ADP 제8회 3번 문제 풀이(text_miming)

Kim Jeong Gyu

2017-09-11

1 ADP 3번 문제

1.1 tvprograms_name.txt 파일을 읽어 그 속에 있는 단어들을 사전에 추가

needed libraries loading

```
library(rJava)
library(stringr)
library(KoNLP)
library(tm)
library(dplyr)
library(ggplot2)
library(scales)
```

- making dictionary : buildDictionay()

```
tvpro_name <- readLines('tvprograms_name.txt')
str(tvpro_name)
```

```
##  chr "한끼줍쇼, 1박 2일, 아는 형님, 정글의 법칙, 복면가왕, 발칙한 동거, 나혼자산다, 무한도전, 삼시세끼 "
```

```
tvpro_name
```

```
## [1] "한끼줍쇼, 1박 2일, 아는 형님, 정글의 법칙, 복면가왕, 발칙한 동거, 나혼자산다, 무한도전, 삼시세끼 "
```

```
tvpro_name_1 <- str_split(tvpro_name, ",")
str(tvpro_name_1)
```

```
## List of 1
##  $ : chr [1:9] "한끼줍쇼" " 1박 2일" " 아는 형님" " 정글의 법칙" ...
```

```
tvpro_name_1 <- as.vector(tvpro_name_1)
tvpro_name_2 <- unlist(tvpro_name_1)
class(tvpro_name_2)
```

```
## [1] "character"
```

```
tvpro_name_2
```

```
## [1] "한끼줍쇼"     " 1박 2일"     " 아는 형님"   " 정글의 법칙"
## [5] " 복면가왕"    " 발칙한 동거" " 나혼자산다"  " 무한도전"
## [9] " 삼시세끼 "
```

```
pro_name <- gsub(" ", '', tvpro_name_2)
pro_name
```

```
## [1] "한끼줍쇼"   "1박2일"     "아는형님"   "정글의법칙" "복면가왕"
## [6] "발칙한동거" "나혼자산다" "무한도전"   "삼시세끼"
```

- get user dictionary as data.frame

```
user_d <- data.frame(term = pro_name, tag = 'ncn')
user_d
```

```
##         term tag
## 1   한끼줍쇼 ncn
## 2     1박2일 ncn
## 3   아는형님 ncn
## 4 정글의법칙 ncn
## 5   복면가왕 ncn
## 6 발칙한동거 ncn
## 7 나혼자산다 ncn
## 8   무한도전 ncn
## 9   삼시세끼 ncn
```

```
dics <- c('sejong')
category <- 'TV 프로그램'
buildDictionary(ext_dic = dics, category_dic_nms = category,
                user_dic = user_d, replace_usr_dic = F)
```

```
## 370986 words dictionary was built.
```

1.2 tvprograms.txt 파일을 읽고 데이터 전처리

1.2.1 tvprogram_name.txt에 있는 프로그램명과 관련이 있는 데이터만 추출

- loading data

```
tvpro <- read.table('tvprograms.txt', header = T, stringsAsFactors = F)
class(tvpro) # data.frame
```

```
## [1] "data.frame"
```

```
names(tvpro)
```

```
## [1] "date"    "title"    "contents"
```

```
head(tvpro, 2)
```

```
##        date
## 1 2017-01-31
## 2 2017-01-31
##                                                           title
## 1                  성소X김종민, `한끼줍쇼` 밥동무 출연.. "대세들의 만남!"
## 2 [리뷰] 안중근 의사의 후예, 안재욱의 뮤지컬 ‘영웅’ 출연은 필연이었을까
##
                                                                                                    contents
## 1 ..끼줍쇼’에는 연예대상을 수상한 김종민과 대세 걸그룹으로 우뚝 선 우주소녀의 성소가 밥동무로 출연을 앞두고 ..‘한끼줍쇼’의 녹화에서는 강력한 예능의 기운을 몰고 온 김종민과 성소의 연희동 한 끼 도전기로 네 사람은 첫 만남부터 예사롭지 않았다고 전해졌다.
## 2                  ▲ 뮤지컬 ‘영웅(연출 윤호진)’ 공.. 최근 인기 예능 프로그램 MBC ‘무한도전’ 팀과 역사 특집 ..형, 이정열, 리사, 박정아, 정재은, 허민진(크레용팝 초아), 이지민 외.<U+00A0> 관람료: VIP석 13만원, R석 11만원, S석 8만원, A석 6만원<U+00A0>
```

- toy data munging : 먼저 title 컬럼에서 구둣점들을 제거하고 여섯 번째 원소까지만 추출한 데이터를 toy 데이터로 만들고 이에 대한 분석 진행

```
rm_punc_title <- gsub('[[:punct:]]+', ' ', tvpro[, 2])
test_2 <- head(rm_punc_title)
test_2
```

```
## [1] "성소X김종민  한끼줍쇼  밥동무 출연  대세들의 만남 "
## [2] " 리뷰  안중근 의사의 후예  안재욱의 뮤지컬  영웅  출연은 필연이었을까"
## [3] "정준영  솔로 앨범  1인칭  발표 복귀 청신호 "
## [4] "정준영 가수로  컴백 첫 정규앨범 발표"
## [5] "주먹쥐고 뱃고동  삼시세끼와 1박2일을 버무린 예능  정규편성 될까 "
## [6] "노홍철  무한도전  복귀  그때는 틀렸지만 지금은 맞다 "
```

```
str(test_2)
```

```
##  chr [1:6] "성소X김종민  한끼줍쇼  밥동무 출연  대세들의 만남 " ...
```

```
class(test_2)
```

```
## [1] "character"
```

```
test_2[1]
```

```
## [1] "성소X김종민  한끼줍쇼  밥동무 출연  대세들의 만남 "
```

```
test_2[2]
```

```
## [1] " 리뷰  안중근 의사의 후예  안재욱의 뮤지컬  영웅  출연은 필연이었을까"
```

```
test_2[3]
```

```
## [1] "정준영  솔로 앨범  1인칭  발표 복귀 청신호 "
```

```
test_2[4]
```

```
## [1] "정준영 가수로  컴백 첫 정규앨범 발표"
```

```
test_2[5]
```

```
## [1] "주먹쥐고 뱃고동  삼시세끼와 1박2일을 버무린 예능  정규편성 될까 "
```

```
test_2[6]
```

```
## [1] "노홍철  무한도전  복귀  그때는 틀렸지만 지금은 맞다 "
```

- 명사 추출 함수를 생성 : ko.words, extractNoun

```
ko.words <- function(doc){
  d <- as.character(doc)
  extractNoun(d)
}
```

