회귀방법

수치 예측을 위한 머신러닝

학습목표

- 회귀(regression)는 수치 관계의 크기와 강도를 모델링하는 통계기법

- 회귀 분석용 데이터 처리와 회귀 모델을 추정하고 해석하는 방법

- 수치 예측 작업에 의사 결정 트리 분류기를 적용하는 회귀 트리와 모델 트리라는 한 쌍의 하이브리드 기법

회귀의 이해

- 회귀
  - 한 개의 수치 종속변수(dependent variable, 예측값)과 한 개 이상의 수치 독립변수(independent variable, 예측변수) 사이의 관계를 명시하는 방법
  - 가장 단순한 형태는 독립변수와 종속변수가 직선 관계라고 가정.
  - 머신이 하는 일은 x와 y의 관계를 가장 잘 나타내도록 a(절편)와 b(기울기)를 구하고 오차(error)의 범위를 정량화하는 방법을 연구.
  - 사례
    - 경제학, 사회학, 심리학, 물리학, 생태학과 같은 다양한 분야의 과학 연구에 사용하고자 측정된 특성에 따라 인구와 개인이 어떻게 변화하는지 검토
    - 임상 의약품 실험, 공학 안전시험, 마케팅 조사와 같은 사건과 반응 간의 인과관계의 정량화
    - 보험금 청구, 자연재해 피해, 선거 결과, 범죄율 예측과 같이 알려진 기준이 있을 때 미래의 행위를 예측하는 데 사용되는 패턴의 식별

$$y = a + bx$$

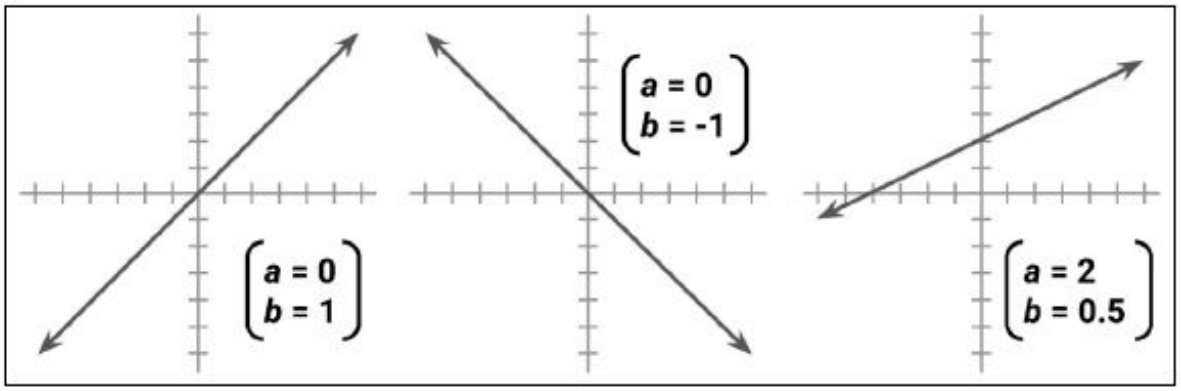


그림 6.1: 여러 기울기와 절편을 가진 직선의 예

단순 선형 회귀

- 예제 : 1986년 미국 우주 왕복선 챌린저의 로켓 부스터 고장으로 인한 사고 원인 분석
  - 원인 : 발사 온도
    - 로켓 연결 부분의 밀봉을 담당하는 패킹용 고무 오링(O-ring)이 40°F 미만에서 테스트 되지 않았다.
    - 발사 당일의 날씨가 평소와 달리 매우 춥고 영하인 상태였다.
  - 가설 : 온도가 낮을 때 부품이 더 잘 부서지고 적절히 밀봉될 수 없게 만들어 연료 유출될 가능성이 높아 사고 위험을 초래한다.
  - 분석
    - 온도와 오링의 고장 사이의 관계 분석
    - 발사 시의 예상 온도에 대한 실패 가능성을 예측할 수 있는 회귀 모델 구축

- 산점도를 통한 변수간 관계 파악
  - 온도와 오링의 손상 관계 파악

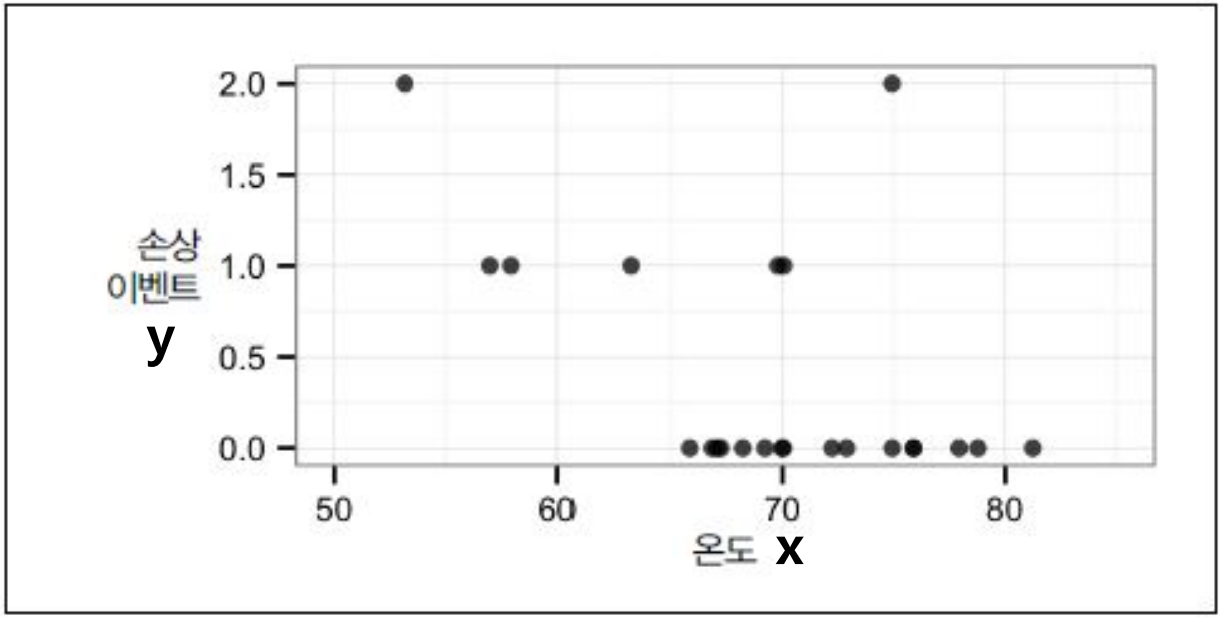


그림 6.2: 우주 왕복선 오링 손상과 발사 온도

고온에서 발사할 때 오링의 손상 정도가 적어지는 것을 알 수 있다.

