13

비정형데이터분석

텍스트 데이터의 통계적 분석(2)

통계·데이터과학과 장영재 교수

학습목차

1. 텍스트 데이터에 대한 군집분석
2. 텍스트 데이터에 대한 분류(classification) 분석

01

텍스트 데이터에 대한 군집분석

1. 텍스트 데이터에 대한 군집분석

1 군집분석에서의 거리 개념 및 비유사성 행렬

1) 군집분석에서의 거리 개념과 비유사성 행렬

- 군집분석은 군집 내의 개체들끼리는 동질적이고, 서로 다른 군집의 개체들끼리는 서로 이질적이 되도록 집단을 나누는 것이 목표

  - 두 개체 x와 y가 각각 벡터 $x = (x_1, x_2, \cdots, x_p)$와 $y = (y_1, y_2, \cdots, y_p)$ 로 표현되어 있다고 할 때 두 개체 사이의 유클리드 거리 $d(x,y)$

$$d(x,y) = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + \cdots + (x_p - y_p)^2} = \sqrt{\sum_{i=1}^{p} (x_i - y_i)^2}$$

1. 텍스트 데이터에 대한 군집분석

- 텍스트 데이터의 경우 코사인 거리도 군집분석에서 사용할 수 있는데 개체 x와 y의 코사인 거리는 1에서 코사인 유사도를 뺀 값으로 정의

$$d^c = 1 - \cos(x, y) = 1 - \frac{x_1 y_1 + x_2 y_2 + \cdots + x_p y_p}{\sqrt{x_1^2 + x_2^2 + \cdots + x_p^2}\sqrt{y_1^2 + y_2^2 + \cdots + y_p^2}}$$

→ 유클리드 거리는 크면 클수록 두 개체 사이의 거리가 먼 것으로 평가하고 코사인 거리는 1에 가까울수록 두 개체 사이의 거리가 먼 것으로 평가

1. 텍스트 데이터에 대한 군집분석

2 군집분석의 방법

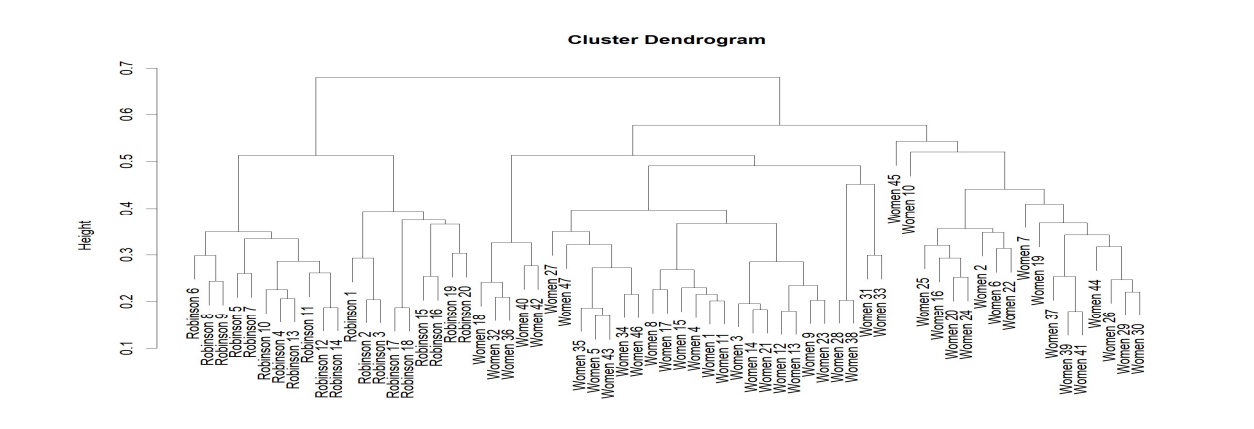
- 대표적인 군집분석 방법은 계층적 군집분석(hierarchical clustering)과 k-평균 군집분석(k-means clustering) 기법
  - 계층적 군집분석 방법에는 분할분석(divisive clustering) 방식과 응집분석(agglomerative clustering) 방식이 존재
  - 계층적 군집분석은 개체의 수가 많아질수록 시간이 지나치게 많이 걸린다는 단점이 있으므로 k-평균 군집분석방법도 이용

