시대별 인기곡 분석

빅데이터 10기
2019-08-29
이광진

1. 분석 기획

2. 데이터 수집 / 정제 / 시각화

3. 결론

1. 분석 기획 - 주제 선정

1. 홍콩 여행 계획

- 홍콩 키워드 SNS 검색 / 데이터 수집
- 주요 관광지, 맛집, 숙소 등 분석
- 여행 계획 수립

2. 커뮤니티 신조어 분석

- 주요 커뮤니티 게시판 글 수집
- 최근 많이 사용되는 신조어 분석

3. 가사 분석

- **시대별 인기곡 차트 수집**
- **노래, 가수, 장르, 가사 분석**

1. 분석 기획 - 분석 목적

1. 시대별 인기 곡 가수 분석

- 그 시대의 인기곡을 가장 많이 부른 가수 Top 10 순위
- 시각화 기법 : 표??

2. 시대별 인기 곡 장르 분석

- 그 시대의 인기곡 장르 별 곡수, 비율
- 시각화 기법 : 스택형 막대그래프

3. 시대별 인기 곡 가사 분석

- 텍스트 마이닝을 통해 시대별 인기 곡들의 가사에는 어떤 단어들이 많이 나오는지
- 가설 : 시대를 막론하고 가장 많은 단어는 '사랑'이다.
- 시각화 기법 : wordcloud

1. 분석 기획 - 분석 계획

1. 데이터 수집

- 음원사이트 이용(멜론, 벅스, 지니 등)
- 시대별, 연도별 인기 음원 차트 수집
- 각 곡의 가사 수집

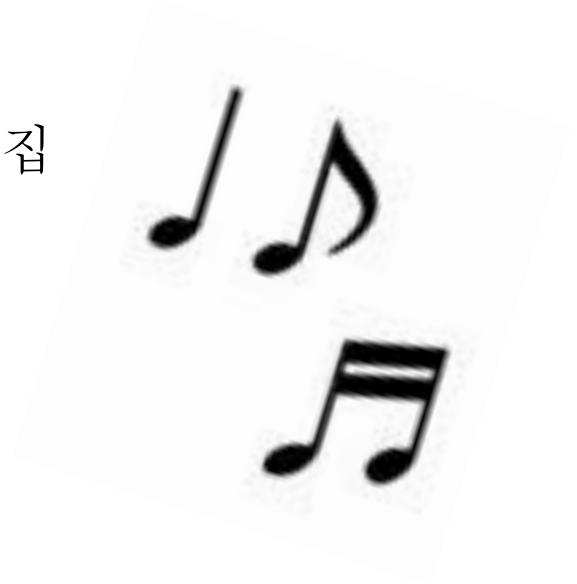
2. 데이터 정제

- 수집된 데이터 확인
- 정규식 표현을 이용, 의미없는 문자 등을 제거

3. 분석 및 시각화

- 분석 목적 및 시각화 수단에 맞는 데이터 분석

2. 데이터 수집 / 정제 / 시각화 - 데이터 수집



1. 데이터 수집

- library(rvest) 이용
- 웹 스크롤링 진행

멜론

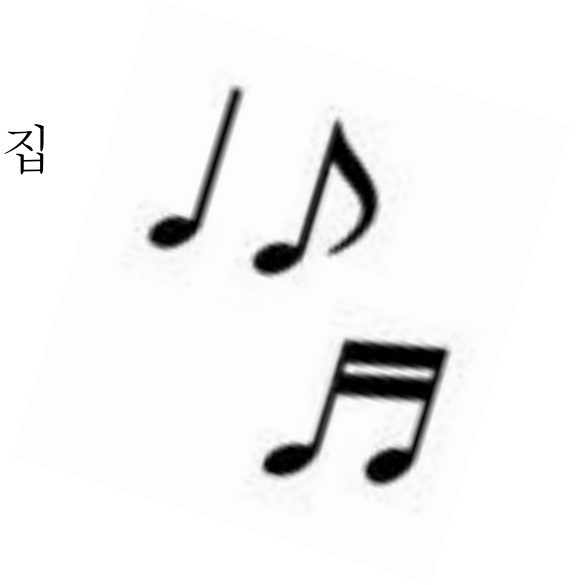
1. 시대별 html

```
> html <-
read_html('https://www.melon.com/chart/age/index.htm?chartType=YE&chartGenre=KPOP&chartDate=1977')
```

```
Error in open.connection(x, "rb") : HTTP error 406.
```

