Naïve Bayes & k-NN classifier

Jehyuk Lee

Department of Data Science

Kookmin University

국민대학교

Contents

1. Naïve Bayes Classifier

Bayesian Classification

- **조건부 확률 (Conditional Probabilty)**
  - 어떤 사건(A)이 발생한 상태에서 다른 사건(B)가 발생할 확률

$$P(B|A) = \frac{P(A,B)}{P(A)}$$

Bayesian Classification

- **Problem Statement (Classification)**
  - 우리에게 n개의 feature $X_1, X_2, \dots, X_n$가 있다고 해볼게요.
  - 이 때, 해당 데이터의 Y의 값을 예측해야 합니다.
  - 이를 수식으로 표현하면, 다음과 같이 나타낼 수 있습니다.

$$argmax_{Y} P(Y|X_1, X_2, \dots, X_n)$$

Bayesian Classification

- **Problem Statement (Classification)**
  - 우리에게 n개의 feature $X_1, X_2, \dots, X_n$가 있다고 해볼게요.
  - 이 때, 해당 데이터의 Y의 값을 예측해야 합니다.
  - 이를 수식으로 표현하면, 다음과 같이 나타낼 수 있습니다.

$$\boxed{\underset{Y}{argmax}}\, P(\boxed{Y} | \boxed{X_1, X_2, \dots, X_n})$$

n개의 feature값이 주어졌을 때,

레이블이 각 Y값이 될 확률을 구하고,

그 확률 값이 가장 큰 Y로 분류하세요

Bayesian Classification

- **Problem Statement (Classification)**
  - 우리에게 n개의 feature $X_1, X_2, \ldots, X_n$가 있다고 해볼게요.
  - 이 때, 해당 데이터의 Y의 값을 예측해야 합니다.
  - 이를 수식으로 표현하면, 다음과 같이 나타낼 수 있습니다.

$$\boxed{\underset{Y}{argmax}}\, P(\boxed{Y} | \boxed{X_1, X_2, \ldots, X_n})$$

n개의 feature값이 주어졌을 때,

레이블이 각 Y값이 될 확률을 구하고,

그 확률 값이 가장 큰 Y로 분류하세요

현재 우리는 이 값을 알지못해요.
(오히려 이 값을 알고 싶어요.)

Bayesian Classification

- **Class Conditional Independence Assumption**
  - 잠시, argmax부분은 빼고 수식만 보겠습니다.
  - Y=i일 때 수식 값은 다음과 같습니다.

$$P(Y = i|X_1, X_2, \dots, X_n) = \frac{P(X_1, X_2, \dots, X_n|Y = i)P(Y = i)}{P(X_1, X_2, \dots, X_n)}$$

Bayesian Classification

- **Class Conditional Independence Assumption**
  - 잠시, argmax부분은 빼고 수식만 보겠습니다.
  - Y=i일 때 수식 값은 다음과 같습니다.

$$\underbrace{\boxed{P(Y=i|X_1,X_2,\dots,X_n)}}_{\text{Posterior}}=\frac{\underbrace{\boxed{P(X_1,X_2,\dots,X_n|Y=i)}}_{\text{Likelihood}}\underbrace{\boxed{P(Y=i)}}_{\text{Prior}}}{P(X_1,X_2,\dots,X_n)}$$

Prior: class가 i일 확률

Likelihood: Class가 i일 때, 데이터가 발생할 확률

Posterior: 데이터가 주어졌을 때, Class가 i일 확률

Bayesian Classification

- **Class Conditional Independence Assumption**
  - 잠시, argmax부분은 빼고 수식만 보겠습니다.
  - Y=i일 때 수식 값은 다음과 같습니다.
  - 분모 부분을 다음과 같이 각 Y에 대해서 조건부 확률을 더해준 값으로 변경 가능합니다.

$$P(Y=i|X_1=x_1,X_2=x_2,\dots,X_n=X_n)=\frac{P(X_1=x_1,X_2=x_2,\dots,X_n=x_n|Y=i)P(Y=i)}{P(X_1=x_1,X_2=x_2,\dots,X_n=x_n)}$$

$$=\frac{P(X_1=x_1,X_2=x_2,\dots,X_n=x_n|Y=i)P(Y=i)}{\boxed{\sum_j P(X_1=x_1,X_2=x_2,\dots,X_n=x_n|Y=j)}}$$

변수 X_1의 값이 x_1, X_2의 값이 $x_2,\dots$일 때,
클래스 Y가 i일 확률

분자에 있는 term을 모든 class에 대해서 합산한 결과일 뿐입니다!

