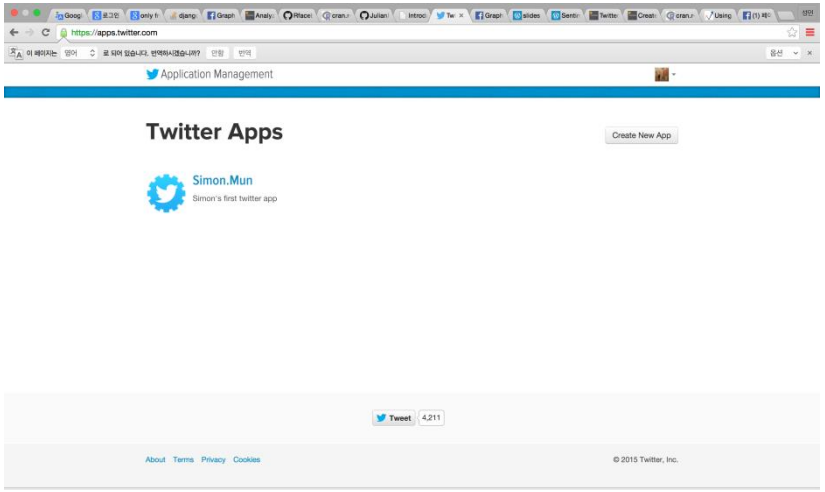
인문학 텍스트 마이닝

Text Analysis utilizing twitter

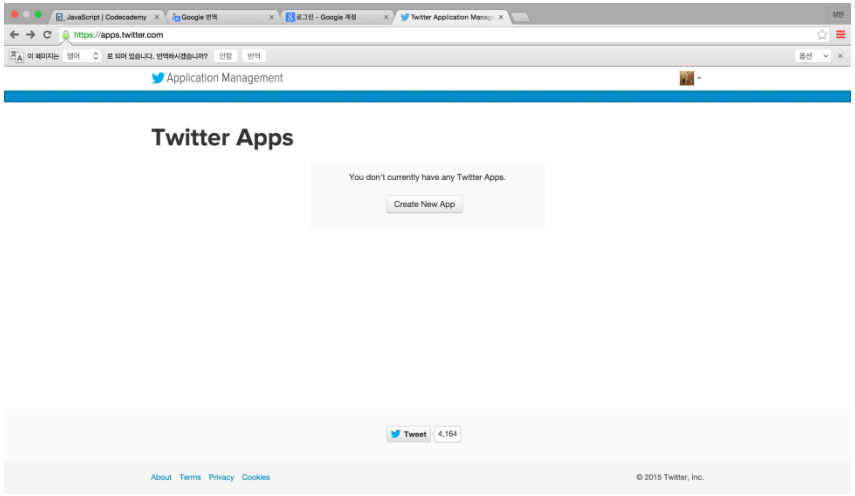
Text Analysis utilizing twitter

- 트위터 계정 생성
  - https://apps.twitter.com

1) Process



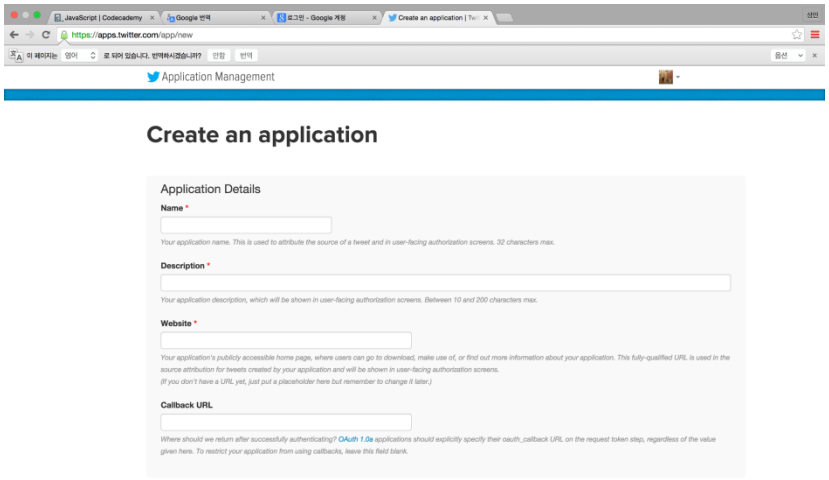
2) Process



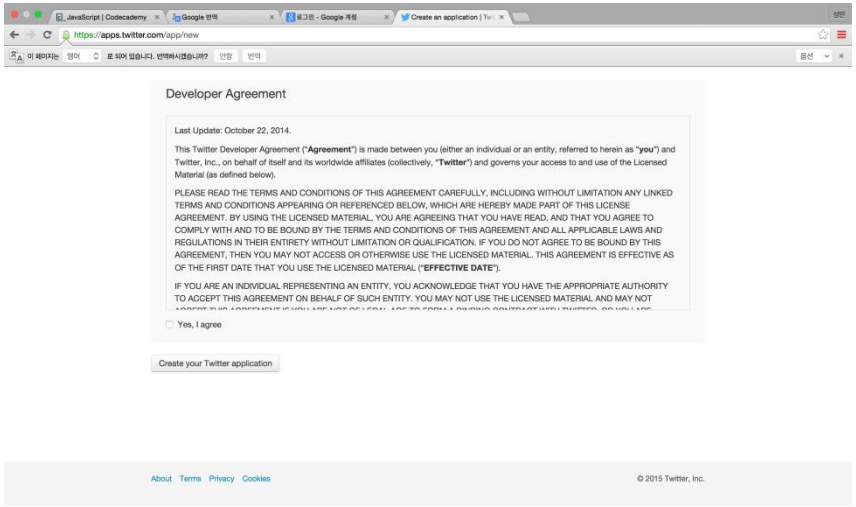
Text Analysis utilizing twitter

- 트위터 계정 생성
  - https://apps.twitter.com

3) Process



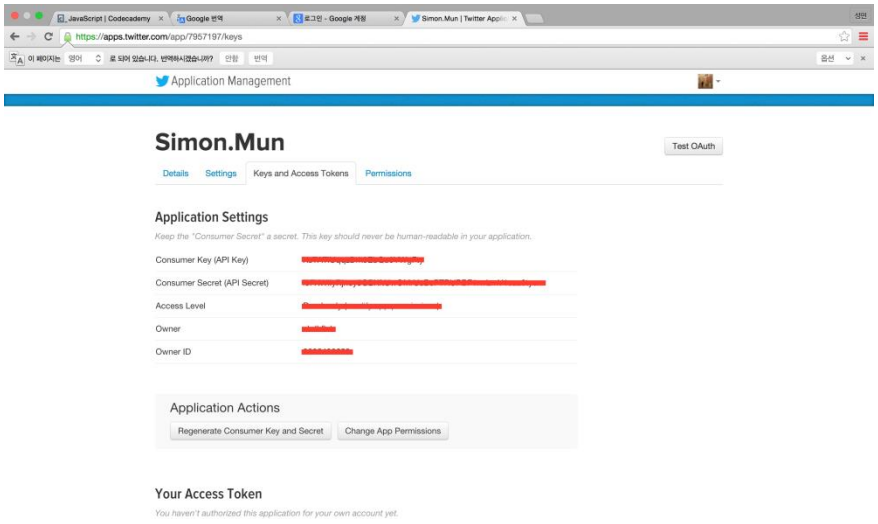
4) Process



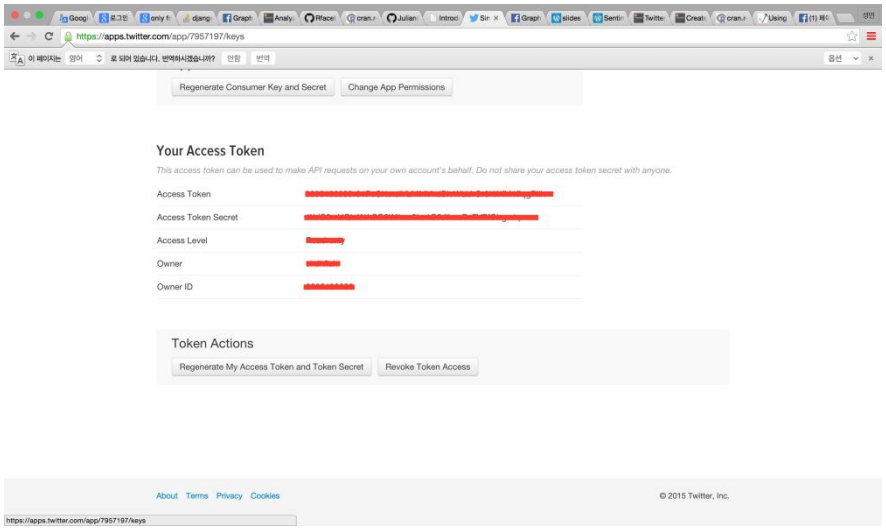
Text Analysis utilizing twitter

- 트위터 계정 생성
  - https://apps.twitter.com

5) Process



6) Process



Text Analysis utilizing twitter

- 라이브러리 설치

```
> library(bitops)
> library(RCurl)
> library(RJSONIO)
> library(twitteR)
> library(ROAuth)
> library(RColorBrewer)
> library(devtools)
> install_github("twitteR", username="geoffjentry")
Downloading github repo geoffjentry/twitteR@master
Installing twitteR
'/Library/Frameworks/R.framework/Resources/bin/R' --vanilla CMD INSTALL  \

'/private/var/folders/28/g8cf_pvx46s5phqgwr6qq7jw0000gn/T/Rtmp8qGMiY/devtoolscb924cc3a7ae/geoffj
entry-twitteR-563a23c'  \
  --library='/Library/Frameworks/R.framework/Versions/3.1/Resources/library'  \