1. 텍스트 데이터에 대한 군집분석

3 코사인 비유사성 행렬 기준 군집분석

- 군집분석에서는 거리의 개념을 사용하므로 코사인 비유사성 행렬인 1-RCLW_CosSim 행렬을 as.dist() 함수에 입력

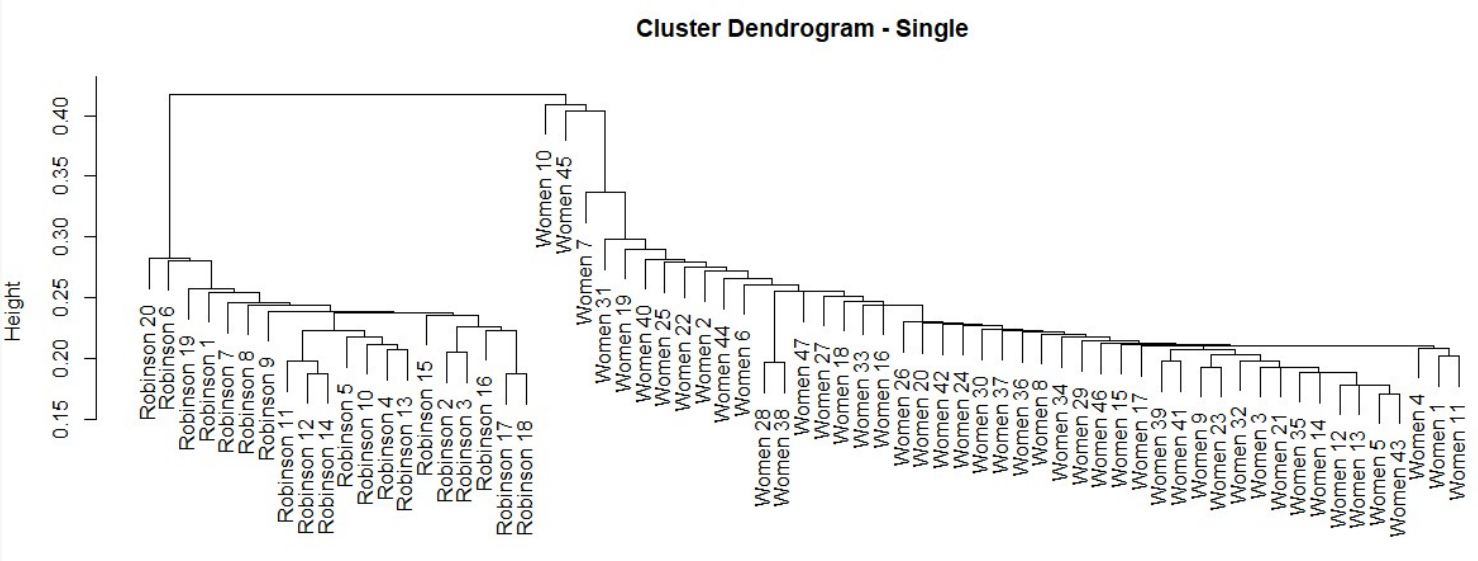
```
> RCLW_clusters <- hclust(as.dist(1-RCLW_CosSim))
> RCLW_clusters$labels <- c(paste("Robinson", c(1:20)), paste("Women", c(1:47)))
> plot(RCLW_clusters)
```



<그림> 코사인 비유사성 행렬을 이용한 응집분석(완전연결법) 덴드로그램

1. 텍스트 데이터에 대한 군집분석

```
> RCLW_single <- hclust(as.dist(1-RCLW_CosSim), method = "single")
> RCLW_single$labels <- c(paste("Robinson", c(1:20)), paste("Women", c(1:47)))
> plot(RCLW_single, main = "Cluster Dendrogram - Single", xlab="", sub="")
```



