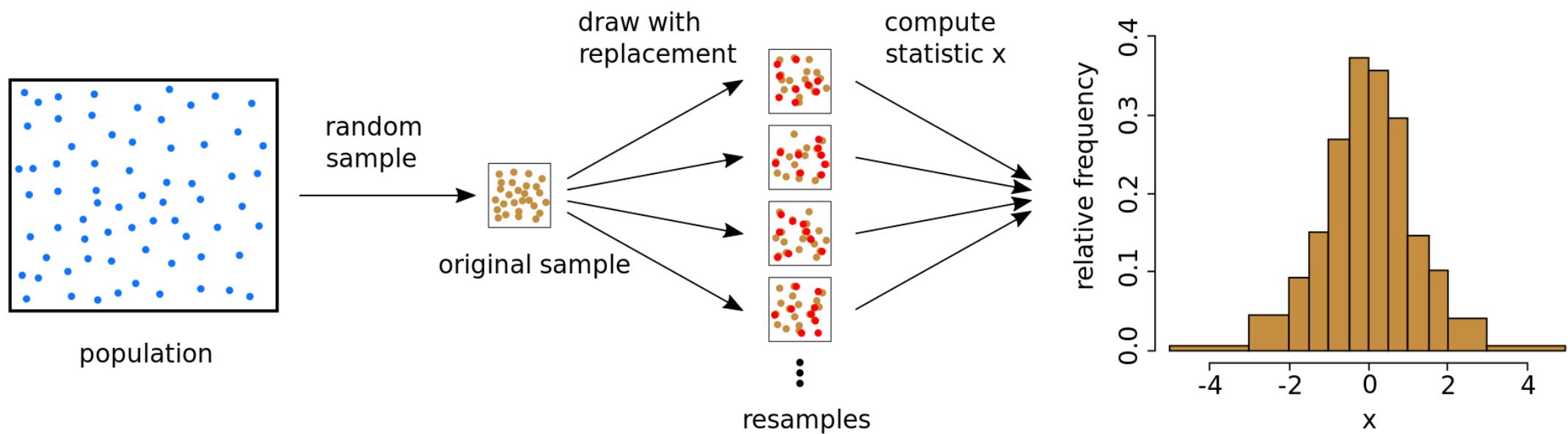


- 동전을 던지는 사건은 명백히 Bernoulli 분포를 따른다. 사건의 분포를 안다면 $Bernoulli(p)$ 의 모수를 추정하기 위해 MLE를 사용할 수 있다.

$$\log \prod_{i=1}^3 p^1 (1-p)^0 = \log p^3 = 3 \log p \equiv 0$$

Frequentist는 위 상황에서 p 의 MLE $\hat{p} = 1$ 로 추정할 것이다. 따라서 앞으로 던지는 동전은 모두 앞면이 나올 것이다. 하지만 이는 합리적인 추론이 아니다. 하지만 어떠한 합리적인 prior를 가진 Bayesian approach는 위와 같은 극단적인 추정을 내리지는 않을 것이다.

Bayesian Curve Fitting



- 그 밖의 자주 사용되는 추정 방법으로는 Bootstrap이 있으며, Frequentist와 Bayesian 방법이 모두 존재.

Bayesian Curve Fitting

- 계수 \mathbf{w} 가 α 라는 precision 값을 가지고 있다고 하자. Prior는 다음과 같다: ($M + 1$ 은 M 차 다항회귀 모형의 계수인 \mathbf{w} 의 성분 수를 의미한다.)

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

- Bayes' theorem을 통해 우리는 다음과 같은 비례 관계를 얻을 수 있다:

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

- Posterior distribution을 극대화시키는 가장 가능성성이 높은 \mathbf{w} 의 값을 찾아보자. 이러한 방법은 *maximum posterior*, 혹은 MAP라 불린다. 위 비례식에서 \mathbf{w}_{MAP} 를 찾기 위해 posterior와 비례하는 likelihood와 prior의 곱이 최대가 될 때를 찾으면 된다. log를 취함으로써 우리는 다음을 얻는다:

$$\log\{p(\mathbf{w}|\mathbf{x}, \mathbf{t}, \alpha, \beta)\} \propto -\frac{\beta}{2}\sum_{n=1}^N \{y(x_n, \mathbf{w}) - t_n\}^2 - \frac{\alpha}{2}\mathbf{w}^T\mathbf{w}.$$

- 즉, \mathbf{w}_{MAP} 를 찾는 것은 $\frac{\beta}{2}\sum_{n=1}^N \{y(x_n, \mathbf{w}) - t_n\}^2 + \frac{\alpha}{2}\mathbf{w}^T\mathbf{w}$ 를 최소화시키는 \mathbf{w} 를 찾는 것과 동치이며, 이는 앞서 살펴본 Regularized loss function (Ridge regression)과 동치이다 ($\lambda = \alpha/\beta$ 로 생각):

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

- 따라서 $\mathbf{w}_{MAP} = (\lambda\mathbf{I} + \mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$ 이다. ($\frac{\partial}{\partial\mathbf{w}}\tilde{E}(\mathbf{w}) \equiv 0$ 이 되게 하는 \mathbf{w} 를 잘 찾아보자.)
- 결론적으로 알 수 있는 것은 ML의 관점에서 loss function을 최소화하는 \mathbf{w} 를 찾나, Frequentist의 관점에서 \mathbf{w}_{MLE} 를 찾나, Bayesian의 관점에서 \mathbf{w}_{MAP} 를 찾나 모두 결과는 비슷하다는 것이다. 다만 \mathbf{w}_{MAP} 는 regularized일 때의 결과를 보여주므로 overfitting 방지 측면에서 더욱 유리할 것이다.

Bayesian Curve Fitting

- 앞서 Curve fitting의 목적은 training data \mathbf{x} 와 \mathbf{t} , 새로운 test point x 가 주어졌을 때, 이에 대응되는 t 를 예측하는 것이었다. 우리는 따라서 predictive distribution인 $p(t|x, \mathbf{x}, \mathbf{t})$ 를 찾고 싶어한다. (α, β are assumed to be known and fixed.)
- 앞선 Bayes' thm.으로부터, 편의성을 위해 α, β 를 생략한다면 우리는 다음과을 얻는다:

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

여기서 $p(\mathbf{w}|\mathbf{x}, \mathbf{t})$ 는 모수에 대한 posterior이고, likelihood와 prior의 곱을 normalizing 하여 얻을 수 있다. 이 posterior는 정규분포를 따른다. (왜 정규분포를 따른는지, 어떻게 normalizing 하는지는 3주차 세션에서 살펴볼 예정)

Bayesian Curve Fitting

- 결론적으로 우리는 다음과 같은 predictive distribution을 얻을 수 있다:

$$p(t|x, \mathbf{x}, \mathbf{t}) \sim N(t|m(x), s^2(x))$$

where

$$m(x) = \beta \boldsymbol{\phi}(x)^T \mathbf{S} \sum_{n=1}^N \boldsymbol{\phi}(x_n) t_n,$$

$$s^2(x) = \beta^{-1} \boldsymbol{\phi}(x)^T \mathbf{S} \boldsymbol{\phi}(x),$$

행렬 \mathbf{S} 는 다음과 같이 정의되며, $\boldsymbol{\phi}(x)$ 는 성분이 $\phi_i(x) = x^i$ 인 벡터이다.
($i = 0, 1, \dots, M$)

$$\mathbf{S}^{-1} = \alpha \mathbf{I} + \beta \sum_{n=1}^N \boldsymbol{\phi}(x_n) \boldsymbol{\phi}(x_n)^T$$

- 중요한 것은 predictive distribution의 평균과 분산이 test point인 x 에 의존한다는 것이다. 즉, Bayesian은 test point의 Uncertainty를 반영 할 수 있음을 보여준다.

