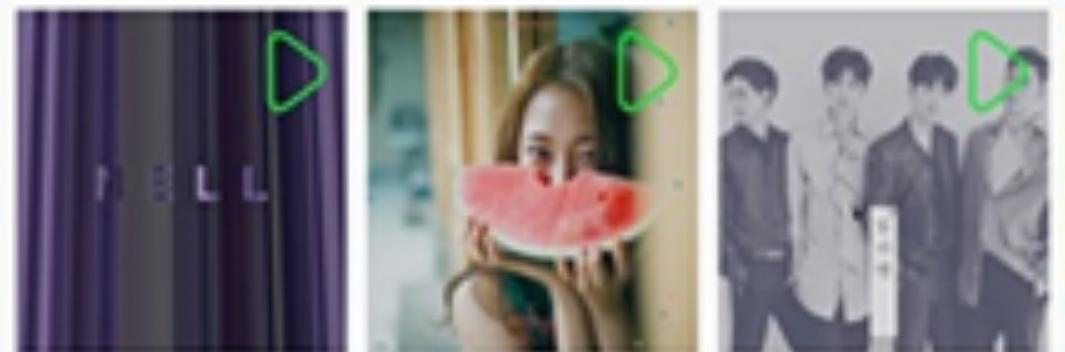


# 멜론 차트 역주행 분석 및 음원 추천 알고리즘

Key word: EDA, Statistics, Big Data, NLP



부서진  
넬(NELL)

맴돌아  
바닐라 어쿠스틱

NU'EST W '있...  
뉴이스트 W

# 목차

## Part 1: 멜론 음원 차트와 빅데이터

- 1.1 프로젝트의 목적
- 1.2 프로젝트의 흐름
- 1.3 데이터 수집 및 크롤링

## Part 3: 자연어처리를 통한 추천 시스템

- 3.1 모델링을 위한 EDA
- 3.2 클러스터링을 통한 음악 추천
- 3.3 Word2Vec을 통한 음악 추천

## Part 2: 멜론 음원 차트와 역주행

- 2.1 역주행의 정의 및 EDA
- 2.2 음원의 역주행 분포 분류
- 2.3 강수량과 역주행
- 2.4 계절성과 역주행

## Part 4: 프로젝트 결론

- 6.1 Part 2 프로젝트 분석 결론
- 6.2 Part 3 프로젝트 분석 결론

## Part 5: 별첨 : 코드 분석 및 참고자료

## Part 1: 멜론 음원 차트와 빅데이터

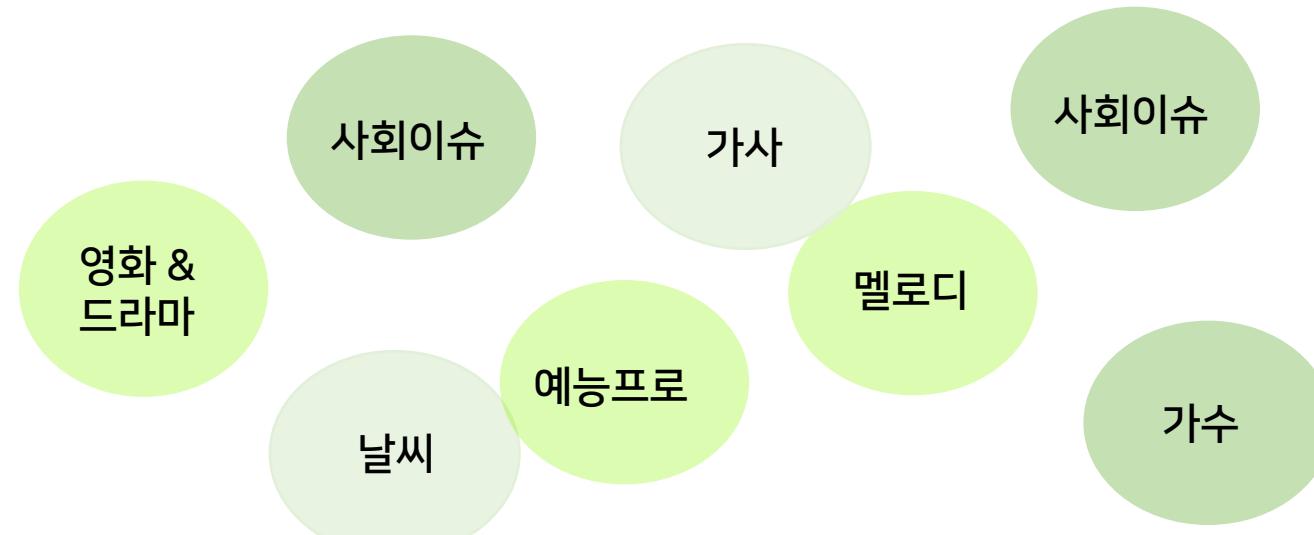
- 1.1 프로젝트의 목적
- 1.2 프로젝트의 흐름
- 1.3 데이터 수집 및 크롤링

## 1.1 프로젝트의 목적

멜론 차트 > 2018.12.05 20:00 기준

| 실시간    | POP                        | ☆ 아티스트 |
|--------|----------------------------|--------|
| 1 - 0  | 아낙네<br>MINO (송민호)          |        |
| 2 - 0  | SOLO<br>제니 (JENNIE)        |        |
| 3 - 0  | YES or YES<br>TWICE (트와이스) |        |
| 4 - 0  | 너를 만나<br>풀칠                |        |
| 5 - 0  | 봄바람<br>Wanna One           |        |
| 6 - 0  | Tempo<br>EXO               |        |
| 7 - 0  | 가을 타나 봐<br>바이브             |        |
| 8 - 0  | 아름답고도 아프구나 비루비<br>비루비      |        |
| 9 - 0  | 삐삐<br>아이유                  |        |
| 10 - 0 | 내 생에 아름다운 케이윌              |        |

“ 음원차트 는 과연 밑을 만한가? ”



- ▶ 과연 멜론 차트 순위가 순수하게 매력 있는 곡, 누구나 좋아할 만한 곡을 나타내는지 알아보자 함.
- ▶ 그 중 가장 특이 케이스인 **역주행** 곡들에 주목하기 시작.

# Part 1 : 멜론 음원 차트와 빅데이터

## 1.1 프로젝트의 목적

멜론 차트 역주행 분석 및 음원 추천 알고리즘

**STEP1:** 역주행 곡 패턴 분석 & 흥행 요인 파악



|F 곡의 흥행이 곡 자체가 아닌 **사회적 요인**으로 인한 것이라면?

**STEP2:** 사회적 요인을 배제하고 곡 가사 자체를  
분석 & 곡 추천 알고리즘 고안



# Part 1 : 멜론 음원 차트와 빅데이터

## 1.3 데이터 수집 및 크롤링

멜론 차트 역주행 분석 및 음원 추천 알고리즘

셀레니움을 이용하여 Melon 웹사이트에서 10년치 주간 top 100 차트 크롤링



```
title = soup.find(attrs={"class": "song_name"}).text.replace('곡명', '')
if '19금' in title:
    title = title.replace('19금', '')
title = re.sub('^\s*|\s+$', '', title)
artist = soup.find(attrs={"class": "artist_name"}).text
album = soup.select('#downloadfrm > div > div > div.entry > div.meta > dl > dd')[0].text
genre = soup.select('#downloadfrm > div > div > div.entry > div.meta > dl > dd')[2].text
release = soup.select('#downloadfrm > div > div > div.entry > div.meta > dl > dd')[1].text

try:
    lyric = soup.find('div', attrs={"class": "lyric"}).get_text()
    lyric = re.sub('<[^>]*>|\s|\n|\r', ' ', lyric)
    lyric = re.sub('^\s*|\s+$', '', lyric)
except:
    print("week_xpath not found")
    continue
```

▲ 멜론 웹사이트 및 멜론 차트 크롤링 파이썬 코드 일부

2011년 1월부터 2018년 10월까지 407주에 대한 총 40797개의 케이스 수집

| artist       | genre         | rank | week        | year |
|--------------|---------------|------|-------------|------|
| 아이유          | Dance         | 1    | 01.02~01.08 | 2011 |
| GD&TOP;      | Rap / Hip-hop | 2    | 01.02~01.08 | 2011 |
| GD&TOP;      | Rap / Hip-hop | 3    | 01.02~01.08 | 2011 |
| 씨스타          | Dance         | 4    | 01.02~01.08 | 2011 |
| 박효신          | Ballad        | 5    | 01.02~01.08 | 2011 |
| 제아           | Ballad        | 6    | 01.02~01.08 | 2011 |
| JOO          | Ballad        | 7    | 01.02~01.08 | 2011 |
| GD&TOP;      | Rap / Hip-hop | 8    | 01.02~01.08 | 2011 |
| 동방신기 (TVXQ!) | Dance         | 9    | 01.02~01.08 | 2011 |
| 티아라          | Dance         | 10   | 01.02~01.08 | 2011 |

▲ 완성된 데이터 프레임의 일부  
: 이외에도 album, id, lyric, release, month 등의 변수가 있다. 특히 id의 경우 unique한 값으로 이후 데이터 프레임의 join 혹은 merge에서 key로 중요한 역할을 함.

