Statistical Learning with Math and R

Jiye Shin

February 12, 2025

Department of Statistics
Sungshin Women's University

Outline

k-Nearest Neighbor Method

- 기존의 데이터에서 어떤 규칙을 만들지 않고, 새로운 데이터를 예측할 때 주변의 가장 가까운 데이터(k개의 이웃)를 참고하여 값을 결정하는 방법
- 데이터 $(x_1, y_1), (x_2, y_2), ..., (x_n, y_n)$이 주어졌을 때, 새로운 데이터 x의 값 y를 예측하고 싶다면, x와 가장 가까운 k개의 데이터를 찾고 그들의 다수결(majority vote)을 따르는 방식
- 거리(distance)는 유클리드 거리(euclidean distance) 를 사용

k-Nearest Neighbor Method 예제

- **주어진 데이터**

x_i	-2.1	-3.7	1.3	0.4	1.5
y_i	-1	1	0	0	1

- **새로운 데이터**

  $x_* = -2.2$

  $k = 2$

- **가장 가까운 두 개의 점**

  $x_1 = -2.1 (y_1 = -1)$

  $x_2 = -3.7 (y_2 = 1)$

- **결정 방법**

  $y_1 = -1,\ y_2 = 1 \rightarrow$ 동점 발생

  더 가까운 $x_1 = -2.1$를 기준으로 결정

KNN 알고리즘을 R 코드로 구현

- x 는 훈련 데이터이고, z 는 예측할 데이터
- 각 훈련 데이터와 z 사이의 유클리드 거리를 계산하여 배열 dis 에 저장
- 가장 가까운 k 개의 이웃 선택
- 가장 많이 등장한 클래스의 이름을 반환

```
knn.1=function(x,y,z,k){
  x=as.matrix(x); n=nrow(x); p=ncol(x); dis=array(dim=n)
  for(i in 1:n)dis[i]=norm(z-x[i,],"2")
  S=order(dis)[1:k]; ## The set of i such that their distances are minimized.
  u=sort(table(y[S]),decreasing=TRUE) ## Most Often y[i] and Frequency
  ## Tie-Breaking START
  while(length(u)>1 && u[1]==u[2]){ k=k-1; S=order(dis)[1:k]; u=sort(table(y[S
  ]),decreasing=TRUE)}
  ## Tie-Breaking END
  return(names(u)[1])
}
```

다중 예측을 위한 KNN 함수 확장

- 여러 개의 데이터 z 를 한 번에 처리하도록 개선한 버전

```
knn=function(x,y,z,k){
n=nrow(z); w=array(dim=n); for(i in 1:n)w[i]=knn.1(x,y,z[i,],k)
return(w)
}
```

Iris 데이터셋을 이용한 예제

- iris 데이터를 훈련/테스트 데이터로 나눔
- KNN 알고리즘 적용
- 예측값(w)과 실제값(ans)의 비교 표(혼동 행렬, confusion matrix)를 생성

```
df=iris;
n=150; train=sample(1:n,n/2,replace=FALSE); test=setdiff(1:n,train)
x=as.matrix(df[train,1:4]); y=as.vector(df[train,5])
z=as.matrix(df[test,1:4]); ans=as.vector(df[test,5])
w=knn(x, y, z, k=3)
table(w,ans)
```

```
##             ans
## w            setosa versicolor virginica
##   setosa         25          0         0
##   versicolor      0         21         4
##   virginica       0          2        23
```

Outline

ROC Curves

- 분류 모델의 성능을 평가하는 데 사용되는 그래프
- True Positive Rate (TPR, 재현율) 와 False Positive Rate (FPR, 위양성률) 간의 관계를 보여주는 그래프
- ROC 곡선 아래 면적(AUC)이 1에 가까울수록 좋은 모델

질병 진단 예제로 본 혼동행렬

	Sick	Healthy
Treating as sick	True Positive	False Positive
Treating as healthy	False Negative	True Negative

- True Positive (TP, 진짜 양성): 실제로 아픈 사람을 병자로 진단한 경우 (정확한 판단)
- False Positive (FP, 거짓 양성, Type I Error): 건강한 사람을 병자로 오진한 경우
- False Negative (FN, 거짓 음성, Type II Error): 실제로 아픈 사람을 건강하다고 판단한 경우
- True Negative (TN, 진짜 음성): 건강한 사람을 건강하다고 정확하게 진단한 경우