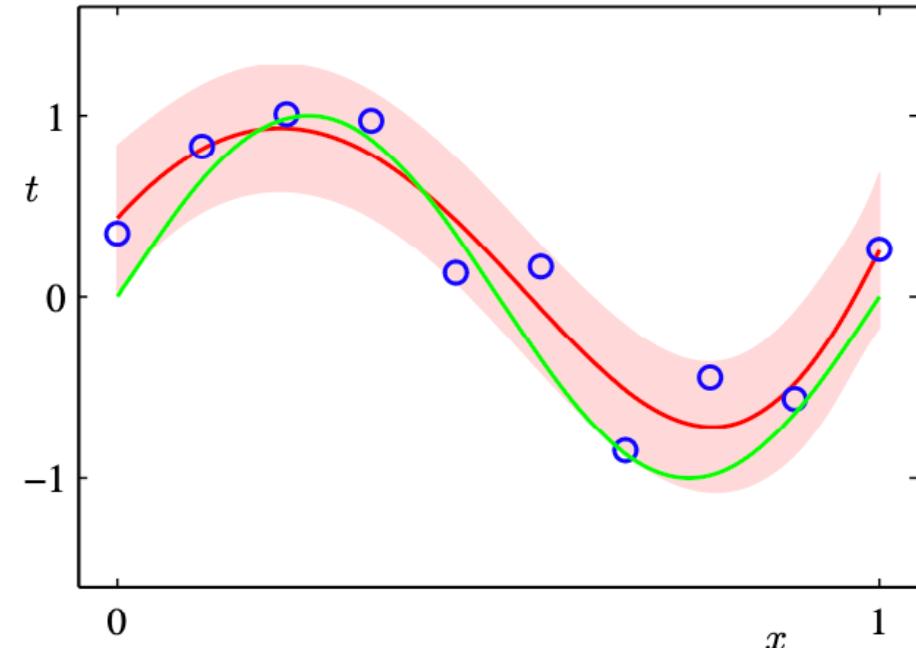


Figure. The plot shows the predictive distribution resulting from a Bayesian treatment of polynomial curve fitting with $M = 9$, $\alpha = 5 \times 10^{-3}$, $\beta = 11.1$, in which the red curve denotes the mean and the red region corresponds to ± 1 s.d. around the mean.

3

Decision Theory

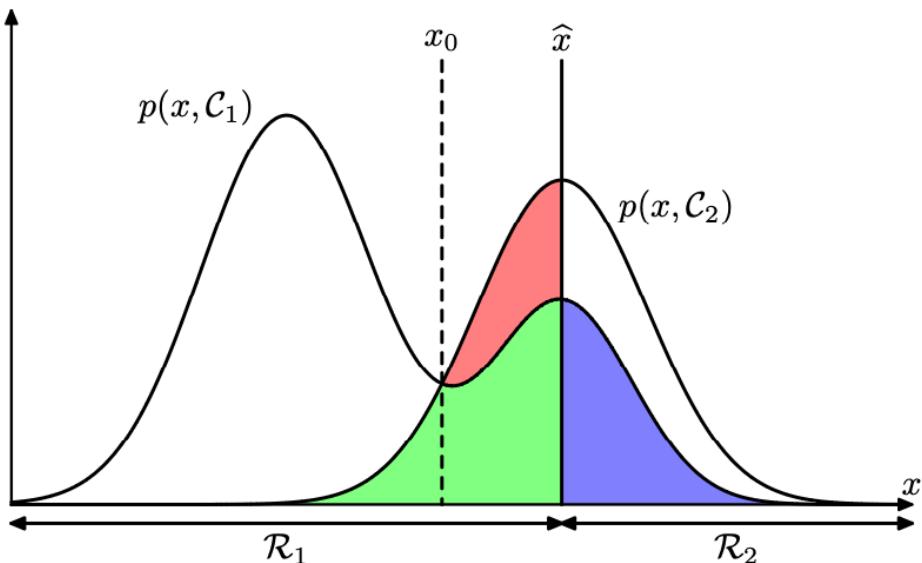
Decision Theory for Situations Involving Uncertainty

Motivation

- 이전과 마찬가지로 training data (\mathbf{x}, \mathbf{t}) 가 주어져 있다고 하자. 우리의 목표가 regression이라면 \mathbf{t} 는 연속적인 변수일 것이고, classification이라면 \mathbf{t} 는 각 class label을 의미하므로 이산적인 변수일 것이다.
- $p(\mathbf{x}, \mathbf{t})$ 는 이러한 변수들과 관련된 uncertainty에 대한 완벽한 요약을 제공한다. 하지만 training data로부터 $p(\mathbf{x}, \mathbf{t})$ 를 찾아내는 것을 inference라 하며, 이는 매우 어려운 일이다.
- 그러나 실전에서는 data의 완전한 분포를 찾지 못하더라도 \mathbf{t} 의 값에 대한 패턴을 찾아냄으로써 어느 정도의 추론이 가능하다. 이것이 decision theory의 주제이다.
- * Decision Theory for Binary Classification Problem
 - 우리가 X-ray 사진을 통해 환자의 암 여부를 판단한다고 하자. $t = 0$ 은 암이 존재함을 의미하며 C_1 라는 클래스에, $t = 1$ 은 암이 없음을 의미하며 C_2 라는 클래스에 대응시키자. 여기에 Bayes' thm.을 적용하면
$$p(C_k|\mathbf{x}) = \frac{p(\mathbf{x}|C_k)p(C_k)}{p(\mathbf{x})}$$
 - 를 얻는다. $p(C_1)$ 은 환자에 암이 존재할 경우, $p(C_2)$ 는 환자에 암이 없을 경우에 대한 prior distribution이다.
 - 수학적, 통계적인 이야기들을 제외하더라도, 상식적으로 X-ray 사진을 보았을 때 암일 확률과 암이 아닐 확률 중 더 커 보이는 것을 암 판정 결과로 내리는 것이 합리적이다. 즉, posterior인 $p(C_k|\mathbf{x})$ 가 큰 것을 고르면 되지 않을까라는 생각을 얻을 수 있다. 그리고 이는 실제로 통계적으로 합리적인 추론임을 후술할 내용을 통해 알 수 있다.

Minimizing the Misclassification Rate

- 우리의 목표가 오분류를 가능한 줄이는 것이라고 하자. 이를 위해 우리는 \mathbf{x} 의 각 값을 class에 분류하는 것에 대한 어떠한 규칙이 필요하다.
- 각 input space를 **decision region**이라 불리는 R_k 로 나누고, 각 class 당 하나의 region을 할당하자. 즉, R_k 에 속하는 모든 input points는 class C_k 로 구분해주는 것이다. 각 region 간 경계를 **decision boundary** 혹은 **decision surface**라 부른다. (각 region이 이어질 필요는 없지만 disjoint여야 한다.)



* Example: Binary Classification

- 오분류는 가령 C_1 에 속해야 할 것을 C_2 로 분류할 때 발생한다. 즉,

$$p(\text{mistake}) = p(\mathbf{x} \in R_1, C_2) + p(\mathbf{x} \in R_2, C_1)$$

$$= \int_{R_1} p(\mathbf{x}, C_2) d\mathbf{x} + \int_{R_2} p(\mathbf{x}, C_1) d\mathbf{x}.$$

- $p(\text{mistake})$ 를 최소화하기 위해서는 각 \mathbf{x} 를 위 적분 값이 더 작은 클래스로 분류되도록 해야 할 것이다. 즉, 어떤 \mathbf{x} 에 대해 $p(\mathbf{x}, C_1) > p(\mathbf{x}, C_2)$ 라면 C_1 로 분류한다.