## Part 2: 멜론 음원 차트와 역주행

2.1 역주행의 정의 및 EDA

2.2 음원의 역주행 분포 분류

2.3 강수량과 역주행

2.4 계절성과 역주행

## Part 2 : 멜론 음원 차트와 역주행

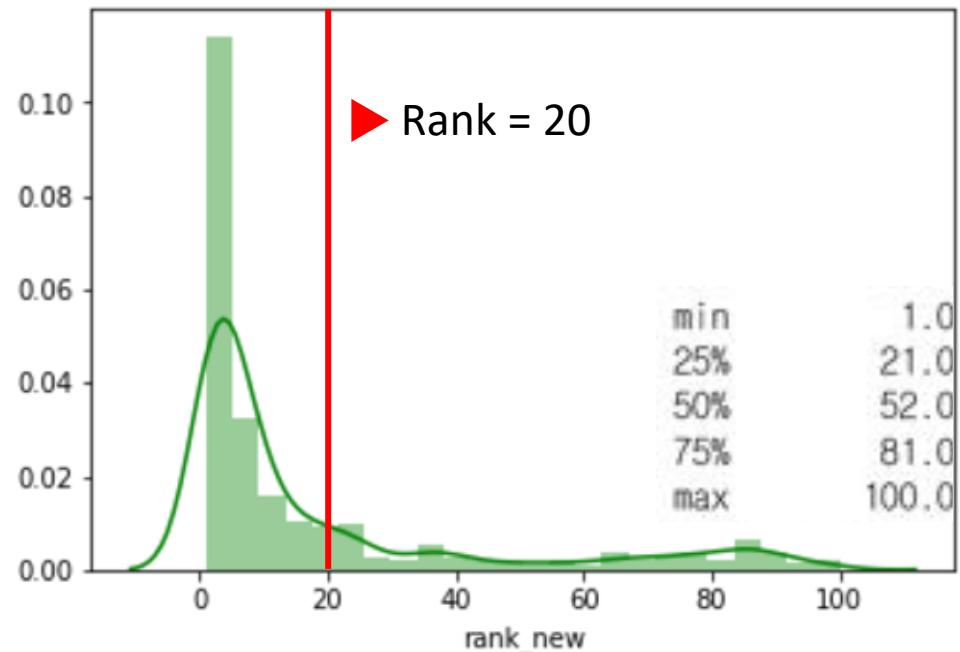
### 2.1 역주행 변수 정의 및 EDA

멜론 차트 역주행 분석 및 음원 추천 알고리즘

차트에 등장한 6037개 음원들 중에서 세 가지 조건으로 역주행 곡 34개 식별

$N = 4826$

조건 1. 진입순위가 20위보다 낮을 것



▲ 진입 순위 분포는 skewed distribution을 보이며, 대략 75% 정도가 20위보다 높은 것을 확인할 수 있음

$N = 134$

조건 2. 최고순위가 5위보다 높을 것

▼ 멜론 모바일 홈 화면에는 실시간 차트 Top 5가 노출됨.



[12.4(목) 21:00]

- 1위 아낙네 (MINO)
- 2위 SOLO (제니)
- 3위 YES or YES (트와이스)
- 4위 봄바람 (워너원)
- 5위 Tempo (EXO)

## Part 2 : 멜론 음원 차트와 역주행

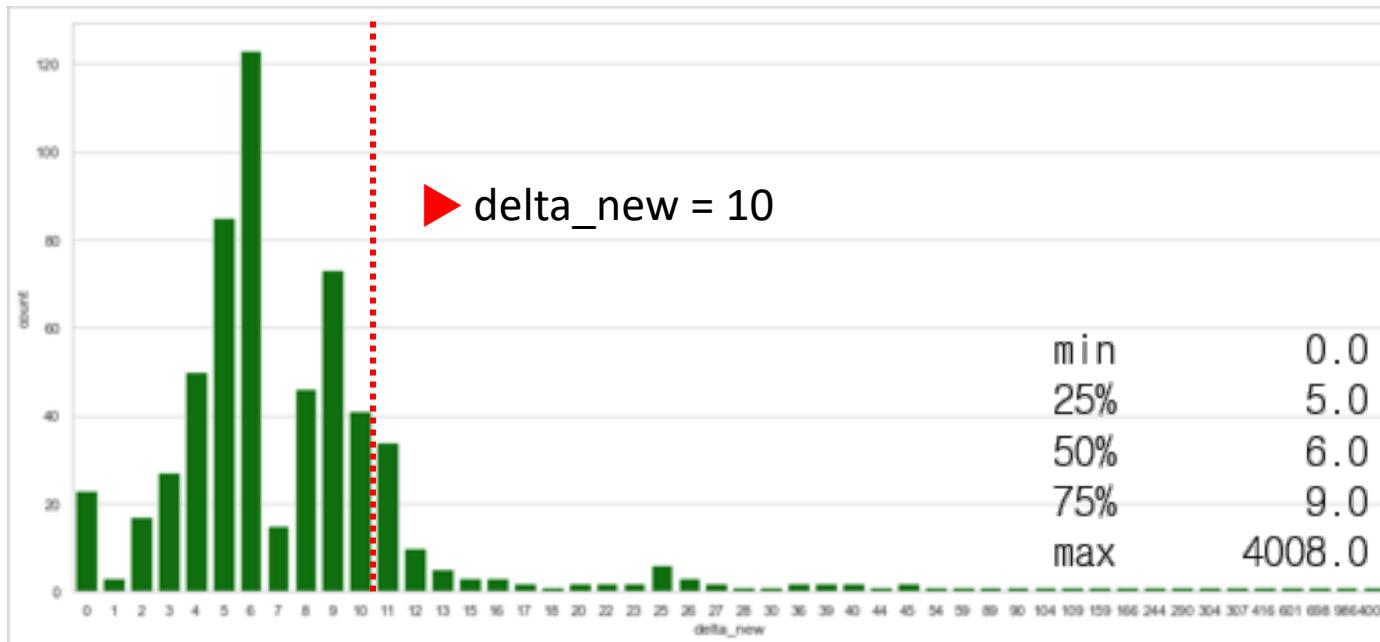
### 2.1 역주행 변수 정의 및 EDA

멜론 차트 역주행 분석 및 음원 추천 알고리즘

차트에 등장한 6037개 음원들 중에서 세 가지 조건으로 역주행 곡 34개 식별

N= 34

조건 3. 발매일로부터 10일 이후에 5위 안으로 진입했을 것



▲ 5위 이내로 진입하기까지 걸린 시간(일) 분포는 긴 꼬리 형태를 보임

#### 역주행곡 리스트

- |                                   |  |
|-----------------------------------|--|
| 1. Black & White                  | 18. OOH-AHH하게                                |
| 2. 톡톡 (Tok Tok)<br>(Feat. SOYA)   | 19. 이 소설의 끝을 다시 써보려 해                        |
| 3. 반짝반짝                           | 20. 우주를 줄게                                   |
| 4. 거울아 거울아                        | 21. Stay With Me                             |
| 5. To Me (내게로..)                  | 22. BLUE MOON<br>(Prod. GroovyRoom)          |
| 6. 아메리카노                          | 23. 처음부터 너와 나                                |
| 7. 애상                             | 24. 매일 듣는 노래<br>(A Daily Song)               |
| 8. 지독하게                           | 25. 좋니                                       |
| 9. Heaven                         | 26. Lonely (Feat. 태연)                        |
| 10. 여수 밤바다                        | 27. 비행운                                      |
| 11. 빠빠빠                           | 28. 그날처럼                                     |
| 12. 신촌을 못가                        | 29. 뽐뽐                                       |
| 13. 위아래                           | 30. 지나오다                                     |
| 14. 이럴거면 그러지말지<br>(Feat. Young K) | 31. 밤 (Time for the moon night)              |
| 15. 와리가리                          | 32. Way Back Home                            |
| 16. 위잉위잉                          | 33. 모든 날, 모든 순간<br>(Every day, Every Moment) |
| 17. 거북선 (Feat. 팔로알토)              | 34. 가을 타나 봐                                  |