<그림> 코사인 비유사성 행렬을 이용한 응집분석(단일연결법) 덴드로그램

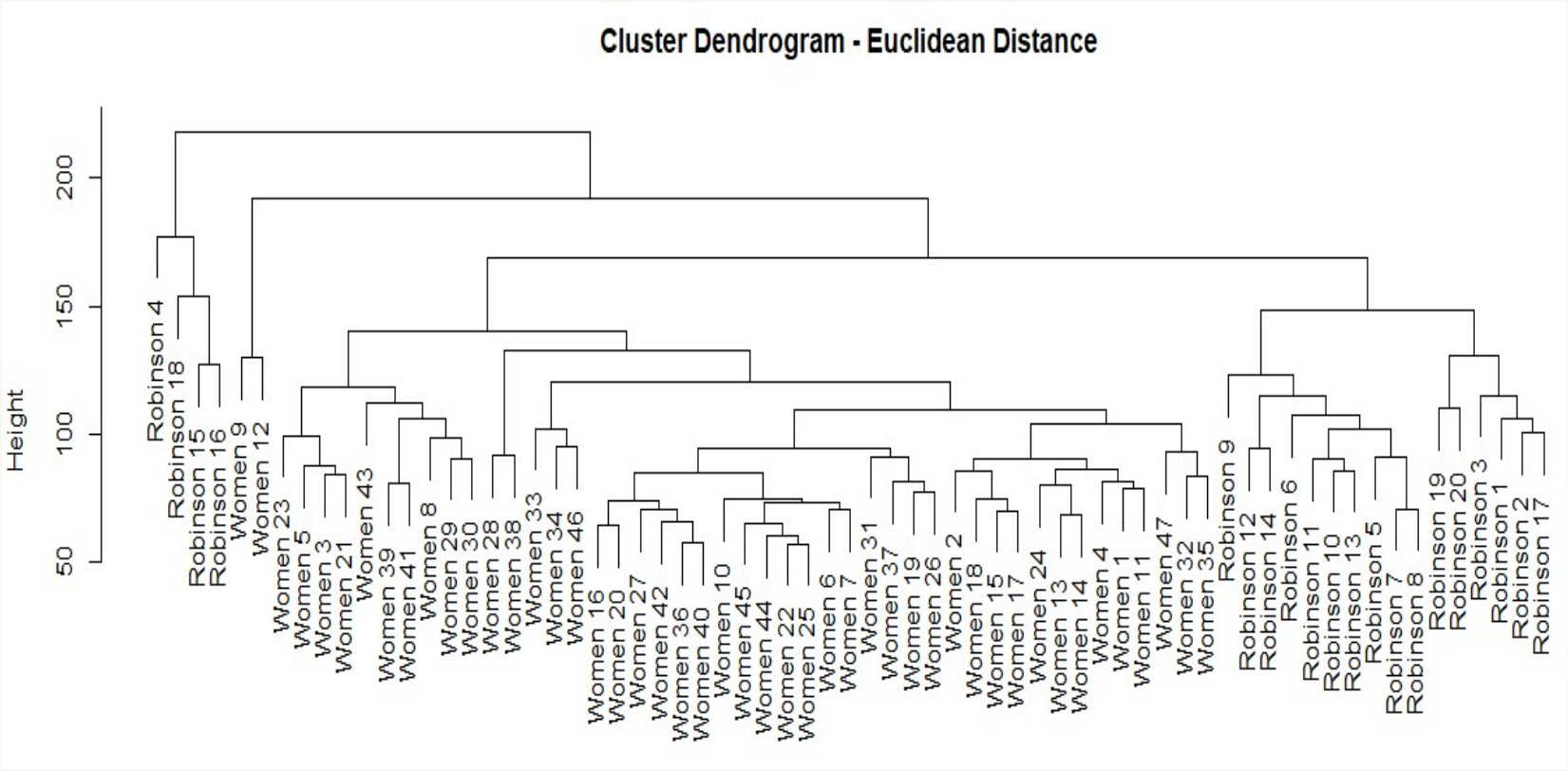
1. 텍스트 데이터에 대한 군집분석

4 유클리드 거리 기준 군집분석

- 코사인 비유사성 행렬 대신 유클리드 거리를 이용하여 군집분석 실행
  - hclust( ) 함수의 인수로 유클리드 거리 dist(RCLW_DTM) 입력

```
> RCLW_euclidean <- hclust(dist(RCLW_DTM))
> RCLW_euclidean$labels <- c(paste("Robinson", c(1:20)), paste("Women", c(1:47)))
> plot(RCLW_euclidean, main = "Cluster Dendrogram - Euclidean Distance", xlab="", sub="")
```

1. 텍스트 데이터에 대한 군집분석



<그림> 유클리드 거리를 이용한 응집분석(완전연결법) 덴드로그램

1. 텍스트 데이터에 대한 군집분석