  --install-tests

* installing *source* package ‘twitteR’ ...
** R
** inst
** preparing package for lazy loading
Creating a generic function for ‘as.data.frame’ from package ‘base’ in package ‘twitteR’
** help
*** installing help indices
** building package indices
** testing if installed package can be loaded
* DONE (twitteR)
Reloading installed twitteR

Attaching package: ‘twitteR’

The following object is masked from ‘package:plyr’:

    id

The following objects are masked from ‘package:dplyr’:

    id, location

경고메시지:
Username parameter is deprecated. Please use geoffjentry/twitteR
```

GitHub에서 twitteR패키지의 최신버젼을 다운로드한다.

Text Analysis utilizing twitter

- 유저 정보 입력

```
> api_key <- "[redacted]"
>
> api_secret <- "[redacted]"
>
> access_token <- "[redacted]"
>
> access_token_secret <- "[redacted]"
>
> setup_twitter_oauth(api_key,api_secret,access_token,access_token_secret)
[1] "Using direct authentication"
```

- https://apps.twitter.com에서 로그인 후 제공받은 api_key, api_secret, access_token, access_token_secret을 입력한다.

Text Analysis utilizing twitter

- Greece에 관련된 텍스트 1000개 크롤링

```
> Greece.tweets = searchTwitter("Greece" , n = 1000)
```

- Greece에 관련된 텍스트만 추출

```
> library(plyr)
>
> Greece.text = laply(Greece.tweets,function(t)t$getText())
```

- 모든 문자 소문자로 변환

```
> Greece.text <- tolower(Greece.text)
>
```

- Rt를 빈공간으로 바꾸기(삭제)

```
> Greece.text <- gsub("rt", "", Greece.text)
>
```

- 유저이름 삭제(@||w+)

```
> Greece.text <- gsub("@\\w+", "", Greece.text)
>
```

Text Analysis utilizing twitter

- 문장 부호 제거

```
> Greece.text <- gsub("[[:punct:]]", "", Greece.text)
>
```

- 링크 제거

```
> Greece.text <- gsub("http\\w+", "", Greece.text)
>
```

- 탭 제거

```
> Greece.text <- gsub("[ |\t]{2,}", "", Greece.text)
>
```

- 시작 부분의 문자 제거

```
> Greece.text <- gsub("^ ", "", Greece.text)
>
```

- 끝 부분의 문자 제거

```
> Greece.text <- gsub(" $", "", Greece.text)
```

Text Analysis utilizing twitter

- Corpus생성

```
>
> Greece.text.corpus <- Corpus(VectorSource(Greece.text))
```

- Tm_map을 활용하여 Stop words 삭제

```
> Greece.text.corpus <- tm_map(Greece.text.corpus, function(x)removeWords(x,stopwords()))
```

- TermDocumentMatrix를 활용하여 수치형 데이터로 형변환

```
> myTdm <- TermDocumentMatrix(Greece.text.corpus, control = list(wordLengths = c(2, Inf)))
>
```