- 오류 발생 / 자료 수집 불가

2. 데이터 수집 / 정제 / 시각화 - 데이터 수집



벅스

1. 연도 html / 각 연도별 앨범 link 정보 / 연도별 앨범명 text(ex, 2018)

```
library(rvest)
#벅스 연도별 html
html <- read_html('https://music.bugs.co.kr/years')
#각 연도별 앨범 link 정보
yurl <- html_nodes(html, xpath='//*[@id="container"]/section/div/ul/li/figure/figcaption/a[1]')%>%
  html_attr('href')
#연도별 앨범명 text
year <- html_nodes(html, xpath='//*[@id="container"]/section/div/ul/li/figure/figcaption/a[1]')%>%
  html_text()
```

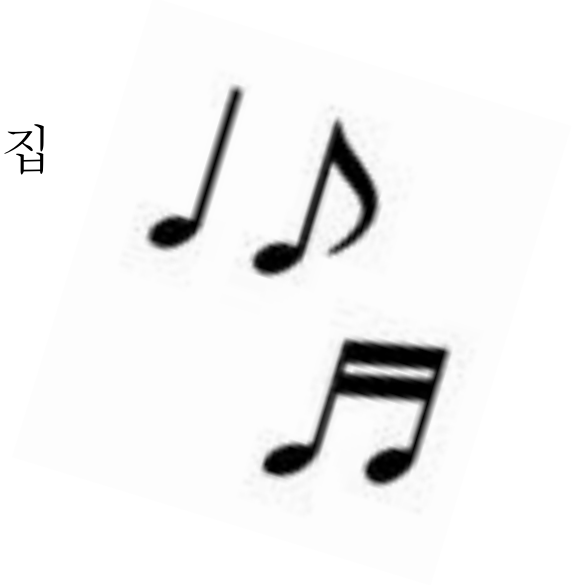
2. 데이터 수집 / 정제 / 시각화 - 데이터 수집

벅스
2. 연도별 앨범에서 각 노래 정보 링크 수집

```
#노래 정보 링크
surl <- html_nodes(album, xpath='//*[@id="ESALBUM30871"]/table/tbody/tr/td[3]/a')%>%
  html_attr('href')
```

- 해당 url 에서 xpath를 이용하여 곡명과 가수 정보를 뽑으려 했으나
  1곡에 2명의 가수가 있는 경우 데이터의 개수가 차이남 (100곡, 104명)
- html 구성이 table 형태로 되어있어 html_table을 이용했지만
  1곡에 2명의 가수가 있는 경우 1번째 가수만 데이터가 2번 뽑힘

2. 데이터 수집 / 정제 / 시각화 - 데이터 수집



벅스
3. 노래 정보 url에서 곡명, 가수, 가사 수집

```
library(stringr)
#1곡 정보 html
songinfo <- read_html(surl[1])
#곡 명
html_nodes(songinfo, xpath='//*[@id="container"]/header/div/h1')%>%
  html_text()%>%
  str_trim()
#아티스트
html_nodes(songinfo, xpath='//*[@id="container"]/section[1]/div/div[1]/table/tbody/tr[1]/td')%>%
  html_text()%>%
  str_trim()%>%
  str_replace_all('[[:space:]]+',' ')
#곡 가사
html_nodes(songinfo, xpath='//*[@id="container"]/section[2]/div/div/xmp')%>%
  html_text()%>%
  str_replace_all('[[:space:]]+',' ')
```