● 유클리드 거리를 이용하여 군집분석 실시할 때 고려할 사항

A : This is a book.

B : This is an interesting book.

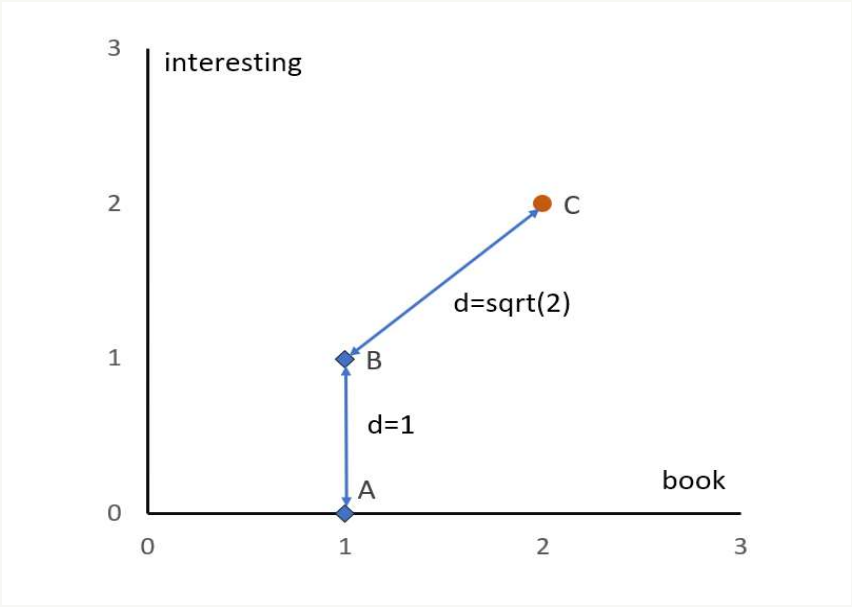
C : This book is interesting. It is an interesting book.

	절대도수		상대도수	
	book	interesting	book	interesting
A	1	0	1	0
B	1	1	0.5	0.5
C	2	2	0.5	0.5

<표> 세 단어주머니의 절대도수와 상대도수

1. 텍스트 데이터에 대한 군집분석

- 유클리드 거리 기준으로 보면 문장 A와 문장 B 사이의 거리는 1, 문장 B와 문장 C 사이의 거리는 $\sqrt{2}$ (포함된 단어 수가 많은 문서는 원점에서 멀리 표현)
- 코사인 유사도 기준으로 볼 때는 B와 C는 각도가 0이므로 코사인 유사도가 1이며 A와 B 또는 A와 C는 각도가 45도이므로 코사인 유사도가 $1/\sqrt{2}$

