Deep Learning Ch.5~5.5

Machine Learning Basics

최건호
ghc0311@naver.com

목차

"Deep learning is a specific kind of machine learning. In order to understand deep learning well, one must have a solid understanding of the basic principles of machine learning."

p.98

딥러닝은 머신러닝의 일환이기 때문에 딥러닝을 이해하려면 머신러닝의 개념들을 잘 알아야 한다.

I. Learning Algorithms

머신러닝 알고리즘은 데이터로부터 학습할 수 있는 알고리즘을 말하는데 이때 "학습"이라고 하는 것은 어떤 의미를 가지고 있을까?

> "A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P , if its performance at tasks in T , as measured by P , improves with experience E."
>
> Mitchell (1997)

프로그램이 학습한다는 것은 어떠한 task(T)에 대해 performance measure(P)로 측정한 수치가 experience(E)를 통해 증가한다는 것을 의미한다.

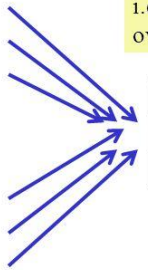
①Classification: 특정 input이 어느 카테고리에 속하는지 구분하는 task
만약 missing value가 있을 경우 probability distribution을 통해 marginalize out 하는 방법도 있음.

Marginalization

- We can also marginalize over more than one variable at once

$$P(X=x) = \Sigma_{z_1 \in dom(Z_1), \ldots, z_n \in dom(Z_n)} P(X=x, Z_1 = z_1, \ldots, Z_n = z_n)$$

Wind	Weather	Temperature	μ(w)
yes	sunny	hot	0.04
yes	sunny	mild	0.09
yes	sunny	cold	0.07
yes	cloudy	hot	0.01
yes	cloudy	mild	0.10
yes	cloudy	cold	0.12
no	sunny	hot	0.06
no	sunny	mild	0.11
no	sunny	cold	0.03
no	cloudy	hot	0.04
no	cloudy	mild	0.25
no	cloudy	cold	0.08



i.e., Marginalization over Temperature and Wind

Weather	μ(w)
sunny	0.40
cloudy	

②Regression: 특정 input이 들어오면 결과값을 수치로 나타내는 task

ex) [시험과목, 공부시간, 집중도] -> 시험점수

③Transcription: 상대적으로 구조화가 덜 된 데이터를 discrete, textual form으로 변환하는 task

ex)Speech Recognition (파동 -> 단어의 나열)
ex)Google Street View에서 address number 뽑아내기

④Machine Translation: 특정 언어로 된 input을 다른 언어로 번역하는 task

ex) 영어 <-> 프랑스어
ex) python <-> c++ 도 가능하려나?

⑤Structured Output: 결과값들이 벡터의 형태를 가지며 상호관계를 가진 task
넓은 개념이고 앞에서 언급한 transcription, translation도 포함

ex) NLP 에서 Parsing
ex) Image Captioning 에서 output (문장구조를 따름)

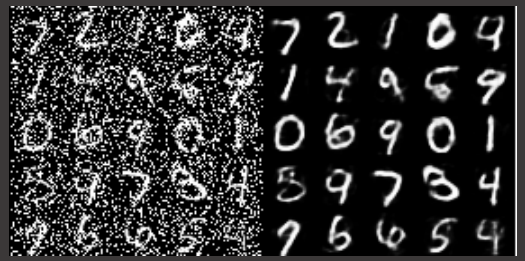
⑥Anomaly Detection: 데이터에서 특이하거나 비정상적인 부분을 찾아내는 task

ex) Fraud Detection

⑦Synthesis & Sampling: 트레이닝 데이터와 유사한 새로운 데이터를 만드는 task

ex) 게임에서 맵을 만들 때 숙련된 디자이너가 나무를 조금씩 다르게
만개를 만드는건 낭비이기 때문에 이런 때 사용됨

⑧Denoising: 노이즈가 있는 input이 들어왔을 때 이를 clean하게 바꿔주는 task

ex) 

⑨Density Estimation: 데이터의 구조, 분포를 파악하는 task

ex) Probability Distribution을 학습하면
missing value를 채우는 것이 가능

알고리즘의 성능을 측정하기 위해서는 quantitative한 척도가 필요하다. 또한 Task 에 따라 적합한 Performance Measure P를 사용해야 함.