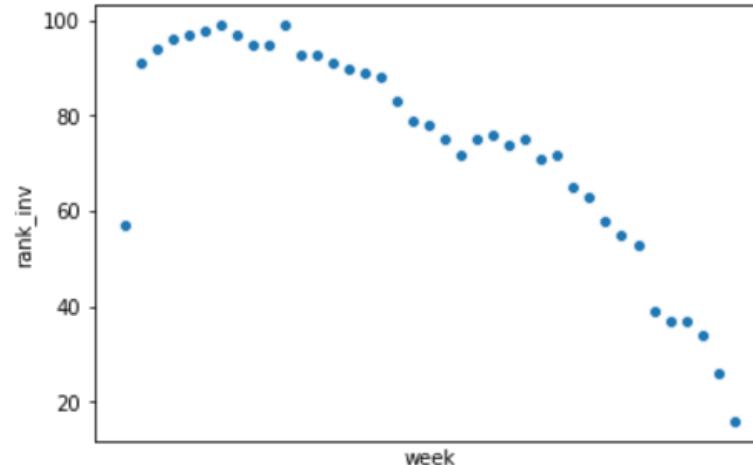
## Part 2 : 멜론 음원 차트와 역주행

### 2.1 역주행 변수 정의 및 EDA

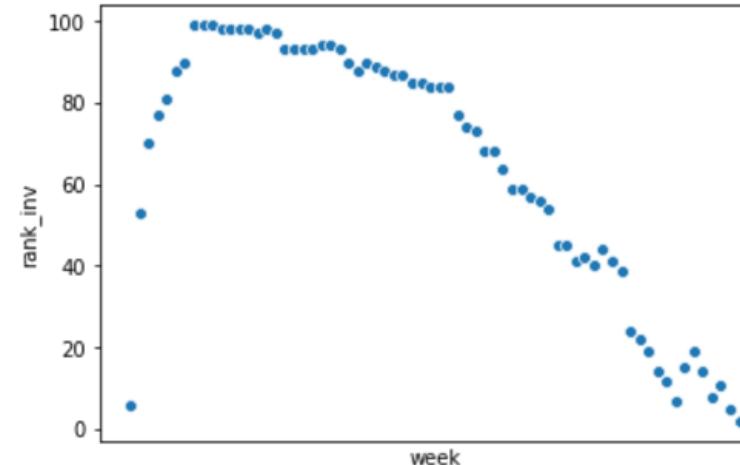
멜론 차트 역주행 분석 및 음원 추천 알고리즘

차트에 등장한 6037개 음원들 중에서 세 가지 조건으로 역주행 곡 34개 식별

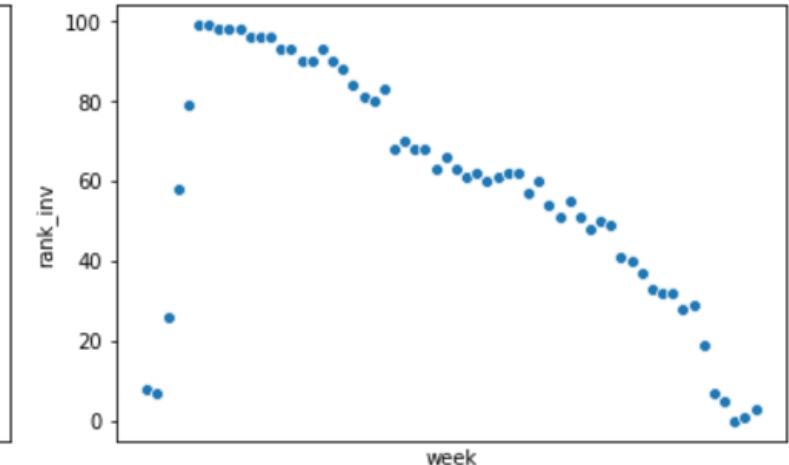
위아래 (EXID)



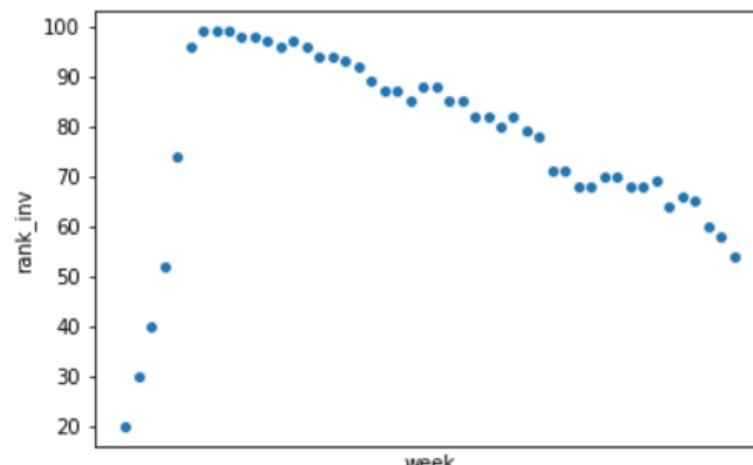
좋니 (윤종신)



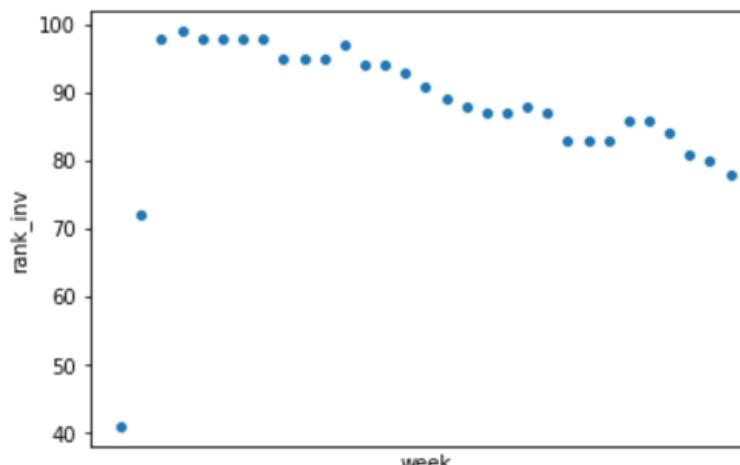
이 소설의 끝을 다시 써보려 해 (한동근)



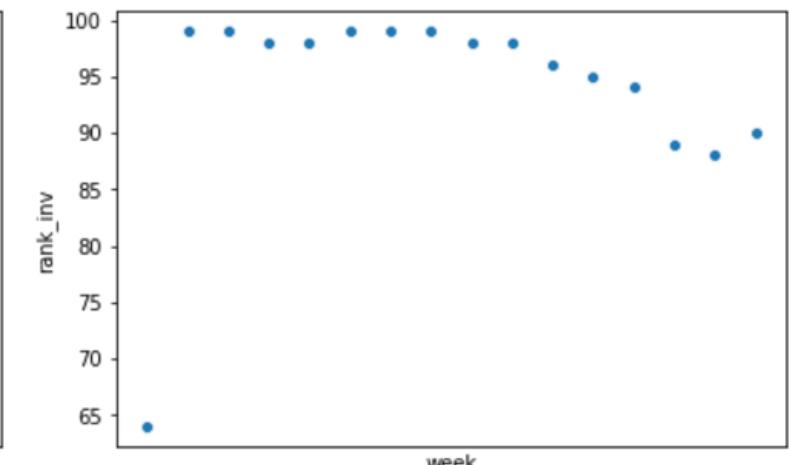
그날처럼 (장덕철)



지나오다 (닐로)



Way Back Home(숀)