Bayesian Classification

- **Class Conditional Independence Assumption**
  - 잠시, argmax부분은 빼고 수식만 보겠습니다.
  - Y=i일 때 수식 값은 다음과 같습니다.
  - 분모 부분을 다음과 같이 각 Y에 대해서 조건부 확률을 더해준 값으로 변경 가능합니다.

$$P(Y=i|X_1=x_1,X_2=x_2,\ldots,X_n=X_n)=\frac{P(X_1=x_1,X_2=x_2,\ldots,X_n=x_n|Y=i)P(Y=i)}{P(X_1=x_1,X_2=x_2,\ldots,X_n=x_n)}$$

이 term을 잠시 보겠습니다!

$$=\frac{\boxed{P(X_1=x_1,X_2=x_2,\ldots,X_n=x_n|Y=i)}P(Y=i)}{\sum_j P(X_1=x_1,X_2=x_2,\ldots,X_n=x_n|Y=j)}$$

Bayesian Classification

- **Class Conditional Independence Assumption**
  - $P(X_1, X_2, \ldots, X_n|Y)$를 직접 모델링하여 구하는 것은 너무 어렵습니다.
    - N이 커질수록 엄청나게 많은 X조합이 생기기 때문이죠.
  - 그래서 각 속성은 서로 독립이라고 가정합니다.
    - 이러한 가정하에, 단변량 조건부 확률의 곱으로 나타냅니다.

$$P(X_1, X_2, \ldots, X_n|Y) = P(X_1|Y)P(X_2|Y)\cdots P(X_n|Y)$$

Under class conditional Independence Assumption

Bayesian Classification

- 자, 다시 원래수식으로 돌아가면…
  - 잠시, argmax부분은 빼고 수식만 보겠습니다.
  - Y=i일 때 수식 값은 다음과 같습니다.
  - 분모 부분을 다음과 같이 각 Y에 대해서 조건부 확률을 더해준 값으로 변경 가능합니다.

$$P(Y=i|X_1=x_1,X_2=x_2,\dots,X_n=X_n)=\frac{P(X_1=x_1,X_2=x_2,\dots,X_n=x_n|Y=i)P(Y=i)}{P(X_1=x_1,X_2=x_2,\dots,X_n=x_n)}$$

$$=\frac{P(X_1=x_1,X_2=x_2,\dots,X_n=x_n|Y=i)P(Y=i)}{\sum_j P(X_1=x_1,X_2=x_2,\dots,X_n=x_n|Y=j)}$$

$$=\frac{\prod_{k=1}^{n} P(X_k=x_k|Y=i)\cdot P(Y=i)}{\sum_j \prod_{k=1}^{n} P(X_k=x_k|Y=j)\cdot P(Y=j)}$$

Bayesian Classification

- **자, 다시 원래수식으로 돌아가면…**
  - 잠시, argmax부분은 빼고 수식만 보겠습니다.
  - Y=i일 때 수식 값은 다음과 같습니다.
  - 분모 부분을 다음과 같이 각 Y에 대해서 조건부 확률을 더해준 값으로 변경 가능합니다.

$$P(Y=i|X_1=x_1,X_2=x_2,\ldots,X_n=X_n)=\frac{P(X_1=x_1,X_2=x_2,\ldots,X_n=x_n|Y=i)P(Y=i)}{P(X_1=x_1,X_2=x_2,\ldots,X_n=x_n)}$$

$$=\frac{\boxed{P(X_1=x_1,X_2=x_2,\ldots,X_n=x_n|Y=i)}P(Y=i)}{\sum_j P(X_1=x_1,X_2=x_2,\ldots,X_n=x_n|Y=j)}$$

$$=\frac{\boxed{\prod_{k=1}^{n}P(X_k=x_k|Y=i)}\cdot P(Y=i)}{\sum_j\prod_{k=1}^{n}P(X_k=x_k|Y=j)\cdot P(Y=j)}$$

Bayesian Classification

- **예제: 스팸메일 분류기**
  - 입력 텍스트(메일 본문)이 주어졌을 때, 해당 메일이 스팸인지 아닌지를 분류
    - $P(Mail_{normal}|Text)$: 입력 텍스트가 주어졌을 때, 정상 메일일 확률
    - $P(Mail_{spam}|Text)$: 입력 텍스트가 주어졌을 때, 스팸 메일일 확률
  - 분류 기준
    - $P(Mail_{normal}|Text) > P(Mail_{spam}|Text)$ → Normal!
    - $P(Mail_{normal}|Text) \leq P(Mail_{spam}|Text)$ → Abnormal!