ex) Classification, Transcription 같은 Task 는 accuracy를 통해 평가함
ex) Density Estimation 같은 경우는 avg. log prob.을 사용함

알고리즘의 목표는 처음 보는 데이터에 대한 성능 향상이기 때문에 Performance -는 Test data에 대한 수치를 의미

보기에는 단순해 보이지만 T에 따른 P를 설정하는 것은 학습에 큰 영향을 끼치기 때문에 꽤나 복잡하고 목표에 따라 비슷한 데이터에도 다른 방식을 사용해야 함

어떠한 Data를 경험하는지에 따라 머신러닝 알고리즘을 구분할 수 있음.

ex) Unsupervised Learning: feature들만을 가지고 학습 P(X)
ex) Supervised Learning: feature & label P(Y|X)

하지만 사실 위의 두 가지의 경계는 모호한데 예를들어 Unsupervised Task는 아래와 같이 바꿔서 Supervised Learnig처럼 학습할 수도 있고, 이 외에도 Semi-Supervised Learning, Reinforcement Learning 등 다양한 형태의 경험이 존재함

$$p(\mathbf{x}) = \prod_{i=1}^{n} p(\mathrm{x}_i \mid \mathrm{x}_1, \ldots, \mathrm{x}_{i-1}).$$

Ⅱ Capacity, Overfitting and Underfitting

Q>Generalization이 뭡니까?

A>머신러닝에서의 주요 목표 중 하나는 학습 때 접하지 못했던 데이터에도 모델이 잘 작동하는 것. 이를 Generalization이라고 함. Test error==Generalization Error

Q>하지만 Training Set만 가지고 Test Error 을 낮출 수 있을까? Training Error 를 낮추는 것이 Test Error 를 낮춘다고 할 수 있을까?

A>Statistical Learning Theory에 의하면 위의 문제를 해결하기 위해서는 3가지의 가정이 필요하다. 동전 던지기를 생각하면 쉽게 이해됨

i: independent
i: identically distributed
d: drawn from the same probability distribution

Ⅱ Capacity, Overfitting and Underfitting

iid의 가정하에서는 train error와 test error가 같아지겠지만 실제 머신러닝에서는 먼저 Training data를 뽑고 여기에서 나온 Probability Distribution으로 test data를 평가한다.

$$\text{Training Error} \leq \text{Test Error}$$

그렇기 때문에 2가지 목표가 생기는데

① Training Error의 최소화 -> 못하면 Underfitting
② Training Error와 Test Error간의 격차 최소화 -> 못하면 Overfitting

우리는 모델의 Capacity를 변화시킴으로써 Underfitting, Overfitting 구간 사이의 목표하는 구간에 다다를 수 있음 (ideally)

Ⅱ Capacity, Overfitting and Underfitting

Q>Capacity가 뭔가요?

A>Capacity == complexity, expressive power, richness, or flexibility
즉, Capacity는 그 모델이 표현할 수 있는 범위라고 할 수 있다.

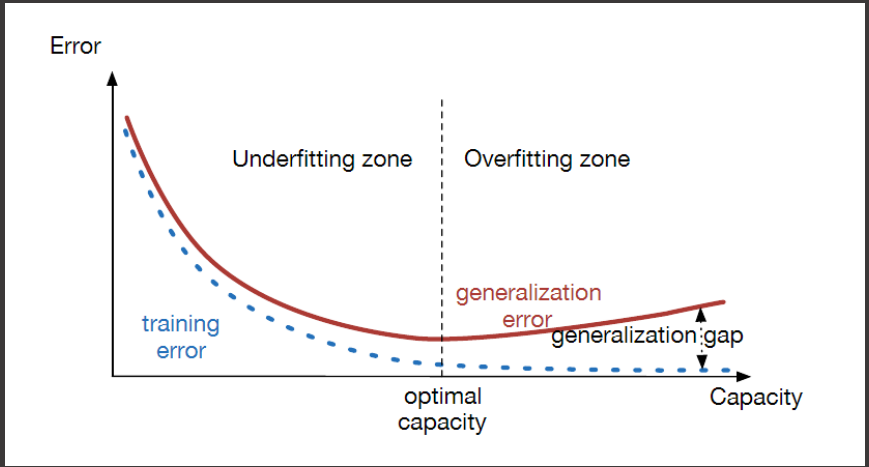
Capacity에 변화를 줄 수 있는 예시로는 Hypothesis Space의 변화가 있다.
Hypothesis space는 "set of possible approximations of function f that the algorithm can create" 으로 정의 될 수 있다.