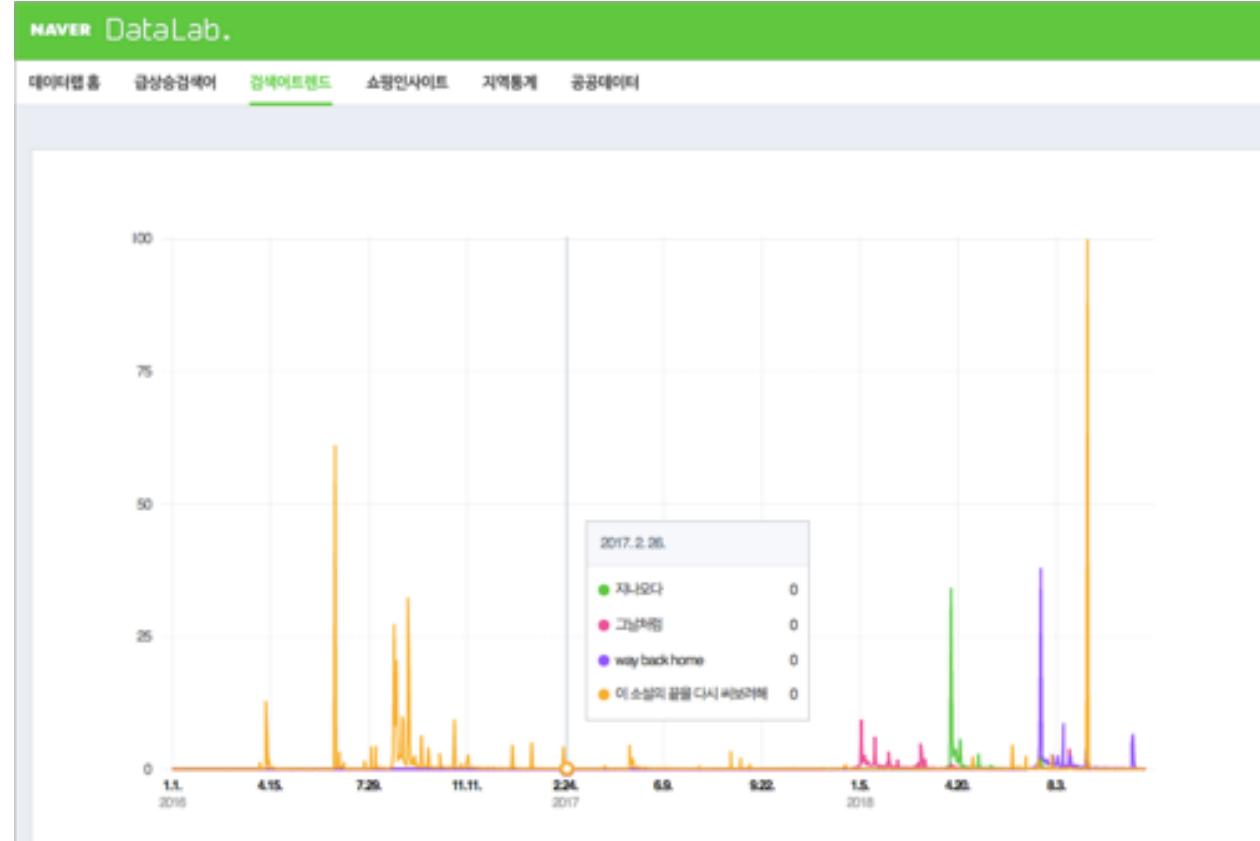


# Part 2 : 멜론 음원 차트와 역주행

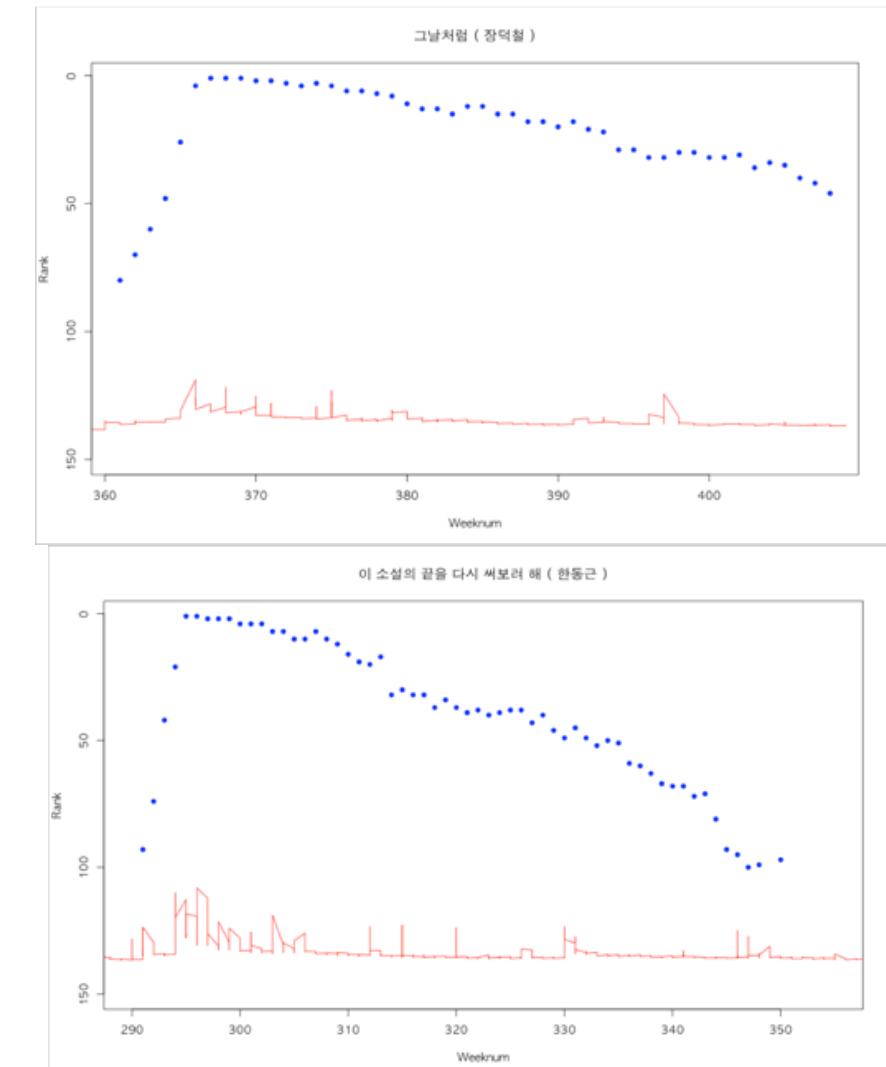
## 2.1 역주행 변수 정의 및 EDA

멜론 차트 역주행 분석 및 음원 추천 알고리즘

네이버 데이터 랩을 통해 역주행 곡에 대한 검색 트랜드 분석 (2016년 이후)



그날처럼 (위), 이 소설의 끝을 다시 써보려 해 (아래) ▶  
: 검색트렌드 상승(빨간색)에 따라 음원의 랭킹(파란색)이 상승



## Part 2 : 멜론 음원 차트와 역주행

### 2.2 음원의 역주행 분포 분류

멜론 차트 역주행 분석 및 음원 추천 알고리즘

#### 5가지로 음원의 역주행 분포 분류

대부분의 음원은 높은 순위에서 시작하여 시간이 흐름에 따라 순위가 낮아짐.