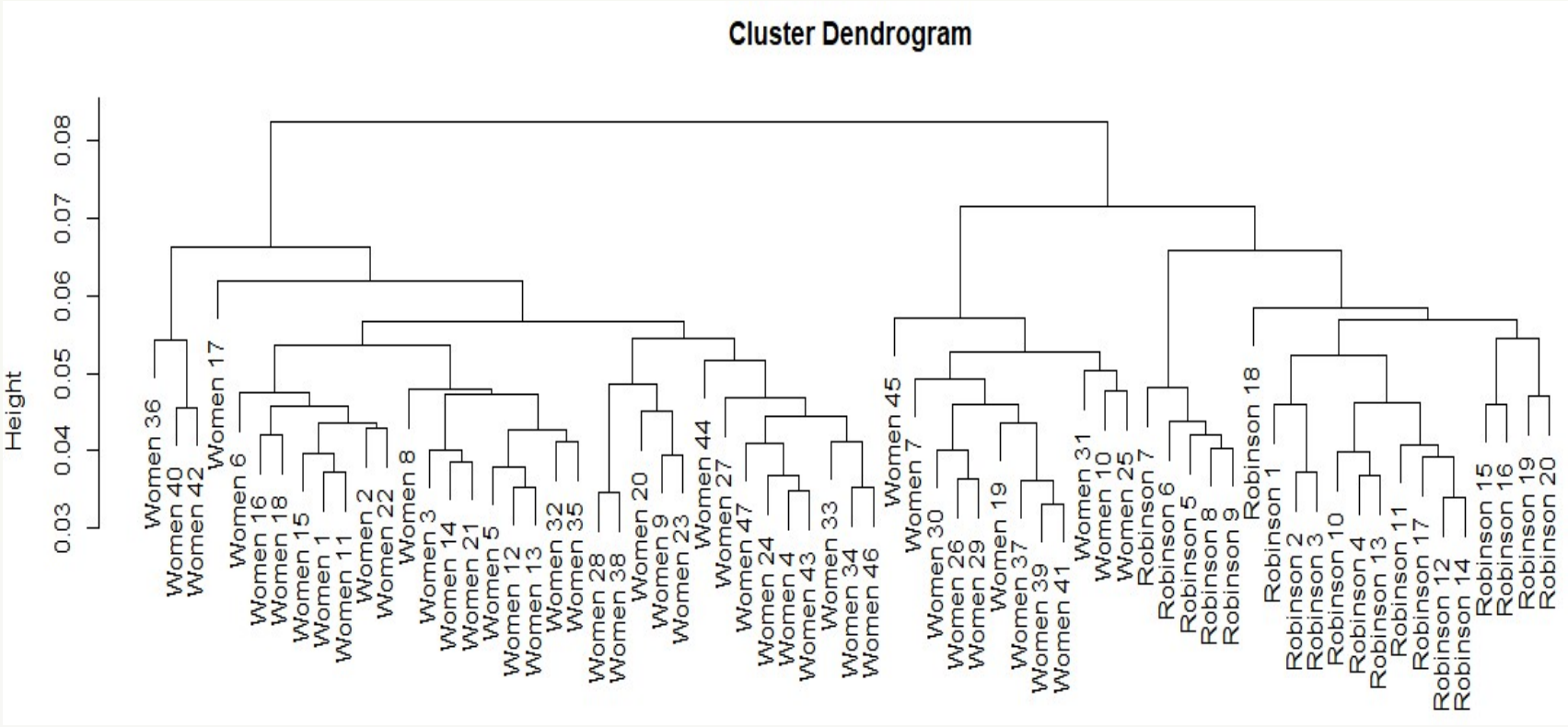


<그림> 세 문장 사이의 유클리드 거리

1. 텍스트 데이터에 대한 군집분석

- 유클리드 거리를 사용하여 군집분석을 할 때에는 많은 경우 표준화과정을 거치기도 하지만, 텍스트 데이터의 경우 문서-단어행렬의 각 원소들이 이산적인 값을 가지므로 표준화과정보다 상대도수 기준 유클리드 거리 이용
  - 상대도수 기준으로는 코사인 비유사성에서와 마찬가지로 B와 C의 거리가 0으로 A와 B의 거리에 비해 가까운 것으로 평가