- 회귀계수 추정
  - 최소 제곱 추정법(OLS, Ordinary Least Squares)
    - 회귀계수(기울기와 절편)은 오차 제곱합(SSE, Sum of the Squared Error)이 최소화되게 선택
    - 오차는 y의 실제값과 예측값 사이의 차이

$$\sum \left(y_i - \hat{y}_i\right)^2 = \sum e_i^2$$

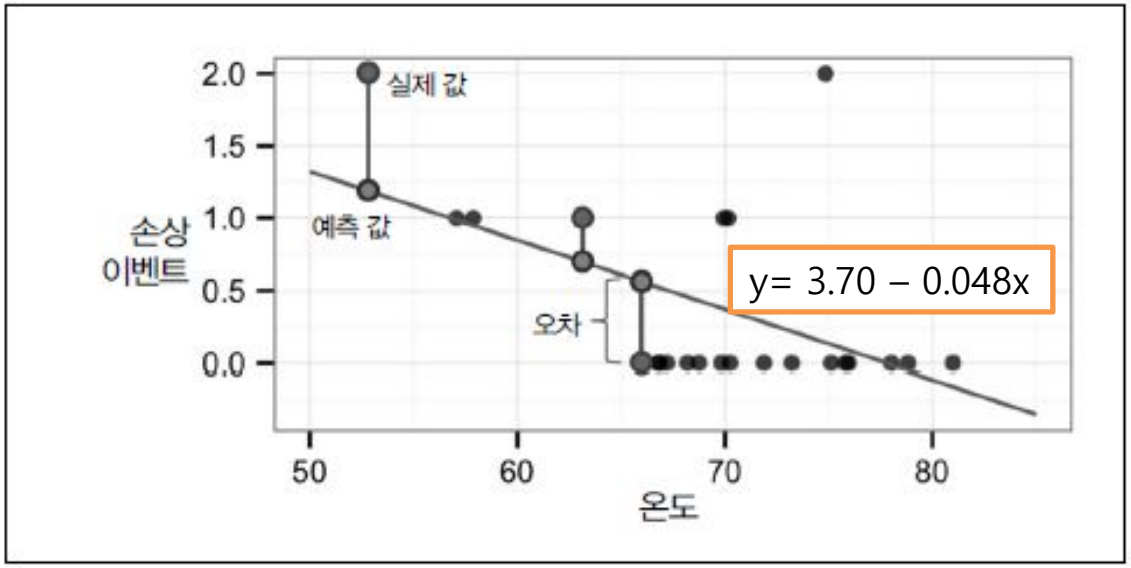


그림 6.4: 회귀선 예측이 실제 값보다 잔차만큼 차이난다.

절편 : $a = \overline{y} - b\overline{x}$

기울기 : $b = \dfrac{\sum (x_i - \overline{x})(y_i - \overline{y})}{\sum (x_i - \overline{x})^2}$ = $b = \dfrac{\mathrm{Cov}(x, y)}{\mathrm{Var}(x)}$

$$\mathrm{Var}(x) = \frac{\sum (x_i - \overline{x})^2}{n} \qquad \mathrm{Cov}(x, y) = \frac{\sum (x_i - \overline{x})(y_i - \overline{y})}{n}$$

- 상관관계
  - 두 변수의 관계가 직선에 가깝게 따르는 정도
  - 0.1 ~ 0.3 : 약한 양의 상관관계
  - 0.3 ~ 0.5 : 보통 양의 상관관계
  - 0.5 ~ : 강한 양의 상관관계

$$\rho_{x,y} = \mathrm{Corr}(x,y) = \frac{\mathrm{Cov}(x,y)}{\sigma_x \sigma_y} \quad = -0.511$$

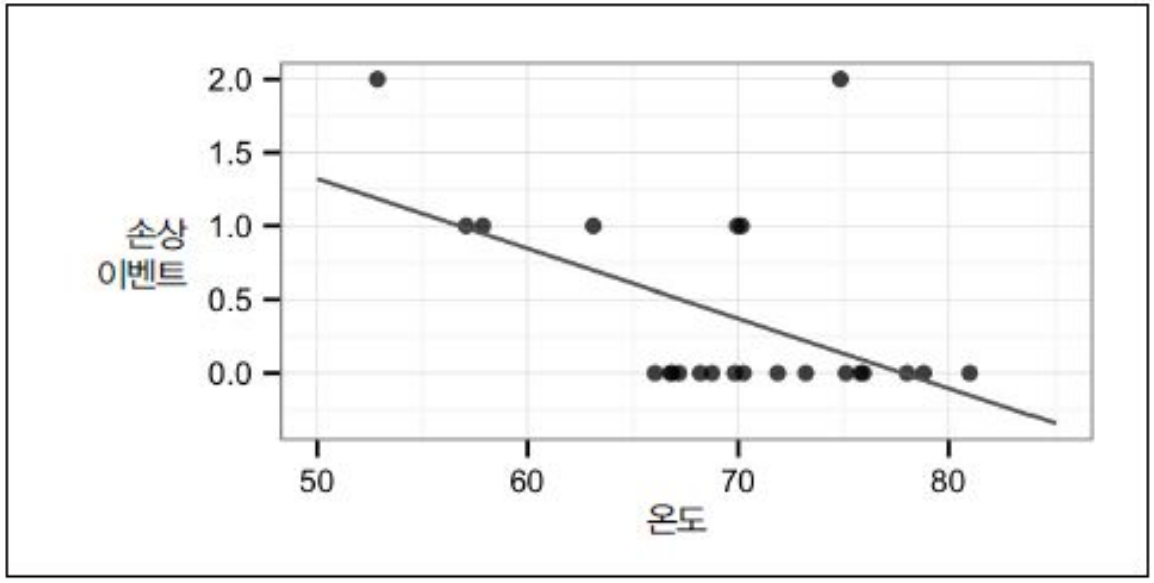
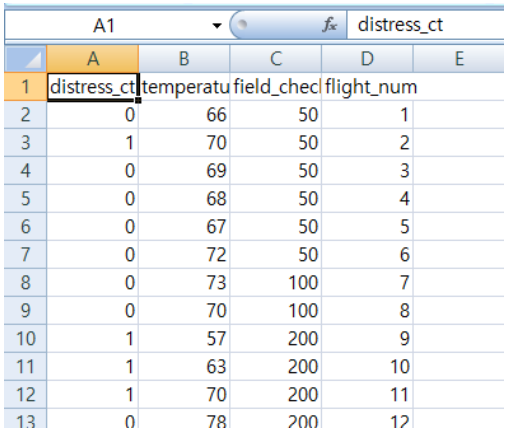