선형 회귀에 음원의 순위를 적합한 후 RMSE가 큰 순서로 산점도를 그려 특이 패턴을 5가지 범주로 분류

1.  
대표적인  
역주행 곡

2.  
아티스트의  
사망에 의한  
역주행

3.  
계절성에  
따른 역주행

4.  
컴백 후  
팬덤에 의한  
역주행

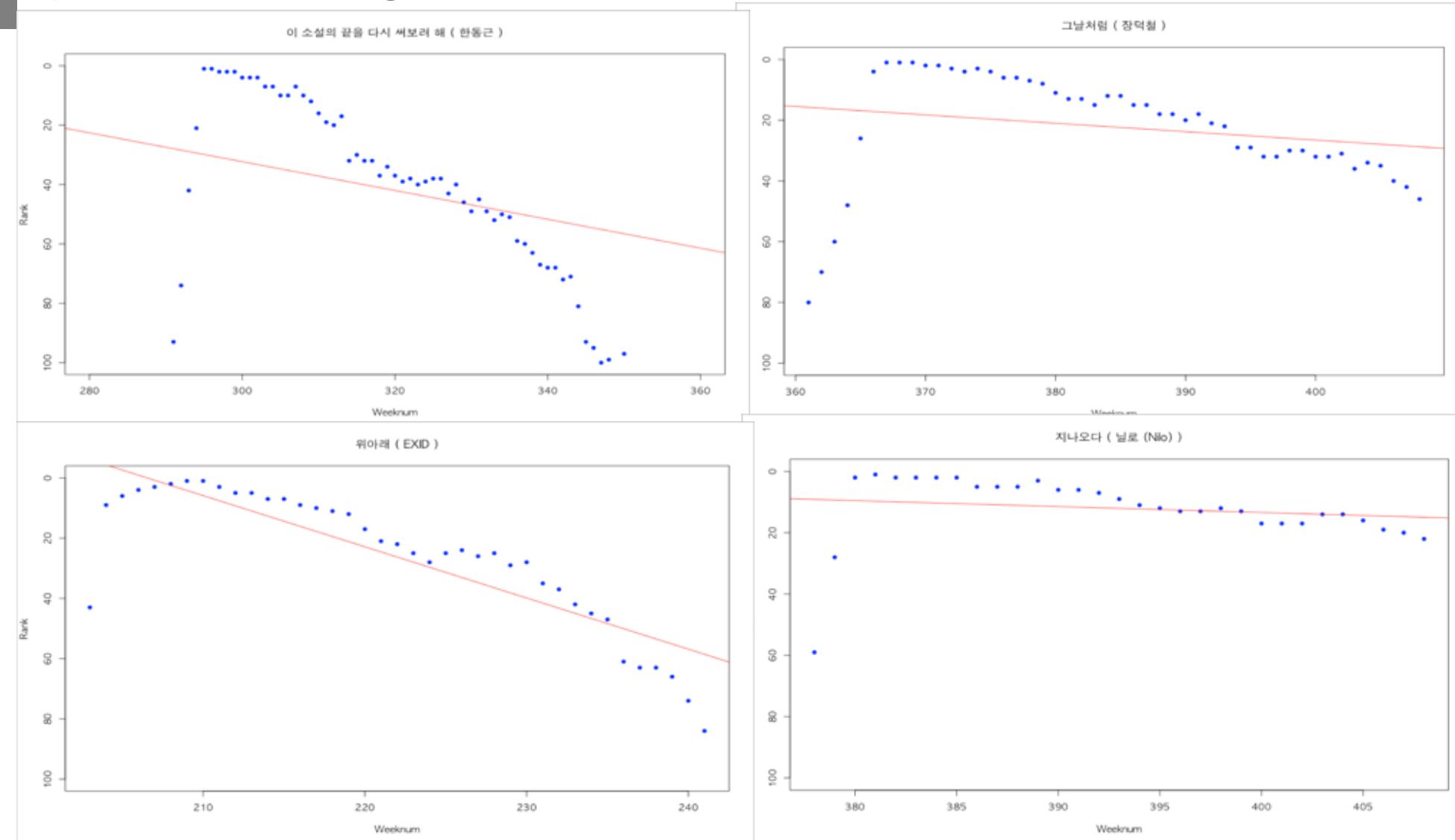
5.  
방송  
출연으로  
인한 역주행

## Part 2 : 멜론 음원 차트와 역주행

### 2.2 음원의 역주행 분포 분류

멜론 차트 역주행 분석 및 음원 추천 알고리즘

#### 특이패턴 1: 대표적 역주행곡들



◀ 이 소설의 끝을  
다시 써보려 해  
(좌측상단),  
그날처럼(우측상단),  
위아래(좌측하단),  
지나오다(우측하단)

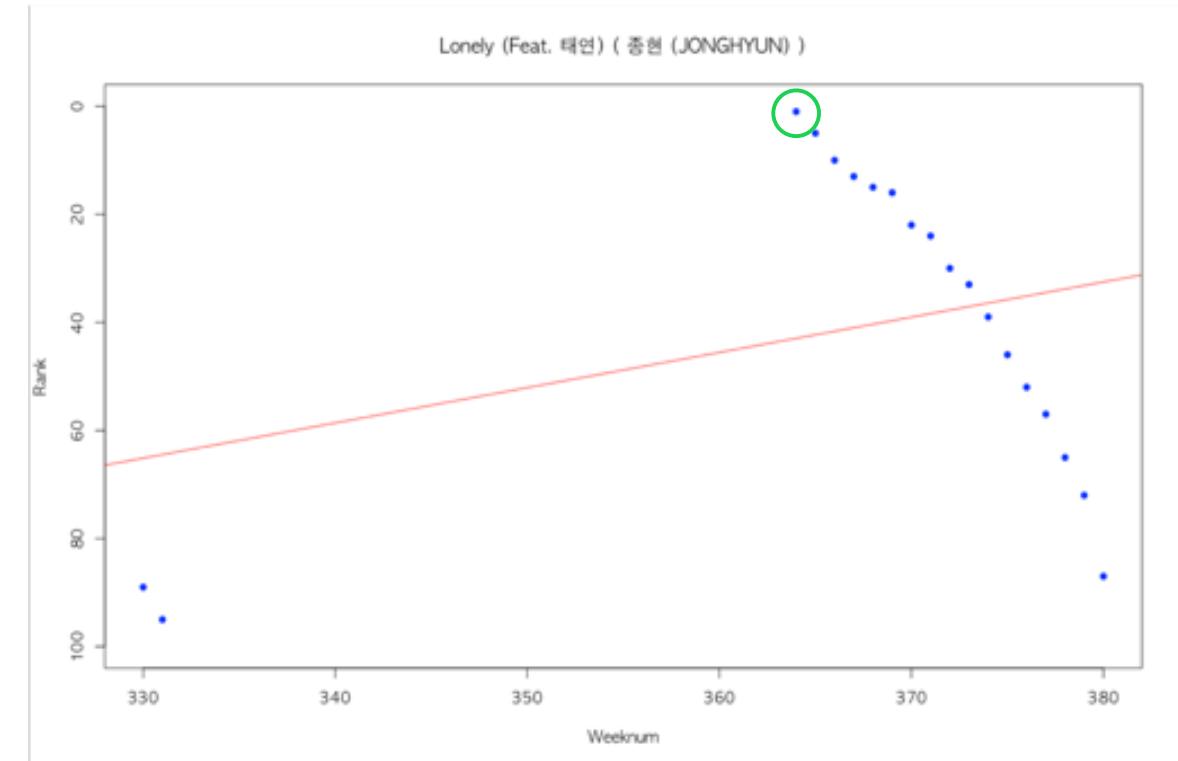
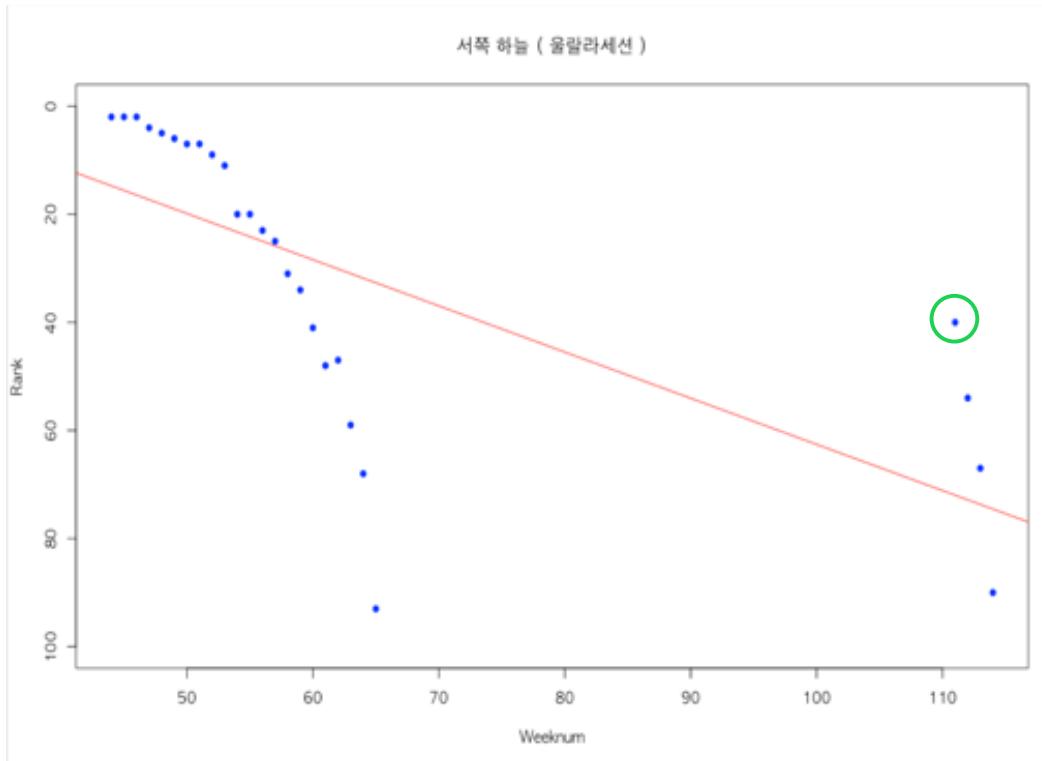
: 빠른 속도로 순위가  
올라간 후 점차  
떨어지는 형태