ex) y=w*x+b의 hypothesis space는 모든 linear function
여기서 hypothesis space를 늘려 y=w1*x^2+w2*x+b로 만들면 linear function으로 표현할 수 없었던 범위까지 표현할 수 있기 때문에 Model Capacity가 늘어난다.
여기서 Capacity를 더 늘리면 Linear -> polynomial로 변화

Ⅱ Capacity, Overfitting and Underfitting

Q>주어진 data의 분포에 가까운 함수를 만드는 방법은 너무나 다양한데 만약 training data에서 같은 Performance를 낸다면 어떤 모델을 선택해야 할까?

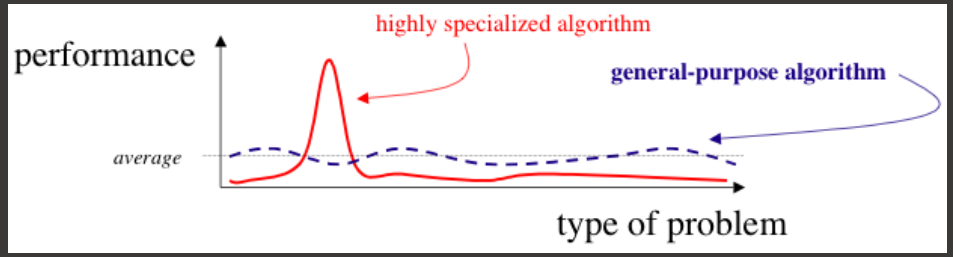
A>Occam's razor 및 확률적으로 생각해봤을 때 단순한 모델일수록 Generalize의 가능성이 높지만 어느 정도 Capacity는 있어야 함.



Ⅱ Capacity, Overfitting and Underfitting

Q>No Free Lunch Theorem이 뭔가요?

A>특정 문제에 최적화된 알고리즘은 다른 문제에 대해 그냥 찍는 것 보다 성능이 잘 안 나올 수도 있다는 정리



Q>Regularization이 뭔가요?

A>머신러닝 알고리즘의 generalization error는 감소시키되 training error에는 변화가 없도록 하는 방법/기법

ex) Linear Regression에서 weight decay $J(\boldsymbol{w}) = \mathrm{MSE}_{\mathrm{train}} + \lambda \boldsymbol{w}^{\top}\boldsymbol{w},$

Q>Hyperparameter란?

A>learning algorithm으로 학습시키지 않는 변수들을 뜻함.
ex)Polynomial Regression에서의 차원이나 Regularization λ

Q>학습을 시키지 않는 이유가 뭘까?

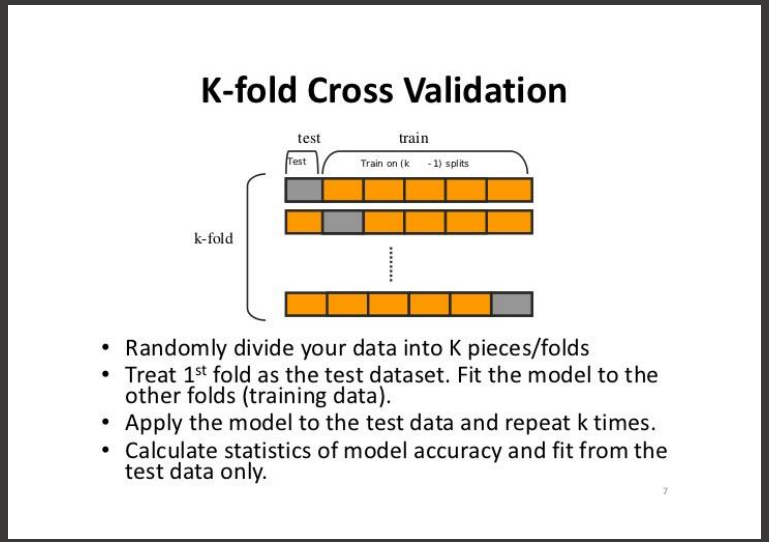
A>model capacity에 영향을 주는 변수도 학습되도록 하면 training set에 맞춰 무조건 capacity를 늘리도록 학습되어 overfitting을 유도하기 때문

Q>그럼 적절한 hyperparameter는 어떻게 찾을 수 있을까?