그림 6.3: 손상 이벤트와 발사 온도 사이의 관계를 모델링하는 회귀선



	A	B	C	D	E
1	distress_ct	temperatu	field_chec	flight_num	
2	0	66	50	1	
3	1	70	50	2	
4	0	69	50	3	
5	0	68	50	4	
6	0	67	50	5	
7	0	72	50	6	
8	0	73	100	7	
9	0	70	100	8	
10	1	57	200	9	
11	1	63	200	10	
12	1	70	200	11	
13	0	78	200	12	

종속변수 : distress_ct(오링의 손상횟수)

독립변수 : temperature(발사온도)

R code :

```
launch <- read.csv("challenger.csv")
str(launch)

#산점도
plot(launch$temperature, launch$distress_ct)
abline(lm(launch$distress_ct ~ launch$temperature))

#회귀계수
b <- cov(launch$temperature, launch$distress_ct) / var(launch$temperature)
a <- mean(launch$distress_ct) - b * mean(launch$temperature)

#상관계수
1.    r <- cov(launch$temperature, launch$distress_ct) / (sd(launch$temperature) * sd(launch$distress_ct))
2. cor(launch$temperature, launch$distress_ct)

#회귀모형
model <- lm(distress_ct ~ temperature, data = launch)
summary(model)
```

```
Call:
lm(formula = distress_ct ~ temperature, data = launch)

Residuals:
    Min      1Q  Median      3Q     Max
-0.5608 -0.3944 -0.0854  0.1056  1.8671

Coefficients:
            Estimate Std. Error t value  Pr(>|t|)
(Intercept)  3.69841    1.21951   3.033  0.00633 **
temperature -0.04754    0.01744  -2.725  0.01268 *

---
Signif. codes:
0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.5774 on 21 degrees of freedom
Multiple R-squared:  0.2613,        Adjusted R-squared:  0.2261
F-statistic: 7.426 on 1 and 21 DF,  p-value: 0.01268
```

다중 선형 회귀

- 하나 이상의 독립변수를 가지는 회귀모형
- 장점
  - 수치 데이터를 모델링하기 위한 가장 일반적인 방법
  - 어떤 모델링 작업에도 적용
  - 특징과 결과 간의 관계에 대한 강도와 크기 추정치 제공
- 단점
  - 데이터에 대한 강한 가정
  - 모델 형태가 사용자에 의해 미리 지정
  - 누락 데이터 처리 안함
  - 수치 특징만 처리, 범주 데이터는 별도 분석
  - 모델의 이해를 위해 통계적 지식이 필요

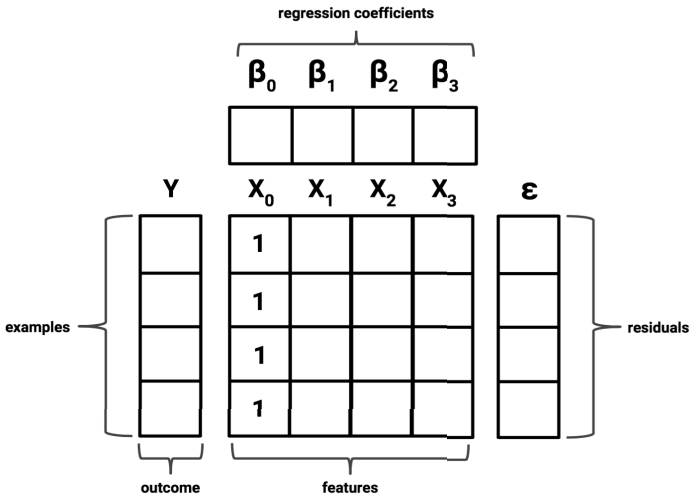
$$y = \alpha + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_i x_i + \varepsilon$$

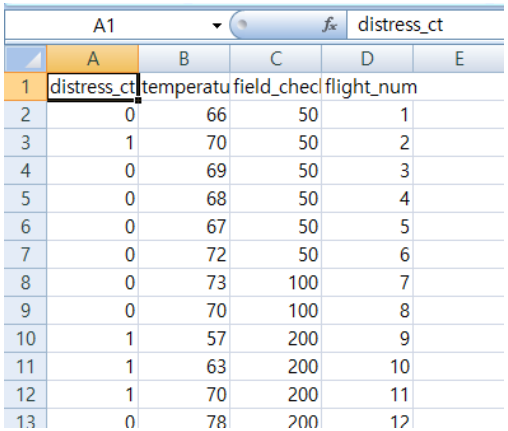
$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_i x_i + \varepsilon$$

$$y = \beta_0 x_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_i x_i + \varepsilon$$

$$\mathrm{Y} = \beta \mathrm{X} + \varepsilon$$

$$\hat{\beta} = \left(\mathrm{X}^{\mathrm{T}}\mathrm{X}\right)^{-1}\mathrm{X}^{\mathrm{T}}\mathrm{Y}$$