- toy 데이터에 대한 TermDocument matrix 생성

```
options(mc.cores=1)
cps <- VCorpus(VectorSource(test_2))
tdm <- TermDocumentMatrix(cps,
                          control=list(tokenize=ko.words,
                                       removePunctuation=T,
                                       wordLengths=c(2, 6),
                                       weighting=weightBin))
```

- why 'read_dic:tag error' message?

```
tdm
```

```
## <<TermDocumentMatrix (terms: 32, documents: 6)>>
## Non-/sparse entries: 37/155
## Sparsity           : 81%
## Maximal term length: 6
## Weighting          : binary (bin)
```

```
tdm.matrix <- as.matrix(tdm)
tdm.matrix
```

```
##              Docs
## Terms         1 2 3 4 5 6
##   1박2일      0 0 0 0 1 0
##   1인칭       0 0 1 0 0 0
##   가수        0 0 0 1 0 0
##   그때        0 0 0 0 0 1
##   노홍철      0 0 0 0 0 1
##   대세        1 0 0 0 0 0
##   리뷰        0 1 0 0 0 0
##   무한도전    0 0 0 0 0 1
##   뮤지컬      0 1 0 0 0 0
##   발표        0 0 1 0 0 0
##   밥동무      1 0 0 0 0 0
##   뱃고동      0 0 0 0 1 0
##   복귀        0 0 1 0 0 1
##   삼시세끼    0 0 0 0 1 0
##   성소x김종민 1 0 0 0 0 0
##   솔로        0 0 1 0 0 0
##   안재욱      0 1 0 0 0 0
##   안중        0 1 0 0 0 0
##   앨범        0 0 1 1 0 0
##   영웅        0 1 0 0 0 0
##   예능        0 0 0 0 1 0
##   의사        0 1 0 0 0 0
##   정규        0 0 0 1 1 0
##   정준영      0 0 1 1 0 0
##   주먹        0 0 0 0 1 0
##   청신        0 0 1 0 0 0
##   출연        1 1 0 0 0 0
##   컴백        0 0 0 1 0 0
##   편성        0 0 0 0 1 0
##   필연        0 1 0 0 0 0
##   한끼줍쇼    1 0 0 0 0 0
##   후예        0 1 0 0 0 0
```

```
nrow(tdm.matrix)
```

```
## [1] 32
```

```
rownames(tdm.matrix)
```

```
##  [1] "1박2일"      "1인칭"       "가수"        "그때"        "노홍철"
##  [6] "대세"        "리뷰"        "무한도전"    "뮤지컬"      "발표"
## [11] "밥동무"      "뱃고동"      "복귀"        "삼시세끼"    "성소x김종민"
## [16] "솔로"        "안재욱"      "안중"        "앨범"        "영웅"
## [21] "예능"        "의사"        "정규"        "정준영"      "주먹"
## [26] "청신"        "출연"        "컴백"        "편성"        "필연"
## [31] "한끼줍쇼"    "후예"
```

```
rownames(tdm.matrix) %in% pro_name
```

```
##  [1]  TRUE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE
## [12] FALSE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [23] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE
```

```
sum(rownames(tdm.matrix) %in% pro_name)
```

```
## [1] 4
```

```
tdm.matrix[rownames(tdm.matrix) %in% pro_name, ]
```

```
##           Docs
## Terms      1 2 3 4 5 6
##   1박2일   0 0 0 0 1 0
##   무한도전 0 0 0 0 0 1
##   삼시세끼 0 0 0 0 1 0
##   한끼줍쇼 1 0 0 0 0 0
```

- '무한도전'은 6번 문건, '삼시세끼'는 5번째 문건, '한끼줍쇼'는 1번째 문건에 해당(제목만 보았을 때)
- real data munging
- title 컬럼에 대한 munging
- title 컬럼 내 NA 처리

```
title <- tvpro$title
title[is.na(title)] <- 'dummy'
title[is.na(title)]
```

```
## character(0)
```

- title 컬럼에 대한 TDM 생성

```
ko.words <- function(doc){
  d <- as.character(doc)
  extractNoun(d)
}

options(mc.cores=1)
cps <- VCorpus(VectorSource(title))
tdm <- TermDocumentMatrix(cps,
                          control=list(tokenize=ko.words,
                                       removePunctuation=T,
                                       wordLengths=c(2, 6),
                                       weighting=weightBin))

tdm_modi <- tdm[dimnames(tdm)$Terms %in% pro_name, ]
title.matrix <- as.matrix(tdm_modi)
```

- 각 방영 프로그램 별 title 컬럼 내 등장횟수 확인

```
str(title.matrix)
```

```
##  int [1:9, 1:9362] 0 0 0 0 0 0 0 0 1 0 ...
##  - attr(*, "dimnames")=List of 2
##   ..$ Terms: chr [1:9] "1박2일" "나혼자산다" "무한도전" "발칙한동거" ...
##   ..$ Docs : chr [1:9362] "1" "2" "3" "4" ...
```

```
rownames(title.matrix)
```

```
## [1] "1박2일"     "나혼자산다" "무한도전"   "발칙한동거" "복면가왕"
## [6] "삼시세끼"   "아는형님"   "정글의법칙" "한끼줍쇼"
```

```
sum(title.matrix[1, ]) ## 1박2일 : 545번
```

```
## [1] 545
```

```
sum(title.matrix[2, ]) ## 나혼자산다 : 200번
```

```
## [1] 200
```

```
sum(title.matrix[3, ]) ## 무한도전 : 721번
```

```
## [1] 721
```

```
sum(title.matrix[4, ]) ## 발칙한동거 : 17번
```

```
## [1] 17
```

```
sum(title.matrix[5, ]) ## 복면가왕 : 468번
```

```
## [1] 468
```

```
sum(title.matrix[6, ]) ## 삼시세끼 : 272번
```

```
## [1] 272
```

```
sum(title.matrix[7, ]) ## 아는형님 : 104번
```

```
## [1] 105
```

```
sum(title.matrix[8, ]) ## 정글의법칙 : 50번
```

```
## [1] 50
```

```
sum(title.matrix[9, ]) ## 한끼줍쇼 : 369번
```

```
## [1] 369
```