Text Analysis utilizing twitter

- 10번 이상 출현한 명사 나타내기

```
> findFreqTerms(myTdm, lowfreq = 10)
 [1] "1greek"       "2015"         "230815"       "24"           "akan"
 [6] "allowed"      "anda"         "atau"         "athens"       "bailout"
[11] "bid"          "border"       "chicago"      "conservative" "copyright"
[16] "coreoo"       "crisis"       "cross"        "dan"          "direction"
[21] "ditangan"     "form"         "gives"        "government"   "greece"
[26] "greek"        "gt"           "holidays"     "just"         "kekalkan"
[31] "last"         "lesvos"       "like"         "macedonia"    "macedonias"
[36] "malaysia"     "migrants"     "mtvhottest"   "najib"        "nasib"
[41] "new"          "night"        "now"          "npr"          "nyc"
[46] "one"          "opposition"   "otrachicago"  "pay"          "refugees"
[51] "refugeesgr"   "ringgit"      "sama"         "seaside"      "see"
[56] "segalanya"    "selamatkan"   "sepei"        "stage"        "syrian"
[61] "tsipras"      "undur"        "update"       "visit"
```

- Bailout과 관련된 명사 찾기

```
> findAssocs(myTdm, "bailout", 0.25)
$bailout
        poll        dutch      weakens     allnight        backs         deal
        0.73         0.61         0.61         0.59         0.59         0.59
      debate        shows        suppo       wilnews   truegreece        costs
        0.59         0.56         0.56         0.55         0.48         0.39
       seats        three          vvd 240815120213    coalition         grip
        0.39         0.39         0.37         0.27         0.27         0.27
    maintain    oligarchs         paid        pnews        yanis
        0.27         0.27         0.27         0.27         0.27
```

- ggplot2를 이용하여 막대 그래프 그리기

- 라이브러리 불러오기, 단어에 따른 빈도수 합하기, 10이상 단어 추출

```
> library(ggplot2)
>
> termFrequency <- rowSums(as.matrix(myTdm))
>
> termFrequency <- subset(termFrequency, termFrequency >= 10)
```

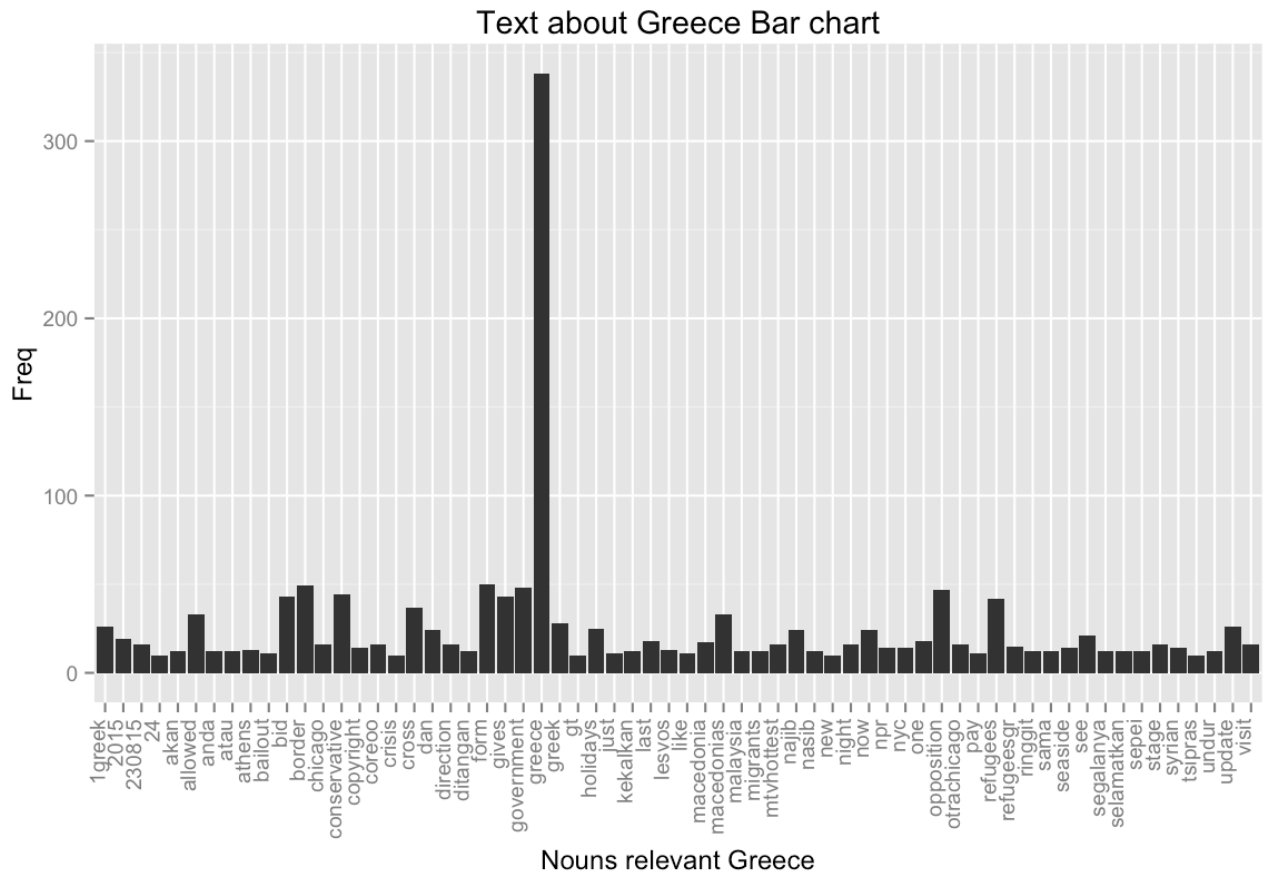
- 데이터 프레임 형태로 형변환, X와 Y생성

```
> termFrequency <- as.data.frame(termFrequency)
>
> X = row.names(termFrequency)
>
> Y = termFrequency$termFrequency
```