	A	B	C	D	E
1	distress_ct	temperatu	field_chec	flight_num	
2	0	66	50	1	
3	1	70	50	2	
4	0	69	50	3	
5	0	68	50	4	
6	0	67	50	5	
7	0	72	50	6	
8	0	73	100	7	
9	0	70	100	8	
10	1	57	200	9	
11	1	63	200	10	
12	1	70	200	11	
13	0	78	200	12	

종속변수 : distress_ct(오링의 손상횟수)

독립변수 : temperature(발사온도)

field check pressure(현장검사압력)

flight_num(발사번호)

R code :

```
launch <- read.csv("challenger.csv")
str(launch)

# 다중회귀모형
model_multi <- lm(distress_ct ~ ., data = launch)
model_multi
summary(model_multi)

# 사용자 정의 함수
reg <- function(y, x) {
  x <- as.matrix(x)
  x <- cbind(Intercept = 1, x)
  b <- solve(t(x) %*% x) %*% t(x) %*% y
  colnames(b) <- "estimate"
  print(b)
}
reg(y = launch$distress_ct, x = launch[2])
reg(y = launch$distress_ct, x = launch[2:4])
```

```
Call:
lm(formula = distress_ct ~ temperature + field_check_pressure +
    flight_num, data = launch)

Residuals:
     Min       1Q   Median       3Q      Max
-0.65003 -0.24414 -0.11219  0.01279  1.67530

Coefficients:
                     Estimate Std. Error t value Pr(>|t|)
(Intercept)          3.527093   1.307024 2.699   0.0142 *
temperature         -0.051386   0.018341 -2.802   0.0114 *
field_check_pressure  0.001757   0.003402 0.517   0.6115
flight_num            0.014293   0.035138 0.407   0.6887

---
Signif. codes:
0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.565 on 19 degrees of freedom
Multiple R-squared:   0.36,         Adjusted R-squared:  0.259
F-statistic: 3.563 on 3 and 19 DF,  p-value: 0.03371
```

실습예제: 선형 회귀를 이용한 의료비 예측

- 분석 목적
  - 환자 데이터를 이용해 인구 집단별로 평균 의료비를 예측하여 연간 보험료를 산정하는데 활용하고자 한다.

- 분석 데이터
  - 미국 통계국의 인구 통계 자료를 이용한 환자의 가상 의료비 데이터(insurance.csv)
  - 1338개의 인스턴스
  - 7개의 특징
    - age(나이), sex(성별), bmi(체질량지수), children(자녀수/부양가족 수), smoker(흡연여부), region(거주 지역)

	A	B	C	D	E	F	G	H
1	age	sex	bmi	children	smoker	region	expenses	
2	19	female	27.9	0	yes	southwest	16884.92	
3	18	male	33.8	1	no	southeast	1725.55	
4	28	male	33	3	no	southeast	4449.46	
5	33	male	22.7	0	no	northwest	21984.47	
6	32	male	28.9	0	no	northwest	3866.86	
7	31	female	25.7	0	no	southeast	3756.62	
8	46	female	33.4	1	no	southeast	8240.59	
9	37	female	27.7	3	no	northwest	7281.51	
10	37	male	29.8	2	no	northeast	6406.41	
11	60	female	25.8	0	no	northwest	28923.14	
12	25	male	26.2	0	no	northeast	2721.32	
13	62	female	26.3	0	yes	southeast	27808.73	
14	23	male	34.4	0	no	southwest	1826.84	
15	56	female	39.8	0	no	southeast	11090.72	
16	27	male	42.1	0	yes	southeast	39611.76	
17	19	male	24.6	1	no	southwest	1837.24	
18	52	female	30.8	1	no	northeast	10797.34	

Multiple regression modeling syntax

using the `lm()` function in the `stats` package

Building the model:

```
m <- lm(dv ~ iv, data = mydata)
```

- `dv` is the dependent variable in the `mydata` data frame to be modeled
- `iv` is an R formula specifying the independent variables in the `mydata` data frame to use in the model
- `data` specifies the data frame in which the `dv` and `iv` variables can be found

Making predictions:

```
p <- predict(m, test)
```

- `m` is a model trained by the `lm()` function
- `test` is a data frame containing test data with the same features as the training data used to build the model.

The function will return a vector of predicted values.

Example:

```
ins_model <- lm(charges ~ age + sex + smoker,
                data = insurance)
ins_pred <- predict(ins_model, insurance_test)
```

```
insurance <- read.csv("insurance.csv", stringsAsFactors = TRUE)

#데이터 구조
str(insurance)

#데이터 탐색
summary(insurance$expenses)
hist(insurance$expenses)
table(insurance$region)

#상관행렬
cor(insurance[c("age", 'bmi','children','expenses')])

#산포도 행렬
pairs(insurance[c("age", 'bmi','children','expenses')])
pairs.panels(insurance[c("age", 'bmi','children','expenses')])

#모델 훈련
ins_model <- lm(expenses ~ ., data=insurance)
ins_model

#모델 평가
summary(ins_model)
```

- Dummy 변수 생성
  - 범주형 자료를 수치형 자료로 변환하기 위해 변수의 범주별로 더미 변수를 생성.
  - 더미 변수는 관측이 명시된 범주에 속하면 1, 그렇지 않으면 0으로 설정 (이진 변수)
    - 성별(sex)
      - sex=female을 기준으로 하면 sexmale=0, sexfemale=1
    - Region=regionnortheast을 기준
    - Smoker=no를 기준
  - 더미 변수 개수 = (명목변수 범주 – 1)개

```
Call:
lm(formula = expenses ~ ., data = insurance)

Residuals:
     Min       1Q   Median       3Q      Max
-11302.7  -2850.9   -979.6   1383.9  29981.7

Coefficients:
                 Estimate Std. Error t value Pr(>|t|)
(Intercept)     -11941.6      987.8 -12.089  < 2e-16 ***
age                256.8       11.9  21.586  < 2e-16 ***
sexmale           -131.3      332.9  -0.395 0.693255
bmi                339.3       28.6  11.864  < 2e-16 ***
children           475.7      137.8   3.452 0.000574 ***
smokeryes        23847.5      413.1  57.723  < 2e-16 ***
regionnorthwest   -352.8      476.3  -0.741 0.458976
regionsoutheast  -1035.6      478.7  -2.163 0.030685 *
regionsouthwest   -959.3      477.9  -2.007 0.044921 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 6062 on 1329 degrees of freedom
Multiple R-squared:  0.7509,  Adjusted R-squared:  0.7494
F-statistic: 500.9 on 8 and 1329 DF,  p-value: < 2.2e-16
```