- contents 컬럼에 대한 munging

```
str(tvpro)
```

```
## 'data.frame':    9362 obs. of  3 variables:
##  $ date    : chr  "2017-01-31" "2017-01-31" "2017-01-31" "2017-01-31" ...
##  $ title   : chr  "성소X김종민, `한끼줍쇼` 밥동무 출연.. W"대세들의 만남!W"" "[리뷰] 안중근 의사의 후예, 안재욱의 뮤지컬 '영웅' 출연은 필연이었을까" "정준영, 솔로 앨범 '1인칭' 발표…복귀 청신호?" "정준영 가수로 '컴백'…첫 정규앨범 발표" ...
##  $ contents: chr  "..끼줍쇼' 에는 연예대상을 수상한 김종민과 대세 걸그룹으로 우뚝 선 우주소녀의 성소가 밥동무로 출연을 앞두고 .. '"| __truncated__ "▲ 뮤지컬 '영웅(연출 윤호진)' 공.. 최근 인기 예능 프로그램 MBC '무한도전' 팀과 역사 특집 ..형, 이정열, 리사"| __truncated__ "가수 정준영이 첫 솔로 정규 앨범을 발표한다. C9 엔터테인먼트 관계자는 정준영 공식 SNS 및 팬카페 등을 통해 오는 7"| __truncated__ "사진/뉴시스 ( <U+00A0> 최근 KBS 2TV 예능 '1박2일'로 연예계 복귀를 알린 가수 정준영(28)이<U+00A0>오는 2월<U+00A0>"| __truncated__ ...
```

```
contents <- tvpro$contents
```

- NA 유무 확인

```
contents[is.na(contents) == TRUE] ## NA 없음
```

```
## character(0)
```

- contents 컬럼에 대한 TDM 생성

```
options(mc.cores=1)
cps <- VCorpus(VectorSource(contents))
tdm <- TermDocumentMatrix(cps,
                          control=list(tokenize=ko.words,
                                       removePunctuation=T,
                                       wordLengths=c(2, 6),
                                       weighting=weightBin))
```

- TDM을 matrix 자료구조로 변환

```
tdm
```

```
## <<TermDocumentMatrix (terms: 15806, documents: 9362)>>
## Non-/sparse entries: 177386/147798386
## Sparsity           : 100%
## Maximal term length: 6
## Weighting          : binary (bin)
```

```
str(tdm)
```

```
## List of 6
##  $ i       : int [1:177386] 1485 1691 2679 3048 3344 3706 4081 4265 6130 7019 ...
##  $ j       : int [1:177386] 1 1 1 1 1 1 1 1 1 1 ...
##  $ v       : int [1:177386] 1 1 1 1 1 1 1 1 1 1 ...
##  $ nrow    : int 15806
##  $ ncol    : int 9362
##  $ dimnames:List of 2
##   ..$ Terms: chr [1:15806] "<U+0301>였는데" "<U+0301>할배들" "■실제론" "■올스타전" ...
##   ..$ Docs : chr [1:9362] "1" "2" "3" "4" ...
##  - attr(*, "class")= chr [1:2] "TermDocumentMatrix" "simple_triplet_matrix"
##  - attr(*, "weighting")= chr [1:2] "binary" "bin"
```

```
tdm_modi <- tdm[dimnames(tdm)$Terms %in% pro_name, ]
contents.matrix <- as.matrix(tdm_modi)
str(contents.matrix)
```

```
##  int [1:9, 1:9362] 0 0 0 0 0 0 0 0 1 0 ...
##  - attr(*, "dimnames")=List of 2
##   ..$ Terms: chr [1:9] "1박2일" "나혼자산다" "무한도전" "발칙한동거" ...
##   ..$ Docs : chr [1:9362] "1" "2" "3" "4" ...
```

- contents 컬럼 내 각 tv program들의 방영 횟수 확인

```
rownames(contents.matrix)
```

```
## [1] "1박2일"     "나혼자산다" "무한도전"   "발칙한동거" "복면가왕"
## [6] "삼시세끼"   "아는형님"   "정글의법칙" "한끼줍쇼"
```

```
sum(contents.matrix[1, ]) ## 1박2일 : 1098번
```

```
## [1] 1098
```

```
sum(contents.matrix[2, ]) ## 나혼자산다 : 229번
```

```
## [1] 229
```

```
sum(contents.matrix[3, ]) ## 무한도전 : 2254번
```

```
## [1] 2254
```

```
sum(contents.matrix[4, ]) ## 발칙한동거 : 9번
```

```
## [1] 9
```

```
sum(contents.matrix[5, ]) ## 복면가왕 : 979번
```

```
## [1] 979
```

```
sum(contents.matrix[6, ]) ## 삼시세끼 : 791번
```

```
## [1] 791
```

```
sum(contents.matrix[7, ]) ## 아는형님 : 169번
```

```
## [1] 169
```

```
sum(contents.matrix[8, ]) ## 정글의법칙 : 84번
```

```
## [1] 84
```

```
sum(contents.matrix[9, ]) ## 한끼줍쇼 : 419번
```

```
## [1] 419
```

- title matrix와 contents matrix 합치기(sum.matrix)

```
dim(title.matrix)
```

```
## [1]    9 9362
```

```
dim(contents.matrix)
```

```
## [1]    9 9362
```

```
sum.matrix <- title.matrix + contents.matrix
```

- 날짜 정보 table 만들기(차후 분석을 위해 별도 저장 관리)

```
head(tvpro$date)
```

```
## [1] "2017-01-31" "2017-01-31" "2017-01-31" "2017-01-31" "2017-01-31"
## [6] "2017-01-31"
```

```
day <- tvpro$date
numb <- as.numeric(c(1:9362))

date.table <- data.frame(number = numb, date = day, stringsAsFactors = F)
head(date.table)
```

```
##   number       date
## 1      1 2017-01-31
## 2      2 2017-01-31
## 3      3 2017-01-31
## 4      4 2017-01-31
## 5      5 2017-01-31
## 6      6 2017-01-31
```

```
nrow(date.table)
```

```
## [1] 9362
```

```
str(date.table)
```

```
## 'data.frame':    9362 obs. of  2 variables:
##  $ number: num  1 2 3 4 5 6 7 8 9 10 ...
##  $ date  : chr  "2017-01-31" "2017-01-31" "2017-01-31" "2017-01-31" ...
```

- sum.matrix 내 모든 원소의 값이 0인 열(column)들을 제외시키기

```
col_sums <- colSums(sum.matrix)
head(col_sums)
```

```
## 1 2 3 4 5 6
## 2 1 1 1 4 1
```

```
str(col_sums)
```

```
##  Named num [1:9362] 2 1 1 1 4 1 2 1 2 2 ...
##  - attr(*, "names")= chr [1:9362] "1" "2" "3" "4" ...
```

```
non_zerosum_col_names <- names(col_sums)[col_sums != 0]

class(non_zerosum_col_names)
```

```
## [1] "character"
```

```
extracted.matrix <- sum.matrix[, non_zerosum_col_names]

extracted.matrix.margin <- rbind(extracted.matrix, colSums(extracted.matrix))

rownames(extracted.matrix.margin)[10] <- 'Sum'

ncol(extracted.matrix.margin) # 6706 개의 열을 가진 행렬
```

```
## [1] 6685
```

- 결론적으로 tvprogrames_name.txt에 있는 프로그램명과 관련이 있는 document의 갯수는 6,706개이며 이를 extracted.matrix.margin에 저장하였음.