TPR과 FPR의 뜻과 식

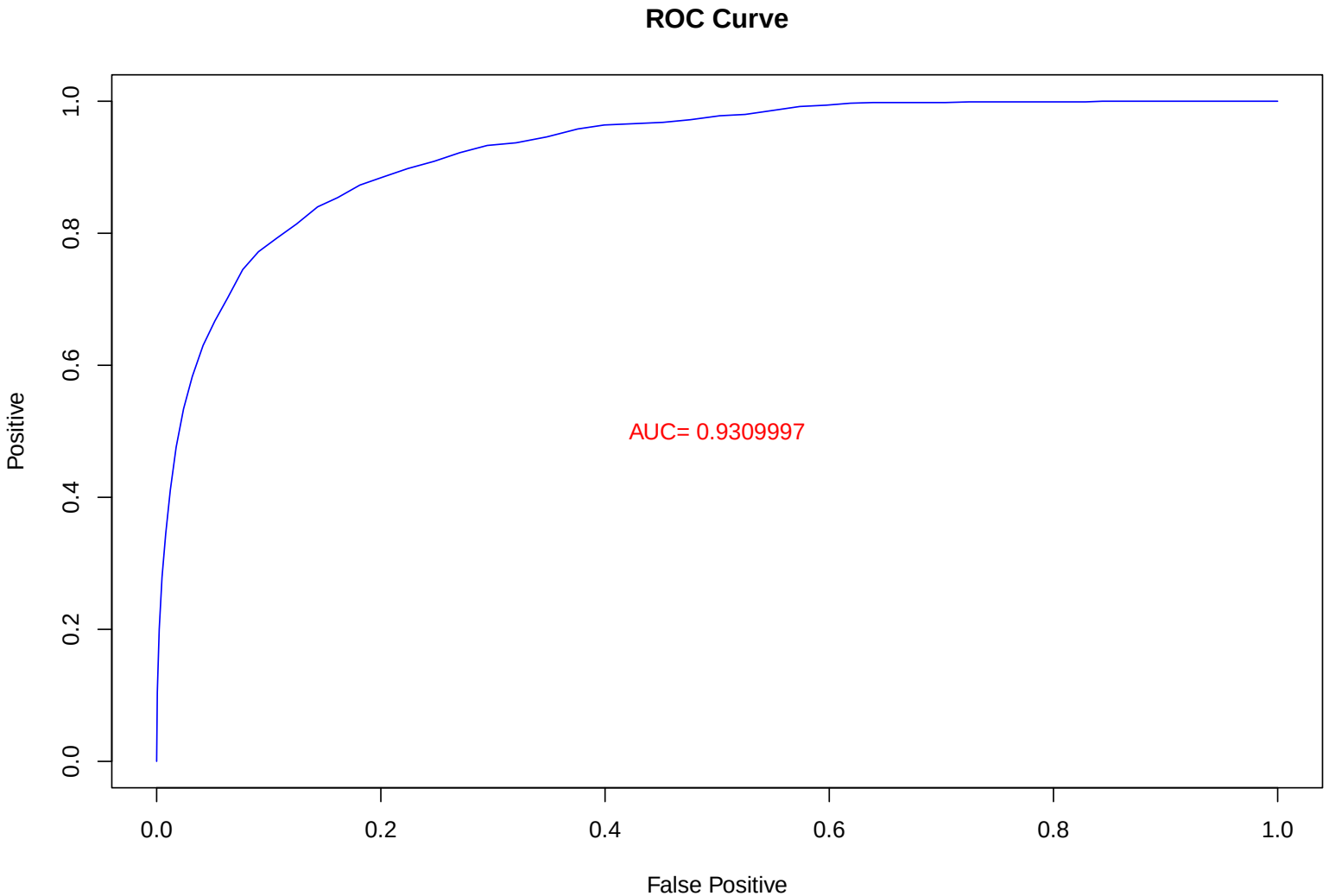
$$Power = \frac{TP}{TP + FN} = 1 - \beta$$

$$FalsePositiveRate = \frac{FP}{FP + TN} = \alpha$$

- True Positive Rate (TPR, Power)는 실제로 아픈 사람을 아프다고 진단하는 확률
- False Positive Rate (FPR)는 실제로 건강한 사람을 병자로 오진하는 비율

질병진단으로 본 ROC곡선 예제

- X축: False Positive Rate (FPR) → 건강한 사람을 병자로 잘못 분류한 비율
- Y축: True Positive Rate (TPR) → 실제 병자를 올바르게 병자로 분류한 비율
- ROC 곡선은 모델의 판별 성능을 보여줌
- AUC 값이 높을수록 모델이 병자와 건강한 사람을 더 잘 구별



감사합니다