2. 데이터 수집 / 정제 / 시각화 - 데이터 수집

벅스

4. 반복문을 통한 시대별 인기곡 정보 수집

- 앞선 script를 모아 반복문을 작성하여 자료 수집하는 중 오류 발생
- 노래정보가 담긴 url의 xpath 중 id의 숫자값이 다음 앨범으로 넘어갈때마다 바뀜
  숫자값은 앨범 url의 마지막 숫자패턴과 동일

```
#변경 전
surl <- html_nodes(album, xpath='//*[@id="ESALBUM30871"]/table/tbody/tr/td[3]/a')%>%
  html_attr('href')
#변경 후
surl <- html_nodes(album,
xpath=paste0('//*[@id="ESALBUM',str_extract(yurl[2],'[0-9]+'),'"]/table/tbody/tr/td[3]/a'))%>%
                html_attr('href')
```

- 이외의 19금 가사 오류 / 재생불가 노래 가사 경로 오류 발생
  xpath 경로 변경, class이름으로 구분하여 해결

2. 데이터 수집 / 정제 / 시각화 - 데이터 수집

벅스

5. 최종 data 수집 script

```
html <- read_html('https://music.bugs.co.kr/years')
yurl <- html_nodes(html, xpath='//*[@id="container"]/section/div/ul/li/figure/figcaption/a[1]')%>%
  html_attr('href')
year <- html_nodes(html, xpath='//*[@id="container"]/section/div/ul/li/figure/figcaption/a[1]')%>%
  html_text()
df <- NULL
for (i in 1:length(yurl)){
  album <- read_html(yurl[i])
  surl <- html_nodes(album, xpath=paste0('//*[@id="ESALBUM',str_extract(yurl[i],'[0-9]+'),'"]/table/tbody/tr/td[3]/a'))%>%
    html_attr('href')
  for (j in 1:length(surl)){
    songinfo <- read_html(surl[j])
    song <- html_nodes(songinfo, xpath='//*[@id="container"]/header/div/h1')%>%
      html_text()%>%
      str_trim()
    artist <- html_nodes(songinfo, xpath='//*[@id="container"]/section[1]/div/div[1]/table/tbody/tr[1]/td')%>%
      html_text()%>%
      str_trim()%>%
      str_replace_all('[[:space:]]+',' ')
    lylics <- html_nodes(songinfo, '.lyricsContainer')%>%
      html_text()%>%
      str_replace_all('[[:space:]]+',' ')
    df <- rbind(df, data.frame(s=song,
                               a=artist,
                               l=lylics,
                               y=year[i]))

  }
}
```

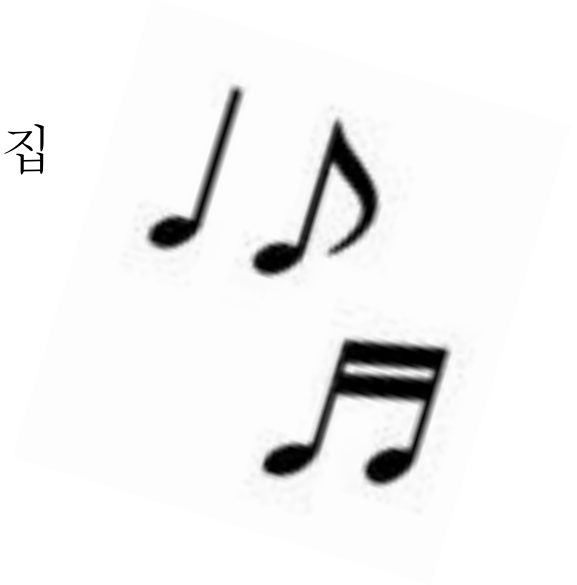
- 벅스에는 노래 장르가 없음…..

2. 데이터 수집 / 정제 / 시각화 - 데이터 수집

지니
1. 연도별 html / 노래 정보 url

```
#지니
html <- read_html('https://www.genie.co.kr/chart/musicHistory?year=2018&category=0&pg=1')
#패턴 : year=yyyy 4글자 / pg=1,2
#곡정보 url
surl <- html_nodes(html, xpath='//*[@id="body-content"]/div[4]/div[1]/table/tbody/tr/td[4]/a')%>%
  html_attr('onclick')
surl <- str_extract(a1, '[0-9]+')
#패턴 : https://www.genie.co.kr/detail/songInfo?xgnm=surl
songinfo <- read_html(paste0('https://www.genie.co.kr/detail/songInfo?xgnm=',surl))
```