Bayesian Classification

- **예제: 스팸메일 분류기**
  - 입력 텍스트(메일 본문)이 주어졌을 때, 해당 메일이 스팸인지 아닌지를 분류
    - $P(Mail_{normal}|Text)$: 입력 텍스트가 주어졌을 때, 정상 메일일 확률
    - $P(Mail_{spam}|Text)$: 입력 텍스트가 주어졌을 때, 스팸 메일일 확률
  - By Bayes Theorem,
    - $P(Mail_{normal}|Text) = \dfrac{P(Text|Mail_{normal})P(Mail_{normal})}{P(Text)}$
    - $P(Mail_{spam}|Text) = \dfrac{P(Text|Mail_{spam})P(Mail_{spam})}{P(Text)}$

Bayesian Classification

- **예제: 스팸메일 분류기**
  - 입력 텍스트(메일 본문)이 주어졌을 때, 해당 메일이 스팸인지 아닌지를 분류
    - $P(Mail_{normal}|Text)$: 입력 텍스트가 주어졌을 때, 정상 메일일 확률
    - $P(Mail_{spam}|Text)$: 입력 텍스트가 주어졌을 때, 스팸 메일일 확률
  - By Class Conditional Independence Assumption, (Text : w_1, w_2, w_3의 세 개의 단어만)
    - $P(Mail_{normal}|Text) \propto \boxed{P(w_1|Mail_{normal})} P(w_2|Mail_{normal}) P(w_3|Mail_{normal}) P(Mail_{normal})$

      일반 메일에서 단어 w_1이 등장할 확률
    - $P(Mail_{spam}|Text) \propto \boxed{P(w_1|Mail_{spam})} P(w_2|Mail_{spam}) P(w_3|Mail_{spam}) P(Mail_{spam})$

      스팸 메일에서 단어 w_1이 등장할 확률

Pros and Cons

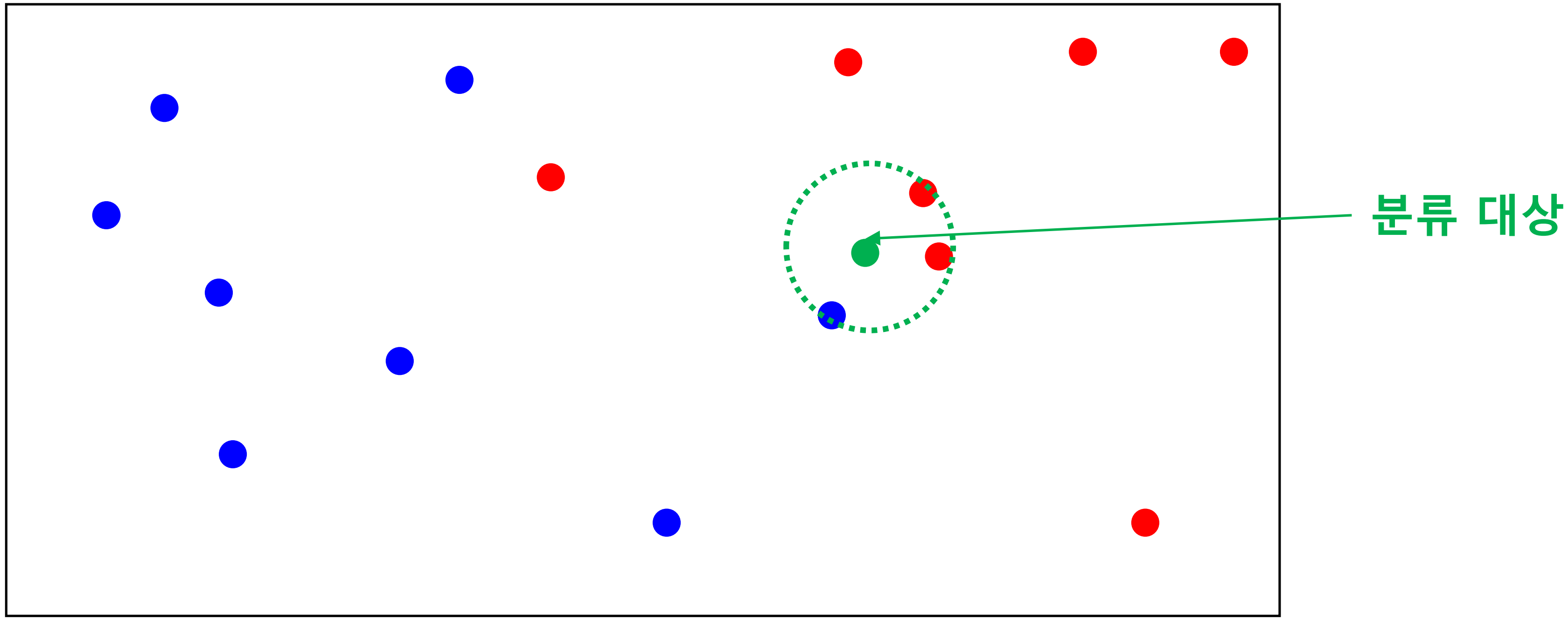
- **Pros**
  - 구현 및 실행이 쉽다
  - 적은 학습 데이터에서도 성능이 괜찮다
    - (Class Conditional Independent Assumption은 매우 이상적인 가정임에도)

- **Cons**
  - Class Conditional Independent Assumption을 만족시키기 어렵다
    - 대부분 변수간의 dependency가 존재
    - 어느정도 성능이 나오는 것이지, 좋은 성능이 나오기는 어렵다.