- $p(\mathbf{x}, C_k) = p(C_k | \mathbf{x})p(\mathbf{x})$ 에서, $p(\mathbf{x})$ 는 클래스와 관계없이 공통적이다. 따라서 $p(\mathbf{x}, C_k)$ 가 가장 큰 클래스로 분류한다는 것은 posterior인 $p(C_k | \mathbf{x})$ 가 가장 큰 클래스로 분류하는 것이 $p(\text{mistake})$ 를 최소화하는 방법이라는 것이다.

Minimizing the Misclassification Rate

- 이번엔 K 개의 class가 존재한다고 하자. $p(\text{correct}) = 1 - p(\text{mistake})$

이므로 이번에는 $p(\text{correct})$ 를 최대화하는 방법을 생각해보자.

- $p(\text{correct})$ 는 다음과 같다:

$$p(\text{correct}) = \sum_{k=1}^K p(\mathbf{x} \in R_k, C_k) = \sum_{k=1}^K \int_{R_k} p(\mathbf{x}, C_k) d\mathbf{x}.$$

- 위 식에서, 각 \mathbf{x} 가 $p(\mathbf{x}, C_k)$ 값이 가장 큰 class에 할당되도록 decision region을 설정하는 것이 $p(\text{correct})$ 를 최대화하는 방법일 것이다.

- $p(\mathbf{x}, C_k) = p(C_k | \mathbf{x})p(\mathbf{x})$ 이므로 앞선 binary case와 마찬가지로 posterior가 가장 큰 class로 \mathbf{x} 를 분류하는 것이 optimization이다.

Minimizing the Expected Loss

- Loss function, 혹은 cost function이라 불리우는 함수를 도입하여 위의 사례를 수식화 해보자. Loss function은 발생가능한 모든 loss의 합이다. 우리의 목표는 total loss를 최소화하는 것이다.

- C_k 가 true class인 \mathbf{x} 가 C_j 로 분류되었다고 하자. ($k = j$ 일 수도 있음.) 이 때 발생할 수 있는 loss를 loss matrix L 의 (k, j) th entry로 설정하자.

- Optimal solution은 loss function을 최소화하는 것이다. 하지만 우리는 true class를 알 수 없다. 이러한 true class의 uncertainty를 $p(\mathbf{x}, C_k)$ 로 표현하고, 대신 average loss를 최소화하도록 하자.

$$E[L] = \sum_k \sum_j \int_{R_j} L_{kj} p(\mathbf{x}, C_k) d\mathbf{x}.$$

- 각 \mathbf{x} 는 decision region들 중 하나로 독립적으로 배정될 것이다. 우리의 목표는 위 $E[L]$ 을 최소화하게끔 region들을 정하는 것이다. 즉, $\sum_k L_{kj} p(\mathbf{x}, C_k)$ 혹은 $\sum_k L_{kj} p(C_k | \mathbf{x})$ 를 최소화하면 된다. Joint dist.를 posterior로 바꿀 수 있는 것은 앞서 언급된 것들과 동일한 이유이다.

* Example: Loss Matrix for Cancer Detection

- 암 여부를 진단할 때 두 종류의 오류가 존재한다:

- i) 암이 없는 환자를 있다고 진단,
- ii) 암이 있는 환자를 없다고 진단.

- 2종 오류는 환자를 죽게 만들 수도 있으므로 2종 오류를 가능한 작게 해야 할 것이다.

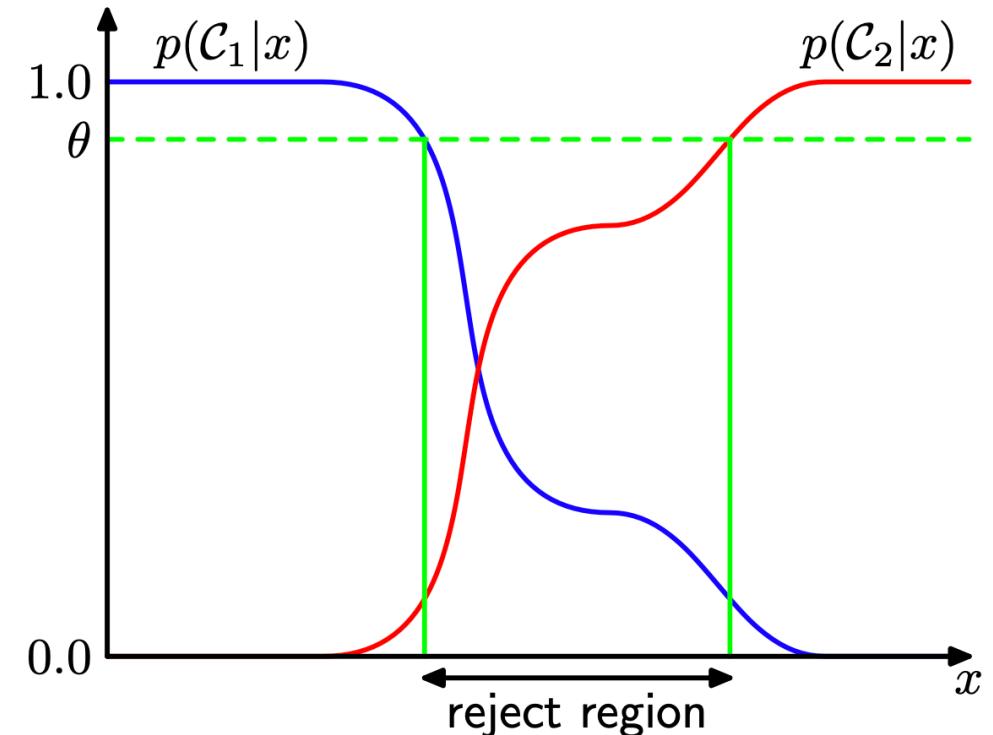
	cancer	normal
cancer	0	1000
normal	1	0

- 각 row는 실제 class를, column은 decision criterion에 따라 설정된 배정된 class를 의미하는 loss matrix이다.

- 실제 암환자가 정상으로 판정될 때의 weight를 다른 것보다 훨씬 크게 설정하였으므로 loss function을 최소화 하기 위해서는 2종 오류가 최우선적으로 작아질 것이다.

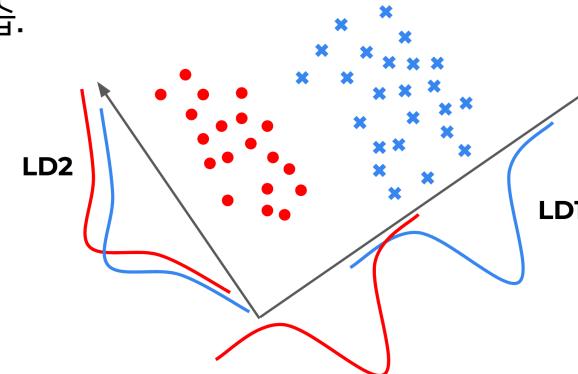
The Reject Option

- What we've learned so far: $p(C_k|\mathbf{x})$ 가 1보다 현저히 작거나, 혹은 $p(\mathbf{x}, C_k)$ 가 유사할 때 분류 오류가 발생한다.
- 위와 같은 상황은 결국 region의 uncertainty가 높은 경우이다. 따라서 이런 경우 우리는 모델로 하여금 error rate가 낮을 것으로 기대되지 않는다면 그러한 데이터의 분류를 유보하게 할 수 있다. 이를 reject option이라 한다.
- 암 진단의 예시에서, 확실한 것은 자동으로 분류하고, 불확실한 것은 의사가 직접 분류하도록 모델을 설정하는 것이 reject option의 사례가 될 것이다.
- 이는 임계값 θ 를 설정하여 $\max_k \{p(C_k|\mathbf{x})\} \leq \theta$ 라면 유보하도록 하여 구현할 수 있다.
- 만약 $\theta = 1$ 이라면 모든 데이터가 rejected될 것이며, K 개의 class가 존재할 때 $\theta < 1/K$ 라면 어떤 데이터도 rejected 되지 않을 것이다.