2. 데이터 수집 / 정제 / 시각화 - 데이터 수집



지니
2. 노래 정보 url에서 곡명, 가수, 가사 수집

```
#곡 명
html_nodes(html1, xpath='//*[@id="body-content"]/div[2]/div[2]/h2')%>%
  html_text()%>%
  str_trim()
#아티스트
html_nodes(html1, xpath='//*[@id="body-content"]/div[2]/div[2]/ul/li[1]/span[2]/a')%>%
  html_text()
#가사
html_nodes(songinfo, xpath='//*[@id="body-content"]/div[4]/div[1]')%>%
  html_text()%>%
  str_replace_all('[[:space:]]+',' ')
#장르
html_nodes(html1, xpath='//*[@id="body-content"]/div[2]/div[2]/ul/li[3]/span[2]')%>%
  html_text()
```

2. 데이터 수집 / 정제 / 시각화 - 데이터 수집

지니

3. 반복문을 통한 시대별 인기곡 정보 수집 (최종 script)

```
#1970년부터 2018년 1page만
df <- NULL
for (i in 1970:2018){
  html <- read_html(paste0('https://www.genie.co.kr/chart/musicHistory?year=',i,'&category=0&pg=1'))
  surl <- html_nodes(html, xpath='//*[@id="body-content"]/div[4]/div[1]/table/tbody/tr/td[4]/a')%>%
    html_attr('onclick')
  surl <- unlist(str_extract_all(surl, '[0-9]+'))
  for (j in surl){
    songinfo <- read_html(paste0('https://www.genie.co.kr/detail/songInfo?xgnm=',j))
    song <- html_nodes(songinfo, xpath='//*[@id="body-content"]/div[2]/div[2]/h2')%>%
      html_text()%>%
      str_trim()
    artist <- html_nodes(songinfo, xpath='//*[@id="body-content"]/div[2]/div[2]/ul/li[1]/span[2]/a')%>%
      html_text()
    lylics <- html_nodes(songinfo, xpath='//*[@id="body-content"]/div[4]/div[1]')%>%
      html_text()%>%
      str_replace_all('[[:space:]]+',' ')
    genre <- html_nodes(songinfo, xpath='//*[@id="body-content"]/div[2]/div[2]/ul/li[3]/span[2]')%>%
      html_text()

    df <- rbind(df, data.frame(곡명=song,
                               아티스트=artist,
                               가사=lylics,
                               장르=genre,
                               연도=i)
    )
  }
}
```

2. 데이터 수집 / 정제 / 시각화 - 데이터 수집

지니

3. 반복문을 통한 시대별 인기곡 정보 수집 (최종 script) -2

```
#1984년도 부터 2page
for (i in 1984:2018){
  html <- read_html(paste0('https://www.genie.co.kr/chart/musicHistory?year=',i,'&category=0&pg=2'))
  surl <- html_nodes(html, xpath='//*[@id="body-content"]/div[4]/div[1]/table/tbody/tr/td[4]/a')%>%
    html_attr('onclick')
  surl <- unlist(str_extract_all(surl, '[0-9]+'))
  for (j in surl){
    songinfo <- read_html(paste0('https://www.genie.co.kr/detail/songInfo?xgnm=',j))
    song <- html_nodes(songinfo, xpath='//*[@id="body-content"]/div[2]/div[2]/h2')%>%
      html_text()%>%
      str_trim()
    artist <- html_nodes(songinfo, xpath='//*[@id="body-content"]/div[2]/div[2]/ul/li[1]/span[2]/a')%>%
      html_text()
    lylics <- html_nodes(songinfo, xpath='//*[@id="body-content"]/div[4]/div[1]')%>%
      html_text()%>%
      str_replace_all('[[:space:]]+',' ')
    genre <- html_nodes(songinfo, xpath='//*[@id="body-content"]/div[2]/div[2]/ul/li[3]/span[2]')%>%
      html_text()

    df <- rbind(df, data.frame(곡명=song,
                               아티스트=artist,
                               가사=lylics,
                               장르=genre,
                               연도=i)
    )
  }
}
```