A>Training data의 일부를 학습 때 쓰지 말고 hyperparameter를 찾는데 쓰고 Validation Set이라 부르자

Q>Cross Validation이 뭡니까?

A>Cross Validation이란 Dataset이 작을 경우 Test set도 작아지는데 그렇게 되면 test set에서의 성능이 일반적인 성능이라고 말하기 힘들다. 그렇기 때문에 전체 데이터를 분할하여 일부는 트레이닝에 쓰고 일부는 테스트에 쓰는 것을 반복하여 평균적인 성능을 일반적인 성능으로 받아들이는 방법



IV Estimators, Bias and Variance

Q>Estimation이란 무엇인가?

A>Point Estimation은 주어진 data를 통해 한 지점을 예측하는 것

$$\hat{\boldsymbol{\theta}}_m = g(\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(m)}).$$

<- unit vector of θ

Q>Function Estimation도 있던데

A>Function Estimation은 한 점이 아닌 함수 자체를 예측하는 것

ex) Linear Regression의 경우 y=w*x+b일때 w,b를 예측하는 것이 예시

IV Estimators, Bias and Variance

Q>Bias란 무엇인가?

A>Bias란 예상 값과 실제 값의 차이를 의미하고 다음과 같이 정의된다.

$$\text{bias}(\hat{\boldsymbol{\theta}}_m) = \mathbb{E}(\hat{\boldsymbol{\theta}}_m) - \boldsymbol{\theta}$$

Q>Variance란 무엇인가?

A>말 그대로 분산을 의미한다. 이 두 가지 지표가 갖는 의미가 있는데
칸이 부족하니 다음 슬라이드에서 설명하는 걸로

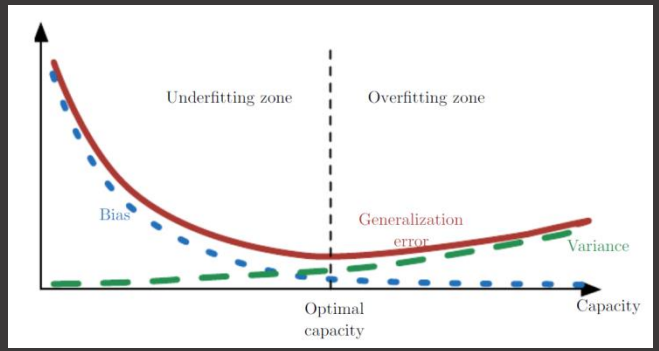
IV Estimators, Bias and Variance

Q>Bias와 Variance가 갖는 의미는 무엇인가?

A>Error를 측정하는 방법으로 Mean Squared Error(MSE)가 있는데 이를 풀어보면 다음과 같은 의미를 가지고 있다.

$$\begin{aligned} \text{MSE} &= \mathbb{E}[(\hat{\theta}_m - \theta)^2] \\ &= \text{Bias}(\hat{\theta}_m)^2 + \text{Var}(\hat{\theta}_m) \end{aligned}$$

두 가지를 가지고 그래프를 그려보면 아래와 같다.



Q> Consistency란 무엇인가?

A> 주로 어떠한 Estimator에 대해 train data가 무한으로 늘어난다면 예측 값이 True Value에 가까워지는가를 표현하는 개념

$$\lim_{m \to \infty} \hat{\theta}_m \xrightarrow{p} \theta.$$

ex) 데이터가 많아져도 실제 값에 수렴하지 않으면 weak consistency 또는 inconsistent / 수렴하면 consistent 하다고 표현

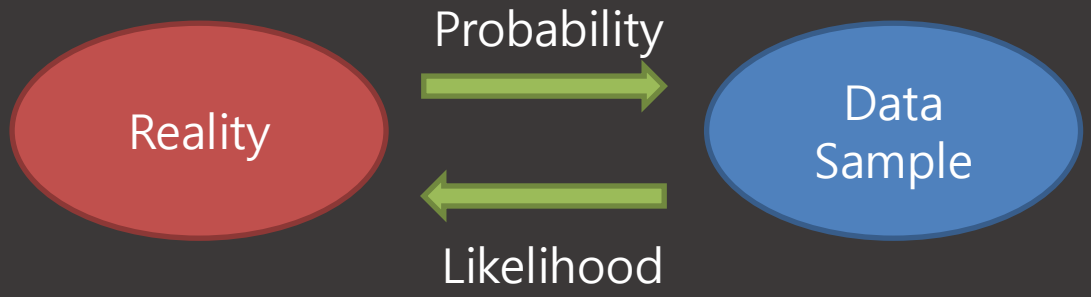
V Maximum Likelihood Estimation

Q>방금 Estimator의 consistency에 대해 배웠는데 수렴하는지도 중요하지만 일단 Estimator 모델이 있어야 consistency를 평가할 수 있지 않나? 모델은 어떻게 찾아야 할까?