Text Analysis utilizing twitter

- 그래프 생성

```
> qplot(x=X,y=Y, geom="bar", stat="identity", xlab="Nouns relevant Greece", ylab="Freq", main
="Text about Greece Bar chart")+ theme(axis.text.x = element_text(angle=90, hjust=1, vjust=0)
)
```



Text Analysis utilizing twitter

- 단어 빈도수에 따른 워드 클라우드 생성

- 라이브러리

```
> library(wordcloud)
>
```

- 행렬로 형변환

```
> m <- as.matrix(myTdm)
>
```

- 빈도 값 합산

```
> wordFreq <- sort(rowSums(m), decreasing = TRUE)
>
```

- 팔레트 생성

```
> pal <- brewer.pal(8, "Dark2")
>
```

Text Analysis utilizing twitter

- 워드 클라우드 생성

```
> wordcloud(words = names(wordFreq), freq = wordFreq, min.freq = 10, random.order = F, rot.pe
r = 0.1, colors = pal)
```

Text Analysis utilizing twitter

- Clustering Analysis

What is Cluster Analysis?

- 개념(Concept)

  - N개의 관찰치를 대상으로 p개의 변수를 측정했을 때, 관측한 p개의 변수 값을 이용하여 N개의 관찰치 사이의 유사성(similarity)의 정도를 측정하여 관찰치들을 가까운 순서대로 군집화하는 통계적 분석 방법이다.
  - 두 관찰치 사이의 유사성을 측정하는 여러 방법(유클리디안, 유클리디안 제곱거리, 코사인값, 상관계수, 체비셰프, 블록, 민코브스키, 커스텀거리,,,) 중 가장 일반적으로 이용되는 방법은 유클리디안 거리 또는 상관계수 공식을 이용한 변수가 유사성 측정이다.
  - 데이터의 특징을 나타내는 변수간에 값의 차이 혹은 단위의 차이가 클 때는 데이터를 표준화시켜 사용하는 것이 일반적이다.

- 유사성 관련 수식(Formula)

$$\sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2 + \ldots + (p_n - q_n)^2} = \sqrt{\sum_{i=1}^{n} (p_i - q_i)^2}$$

유클리디안 거리 측정 공식

$$r_{xy} = \frac{\sum_{i=1}^{n} (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n} (x_i - \bar{x})^2 \sum_{i=1}^{n} (y_i - \bar{y})^2}}$$

상관계수 거리 측정 공식

What is Cluster Analysis?