1.3 월별 분석이 가능하도록 데이터 전처리

- 상기 과정에서 추출된 extracted.matrix.margin 행렬의 역행렬 구하기

```
reverse.matrix <- t(extracted.matrix.margin)

reverse.df <- as.data.frame(reverse.matrix)
head(reverse.df)
```

```
##   1박2일 나혼자산다 무한도전 발칙한동거 복면가왕 삼시세끼 아는형님
## 1      0          0        0          0        0        0        0
## 2      0          0        1          0        0        0        0
## 3      1          0        0          0        0        0        0
## 4      1          0        0          0        0        0        0
## 5      2          0        0          0        0        2        0
## 6      0          0        1          0        0        0        0
##   정글의법칙 한끼줍쇼 Sum
## 1          0        2   2
## 2          0        0   1
## 3          0        0   1
## 4          0        0   1
## 5          0        0   4
## 6          0        0   1
```

```
class(rownames(reverse.df))
```

```
## [1] "character"
```

```
class(as.numeric(rownames(reverse.df)))
```

```
## [1] "numeric"
```

```
number_id <- as.numeric(rownames(reverse.df))
df <- cbind(number_id, reverse.df)
head(df)
```

```
##   number_id 1박2일 나혼자산다 무한도전 발칙한동거 복면가왕 삼시세끼
## 1         1      0          0        0          0        0        0
## 2         2      0          0        1          0        0        0
## 3         3      1          0        0          0        0        0
## 4         4      1          0        0          0        0        0
## 5         5      2          0        0          0        0        2
## 6         6      0          0        1          0        0        0
##   아는형님 정글의법칙 한끼줍쇼 Sum
## 1        0          0        2   2
## 2        0          0        0   1
## 3        0          0        0   1
## 4        0          0        0   1
## 5        0          0        0   4
## 6        0          0        0   1
```

```
str(df)
```

```
## 'data.frame':    6685 obs. of  11 variables:
##  $ number_id : num  1 2 3 4 5 6 7 8 9 10 ...
##  $ 1박2일    : num  0 0 1 1 2 0 0 0 0 0 ...
##  $ 나혼자산다: num  0 0 0 0 0 0 0 0 0 0 ...
##  $ 무한도전  : num  0 1 0 0 0 1 0 1 0 0 ...
##  $ 발칙한동거: num  0 0 0 0 0 0 0 0 0 0 ...
##  $ 복면가왕  : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ 삼시세끼  : num  0 0 0 0 2 0 0 0 0 0 ...
##  $ 아는형님  : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ 정글의법칙: num  0 0 0 0 0 0 0 0 0 0 ...
##  $ 한끼줍쇼  : num  2 0 0 0 0 0 2 0 2 2 ...
##  $ Sum       : num  2 1 1 1 4 1 2 1 2 2 ...
```

- join 연산 수행을 위해 날짜 정보 데이블의 이름 변경

```
head(date.table)
```

```
##   number       date
## 1      1 2017-01-31
## 2      2 2017-01-31
## 3      3 2017-01-31
## 4      4 2017-01-31
## 5      5 2017-01-31
## 6      6 2017-01-31
```

```
str(date.table)
```

```
## 'data.frame':    9362 obs. of  2 variables:
##  $ number: num  1 2 3 4 5 6 7 8 9 10 ...
##  $ date  : chr  "2017-01-31" "2017-01-31" "2017-01-31" "2017-01-31" ...
```

```
colnames(date.table) <- c('number_id' , 'DATE')
str(date.table)
```

```
## 'data.frame':    9362 obs. of  2 variables:
##  $ number_id: num  1 2 3 4 5 6 7 8 9 10 ...
##  $ DATE     : chr  "2017-01-31" "2017-01-31" "2017-01-31" "2017-01-31" ...
```

- left join

```
df_1 <- left_join(df, date.table)
head(df_1)
```

```
##   number_id 1박2일 나혼자산다 무한도전 발칙한동거 복면가왕 삼시세끼
## 1         1      0          0        0          0        0        0
## 2         2      0          0        1          0        0        0
## 3         3      1          0        0          0        0        0
## 4         4      1          0        0          0        0        0
## 5         5      2          0        0          0        0        2
## 6         6      0          0        1          0        0        0
##   아는형님 정글의법칙 한끼줍쇼 Sum       DATE
## 1        0          0        2   2 2017-01-31
## 2        0          0        0   1 2017-01-31
## 3        0          0        0   1 2017-01-31
## 4        0          0        0   1 2017-01-31
## 5        0          0        0   4 2017-01-31
## 6        0          0        0   1 2017-01-31
```

```
str(df_1)
```

```
## 'data.frame':    6685 obs. of  12 variables:
##  $ number_id : num  1 2 3 4 5 6 7 8 9 10 ...
##  $ 1박2일    : num  0 0 1 1 2 0 0 0 0 0 ...
##  $ 나혼자산다: num  0 0 0 0 0 0 0 0 0 0 ...
##  $ 무한도전  : num  0 1 0 0 0 1 0 1 0 0 ...
##  $ 발칙한동거: num  0 0 0 0 0 0 0 0 0 0 ...
##  $ 복면가왕  : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ 삼시세끼  : num  0 0 0 0 2 0 0 0 0 0 ...
##  $ 아는형님  : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ 정글의법칙: num  0 0 0 0 0 0 0 0 0 0 ...
##  $ 한끼줍쇼  : num  2 0 0 0 0 0 2 0 2 2 ...
##  $ Sum       : num  2 1 1 1 4 1 2 1 2 2 ...
##  $ DATE      : chr  "2017-01-31" "2017-01-31" "2017-01-31" "2017-01-31" ...
```

```
df_2 <- df_1[, c(1, 12, 2:11)]
head(df_2)
```

```
##   number_id       DATE 1박2일 나혼자산다 무한도전 발칙한동거 복면가왕
## 1         1 2017-01-31      0          0        0          0        0
## 2         2 2017-01-31      0          0        1          0        0
## 3         3 2017-01-31      1          0        0          0        0
## 4         4 2017-01-31      1          0        0          0        0
## 5         5 2017-01-31      2          0        0          0        0
## 6         6 2017-01-31      0          0        1          0        0
##   삼시세끼 아는형님 정글의법칙 한끼줍쇼 Sum
## 1        0        0          0        2   2
## 2        0        0          0        0   1
## 3        0        0          0        0   1
## 4        0        0          0        0   1
## 5        2        0          0        0   4
## 6        0        0          0        0   1
```