A>최적의 모델을 찾는 원리로는 Maximum Likelihood Principle이 있다.

Q>Likelihood가 뭐지?

A>Probability의 반대 개념



V Maximum Likelihood Estimation

Q>그렇다면 Maximum Likelihood Estimation은?

A>Data sample을 가지고 만든 모델로 예측했을 때 실제 값과 일치할 확률로 모델의 성능을 평가하는 방법. θ에 대한 Maximum Likelihood Estimator는 아래와 같이 정의된다.

$$\begin{aligned}\boldsymbol{\theta}_{\mathrm{ML}} &= \arg\max_{\boldsymbol{\theta}} p_{\mathrm{model}}(\mathbb{X};\boldsymbol{\theta}) \\ &= \arg\max_{\boldsymbol{\theta}} \prod_{i=1}^{m} p_{\mathrm{model}}(\boldsymbol{x}^{(i)};\boldsymbol{\theta})\end{aligned}$$

하지만 확률의 곱이기 때문에 underflow같은 문제가 발생할 가능성이 있어서 합으로 변형해서 쓴다.

$$\boldsymbol{\theta}_{\mathrm{ML}} = \arg\max_{\boldsymbol{\theta}} \sum_{i=1}^{m} \log p_{\mathrm{model}}(\boldsymbol{x}^{(i)};\boldsymbol{\theta}).$$

V Maximum Likelihood Estimation

Q>실제와 가장 가까운 모델을 구했다고 해서 실제와 일치하는 경우는 거의 없을 텐데 모델의 error는 어떻게 구하는가?

A>Estimation으로 구한 값을 Kullback-Leibler(KL) Divergence를 통해 실제 값과 비교함으로써 얼마나 차이가 발생했는지 알 수 있다. KL Divergence의 정의는 다음과 같다.

$$D_{\mathrm{KL}}\left(\hat{p}_{\mathrm{data}} \| p_{\mathrm{model}}\right) = \mathbb{E}_{\mathbf{x} \sim \hat{p}_{\mathrm{data}}}\left[\log \hat{p}_{\mathrm{data}}(\boldsymbol{x}) - \log p_{\mathrm{model}}(\boldsymbol{x})\right].$$

이중에 모델 개선을 통해 바뀔 수 있는 부분만 보면

$$-\mathbb{E}_{\mathbf{x} \sim \hat{p}_{\mathrm{data}}}\left[\log p_{\mathrm{model}}(\boldsymbol{x})\right]$$

위와 같은 식이 나오는데 신기하게도 Cross Entropy Loss와 일치한다.
하지만 KL Divergence와 Cross Entropy Loss는 엄연히 다른 수식이다.

V Maximum Likelihood Estimation

Q>Maximum Likelihood Estimation을 conditional probability에 적용시킬 수 있나?

A>그렇다. 앞에서 배운 식을 conditional probability에 적용시키면 다음과 같다.

$$\boldsymbol{\theta}_{\mathrm{ML}} = \arg\max_{\boldsymbol{\theta}} P(\boldsymbol{Y} \mid \boldsymbol{X}; \boldsymbol{\theta}).$$

$$\boldsymbol{\theta}_{\mathrm{ML}} = \arg\max_{\boldsymbol{\theta}} \sum_{i=1}^{m} \log P(\boldsymbol{y}^{(i)} \mid \boldsymbol{x}^{(i)}; \boldsymbol{\theta}).$$

i.i.d.하다면
다음도 성립

Q>Maximum Likelihood Estimator는 Consistent 한가?

A>두 가지 조건을 만족할 때 consistent 하다.

①실제 P_data의 분포가 예측하는 model family P_model(;θ)에 속해 있어야 함
②하나의 θ에 대하여 실제 P_data의 분포가 하나의 값만 가져야 함

Q & A