## Part 2 : 멜론 음원 차트와 역주행

### 2.2 음원의 역주행 분포 분류

멜론 차트 역주행 분석 및 음원 추천 알고리즘

#### 특이패턴 2 : 가수의 사망에 의한 역주행



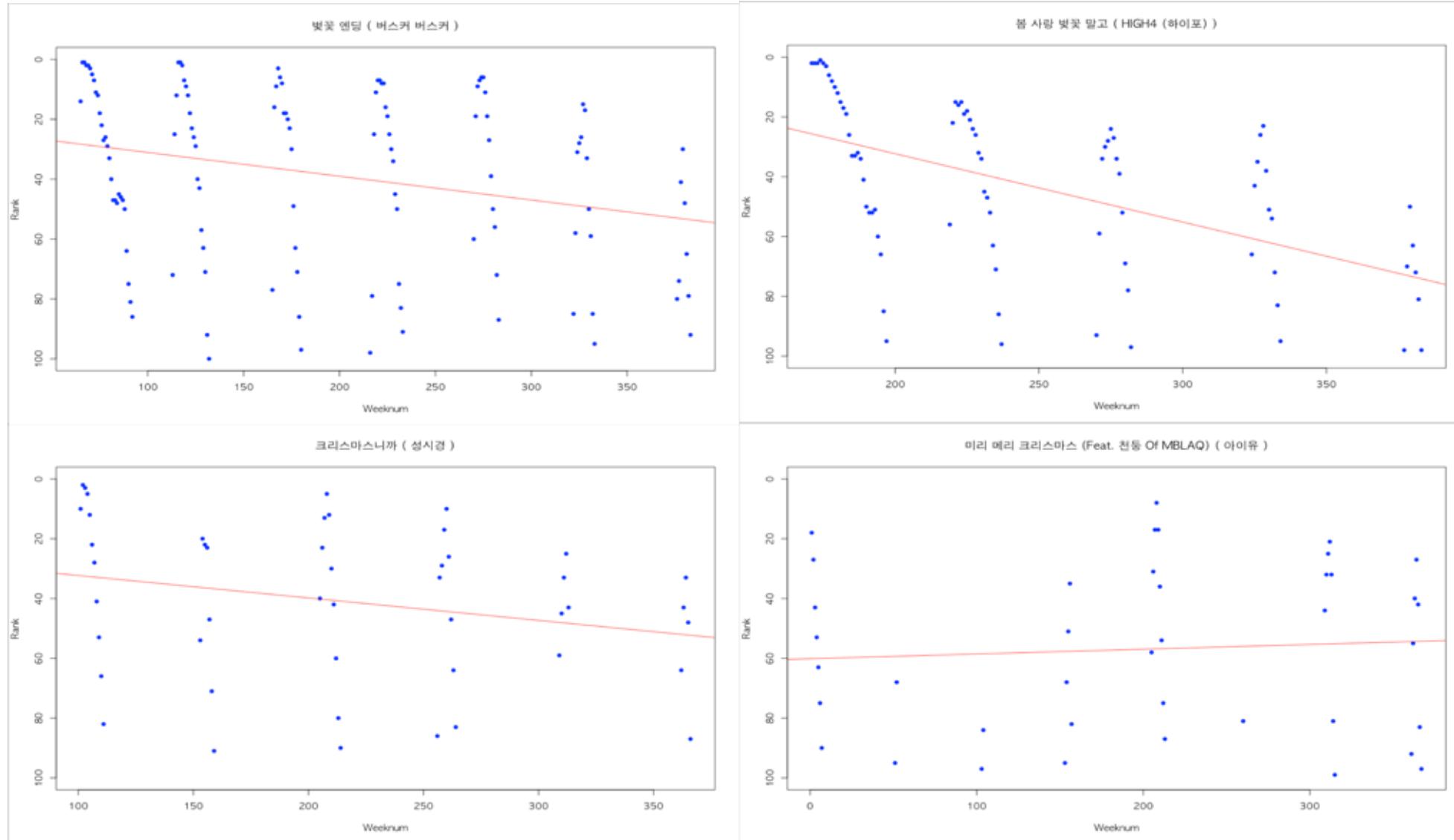
▲ 서쪽 하늘 (좌), Lonely (우) : 가수가 사망한 시점에서 갑자기 높은 순위에 등장

## Part 2 : 멜론 음원 차트와 역주행

### 2.2 음원의 역주행 분포 분류

멜론 차트 역주행 분석 및 음원 추천 알고리즘

#### 특이 패턴 3 : 계절에 의한 역주행



◀ 벚꽃 엔딩  
(좌상단), 봄 사랑  
벚꽃 말고 (우상단),  
크리스마스니까  
(좌하단), 미리 메리  
크리스마스  
(우하단)

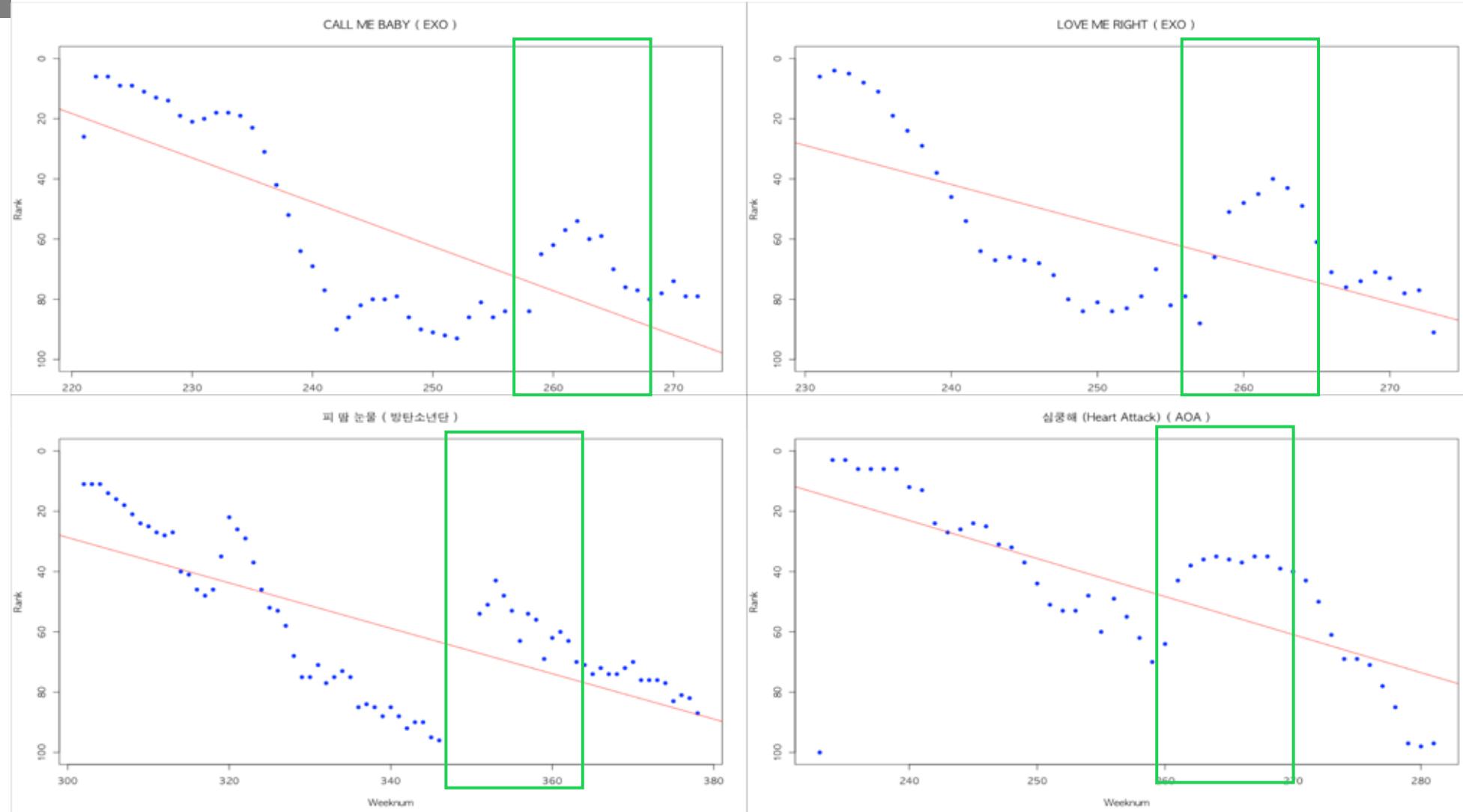
: 계절에 따라 차트에  
다시 등장하는 패턴

## Part 2 : 멜론 음원 차트와 역주행

### 2.2 음원의 역주행 분포 분류

멜론 차트 역주행 분석 및 음원 추천 알고리즘

#### 특이 패턴 4 : 컴백 후 팬덤에 의한 역주행



◀ CALL ME BABY  
(좌상단), LOVE ME  
RIGHT (우상단), 피  
땀 눈물 (좌하단),  
심쿵해 (우하단)