- 날짜 정보를 이용 하고 월별 분석이 가능하도록 'year_month' 변수를 factor형으로 생성

```
df_2 <- df_2 %>% mutate( year = substr(DATE, 1, 4),
                         month = substr(DATE, 6, 7),
                         day = substr(DATE, 9, 10),
                         year_month = substr(DATE, 1, 7))
str(df_2)
```

```
## 'data.frame':    6685 obs. of  16 variables:
##  $ number_id : num  1 2 3 4 5 6 7 8 9 10 ...
##  $ DATE      : chr  "2017-01-31" "2017-01-31" "2017-01-31" "2017-01-31" ...
##  $ 1박2일    : num  0 0 1 1 2 0 0 0 0 0 ...
##  $ 나혼자산다: num  0 0 0 0 0 0 0 0 0 0 ...
##  $ 무한도전  : num  0 1 0 0 0 1 0 1 0 0 ...
##  $ 발칙한동거: num  0 0 0 0 0 0 0 0 0 0 ...
##  $ 복면가왕  : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ 삼시세끼  : num  0 0 0 0 2 0 0 0 0 0 ...
##  $ 아는형님  : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ 정글의법칙: num  0 0 0 0 0 0 0 0 0 0 ...
##  $ 한끼줍쇼  : num  2 0 0 0 0 0 2 0 2 2 ...
##  $ Sum       : num  2 1 1 1 4 1 2 1 2 2 ...
##  $ year      : chr  "2017" "2017" "2017" "2017" ...
##  $ month     : chr  "01" "01" "01" "01" ...
##  $ day       : chr  "31" "31" "31" "31" ...
##  $ year_month: chr  "2017-01" "2017-01" "2017-01" "2017-01" ...
```

```
df_3 <- df_2[ , c(1, 16, 3:12)]
head(df_3)
```

```
##   number_id year_month 1박2일 나혼자산다 무한도전 발칙한동거 복면가왕
## 1         1    2017-01      0          0        0          0        0
## 2         2    2017-01      0          0        1          0        0
## 3         3    2017-01      1          0        0          0        0
## 4         4    2017-01      1          0        0          0        0
## 5         5    2017-01      2          0        0          0        0
## 6         6    2017-01      0          0        1          0        0
##   삼시세끼 아는형님 정글의법칙 한끼줍쇼 Sum
## 1        0        0          0        2   2
## 2        0        0          0        0   1
## 3        0        0          0        0   1
## 4        0        0          0        0   1
## 5        2        0          0        0   4
## 6        0        0          0        0   1
```

```
str(df_3)
```

```
## 'data.frame':    6685 obs. of  12 variables:
##  $ number_id : num  1 2 3 4 5 6 7 8 9 10 ...
##  $ year_month: chr  "2017-01" "2017-01" "2017-01" "2017-01" ...
##  $ 1박2일    : num  0 0 1 1 2 0 0 0 0 0 ...
##  $ 나혼자산다: num  0 0 0 0 0 0 0 0 0 0 ...
##  $ 무한도전  : num  0 1 0 0 0 1 0 1 0 0 ...
##  $ 발칙한동거: num  0 0 0 0 0 0 0 0 0 0 ...
##  $ 복면가왕  : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ 삼시세끼  : num  0 0 0 0 2 0 0 0 0 0 ...
##  $ 아는형님  : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ 정글의법칙: num  0 0 0 0 0 0 0 0 0 0 ...
##  $ 한끼줍쇼  : num  2 0 0 0 0 0 2 0 2 2 ...
##  $ Sum       : num  2 1 1 1 4 1 2 1 2 2 ...
```

```
df_3$year_month <- as.factor(df_3$year_month)
```

- 결론적으로 '월별' 분석이 가능하도록 데이터 전처리하였으며, 최종 전처리 결과를 df_3에 저장하였음.

1.4 월별/프로그램별 나온 단어들의 빈도수 분석

```
head(df_3)
```

```
##   number_id year_month 1박2일 나혼자산다 무한도전 발칙한동거 복면가왕
## 1         1    2017-01      0          0        0          0        0
## 2         2    2017-01      0          0        1          0        0
## 3         3    2017-01      1          0        0          0        0
## 4         4    2017-01      1          0        0          0        0
## 5         5    2017-01      2          0        0          0        0
## 6         6    2017-01      0          0        1          0        0
##   삼시세끼 아는형님 정글의법칙 한끼줍쇼 Sum
## 1        0        0          0        2   2
## 2        0        0          0        0   1
## 3        0        0          0        0   1
## 4        0        0          0        0   1
## 5        2        0          0        0   4
## 6        0        0          0        0   1
```

```
table(df_3$year_month) ## 16.10월, 16.11월, 16.12월, 17.1월 .... 4개 달에 대한 월별 분석 진행
```

```
##
## 2016-10 2016-11 2016-12 2017-01
##    2094    1672    1455    1464
```

```
df_3 %>%
  group_by(year_month) %>%
    summarise(freq.1박2일 = sum(`1박2일`),
              freq.나혼자산다 = sum(나혼자산다),
              freq.무한도전 = sum(무한도전),
              freq.발칙한동거 = sum(발칙한동거),
              freq.복면가왕 = sum(복면가왕),
              freq.삼시세끼 = sum(삼시세끼),
              freq.아는형님 = sum(아는형님),
              freq.정글의법칙 = sum(정글의법칙),
              freq.한끼줍쇼 = sum(한끼줍쇼))
```

```
## # A tibble: 4 x 10
##   year_month freq.1박2일 freq.나혼자산다 freq.무한도전 freq.발칙한동거
##       <fctr>       <dbl>          <dbl>         <dbl>           <dbl>
## 1    2016-10         375            118           900               0
## 2    2016-11         309            138           756               0
## 3    2016-12         504             79           582               0
## 4    2017-01         455             94           737              26
## # ... with 5 more variables: freq.복면가왕 <dbl>, freq.삼시세끼 <dbl>,
## #   freq.아는형님 <dbl>, freq.정글의법칙 <dbl>, freq.한끼줍쇼 <dbl>
```

```
month_tvpro <- df_3 %>%
  group_by(year_month) %>%
  summarise(freq.1박2일 = sum(`1박2일`),
            freq.나혼자산다 = sum(나혼자산다),
            freq.무한도전 = sum(무한도전),
            freq.발칙한동거 = sum(발칙한동거),
            freq.복면가왕 = sum(복면가왕),
            freq.삼시세끼 = sum(삼시세끼),
            freq.아는형님 = sum(아는형님),
            freq.정글의법칙 = sum(정글의법칙),
            freq.한끼줍쇼 = sum(한끼줍쇼))

month_tvpro_df <- as.data.frame(month_tvpro)
month_tvpro_df
```

```
##   year_month freq.1박2일 freq.나혼자산다 freq.무한도전 freq.발칙한동거
## 1    2016-10         375            118           900               0
## 2    2016-11         309            138           756               0
## 3    2016-12         504             79           582               0
## 4    2017-01         455             94           737              26
##   freq.복면가왕 freq.삼시세끼 freq.아는형님 freq.정글의법칙 freq.한끼줍쇼
## 1           367           587            89              13           345
## 2           479           176            84              62           134
## 3           334           158            40              20           130
## 4           267           142            61              39           179
```

- 결론적으로 '월별' / '프로그램' 별 나온 단어들의 빈도수를 계산하여 month_tvpro_df에 저장하였음.