2. 데이터 수집 / 정제 / 시각화 - 데이터 정제

1. 시대별 인기 곡 가수 분석

```
library(stringr)
df <- read.csv('c:/data/시대별차트(지니)1.csv', header=T, stringsAsFactors=F, quote="\"'")
df$연도 <- str_replace_all(df$연도,'[0-9]$','0')
df <- unique(df)
df <- df[,c('아티스트','연도')]
total <- table(df$연도) #연도별 곡수
t <- table(df)
t
g_1970_a <- head(sort(t[,1], decreasing=T), 10)
g_1980_a <- head(sort(t[,2], decreasing=T), 10)
g_1990_a <- head(sort(t[,3], decreasing=T), 10)
g_2000_a <- head(sort(t[,4], decreasing=T), 10)
g_2010_a <- head(sort(t[,5], decreasing=T), 10)

names(g_1970_a)
gya <- data.frame(g_1970=paste(names(g_1970_a),g_1970_a),
                  g_1980=paste(names(g_1980_a),g_1980_a),
                  g_1990=paste(names(g_1990_a),g_1990_a),
                  g_2000=paste(names(g_2000_a),g_2000_a),
                  g_2010=paste(names(g_2010_a),g_2010_a),
                  stringsAsFactors=F)
gya
gya <- rbind(gya, total)

#연도별 아티스트 수
df <- unique(df)
table(df)
a_total<- colSums(table(df))
a_total

gya <- rbind(gya, a_total)
gya
```

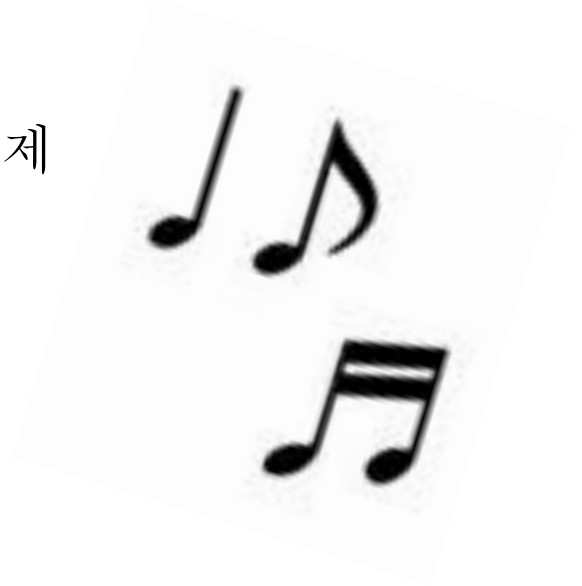
2. 데이터 수집 / 정제 / 시각화 - 시각화

1. 시대별 인기 곡 가사 분석 시각화

- 표 : 열이름 변경 및 데이터 가공

```
dimnames(gya)
colnames(gya) <- paste0(substr(colnames(gya),3,6),'년도')
gya <- cbind(순위=paste0(rownames(gya),'위'), gya)
gya[11,] <- paste0(gya[12,],'명 ',gya[11,],'곡')
gya <- gya[-12,]
gya$순위 <- as.vector(gya$순위)
gya$순위[11] <- 'Total'
gya
gya$`1970년도`[-11] <- paste0(gya$`1970년도`[-11],'곡')
gya$`1980년도`[-11] <- paste0(gya$`1980년도`[-11],'곡')
gya$`1990년도`[-11] <- paste0(gya$`1990년도`[-11],'곡')
gya$`2000년도`[-11] <- paste0(gya$`2000년도`[-11],'곡')
gya$`2010년도`[-11] <- paste0(gya$`2010년도`[-11],'곡')
gya
View(gya)
```