- 모델 성능 개선
  - 나이에 대한 비선형 항목 추가(다항회귀식)

```
insurance$age2 <- insurance$age^2
```

  - 비만에 대한 이진 지시 변수 생성
  - BMI가 30이상(비만)인 사람의 의료비에 대한 영향

```
insurance$bmi30 <- ifelse(insurance$bmi >= 30, 1, 0)
```

  - 비만과 흡연 간의 상호작용 명시

```
bmi30*smoker
```

최종 개선 모형

```
ins_model2 <- lm(expenses ~ age + age2 + children + bmi + sex + bmi30*smoker +
                                                              region, data = insurance)

summary(ins_model2)
```

```
Call:
lm(formula = expenses ~ age + age2 + children + bmi + sex + bmi30 *
    smoker + region, data = insurance)

Residuals:
     Min       1Q   Median       3Q      Max
-17297.1  -1656.0  -1262.7   -727.8  24161.6

Coefficients:
                  Estimate Std. Error t value Pr(>|t|)
(Intercept)       139.0053  1363.1359   0.102 0.918792
age               -32.6181    59.8250  -0.545 0.585690
age2                3.7307     0.7463   4.999 6.54e-07 ***
children          678.6017   105.8855   6.409 2.03e-10 ***
bmi               119.7715    34.2796   3.494 0.000492 ***
sexmale          -496.7690   244.3713  -2.033 0.042267 *
bmi30            -997.9355   422.9607  -2.359 0.018449 *
smokeryes       13404.5952   439.9591  30.468  < 2e-16 ***
regionnorthwest  -279.1661   349.2826  -0.799 0.424285
regionsoutheast  -828.0345   351.6484  -2.355 0.018682 *
regionsouthwest -1222.1619   350.5314  -3.487 0.000505 ***
bmi30:smokeryes 19810.1534   604.6769  32.762  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 4445 on 1326 degrees of freedom
Multiple R-squared:  0.8664,  Adjusted R-squared:  0.8653
F-statistic: 781.7 on 11 and 1326 DF,  p-value: < 2.2e-16
```

- 회귀 모델로 예측
  - 미래의 건강 보험 등록자의 비용을 예측해보자.

```
#회귀모델로 예측
insurance$pred <- predict(ins_model2, insurance)
cor(insurance$pred, insurance$expenses)
plot(insurance$pred, insurance$expenses)
abline(a=0, b=1, col='red', lwd=3, lty=2)

#northeast에 사는 두 자녀를 둔 30세 과제중 비흡연 남자의 보험 비용 추정
pred_male <- predict(ins_model2, data.frame(age=30, age2=30^2, children=2, bmi=30,
sex='male', bmi30=1, smoker='no', region='northeast'))

#동일 조건의 여성의 경우
pred_female <- predict(ins_model2, data.frame(age=30, age2=30^2, children=2, bmi=30,
sex='female', bmi30=1, smoker='no', region='northeast'))

pred_male - pred_female

#동일 조건에서 자녀가 없다면
pred_child <- predict(ins_model2, data.frame(age=30, age2=30^2, children=0, bmi=30,
sex='female', bmi30=1, smoker='no', region='northeast'))

pred_child

pred_female - pred_child
678.601683*2
```

회귀 트리와 모델 트리

- 수치 예측용 트리
  - 회귀 트리(regression trees)
    - CART(Classification and Regression Tree)알고리즘
    - 선형 회귀 방법을 사용하지 않고 leaf노드에 도달하는 인스턴스의 평균값으로 예측
    - 분할기준 (SDR: Standard Deviation Reduction)
      - 표준 편차 축소 $\mathrm{SDR} = sd(T) - \sum_i \frac{|T_i|}{|T|} \times sd(T_i)$
      - sd(T) 함수는 전체집합 T의 표준 편차
      - T_i는 T의 분할된 집합, |T| 는 집합 T의 관측 개수
      - 분할 이전의 표준 편차와 분할 이후의 가중 표준 편차를 비교해서 표준 편차의 축소를 측정
  - 모델 트리(model trees)
    - 회귀 트리와 거의 같은 방식으로 성장하지만, leaf 노드에 도달한 인스턴스로 다중 회귀 모델을 구축.
    - leaf 노드에 따라 몇 십 개 또는 몇 백 개의 다중 회귀 모델을 구축 가능.
    - 회귀 트리보다 해석은 복잡하지만 모델의 정확성이 높아짐.

회귀 트리와 모델 트리의 특징

장점	단점
의사 결정 트리의 장점과 수치 데이터를 모델링하는 능력을 결합	선형 회귀만큼 잘 알려져 있지 않다.
사용자가 모델을 미리 명시하지 않아도 된다.	많은 양의 훈련 데이터가 필요하다.
자동 특징 선택을 사용하기 때문에 아주 많은 개수의 특징이 이 방식에 사용될 수 있다.	결과에 대한 개별 특징의 전체적인 순영향을 알아내기가 어렵다.
선형 회귀보다 일부 데이터 타입에 아주 잘 맞는다.	큰 트리는 회귀 모델보다 해석하기가 좀 더 어려워질 수 있다.
모델을 해석하는 데 많은 통계 지식이 없어도 가능하다.	