1.5 월별 프로그램 방영 비율 분석, 동 비율에 대한 그래프 그리기

- dataframe 형식의 데이터를 prop.table 으로 전환하는 test

```
x <- data.frame(id=letters[1:3],val0=1:3,val1=4:6,val2=7:9)
x
```

```
##   id val0 val1 val2
## 1  a    1    4    7
## 2  b    2    5    8
## 3  c    3    6    9
```

```
prop.table(as.matrix(x[-1]),margin=1)
```

```
##           val0      val1      val2
## [1,] 0.08333333 0.3333333 0.5833333
## [2,] 0.13333333 0.3333333 0.5333333
## [3,] 0.16666667 0.3333333 0.5000000
```

- making a prop.table for month_tvpro_df

```
month_tvpro_df
```

```
##   year_month freq.1박2일 freq.나혼자산다 freq.무한도전 freq.발칙한동거
## 1    2016-10         375            118           900               0
## 2    2016-11         309            138           756               0
## 3    2016-12         504             79           582               0
## 4    2017-01         455             94           737              26
##   freq.복면가왕 freq.삼시세끼 freq.아는형님 freq.정글의법칙 freq.한끼줍쇼
## 1           367           587           89             13           345
## 2           479           176           84             62           134
## 3           334           158           40             20           130
## 4           267           142           61             39           179
```

```
p.table <- prop.table(as.matrix(month_tvpro_df[-1]),margin=1)*100
p.table
```

```
##      freq.1박2일 freq.나혼자산다 freq.무한도전 freq.발칙한동거
## [1,]    13.42162        4.223336      32.21188             0.0
## [2,]    14.45276        6.454630      35.36015             0.0
## [3,]    27.28749        4.277206      31.51056             0.0
## [4,]    22.75000        4.700000      36.85000             1.3
##      freq.복면가왕 freq.삼시세끼 freq.아는형님 freq.정글의법칙
## [1,]      13.13529     21.009306      3.185397       0.4652827
## [2,]      22.40412      8.231993      3.928906       2.8999065
## [3,]      18.08338      8.554413      2.165674       1.0828370
## [4,]      13.35000      7.100000      3.050000       1.9500000
##      freq.한끼줍쇼
## [1,]     12.347888
## [2,]      6.267540
## [3,]      7.038441
## [4,]      8.950000
```

- p.table 에 이름부여

```
rownames(p.table) <- month_tvpro_df$year_month
str(p.table)
```

```
##  num [1:4, 1:9] 13.42 14.45 27.29 22.75 4.22 ...
##  - attr(*, "dimnames")=List of 2
##   ..$ : chr [1:4] "2016-10" "2016-11" "2016-12" "2017-01"
##   ..$ : chr [1:9] "freq.1박2일" "freq.나혼자산다" "freq.무한도전" "freq.발칙한동거" ...
```

```
colnames(p.table) <- c('percent.1박2일', 'percent.나혼자산다', 'percent.무한도전', 'percent.발착한동거',
                       'percent.복면가왕', 'percent.삼시세끼', 'percent.아는형님', 'percent.정글의법칙', 'percent.한끼줍
쇼')
str(p.table)
```

```
##  num [1:4, 1:9] 13.42 14.45 27.29 22.75 4.22 ...
##  - attr(*, "dimnames")=List of 2
##   ..$ : chr [1:4] "2016-10" "2016-11" "2016-12" "2017-01"
##   ..$ : chr [1:9] "percent.1박2일" "percent.나혼자산다" "percent.무한도전" "percent.발착한동거" ...
```

- 동 비율에 대한 그래프 그리기

```
p.table
```

```
##         percent.1박2일 percent.나혼자산다 percent.무한도전
## 2016-10       13.42162           4.223336         32.21188
## 2016-11       14.45276           6.454630         35.36015
## 2016-12       27.28749           4.277206         31.51056
## 2017-01       22.75000           4.700000         36.85000
##         percent.발착한동거 percent.복면가왕 percent.삼시세끼
## 2016-10                0.0         13.13529        21.009306
## 2016-11                0.0         22.40412         8.231993
## 2016-12                0.0         18.08338         8.554413
## 2017-01                1.3         13.35000         7.100000
##         percent.아는형님 percent.정글의법칙 percent.한끼줍쇼
## 2016-10         3.185397          0.4652827        12.347888
## 2016-11         3.928906          2.8999065         6.267540
## 2016-12         2.165674          1.0828370         7.038441
## 2017-01         3.050000          1.9500000         8.950000
```

```
df.table <- as.data.frame(p.table)
df.table
```

```
##         percent.1박2일 percent.나혼자산다 percent.무한도전
## 2016-10       13.42162           4.223336         32.21188
## 2016-11       14.45276           6.454630         35.36015
## 2016-12       27.28749           4.277206         31.51056
## 2017-01       22.75000           4.700000         36.85000
##         percent.발착한동거 percent.복면가왕 percent.삼시세끼
## 2016-10                0.0         13.13529        21.009306
## 2016-11                0.0         22.40412         8.231993
## 2016-12                0.0         18.08338         8.554413
## 2017-01                1.3         13.35000         7.100000
##         percent.아는형님 percent.정글의법칙 percent.한끼줍쇼
## 2016-10         3.185397          0.4652827        12.347888
## 2016-11         3.928906          2.8999065         6.267540
## 2016-12         2.165674          1.0828370         7.038441
## 2017-01         3.050000          1.9500000         8.950000
```

```
rownames(df.table)
```

```
## [1] "2016-10" "2016-11" "2016-12" "2017-01"
```

```
colnames(df.table)
```

```
## [1] "percent.1박2일"     "percent.나혼자산다" "percent.무한도전"
## [4] "percent.발착한동거" "percent.복면가왕"   "percent.삼시세끼"
## [7] "percent.아는형님"   "percent.정글의법칙" "percent.한끼줍쇼"
```