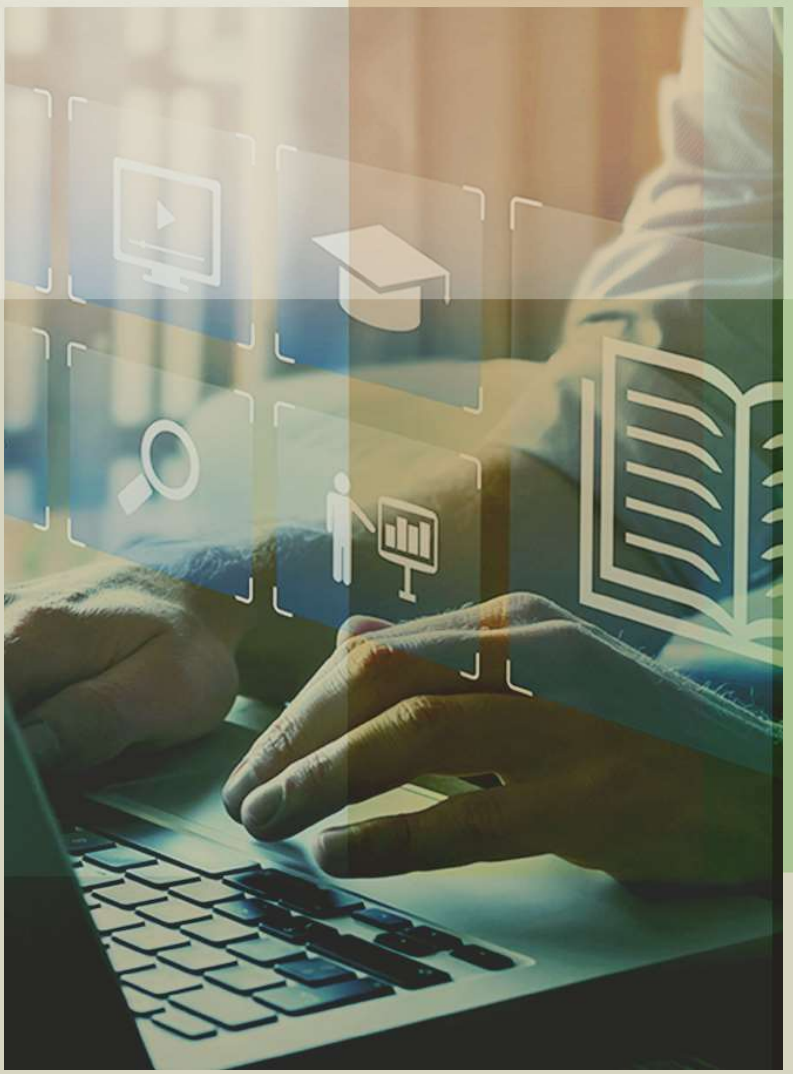
```
> RCLW_relfreq <- hclust(dist(RCLW_DTM/rowSums(RCLW_DTM)))
> RCLW_relfreq$labels <- c(paste("Robinson", c(1:20)), paste("Women", c(1:47)))
> plot(RCLW_relfreq, xlab="", sub="")
```



<그림> 상대도수의 유클리드 거리를 이용한 응집분석(완전연결법) 덴드로그램

02

텍스트 데이터에 대한 분류(classification) 분석



2. 텍스트 데이터에 대한 분류(classification) 분석

1 분류나무 모형

- 분류나무 모형은 전체를 하나의 집단으로 보고 이 집단을 가장 동질적인 두 집단으로 분할할 수 있는 방법을 찾아 분할 과정을 반복해서 결과를 도출
  - 가능한 집단 내의 불순도(impurity)를 낮출 수 있는 변수를 찾아 집단을 분할