2. 데이터 수집 / 정제 / 시각화 - 데이터 정제



2. 시대별 인기 곡 장르 분석

```
df <- read.csv('c:/data/시대별차트(지니)1.csv', header=T, quote="\"'")
df$대분류 <- str_extract(df$장르,'^.+?/ ')
df$소분류 <- str_extract(df$장르, ' /.+')

df$대분류 <- substr(df$대분류,1,nchar(df$대분류)-3)
df$소분류 <- substr(df$소분류,4,nchar(df$소분류))

View(df)
str(df)
df$곡명 <- as.vector(df$곡명)
df$아티스트 <- as.vector(df$아티스트)
df <- df[,c(2,3,5,6,7)]
df$연도 <- str_replace(df$연도, '.$', '0')
df$연도 <- as.integer(df$연도)
str(df)

df$소분류 <- str_replace(df$소분류, '전체', '미분류')
df[df$대분류 != '가요','소분류'] <- df[df$대분류 != '가요','대분류']
df$소분류 <- str_replace(df$소분류, '동요/태교', '기타')
df$소분류 <- str_replace(df$소분류, '그외장르', '기타')
df$소분류 <- str_replace(df$소분류, '인디', '기타')
df$소분류 <- str_replace(df$소분류, '재즈', '기타')
df$소분류 <- str_replace(df$소분류, '일렉트로니카', '기타')
df$소분류 <- str_replace(df$소분류, 'CCM', '기타')
df$소분류 <- str_replace(df$소분류, '국악', '기타')

View(df)

t <- tapply(df$곡명, list(df$연도,df$소분류),length, default=0)
t
str(t)

prop <- t/rowSums(t)
g <- t(prop)
```

2. 데이터 수집 / 정제 / 시각화 - 시각화

2. 시대별 인기 곡 장르 분석 시각화

- 스택형 막대 그래프 : ggplot2 이용

```
dg <-data.frame(장르=rownames(g), g)
rownames(dg) <-NULL
library(reshape2)
dg <- reshape2::melt(dg,id='장르')
dg
colnames(dg)[2:3] <-c('연도', '비율')
dg$연도 <- substr(dg$연도,2,5)
library(ggplot2)
ggplot(dg, aes(x=연도, y=비율, fill=장르))+
  geom_bar(stat='identity')+
  geom_text(aes(y=비율, label=paste0(round(비율),'%')),
            col='black', size=4,
            position=position_stack(vjust=0.5))
```

2. 데이터 수집 / 정제 / 시각화 - 데이터 정제

3. 시대별 인기 곡 가사 분석 -1

- 1차 정제 작업

```
#지워야할 글자
# 처음 ' 전체선택 프린트 '
# 마지막 ' 듣기 담기 다운로드 더보기 다운로드 선물하기 공유하기 오류신고하기 '
# 가사 앞 제목과 숫자 패턴 예) LUCIFER - 03:54
# 청소년 이용제한 가사 삭제
g2010 <- gsub(' 전체선택 프린트 ', ' ', g2010)
g2010 <- gsub(' 듣기 담기 다운로드 더보기 다운로드 선물하기 공유하기 오류신고하기 ', ' ', g2010)
head(g2010)
library(stringr)
g2010 <- str_replace_all(g2010, '^.+\\:[0-9]{2,}', '')
g2010[grep('청소년 이용제한', g2010)] <- NA
g2010 <- na.omit(g2010)

library(tm)
corp1_g2010 <- VCorpus(VectorSource(g2010))
summary(corp1_g2010)
inspect(corp1_g2010)
corp1_g2010[[1]]$content
corp2_g2010 <- tm_map(corp1_g2010, removeNumbers)
corp2_g2010 <- tm_map(corp2_g2010, removePunctuation)
```

2. 데이터 수집 / 정제 / 시각화 - 데이터 정제

3. 시대별 인기 곡 가사 분석 -2

```
library(KoNLP)
useSejongDic()
library(stringr)

ko.words <- function(doc){
  d <- as.character(doc)
  s <- SimplePos22(d)
  a <- as.vector(na.omit(str_match(s, '([A-z가-힣]+)/NC')[,2]))
  unique(a)
}
stop_w <- c('날','널','때','속','니','니가','이젠','앞','위','우린','곳','안','뒤','내겐','그댈','밖','애',
            '옆','어','그때','m','t','라')
tdm_g2010 <- TermDocumentMatrix(corp2_g2010, control=list(tokenize=ko.words, wordLengths=c(1,Inf),
                                                          stopwords=stop_w))
m_g2010 <- as.matrix(tdm_g2010)
wc_g2010 <- sort(rowSums(m_g2010), decreasing=T)

head(wc_g2010, 100)
aa <- wc_g2010[wc_g2010>1]
```