Inference and Decision

- 분류 문제는 크게 두 단계로 나뉠 수 있다:
 - i) Inference stage: training data를 통해 $p(C_k|\mathbf{x})$ 를 학습,
 - ii) Decision stage: 최적화된 class 할당을 위해 posterior 이용
- 그러나 실제로 decision problem을 위한 방법은 총 세가지가 존재한다.
하나씩 살펴보자. (복잡도 기준 내림차순)
 1. Generative Models — Naive Bayes, Gaussian Mixture Models
 - 모든 클래스에 대해 $p(\mathbf{x}|C_k)$ 를 찾는 inference stage 진행.
 - Prior class probabilities $p(C_k)$ 에 대한 Inference stage 따로 진행.
 - Bayes' thm.을 이용하여 posterior class probabilities $p(C_k|\mathbf{x})$ 찾음.
(* 분모는 $p(\mathbf{x}) = \sum_k p(\mathbf{x}|C_k)p(C_k)$ 를 통해 구할 수 있음.)
 - 이후 decision theory를 이용해 새로운 Input에 대한 class 할당을 결정.
 - 이렇게 입력과 출력에 대한 분포를 학습하는 모델은 샘플링을 통해 input data를 생성할 수 있기 때문에 생성 모델이라 불림.
 2. Discriminative Models — Logistic Regression, SVM
 - posterior를 찾는 Inference stage
 - 각각의 새로운 input data를 class들 중 하나에 할당하기 위해 decision theory 이용
 - 바로 posterior를 학습하려는 이러한 모델은 결국 클래스 간의 결정 경계를 학습하여 새로운 입력이 어떤 클래스에 속하는지를 예측하려는 것으로 판별 모델이라 불림.
 3. Linear Discriminant Analysis
 - **discriminant function**이라 불리는 $f(\mathbf{x}): \mathbf{x} \mapsto C_k$ 로 mapping하는 함수를 바로 학습.



Inference and Decision

1. Generative Models — Naive Bayes, Gaussian Mixture Models

- 모든 클래스에 대해 $p(\mathbf{x}|C_k)$ 를 찾는 inference stage 진행.
- Prior class probabilities $p(C_k)$ 에 대한 Inference stage 따로 진행.
- Bayes' thm.을 이용하여 posterior class probabilities $p(C_k|\mathbf{x})$ 찾음.
(* 분모는 $p(\mathbf{x}) = \sum_k p(\mathbf{x}|C_k)p(C_k)$ 를 통해 구할 수 있음.)
- 이후 decision theory를 이용해 새로운 Input에 대한 class 할당을 결정.
- 이렇게 입력과 출력에 대한 분포를 학습하는 모델은 샘플링을 통해 input data를 생성할 수 있기 때문에 생성 모델이라 불림.

- Pros: likelihood와 prior를 모두 계산하므로, 이를 통해 data의 marginal density인 $p(\mathbf{x})$ 를 찾을 수 있음. 이를 통해 학습된 모델 하에서 발생 확률이 낮은, 즉 예측 정확도가 낮은 새로운 data point를 찾아낼 수 있음. 즉, outlier detection이 가능.
(Prior는 보통 training data의 class fraction으로 찾음.)

- Cons: 계산해야하는 분포가 많으므로 복잡도가 높음. 모든 class에 대해 정확도가 괜찮은 $p(\mathbf{x}|C_k)$ 를 찾아내기 위해 대량의 training data를 필요.

Inference and Decision

2. Discriminative Models — Logistic Regression, SVM

- posterior를 찾는 Inference stage

- 각각의 새로운 input data를 class들 중 하나에 할당하기 위해 decision theory 이용

- 바로 posterior를 학습하려는 이러한 모델은 결국 클래스 간의 결정 경계를 학습하여 새로운 입력이 어떤 클래스에 속하는지를 예측하려는 것이므로 판별 모델이라 불림.

- Pros: Decision theory가 말해주듯이 classification을 위해서는 joint dist.가 아닌 posterior만 찾으면 됨. Discriminative Model은 posterior를 직접 찾는데, 이는 joint distribution을 찾는 Generative model에 비해 훨씬 적은 computational resource를 필요로 함.

- Cons: 이상치 탐지 어려움, missing data 찾기 어려움, 데이터 분포에 대한 이해도 상대적으로 부족.

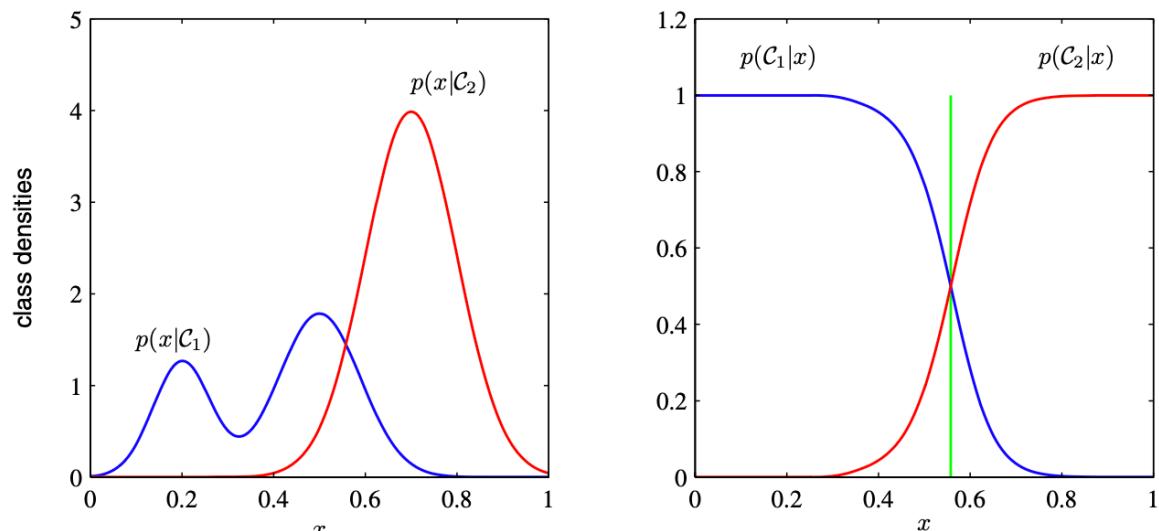


Figure. This figure shows that class-conditional density has no effect on the posterior probabilities. Compare the two blue curves on each plot.

Inference and Decision

3. Linear Discriminant Analysis

- discriminant function이라 불리는 $f(\mathbf{x}): \mathbf{x} \mapsto C_k$ 로 mapping하는 함수를 바로 학습.

- Pros: Inference stage와 Decision stage를 한번에 진행함으로 간결함.

$f(\mathbf{x})$ 를 찾는다는 것은 아래 Firgure의 우측 plot에서 minimum misclassification probability를 가지는 decision boundary인 초록색 수직선을 찾는 것을 의미함.

- Cons: posterior probability에 대한 어떠한 접근도 불가능. (즉, 분류 외에 어떤 것도 할 수 없음.)