		예측 결과		합계
		양(positive)	음(negative)	
실제값	양(positive)	TP (true positive) n_{11}	FN (false negative) n_{10}	$n_{1.}$
	음(negative)	FP (false positive) n_{01}	TN (true negative) n_{00}	$n_{0.}$
	합계	$n_{.1}$	$n_{.0}$	

2. 텍스트 데이터에 대한 분류(classification) 분석

2 분류모형에 대한 예측오차 평가

- 이항분류모형의 분류결과는 참 또는 거짓으로 주어지므로 정오표(confusion matrix)를 작성하여 분류 성능을 평가
  - 성능을 평가하는 지표로 정확도(precision) 정의

$$precision = \frac{TP + TN}{TP + TN + FP + FN} = \frac{n_{11} + n_{00}}{n}$$

  - 불균형 데이터일 경우 정확도는 큰 의미가 없으므로 민감도(sensitivity), 특이도(specificity)도 함께 고려

$$sensitivity = \frac{TP}{TP + FN} = \frac{n_{11}}{n_{1.}}, \text{specificity} = \frac{TN}{TN + FP} = \frac{n_{00}}{n_{0.}}$$

2. 텍스트 데이터에 대한 분류(classification) 분석

- 민감도와 특이도는 분류모형의 임계치를 어떻게 정하는가에 따라 달라지게 되므로 ROC(receiver operating characteristic) 곡선을 이용하여 분류모형의 성능을 평가하기도 함

2. 텍스트 데이터에 대한 분류(classification) 분석

3 교차검증(cross-validation, CV)

- 모형 구축을 위해 사용된 데이터(in-sample)에 대한 설명력은 높지만 다른 데이터(out-of-sample)에 적용했을 때에는 모형의 성능이 좋지 않게 나타나는 경우가 많으므로 교차검증을 실시

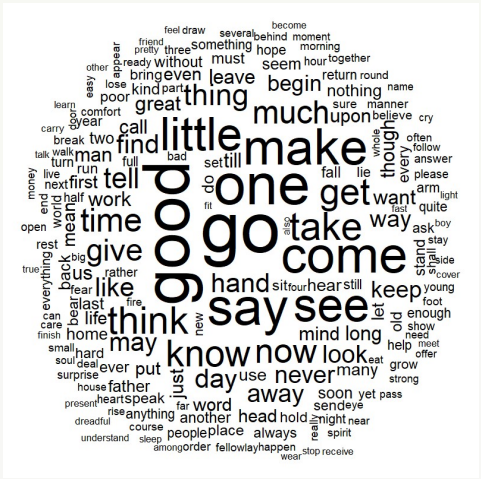
  - 전체 데이터를 k개의 집단으로 랜덤으로 나누고 이 집단들 중에서 k-1개의 집단을 사용하여 모형을 구축한 후 나머지 하나의 집단을 사용하여 모형의 예측오차를 평가(k-fold cross-validation)

  → k번 반복하여 표본 외 오차(out-of-sample error)를 가능한 작게 만드는 모형을 선택

2. 텍스트 데이터에 대한 분류(classification) 분석

4 R을 이용한 텍스트 데이터의 분류분석

- 텍스트 데이터의 분류를 위해 목표변수의 불순도를 가능한 낮출 수 있는 변수들을 분류모형의 설명변수로 사용
  - 두 소설의 공통단어(좌)와 공통되지 않는 단어(우) 워드클라우드



2. 텍스트 데이터에 대한 분류(classification) 분석

- RCLW_DTM 행렬로 두 소설의 각 장에서 "jo"가 사용되었는지를 확인
  - RCLW_lev=="jo"로 "jo"에 해당되는 열의 위치를 찾고 「로빈슨 크루소」와 「작은 아씨들」에 해당 부분을 1:20, 21:67로 선택

```
> RCLW_DTM[1:20, RCLW_lev=="jo"]
 [1] 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
> RCLW_DTM[21:67, RCLW_lev=="jo"]
 [1] 40 20 53 26 50 11  9 61 20  8 38 57 24 39 25 14 29 31  4 17 60 15 31 24  4
[26] 13 27 10 46 39  7 57  2 51 37 31  2  2 13 10 10 30 56  8  5 48 31
```