- 표준화 관련 수식(Formula)

$$Z_i = \frac{W_i - \bar{W}}{S_W} \qquad \bar{W} = \frac{\sum_{i=1}^{n} W_i}{n} \qquad S_W = \sqrt{\frac{\sum (W_i - \bar{W})^2}{n}}$$

- 군집화 방법의 개념(Concept)

  - 계층적 군집 분석 : 각 관찰치들 사이의 유사성, 거리 행렬을 구한 뒤에 관찰치들을 가까운 순서대로 연결해 가는 방법(최단연결법, 최장연결법, 중심연결법, 평균연결법, 워드의 방법 등이 있다.)

  - 비계층적 군집 분석 : 비 계층 적 군집 방법은 K-means clustering으로 불리며 관찰치들이 속할 군집의 수(K)를 미리 정한 뒤 정해진 군집으로 관찰치들을 포함시키는 방법이다.

- 단어 빈도수에 따른 군집 분석

- removeSparseTerms를 활용하여 Corpus상의 0값 제거

```
> myTdm2 <- removeSparseTerms(myTdm, sparse = 0.95)
>
```

- 행렬로 형변환

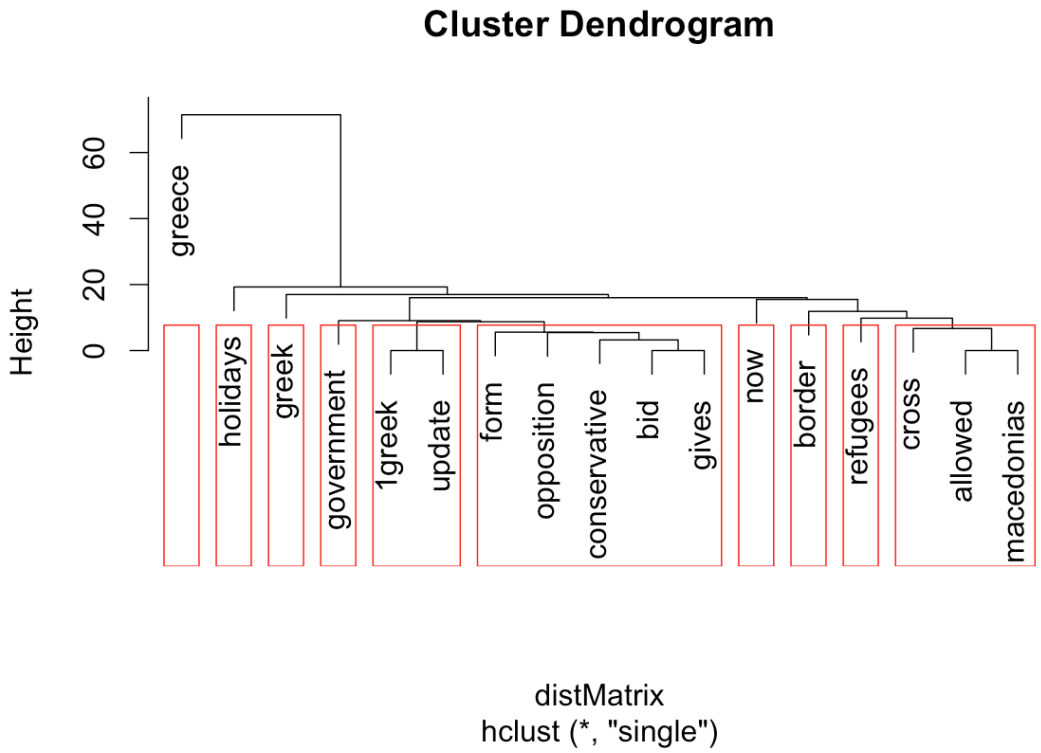
```
> m2 <- as.matrix(myTdm2)
>
```

- 유클리디안 매트릭스 생성

```
> distMatrix <- dist(scale(m2),method="euclidean")
```

- 유클리디안, 최단거리 연결법을 활용한 군집 수형도(덴드로그램)

```
> single <- hclust(distMatrix, method = "single")
>
> plot(single)
>
> rect.hclust(single, k = 10)
```



- 유클리디안, 최장거리 연결법을 활용한 군집 수형도(덴드로그램)

```
> complete <- hclust(distMatrix, method = "complete")
>
> plot(complete)
>
> rect.hclust(complete, k = 10)
```