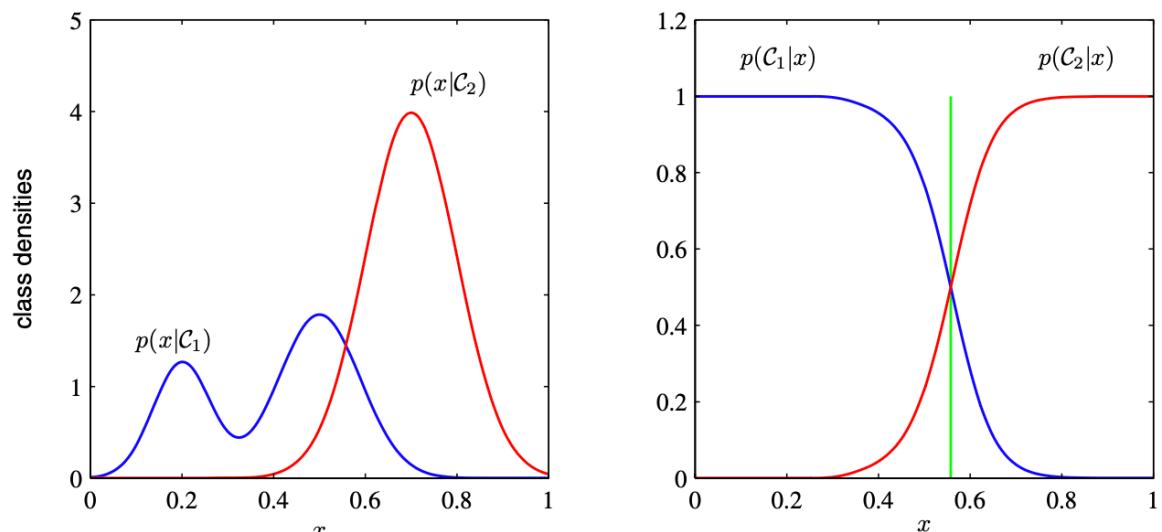


Figure. This figure shows that class-conditional density has no effect on the posterior probabilities. Compare the two blue curves on each plot.

Loss Functions for Regression

- Regression에서의 loss를 $L(t, y(\mathbf{x}))$ 라 표기하고 classification의 사례와 마찬가지로 average loss를 정의하자:

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

- Loss로서 Squared loss를 이용한다면 average loss는 다음과 같다:

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

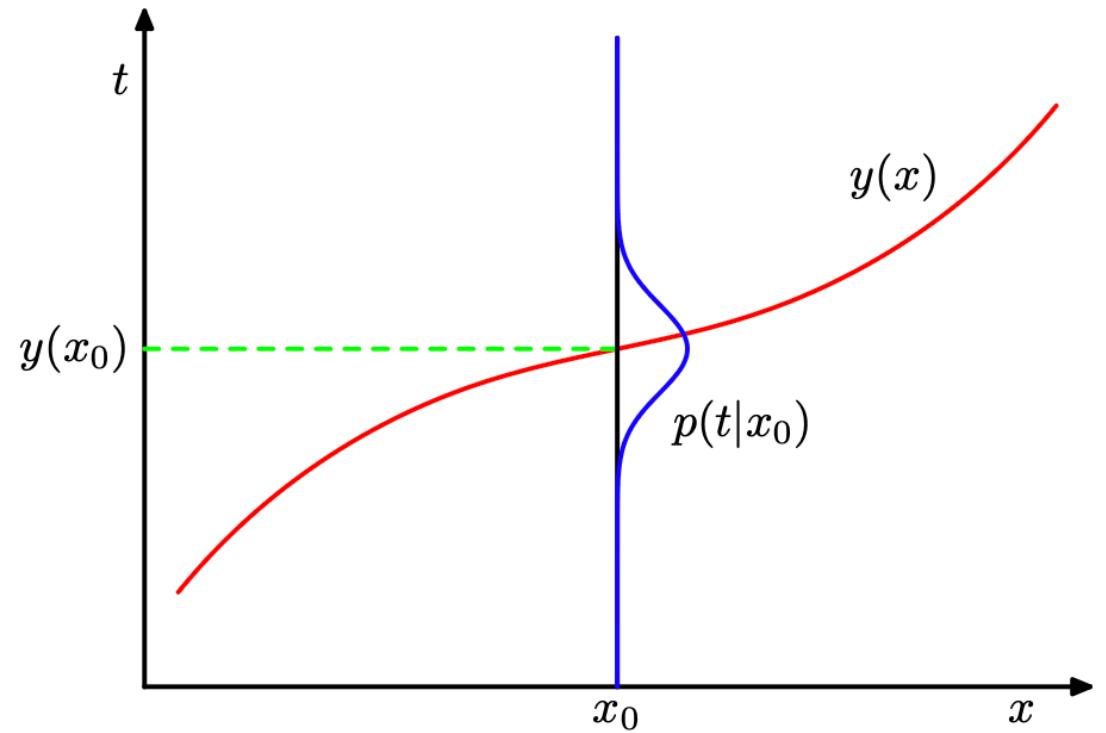
- 우리의 목표는 $E[L]$ 을 최소화하는 $y(\mathbf{x})$ 를 찾는 것이다. 즉,

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

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

를 얻을 수 있다.

- 이는 t 가 주어졌을 때 \mathbf{x} 의 조건부 평균이고, regression function이라 알려진 것과 동일한 결과를 보여준다.
- 이를 target variable이 여러개일 때로 확장하더라도, optimal point estimation은 마찬가지로 $y(\mathbf{x}) = E[t|\mathbf{x}]$ 일 때일 것이다.



Loss Functions for Regression

- Square term은 아래와 같이 확장할 수 있다:

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

- 이를 loss function에 넣고 t 에 대해 적분을 진행하면,

$$E[L] = \int \{y(\mathbf{x}) - E[t|\mathbf{x}]\}^2 p(\mathbf{x}) d\mathbf{x} + \int \text{var}[t|\mathbf{x}] p(\mathbf{x}) d\mathbf{x}.$$

- 우리가 최소화하고자 하는 $y(\mathbf{x})$ 는 첫번째 term에만 존재하고, $y(\mathbf{x}) = E[t|\mathbf{x}]$ 일 때 최소화되므로, 마찬가지로 앞서 미분을 통해 찾은 결과와 동일하다.

- 하지만 $y(\mathbf{x}) = E[t|\mathbf{x}]$ 이더라도 두번째 term은 사라지지 않는 것을 볼 수 있는데, 이는 $\int \text{var}[t|\mathbf{x}] p(\mathbf{x}) d\mathbf{x} = E[\text{var}[t|\mathbf{x}]]$ 이며, target data에 내재된 변동의 평균을 의미한다. 즉, 데이터에 포함된 noise로서, $y(\mathbf{x})$ 와 무관하므로 줄일 수 없는 loss function의 최솟값을 나타낸다.

* Three Approaches for Solving Regression Problems

(in order of decreasing complexity)

1. $p(\mathbf{x}, t)$ 를 위한 inference stage. 이후 $p(t|\mathbf{x})$ 를 찾아 $E[t|\mathbf{x}]$ 계산.
2. $p(t|\mathbf{x})$ 를 찾기 위한 inference stage. 이후 $E[t|\mathbf{x}]$ 계산.
3. Training data로부터 곧바로 $y(\mathbf{x})$ 계산.

각각의 장단점은 classification problem의 경우들과 동일함.