2. 텍스트 데이터에 대한 분류(classification) 분석

- 일반적인 분류모형을 작성하는 절차는 다음과 같이 정리할 수 있음

① 두 소설에 공통적으로 사용된 단어 중에서 25개의 단어만 추출하여 분류모형을 작성

```
> RCLW_DTMs <- rbind(colSums(RCLW_DTM[1:20,]), colSums(RCLW_DTM[21:67,]))
> sum(RCLW_DTMs[1,]>0 & RCLW_DTMs[2,]>0)
[1] 2707
> sample_words <- sample(which(RCLW_DTMs[1,]*RCLW_DTMs[2,]>0), 25)
> RCLW_DTMs[1:2,sort(sample_words)]   # 랜덤 25단어 추출
```

2. 텍스트 데이터에 대한 분류(classification) 분석

② 문서-단어행렬에서 랜덤추출된 단어들의 열만 선택하여 부분행렬을 작성하고 분류나무모형의 목표변수에 해당되는 벡터를 생성

```
> RCLW_DTM_smpl <- RCLW_DTM[,sample_words]
> colnames(RCLW_DTM_smpl) <- RCLW_lev[sample_words]
> RCLW_target <- c(rep("Robin",20),rep("Women",47))
```

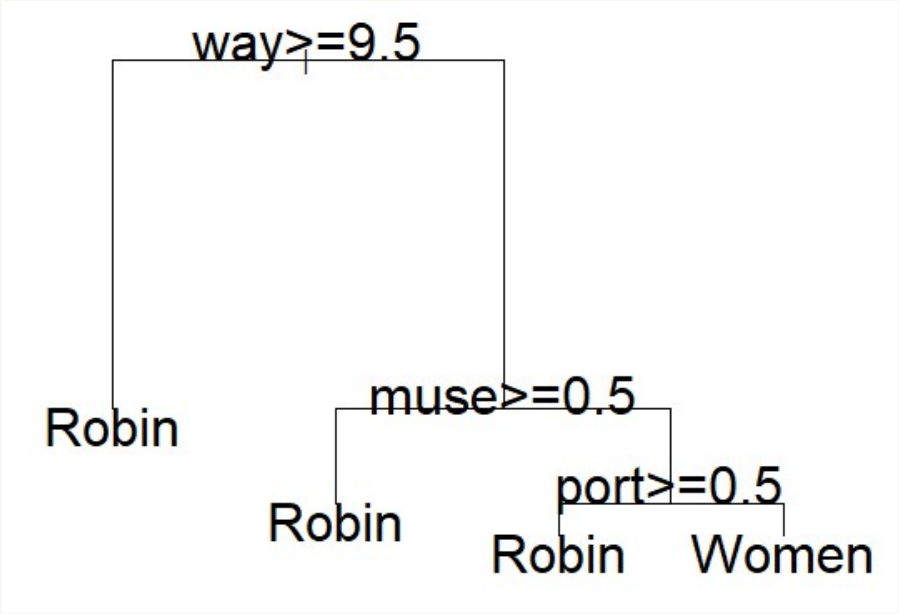
③ 제약조건을 지정한 후 분류모형 적합

```
> library(rpart)
> ctrl <- rpart.control(minsplit = 5, cp = -0.01, xval = 10)
> fit_tree <- rpart(RCLW_target ~ ., data = data.frame(RCLW_DTM_smpl),
method = "class", control = ctrl)
```

2. 텍스트 데이터에 대한 분류(classification) 분석

④ 가지치기 단계를 수행

```
> prune_tree <- prune(fit_tree, cp=0)
> plot(prune_tree, margin = 0.1)
> text(prune_tree, cex = 2)
```



<그림> 분류모형의 가지치기 결과

실습하기

다음시간안내

14

텍스트 데이터 분석 사례(1)