2. k-NN Classifier

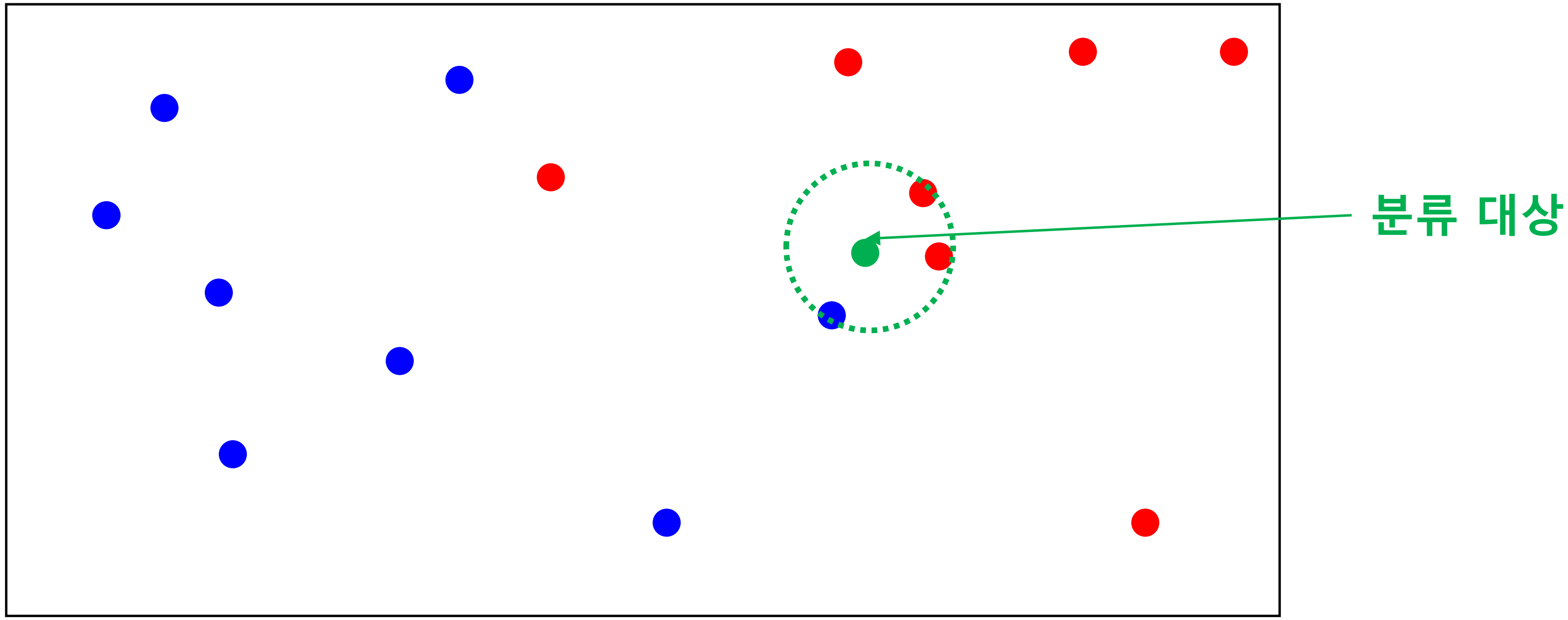
K-nearest neighbor(k-NN) Classifier

- **특징**
  - 분류하려는 대상과 가장 가까운 k개의 데이터를 찾은 뒤, label들을 voting하여 분류



K-nearest neighbor(k-NN) Classifier

- **특징**
  - Distance Measure로 임의의 measure 함수를 사용 가능
    - Euclidean Distance
    - Cosine Distance



Pros and Cons

- **Pros**
  - 간단한 모델이며 매우 직관적이다
- **Cons**
  - 검색 비용이 크다
    - 학습 데이터의 수만큼 거리 계산을 함
  - Distance 함수를 잘 정의해야 함
    - 문제에 적합한 Distance 함수를 잘 정의하는게 중요

Summary

- **Naïve Bayes**
  - Classification 문제: $arg \max_{Y} P(Y|X_1, X_2, \ldots, X_n)$
  - Bayes Theorem을 이용하여, $Posterior \propto Prior \cdot Likelihood$로 표현
  - Class Conditional Independence 가정하에서 Likelihood를 쉽게 계산이 가능하도록 함

- **k-NN**
  - 분류하려는 대상과 가장 가까운 k개의 데이터를 찾은 뒤, label들을 voting하여 분류
  - 임의의 distance measure를 사용할 수 있음 → 정교한 정의가 필요

End of the documents