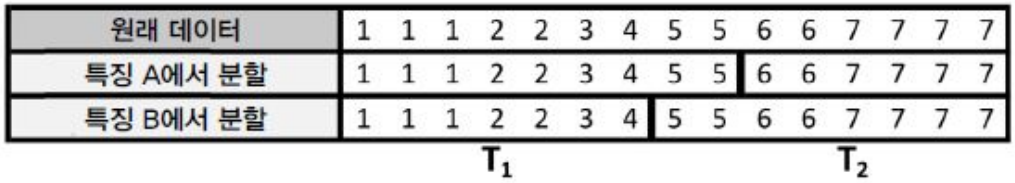


그림 6.10: 알고리즘은 특징 A와 B에 대한 분할을 고려하고, 이는 서로 다른 T_1과 T_2 그룹을 생성한다.

```
# set up the data
tee <- c(1, 1, 1, 2, 2, 3, 4, 5, 5, 6, 6, 7, 7, 7, 7)
at1 <- c(1, 1, 1, 2, 2, 3, 4, 5, 5)
at2 <- c(6, 6, 7, 7, 7, 7)
bt1 <- c(1, 1, 1, 2, 2, 3, 4)
bt2 <- c(5, 5, 6, 6, 7, 7, 7, 7)

# compute the SDR
sdr_a <- sd(tee) - (length(at1) / length(tee) * sd(at1) + length(at2) / length(tee) * sd(at2))
sdr_b <- sd(tee) - (length(bt1) / length(tee) * sd(bt1) + length(bt2) / length(tee) * sd(bt2))

# compare the SDR for each split
sdr_a
sdr_b
```

- sdr_a = 1.202815
- sdr_b = 1.392751
  - B 방법을 선택 : 표준편차가 더 많이 감소되어 동질성이 증가

- 단 한 번의 분할로 트리 성장이 멈췄다고 가정했을 때
  - 회귀 트리의 예측
    - 인스턴스가 T1에 배치 : 모델은 mean(bt1)를 예측
    - 인스턴스가 T2에 배치 : 모델은 mean(bt2)를 예측
  - 모델 트리의 예측
    - 회귀 트리에서 한 단계 더 진행
      - T1과 T2에 대한 각각의 선형 회귀 모델을 구축
      - 두 선형 모델 중 하나를 이용해 새로운 예시에 대한 예측

예제 : 와인 품질 평가

- 분석 목적
  - 와인 평가 모델 개발

- 데이터
  - 화이트 와인 데이터
  - 4898개의 와인 샘플, 11가지 화학 속성
  - 산도(acidity), 당 함유량(sugar content), 염화물(chlorides), 황(sulfur), 알코올(alcohol), pH, 밀도(density)
  - 종속변수
    - 와인 품질(quality) 척도 : 0(매우 나쁨) ~ 10(매우 우수)

- 분석 알고리즘
  - 회귀트리(rpart 함수 사용)
  - 모델트리(cubist 함수 사용)

```
# 와인 평가 모델 개발
wine <- read.csv("whitewines.csv")

# 데이터 탐색 및 준비
str(wine)
table(wine$quality)
hist(wine$quality)
summary(wine)

# 훈련 데이터 및 테스트 데이터 생성
wine_train <- wine[1:3750,]
wine_test <- wine[3751:4898,]

# 회귀트리 모델 훈련
install.packages("rpart")
library(rpart)
m.rpart <- rpart(quality ~ ., data=wine_train)
m.rpart
summary(m.rpart)
```