4

Information Theory

Decision Theory for Situations Involving Uncertainty

Measure of Information

- 정보의 양을 어떻게 측정하면 좋을까? 정보의 양은 'degree of surprise'로 생각될 수 있다. 예를 들어 우리가 X 라는 이산확률변수의 값을 관찰한다고 하자. 만약 우리가 예상했던 것과 값이 비슷하다면 별로 놀랍지 않을 것이다. 반대로 예상을 벗어난다면 우리가 그 관찰로부터 알게된 정보의 양은 매우 많을 것이다.
- 그렇다면 어떤 확률변수의 값이 $X = x$ 일 것으로 예상된다는 것은 결국 $p(X = x)$ 가 높다는 것이고, 예상 외라는 것은 $p(X = x)$ 가 작다는 것이다.
- 따라서 정보의 양을 나타내주는 함수로 $h(x)$ 를 정의한다면, 결국 $h(x)$ 는 $p(x)$ 에 대해 monotonic한 함수로 표현될 것이다.
- $\text{Cov}(x, y) = 0$ 인 x 와 y 가 있다. $p(x, y) = p(x)p(y)$ 이고, 따라서 $h(x, y) = h(x) + h(y)$ 일 것이다. 이러한 조건을 만족시키면서 $p(x)$ 와 $h(x)$ 가 서로 반대로 움직이는 단조 관계이도록 하려면 둘의 관계를 어떻게 정의하면 좋을까?

Measure of Information

- 바로 logarithm 관계를 이용하면 된다. 즉,

$$h(x) = -\log_2 p(x).$$

- 로그의 밑은 무엇으로 하든 상관없으며, 위의 예시에서 $h(x)$ 의 단위는 'bits'이다.

- 이제 확률변수의 값을 전송한다고 하자. 보내지는 정보의 평균량은 $E[h(x)]$ 일 것이며, 이를 $H[x]$ 라 정의하자:

$$H[x] = -\sum_x p(x) \log_2 p(x).$$

- $H[x]$ 는 확률변수 x 의 **entropy**라 불리우며, $\lim_{p \rightarrow 0} p \ln p = 0$ 으로 $p(x) = 0$ 인 x 의 $p(x) \ln p(x) := 0$ 으로 생각한다.

* Example: Transmission of x with 8 Possible States

1. If each of states is equally likely:

$$H[x] = -8 \times \frac{1}{8} \log_2 \frac{1}{8} = 3 \text{ bits.}$$

2. If $x \in \{a, b, c, d, e, f, g, h\}$ for which the respective probabilities are given by $\left\{\frac{1}{2}, \frac{1}{4}, \frac{1}{8}, \frac{1}{16}, \frac{1}{64}, \frac{1}{64}, \frac{1}{64}, \frac{1}{64}\right\}$:

$$\begin{aligned} H[x] &= -\frac{1}{2} \log_2 \frac{1}{2} - \frac{1}{4} \log_2 \frac{1}{4} - \frac{1}{8} \log_2 \frac{1}{8} - \frac{1}{16} \log_2 \frac{1}{16} - \frac{4}{64} \log_2 \frac{1}{64} \\ &= 2 \text{ bits.} \end{aligned}$$

Measure of Information

- 앞선 예시에서 확률이 모두 동일할 때보다 그렇지 않을 때의 entropy가 더 작음을 확인할 수 있었다.

- 평균 코드 길이를 짧게 하기 위하여 발생 확률이 작은 사건일수록 긴 코드를 할당하여 보자. $a \sim h$ 에 대하여 0, 10, 110, 1110, 111100, 111101, 111110, 111111을 각각 할당한다면, 전송되는 평균 코드 길이는 다음과 같다:

$$\begin{aligned}\text{average code length} &= \frac{1}{2} \times 1 + \frac{1}{4} \times 2 + \frac{1}{8} \times 3 + \frac{1}{16} \times 4 + 4 \times \frac{1}{64} \times 6 \\ &= 2 \text{ bits.}\end{aligned}$$

- 이는 확률변수 x 의 엔트로피와 동일한 결과이다. 이보다 더 짧은 code string은 불가능한데, 예를 들어 11001110은 c, a, d 로 복호화될 것이기 때문이다.

- *noiseless coding theorem* (Shannon, 1948)에 따르면

$$H[x] = \inf |\text{bits to transmit } x|$$

이므로 결국 위 평균 코드 길이는 $H[x]$ 보다 작아질 수 없다.

* Example: Transmission of x with 8 Possible States

1. If each of states is equally likely:

$$H[x] = -8 \times \frac{1}{8} \log_2 \frac{1}{8} = 3 \text{ bits.}$$

2. If $x \in \{a, b, c, d, e, f, g, h\}$ for which the respective probabilities are given by $\left\{\frac{1}{2}, \frac{1}{4}, \frac{1}{8}, \frac{1}{16}, \frac{1}{64}, \frac{1}{64}, \frac{1}{64}, \frac{1}{64}\right\}$:

$$\begin{aligned}H[x] &= -\frac{1}{2} \log_2 \frac{1}{2} - \frac{1}{4} \log_2 \frac{1}{4} - \frac{1}{8} \log_2 \frac{1}{8} - \frac{1}{16} \log_2 \frac{1}{16} - \frac{4}{64} \log_2 \frac{1}{64} \\ &= 2 \text{ bits.}\end{aligned}$$

Measure of Information

- 총 N 개의 공을 바구니에 옮겨 담을 때, i 번째 바구니에 담긴 공의 개수를 n_i 라 하자. ($\sum n_i = N$) 그렇다면 공을 분배하는 경우의 수는

$$W = \frac{N!}{\prod_i n_i!}$$

이며, 이를 multiplicity라 부른다. Entropy는 어떤 특정한 상수에 의해 scaled된 multiplicity의 로그값이다. 즉,

$$H = \frac{1}{N} \ln W = \frac{1}{N} \ln N! - \frac{1}{N} \sum_i \ln n_i !$$

이다. $N \rightarrow \infty$ 일 때, Stirling's approximation에 의하면 $\ln N! \simeq N \ln N - N$ 이다. 따라서

$$H = - \lim_{N \rightarrow \infty} \sum_i \left(\frac{n_i}{N} \right) \ln \left(\frac{n_i}{N} \right) = - \sum_i p_i \ln p_i$$

이며, 이를 통해 우리는 이항분포를 포함한 어떤 이산확률변수에 대해서도 entropy를 정의할 수 있다.

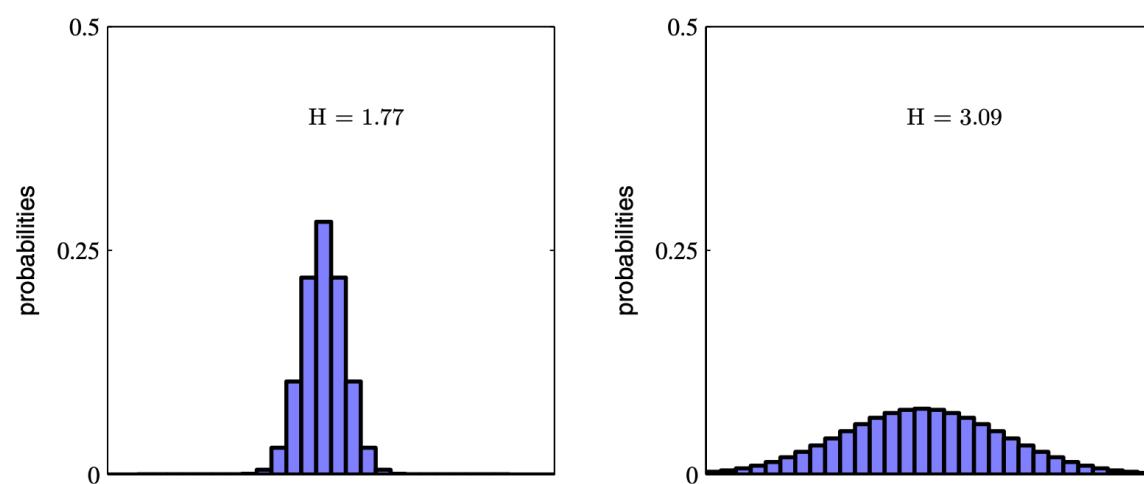
$$H[p] = - \sum_i p(x_i) \ln p(x_i) \text{ where } p(X = x_i) = p_i$$