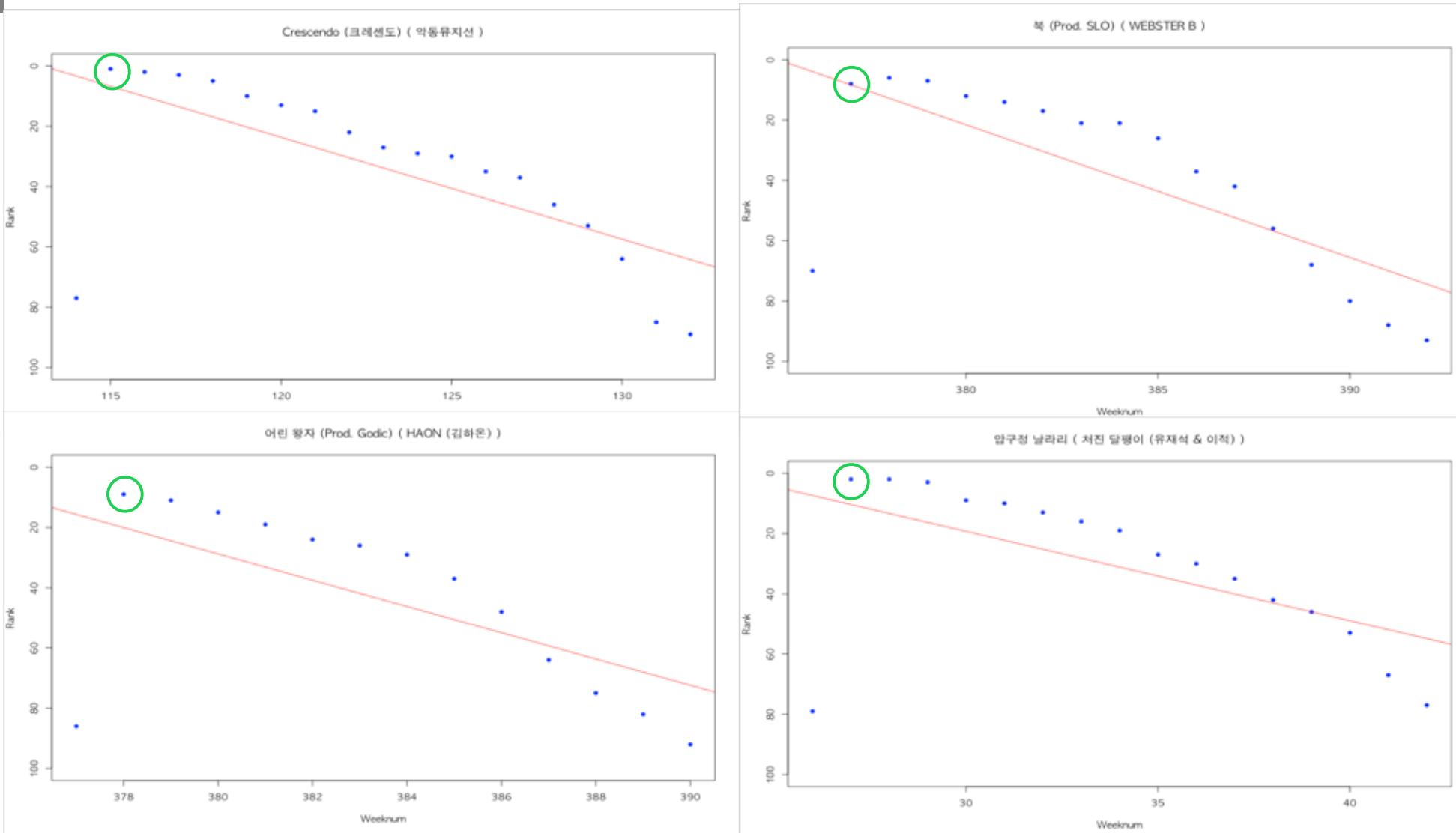
: 팬덤이 쎈 아이돌  
곡의 경우 새 앨범  
발매 시점에서 기존의  
곡들의 순위가 급상승

## Part 2 : 멜론 음원 차트와 역주행

### 2.2 음원의 역주행 분포 분류

멜론 차트 역주행 분석 및 음원 추천 알고리즘

#### 특이 패턴 5 : 방송을 통해 인기를 얻게 된 곡들



◀ Crecendo (좌상단),  
북 (우상단), 어린왕자  
(좌하단), 암구정 날라리  
(우하단)

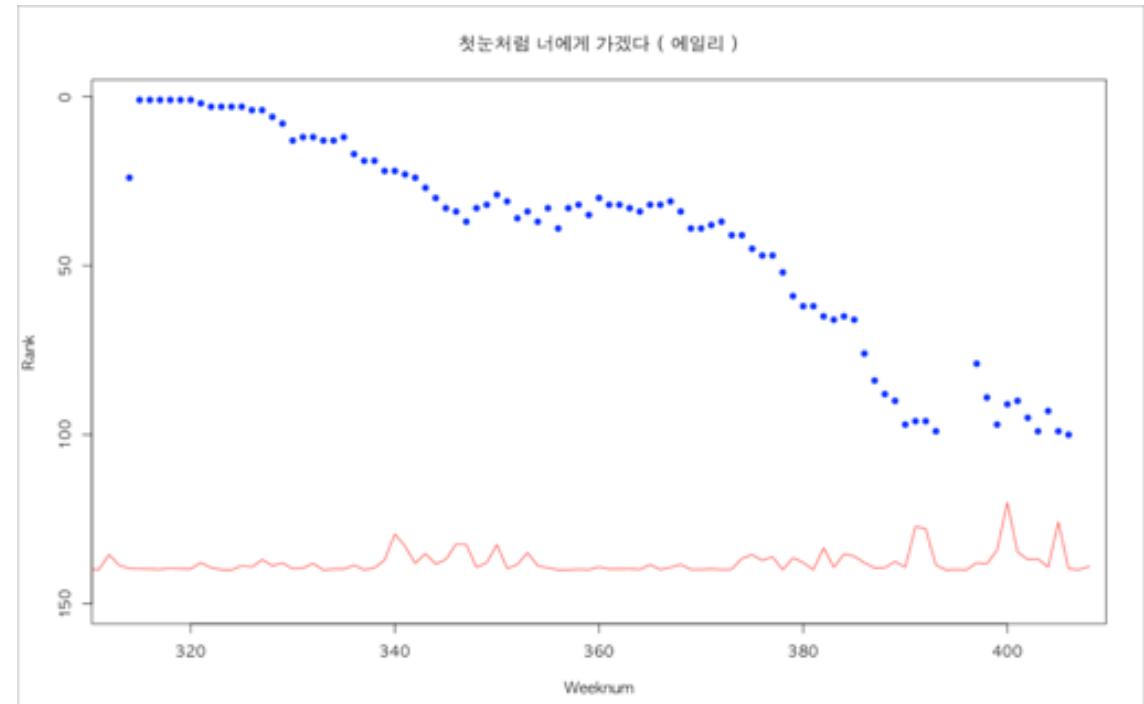
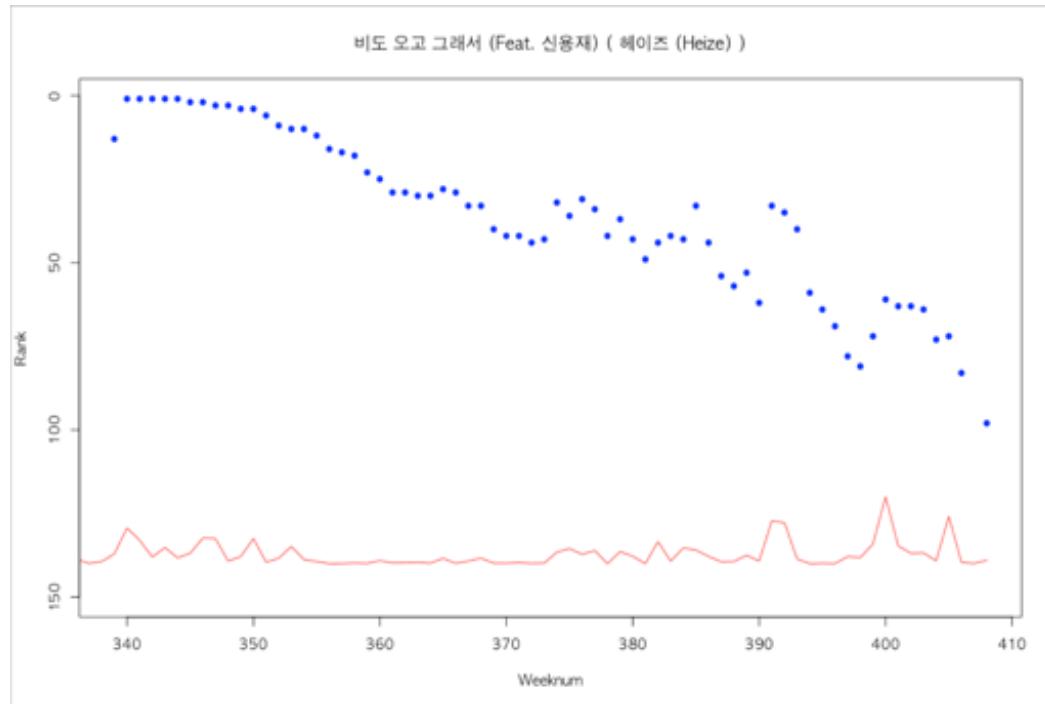
: 처음 낮은 순위로  
차트에 진입하여 일주일  
만에 순위가 급상승하는  
패턴

## Part 2 : 멜론 음원 차트와 역주행

### 2.3 강수량과 역주행

멜론 차트 역주행 분석 및 음원 추천 알고리즘

가설: 비, 눈 등의 키워드를 가진 곡들은 강수량이 많을 때 차트 순위가 상승할 것이다.