3. 시대별 인기 곡 가사 분석 시각화

- 워드클라우드 : wordcloud2 이용

```
library(wordcloud2)
wordcloud2(data.frame(names(aa),aa))
```

3. 결론

1. 시대별 인기 곡 가사 분석

- 1970년대

지니	순위	벅스
송창식 11곡	1위	나훈아 16곡
양희은 10곡	2위	송창식 16곡
나훈아 9곡	3위	양희은 15곡
패티김 7곡	4위	남진 13곡
김추자 6곡	5위	이미자 11곡
서유석 6곡	6위	김세환 10곡
혜은이 6곡	7위	박인희 10곡
김세환 5곡	8위	서유석 9곡
남진 5곡	9위	문주란 8곡
이장희 5곡	10위	조영남 8곡
117명 264곡	Total	193명 507곡

- 1980년대

지니	순위	벅스
조용필 32곡	1위	조용필 32곡
이선희 16곡	2위	전영록 16곡
김수철 14곡	3위	이문세 15곡
전영록 14곡	4위	이선희 15곡
이문세 12곡	5위	산울림 12곡
김범룡 11곡	6위	김범룡 11곡
김수희 11곡	7위	김수희 10곡
산울림 11곡	8위	송골매 10곡
해바라기 9곡	9위	정수라 10곡
민해경 8곡	10위	해바라기 10곡
226명 599곡	Total	281명 747곡

- 1990년대

지니	순위	벅스
신승훈 14곡	1위	신승훈 16곡
김종서 12곡	2위	김건모 15곡
이승환 11곡	3위	이승환 14곡
강수지 10곡	4위	서태지와 아이들 12곡
김건모 10곡	5위	김종서 11곡
김원준 10곡	6위	김현철 11곡
015B 9곡	7위	강수지 10곡
김현철 9곡	8위	김광석 10곡
이문세 9곡	9위	이문세 10곡
이승철 9곡	10위	이승철 10곡
300명 761곡	Total	386명 978곡

3. 결론

1. 시대별 인기 곡 가사 분석

- 2000년대

지니	순위	벅스
이수영 16곡	1위	SG워너비 19곡
엠씨더맥스 (M.C THE MAX) 13곡	2위	이수영 14곡
지오디 (god) 13곡	3위	MC 몽 13곡
KYT (코요태) 11곡	4위	성시경 12곡
SG워너비 11곡	5위	지오디(god) 12곡
성시경 11곡	6위	코요태 12곡
이승기 11곡	7위	김종국 11곡
신화 10곡	8위	엠씨더맥스(M.C THE MAX) 11곡
씨야 (SeeYa) 10곡	9위	쿨(COOL) 11곡
BIGBANG 9곡	10위	휘성(Realslow) 11곡
351명 1000곡	Total	355명 1000곡