Measure of Information

- X 의 Entropy는 분포가 uniform할수록 커지며, $p_i = 1, p_{j \neq i} = 0$ 일 때 0으로 최소이다.
- Entropy를 최대화하기 위해서는 Lagrange multiplier를 이용하여 계산할 수 있다:
- States x_i 의 개수가 M 이라 하자. 확률이 모두 uniform할 때 Entropy가 최대일 것이며, 각 $p(x_i) = 1/M$ 일 것이다. 이때 $H = \ln M$ 이다.
- 실제로 이 entropy 값이 최대인지 알아보기 위하여, 엔트로피의 이계도함수를 구해보자:

$$\tilde{H} = - \sum_i p(x_i) \ln p(x_i) + \lambda \left(\sum_i p(x_i) - 1 \right).$$

$$\frac{\partial \tilde{H}}{\partial p(x_i) \partial p(x_j)} = -\delta_{ij} \frac{1}{p_i} \quad (\delta_{ij} \text{ := Kronecker delta})$$



임을 알 수 있다. 이계도함수가 0보다 작으므로, 확률이 모두 uniform할 때 maximum임을 확인할 수 있다.

Measure of Information

- 만약 x 가 연속확률변수라면 어떻게 entropy를 정의할 수 있을까?
앞서 이산확률변수를 위해 사용한 바구니 예시를 변형하여 살펴보자.
- x 의 PDF를 Δ 의 길이를 가진 uniform partition들로 나누어주자. 만약 Δ 의 길이가 충분히 작다면, 평균값 정리에 의하여

$$\int_{i\Delta}^{(i+1)\Delta} p(x) dx = p(x_i)\Delta$$

를 만족시키는 x_i 가 존재할 것이다. 이를 통해 우리는 x 가 i^{th} 막대에 위치할 확률을 $p(x_i)\Delta$ 라고 표현할 수 있다. 즉, 연속확률변수를 이산확률변수로서 표현할 수 있게 된 것이다.

- 앞서 살펴본 이산확률변수에서의 entropy의 정의에 대입하면

$$H_\Delta = - \sum_i p(x_i)\Delta \ln(p(x_i)\Delta) = - \sum_i p(x_i)\Delta \ln p(x_i) - \ln \Delta$$

을 얻을 수 있다. 이제 $\Delta \rightarrow 0$ 이라면,

$$\lim_{\Delta \rightarrow 0} \left\{ \sum_i p(x_i)\Delta \ln p(x_i) \right\} = - \int p(x) \ln p(x) dx$$

가 되며, RHS를 **differential entropy**라 부른다.

- 그렇다면 이산일 때의 entropy와 연속일 때의 entropy는 $\ln \Delta$ 만큼의 차 이를 갖게 되는데, 이는 $\Delta \rightarrow 0$ 일 때 발산한다. 이것은 우리가 연속적인 정보를 매우 정확하게 보내기 위해서는 엄청난 양의 정보를 필요로 한다는 것을 의미한다.
- 만약 \mathbf{x} 가 multivariate continuous r.v.라면 differential entropy는 위 univariate case에서 x 를 \mathbf{x} 로 바꿔주면 된다.

Measure of Information

- x 가 연속확률변수일 때 entropy의 최대값은 어떤 경우일까? 이를 위해 우선 $p(x)$ 는 확률이므로 다음과 같은 conditions을 충족해야 한다:

$$\int_x p(x) = 1$$

$$\int_x xp(x) = \mu$$

$$\int_x (x - \mu)^2 p(x) = \sigma^2$$

- 이제 Lagrange multipliers를 이용하여 constrained maximization을 구해주자:

$$-\int p(x) \ln p(x) dx + \lambda_1 \left(\int p(x) dx - 1 \right)$$

$$+ \lambda_2 \left(\int xp(x) dx - \mu \right) + \lambda_3 \left(\int (x - \mu)^2 p(x) dx - \sigma^2 \right)$$

- 확률에 대한 편미분값이 0이 되도록 설정하여,

$$p^*(x) = \exp\{-1 + \lambda_1 + \lambda_2 x + \lambda_3 (x - \mu)^2\}$$

임을 찾을 수 있다. $p^*(x)$ 를 다시 위의 Lagrange multiplier의 $p(x)$ 자리에 넣어주면, 다음 결과를 얻는다:

$$p^*(x) = \frac{1}{(2\pi\sigma^2)^{1/2}} \exp\left\{-\frac{(x - \mu)^2}{2\sigma^2}\right\}.$$

Measure of Information

- 즉, 연속확률변수의 differential entropy는 그 변수가 정규분포를 따를 때 최대가 된다는 것을 확인할 수 있다.
 - 정규분포를 따를 때의 differential entropy는
- * Conditional Entropy
 - \mathbf{x} 가 이미 알고있을 때 \mathbf{y} 를 얻는데 필요한 추가적인 정보의 양은 아래와 같은 conditional entropy를 통해 정의된다:

$$H[x] = \frac{1}{2} \{1 + \ln(2\pi\sigma^2)\}$$

이며, 평균과는 관계 없이 분산이 증가할수록, H 도 증가함을 확인할 수 있다.

- 위의 H 에서, $\sigma^2 < 1/(2\pi e)$ 일 때, $H(x) < 0$ 이 될 것은 자명하다. 즉, differential entropy는 discrete일 때와 다르게 음수일 수 있다.

$$H[\mathbf{y}|\mathbf{x}] = - \iint p(\mathbf{y}, \mathbf{x}) \ln p(\mathbf{y}|\mathbf{x}) d\mathbf{y} d\mathbf{x}.$$

- Conditional entropy는 product rule에 의해 다음과 같은 관계를 만족해야 한다:

$$H[\mathbf{x}, \mathbf{y}] = H[\mathbf{y}|\mathbf{x}] + H[\mathbf{x}]$$

즉, \mathbf{x} 와 \mathbf{y} 를 모두 설명하기 위해 필요한 정보의 양은 \mathbf{x} 를 알고, 추가적으로 \mathbf{y} 를 얻기 위해 필요한 정보량의 합과 같다.

KL Divergence and Mutual Information

- 어떠한 unknown distribution $p(\mathbf{x})$ 가 있고, 이를 모델링하여 근사한 $q(\mathbf{x})$ 가 있다고 하자. \mathbf{x} 를 전송하기 위해서, $p(\mathbf{x})$ 가 아닌 $q(\mathbf{x})$ 를 사용한 것으로 인하여 추가적인 정보량이 필요하다. 이때 평균 추가 정보량은 다음과 같다:

$$KL(p||q) = - \int p(\mathbf{x}) \ln q(\mathbf{x}) d\mathbf{x} - \left(- \int p(\mathbf{x}) \ln p(\mathbf{x}) d\mathbf{x} \right)$$

$$= - \int p(\mathbf{x}) \ln \left\{ \frac{q(\mathbf{x})}{p(\mathbf{x})} \right\} d\mathbf{x}.$$

- 이 식은 $p(\mathbf{x})$ 와 $q(\mathbf{x})$ 사이의 relative entropy, Kullback-Leibler divergence, 혹은 KL divergence라고 불리며, $KL(p||q) \neq KL(q||p)$ 임에 유의하자.