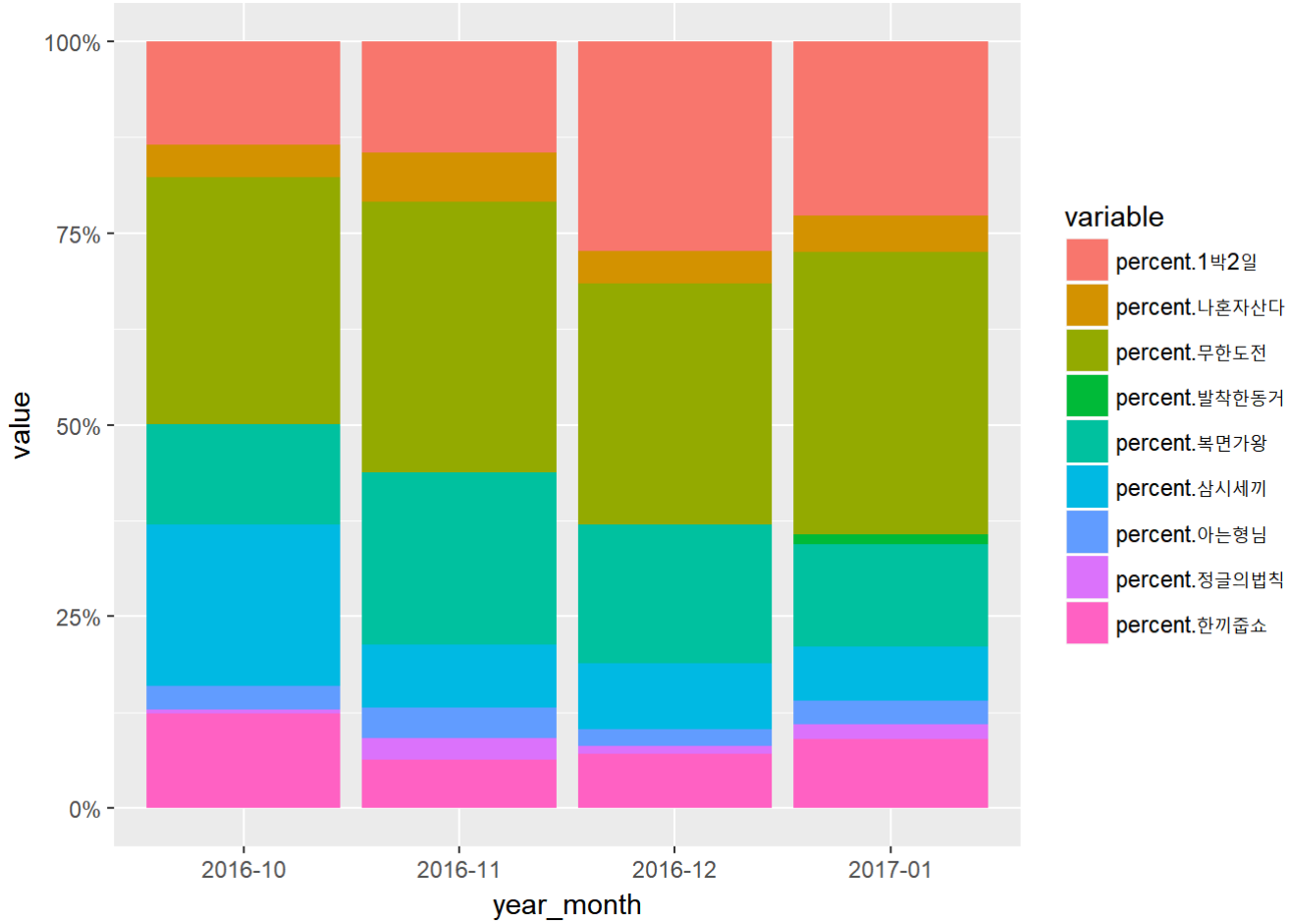
```
library(reshape2)

melted <- melt(cbind(df.table, year_month = rownames(df.table)), id.vars = c('year_month'))
head(melted)
```

```
##   year_month          variable     value
## 1    2016-10     percent.1박2일 13.421618
## 2    2016-11     percent.1박2일 14.452760
## 3    2016-12     percent.1박2일 27.287493
## 4    2017-01     percent.1박2일 22.750000
## 5    2016-10 percent.나혼자산다  4.223336
## 6    2016-11 percent.나혼자산다  6.454630
```

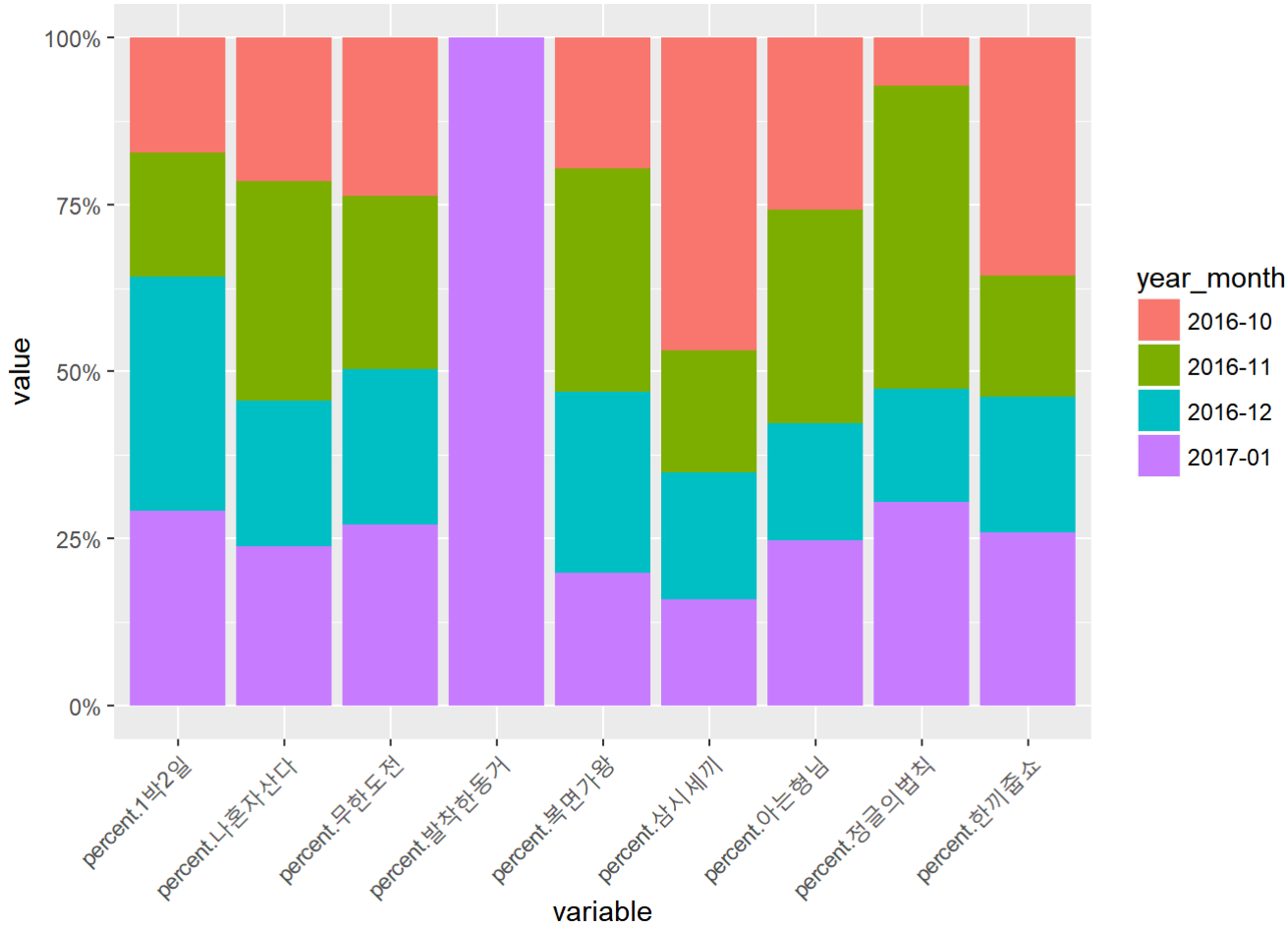
```
library(ggplot2)
library(scales)

ggplot(melted, aes(x = year_month, y = value, fill = variable)) +
  geom_bar(position = "fill",stat = "identity") +
  scale_y_continuous(labels = percent_format())
```