3. 결론

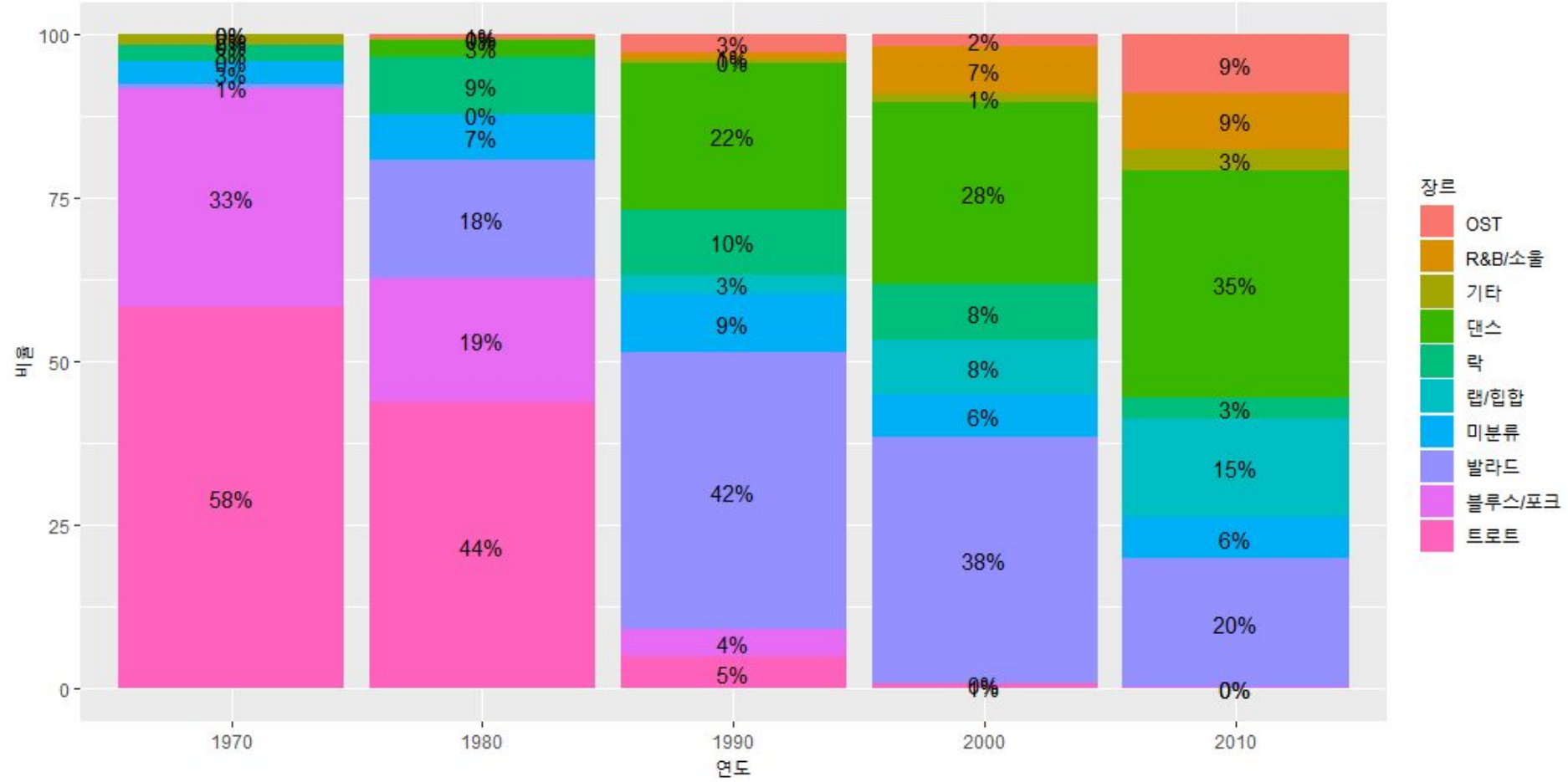
1. 시대별 인기 곡 가사 분석

- 2010년대

지니	순위	벅스
아이유 (IU) 24곡	1위	아이유(IU) 21곡
BIGBANG 16곡	2위	BIGBANG 19곡
다비치 15곡	3위	다비치 18곡
볼빨간사춘기 14곡	4위	버스커 버스커(Busker Busker) 15곡
2NE1 13곡	5위	2NE1 13곡
악동뮤지션 13곡	6위	악동뮤지션(AKMU) 13곡
비스트 (Beast) 11곡	7위	볼빨간사춘기 12곡
씨스타 (Sistar) 11곡	8위	비스트(Beast) 12곡
TWICE (트와이스) 10곡	9위	씨스타(Sistar) 12곡
소녀시대 (GIRLS' GENERATION) 10곡	10위	케이윌 10곡
351명 900곡	Total	323명 867곡

3. 결론

2. 시대별 인기 곡 장르 분석



3. 결론

3. 시대별 인기 곡 가사 분석 (지니)

1970년대

1980년대

1990년대

2000년대

2010년대

3. 결론

3. 시대별 인기 곡 가사 분석 (벅스)

1970년대

1980년대

1990년대

2000년대

2010년대

※출처

1. 벅스

   - https://music.bugs.co.kr/

2. 지니뮤직

   - https://www.genie.co.kr/

THANK YOU