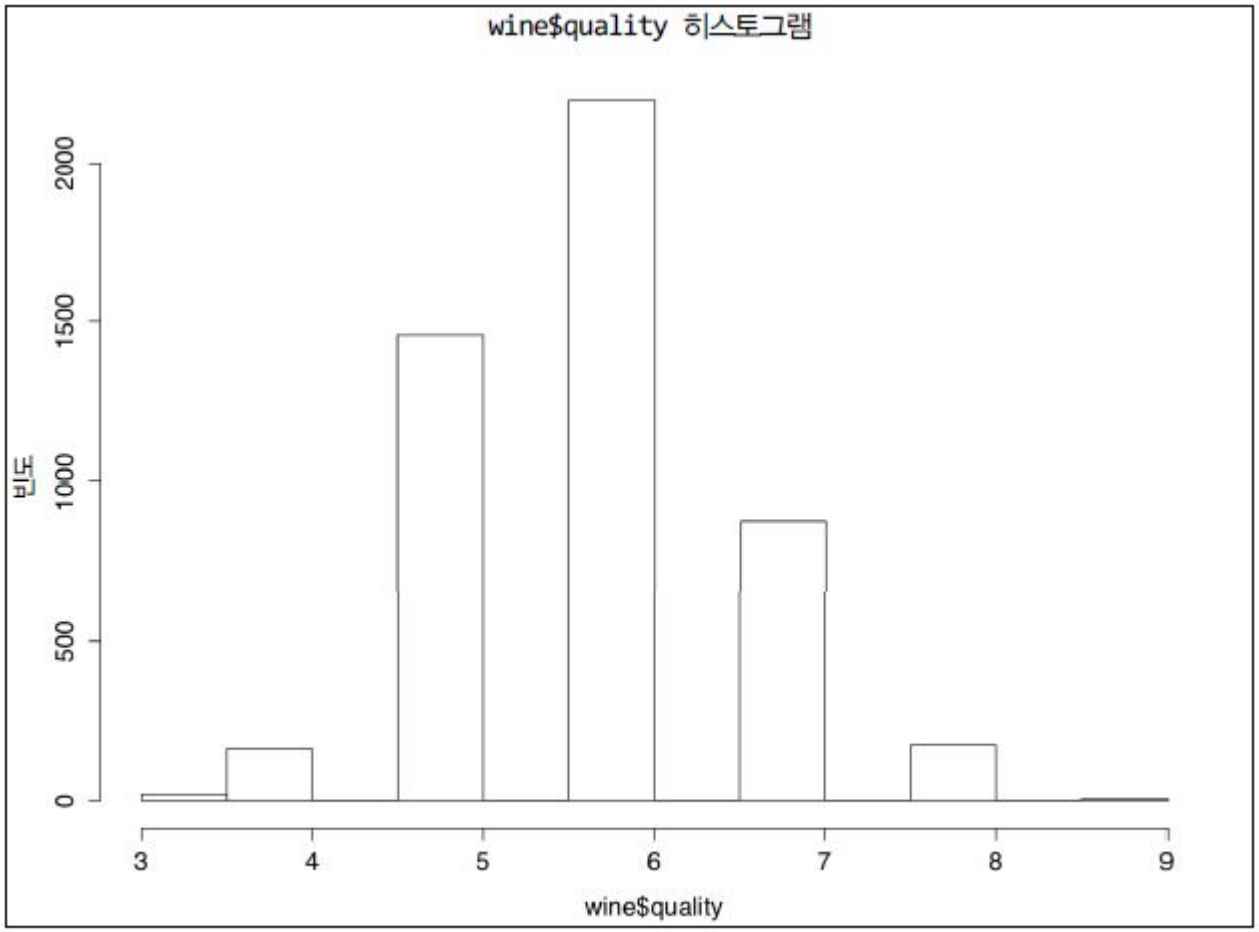


그림 6.11: 백포도주의 품질 등급 분포

Regression trees syntax

using the `rpart()` function in the `rpart` package

Building the model:

```
m <- rpart(dv ~ iv, data = mydata)
```

- `dv` is the dependent variable in the `mydata` data frame to be modeled
- `iv` is an R formula specifying the independent variables in the `mydata` data frame to use in the model
- `data` specifies the data frame in which the `dv` and `iv` variables can be found

The function will return a regression tree model object that can be used to make predictions.

Making predictions:

```
p <- predict(m, test, type = "vector")
```

- `m` is a model trained by the `rpart()` function
- `test` is a data frame containing test data with the same features as the training data used to build the model
- `type` specifies the type of prediction to return, either `"vector"` (for predicted numeric values), `"class"` for predicted classes, or `"prob"` (for predicted class probabilities)

The function will return a vector of predictions depending on the `type` parameter.

Example:

```
wine_model <- rpart(quality ~ alcohol + sulfates,
                    data = wine_train)
wine_predictions <- predict(wine_model, wine_test)
```

```
n= 3750

node), split, n, deviance, yval
      * denotes terminal node

 1) root 3750 2945.53200 5.870933
   2) alcohol< 10.85 2372 1418.86100 5.604975
     4) volatile.acidity>=0.2275 1611  821.30730 5.432030
       8) volatile.acidity>=0.3025 688  278.97670 5.255814 *
       9) volatile.acidity< 0.3025 923  505.04230 5.563380 *
     5) volatile.acidity< 0.2275 761  447.36400 5.971091 *
   3) alcohol>=10.85 1378 1070.08200 6.328737
     6) free.sulfur.dioxide< 10.5 84   95.55952 5.369048 *
     7) free.sulfur.dioxide>=10.5 1294  892.13600 6.391036
      14) alcohol< 11.76667 629  430.11130 6.173291
        28) volatile.acidity>=0.465 11   10.72727 4.545455 *
        29) volatile.acidity< 0.465 618  389.71680 6.202265 *
      15) alcohol>=11.76667 665  403.99400 6.596992 *
```

```
# 의사결정트리 시각화
install.packages("rpart.plot")
library(rpart.plot)
rpart.plot(m.rpart, digits = 3)
rpart.plot(m.rpart, digits = 4, fallen.leaves = TRUE, type = 3, extra = 101)

# 모델성능평가
p.rpart <- predict(m.rpart, wine_test)
p.rpart
summary(p.rpart)
summary(wine_test$quality)
cor(p.rpart, wine_test$quality)

# 평균절대오차로 성능 측정
MAE <- function(actual, predicted){
  mean(abs(actual - predicted))
}
MAE(p.rpart, wine_test$quality)

# 훈련 데이터의 평균 품질 평가
mean(wine_train$quality)
MAE(5.87, wine_test$quality)
```