KL Divergence and Mutual Information

- KL divergence는 $KL(p||q) \geq 0$ 이며, $p(\mathbf{x}) = q(\mathbf{x})$ 일 때 등식을 만족시

킨다는 성질을 가진다.

- 이 성질을 이해하기 위해 convexity에 대해 알아보자. 다음 부등식을 만족하는 함수 $f(x)$ 를 convex function이라 한다:

$$\forall a, b \in \text{dom}f, \forall \lambda \in [0,1],$$

$$f(x) \text{ satisfies } f(\lambda a + (1 - \lambda)b) \leq \lambda f(a) + (1 - \lambda)f(b).$$

- 그렇다면 convex function $f(x)$ 는 귀납적으로

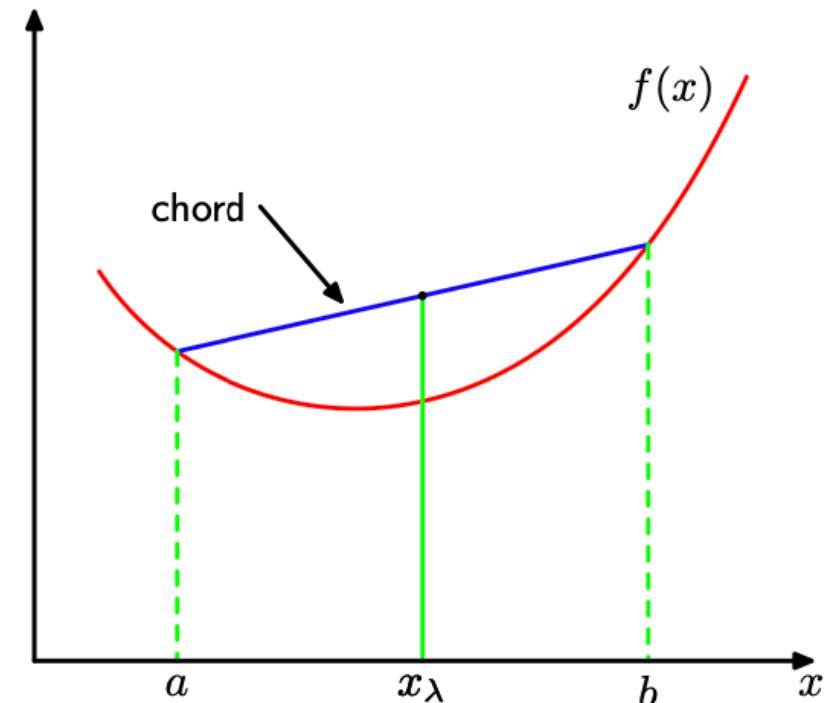
$$f\left(\sum_{i=1}^M \lambda_i x_i\right) \leq \sum_{i=1}^M \lambda_i f(x_i)$$

를 만족시킬 것이다. 이때 이산확률변수 x 에 대해 $\lambda_i := p(x_i)$ 라 정의하면

$$f(E[x]) \leq E[f(x)],$$

혹은 연속확률변수의 경우, Jensen's inequality에 의해 다음과 같다:

$$f\left(\int \mathbf{x} p(\mathbf{x}) d\mathbf{x}\right) \leq \int f(\mathbf{x}) p(\mathbf{x}) d\mathbf{x}.$$



KL Divergence and Mutual Information

- $-\ln x$ 가 convex function이므로 KL divergence에 Jensen's inequality를 적용한다면,

$$KL(p||q) = - \int p(\mathbf{x}) \ln \left\{ \frac{q(\mathbf{x})}{p(\mathbf{x})} \right\} d\mathbf{x} \geq - \ln \int q(\mathbf{x}) d\mathbf{x} = 0$$

을 얻는다. 등호는 $p(\mathbf{x}) = q(\mathbf{x})$ 일 때 성립할 것이다. 이는 KL divergence를 $p(\mathbf{x})$ 와 $q(\mathbf{x})$ 사이의 차이에 대한 측도라고 해석할 수 있음을 보여준다.

- $p(\mathbf{x})$ 를 찾기 위해 θ 를 모수로 가지는 $q(\mathbf{x}|\theta)$ 를 근사분포로서 찾고자 한다. 이때 가장 적합한 θ 를 찾는 방법은, $p(\mathbf{x})$ 와 $q(\mathbf{x}|\theta)$ 사이의 KL divergence를 최소화하는 θ 를 찾는 것이다. 그러나 $p(\mathbf{x})$ 를 모르기 때문에 이 방법은 불가능하다.

- 우리가 가진 training points $x_n (n = 1, \dots, N)$ 들은 $p(\mathbf{x})$ 에서 뽑힌 sample들이다. 이때 $E[\mathbf{x}]$ 는 $\sum_{n=1}^N x_n / N$ 으로 추정할 수 있고, 따라서

$$KL(p||q) \simeq \sum_{n=1}^N \{-\ln q(x_n|\theta) + \ln p(x_n)\}$$

와 같이 KL divergence를 근사할 수 있다. 이때 우변의 두번째 항은 θ 와 무관하고, 첫번째 항은 q 의 negative log-likelihood function과 같다. 따라서 KL divergence를 최소화하는 것은 결국 log-likelihood를 최대화하는, 즉, θ 의 MLE를 찾는 것과 동치임을 확인할 수 있다.

KL Divergence and Mutual Information

- 확률 변수 \mathbf{x} 와 \mathbf{y} 가 서로 독립이라 하자. 그렇다면 당연히 $p(\mathbf{x}, \mathbf{y}) = p(\mathbf{x})p(\mathbf{y})$ 가 성립할 것이다. 하지만 독립이 아니라면 성립하지 않는다.
- 우리는 KL divergence를 이용하여 두 변수가 얼마나 독립에 가까운지 를 수치화할 수 있는데,
- Bayesian 관점에서, $p(\mathbf{x})$ 를 prior, $p(\mathbf{x}|\mathbf{y})$ 를 posterior로 해석할 수 있다. 따라서 mutual information은 \mathbf{y} 를 새로이 관측하면서 감소된 \mathbf{x} 의 불확실성의 감소 정도이다.

$$I[\mathbf{x}, \mathbf{y}] := KL(p(\mathbf{x}, \mathbf{y}) || p(\mathbf{x})p(\mathbf{y})) = - \iint p(\mathbf{x}, \mathbf{y}) \ln \left(\frac{p(\mathbf{x})p(\mathbf{y})}{p(\mathbf{x}, \mathbf{y})} \right) d\mathbf{x}d\mathbf{y}$$

로 정의되는 $I[\mathbf{x}, \mathbf{y}]$ 는 변수 \mathbf{x} 와 \mathbf{y} 사이의 mutual information이라 불린다. KL divergence의 성질에 따라 $I[\mathbf{x}, \mathbf{y}] \geq 0$ 이며, \mathbf{x} 와 \mathbf{y} 가 독립일 때 등호가 성립한다.

- 또한 우리는 mutual information을 conditional entropy의 관점에서 해석할 수도 있는데,

$$I[\mathbf{x}, \mathbf{y}] = H[\mathbf{x}] - H[\mathbf{x}|\mathbf{y}] = H[\mathbf{y}] - H[\mathbf{y}|\mathbf{x}]$$

이므로 $I[\mathbf{x}, \mathbf{y}]$ 는 \mathbf{y} 에 대한 정보가 주어졌을 때 \mathbf{x} 에 대한 불확실성 감소 정도 (or vice versa)로 해석할 수 있다.

감사합니다