- 혹시나 이런 거꾸로 그래프도 필요한지...(출제자 의도를 잘 몰라서리...)

```
ggplot(melted, aes(x = variable, y = value, fill = year_month)) +
  geom_bar(position = "fill",stat = "identity") +
  scale_y_continuous(labels = percent_format()) +
  theme(axis.text.x=element_text(angle=45,hjust=1))
```

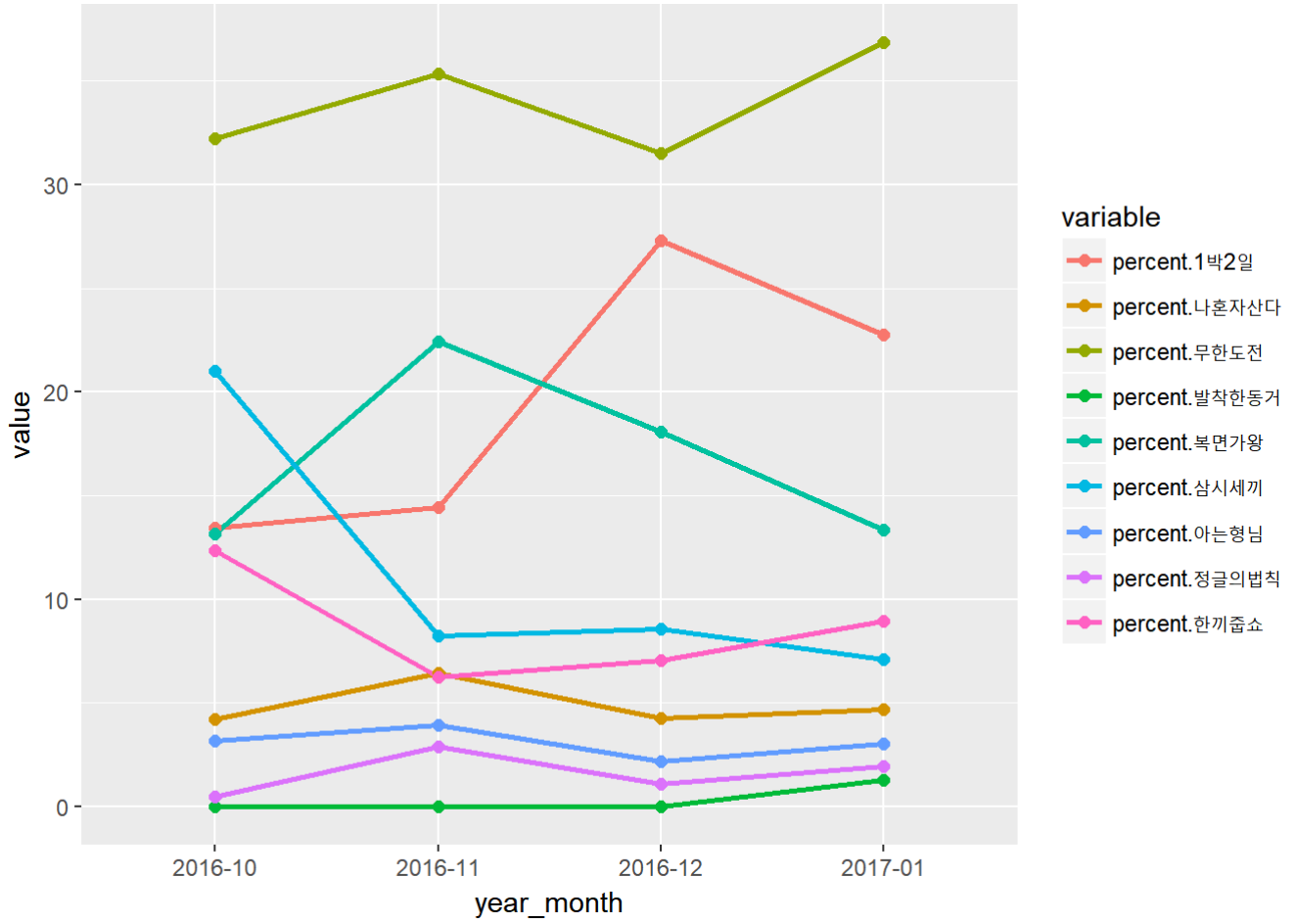


- using geom_line()

```
head(melted)
```

```
##   year_month           variable     value
## 1    2016-10     percent.1박2일 13.421618
## 2    2016-11     percent.1박2일 14.452760
## 3    2016-12     percent.1박2일 27.287493
## 4    2017-01     percent.1박2일 22.750000
## 5    2016-10 percent.나혼자산다  4.223336
## 6    2016-11 percent.나혼자산다  6.454630
```

```
ggplot(melted, aes(x = year_month, y = value, group = variable, colour = variable)) +
  geom_line(size = 1) + geom_point(size = 2)
```


value
30
20
10
0
2016-10
2016-11
2016-12
2017-01
year_month
variable
percent.1박2일
percent.나혼자산다
percent.무한도전
percent.발착한동거
percent.복면가왕
percent.삼시세끼
percent.아는형님
percent.정글의법칙
percent.한끼줍쇼