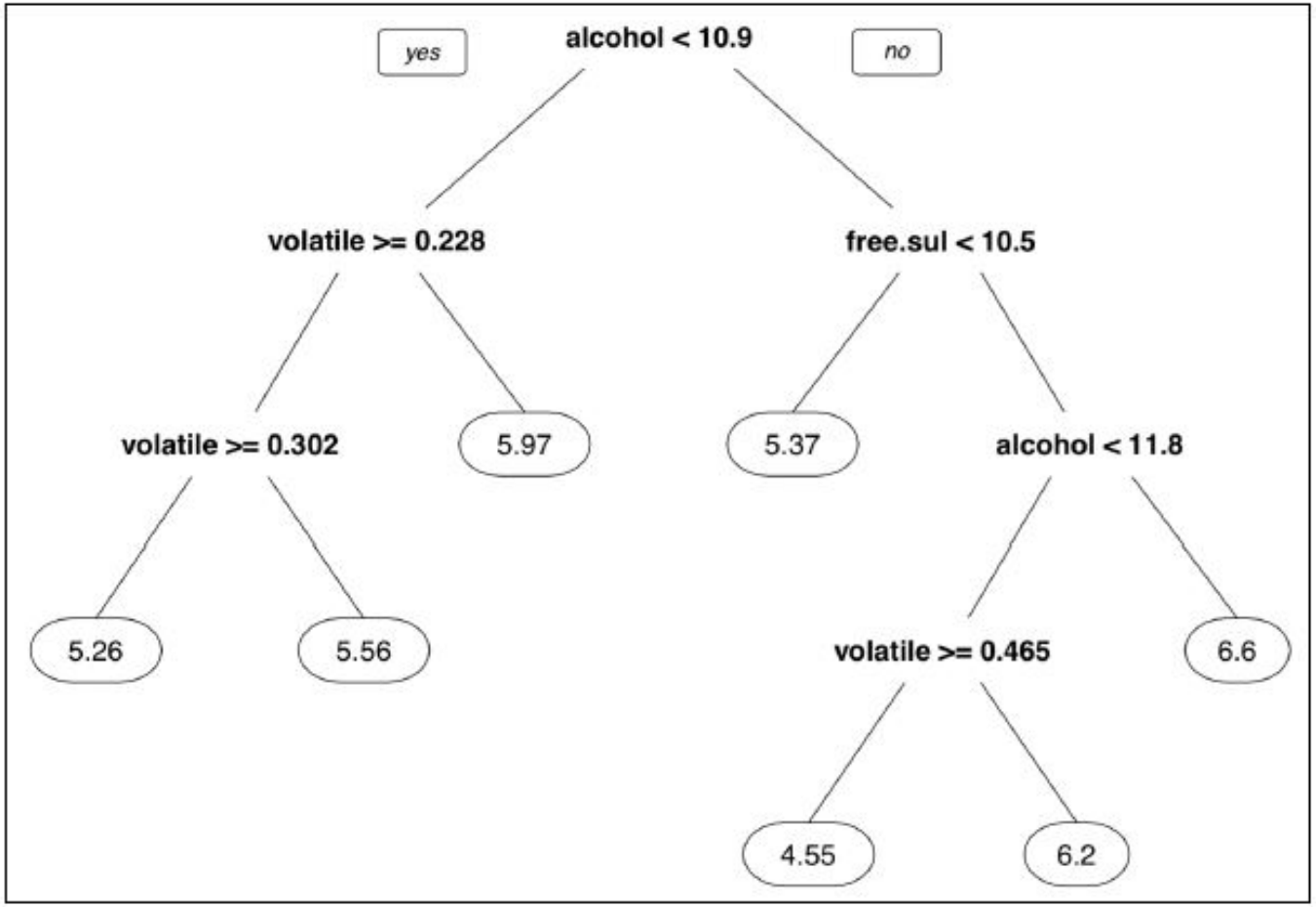


그림 6.12: 와인 품질 회귀 트리 모델의 시각화

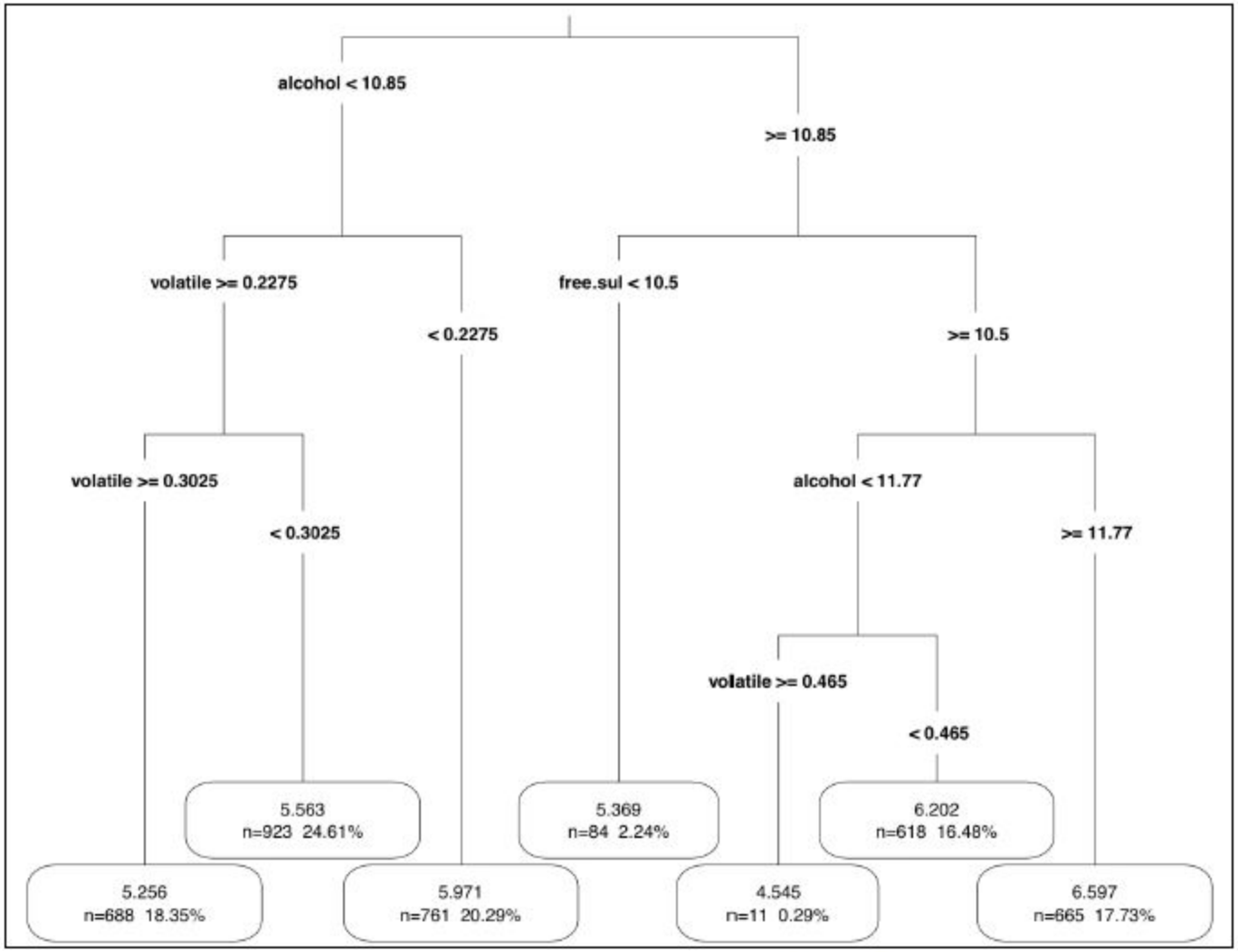


그림 6.13: 도식화 함수 파라미터를 변경하면 트리 시각화를 사용자 정의할 수 있다.

```
# 모델 성능 개선

install.packages("Cubist")

library(Cubist)

#cubist(독립변수, 종속변수)

m.cubist <- cubist(x=wine_train[-12], y=wine_train$quality)

m.cubist

summary(m.cubist)

p.cubist <- predict(m.cubist, wine_test)

summary(p.cubist)

cor(p.cubist, wine_test$quality)

MAE(wine_test$quality, p.cubist)
```