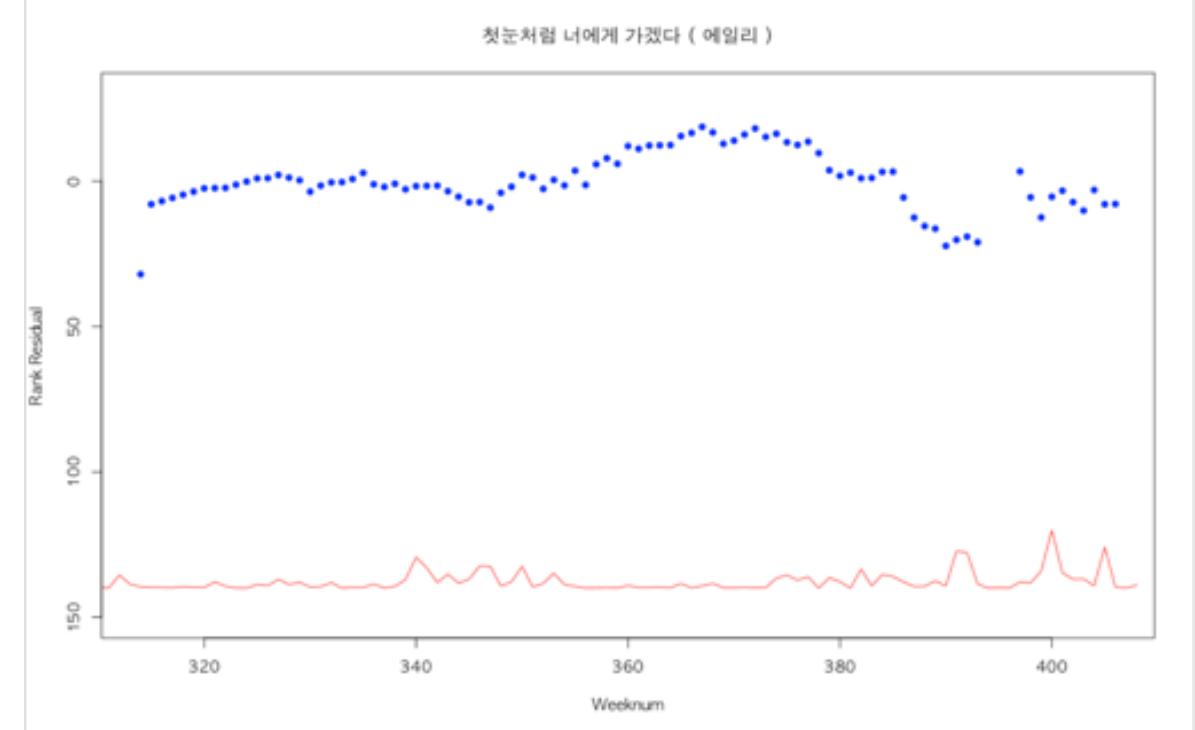
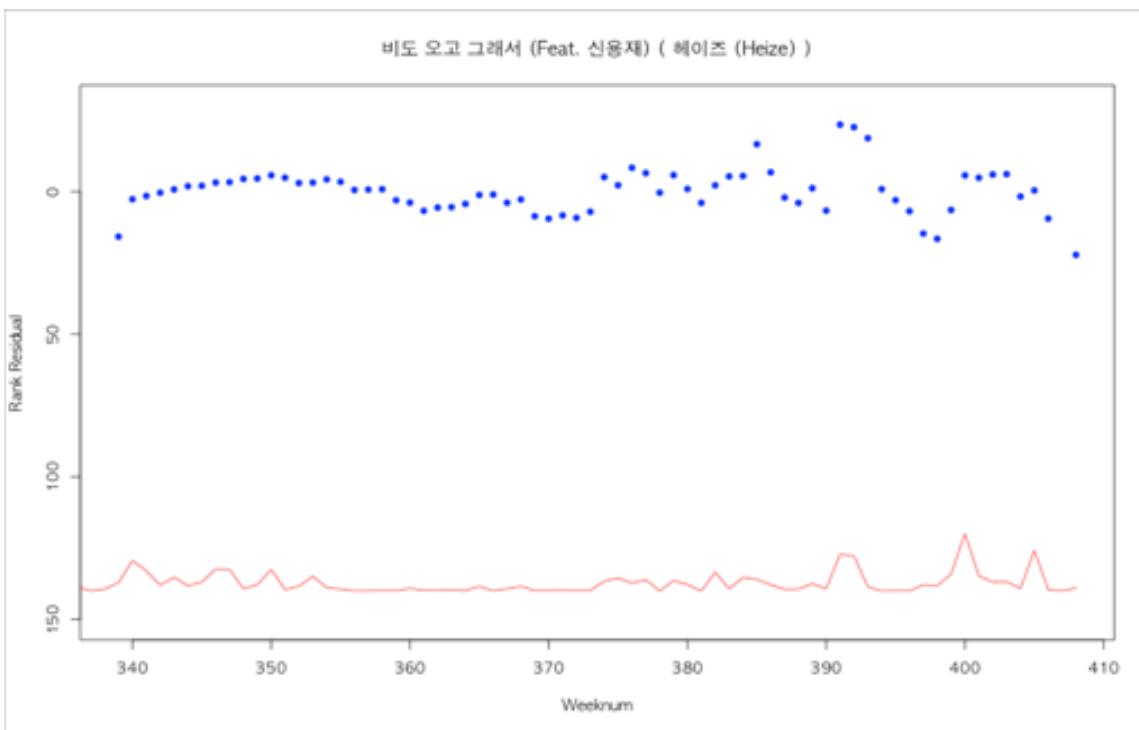
▲ 비도 오고 그래서 (좌), 첫눈처럼 너에게 가겠다 (우)  
: 해당 곡의 차트 순위(파란색), 해당 시점의 전국 광역시와 제주도의 강수량 합 (빨간색)

## Part 2 : 멜론 음원 차트와 역주행

### 2.3 강수량과 역주행

멜론 차트 역주행 분석 및 음원 추천 알고리즘

가설: 비, 눈 등의 키워드를 가진 곡들은 강수량이 많을 때 차트 순위가 상승할 것이다.



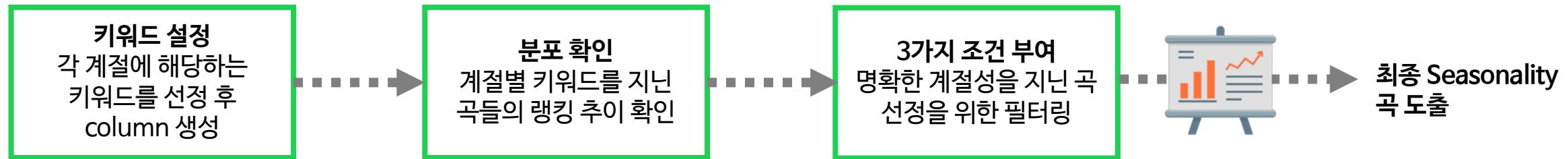
▲ 비도 오고 그래서 (좌), 첫눈처럼 너에게 가졌다 (우)  
: 선형 회귀 적합 후 잔차(파란색), 해당 시점의 전국 광역시와 제주도의 강수량 합 (빨간색)

## Part 2 : 멜론 음원 차트와 역주행

멜론 차트 역주행 분석 및 음원 추천 알고리즘

### 2.4 계절성과 역주행

가설: 계절성을 가지고 있는 곡들은 매년 해당 계절에 랭킹상승을 보일 것이다.



#### 1) 키워드 설정 및 라벨링

| title               | week        | year | realmonth | releaseyear | jointdate | dateorder | season | keyword | realseason |
|---------------------|-------------|------|-----------|-------------|-----------|-----------|--------|---------|------------|
| 꽃                   | 05.19~05.25 | 2014 | 05        | 2014        | 20140519  | 176.0     | 봄      | 꽃       | 봄          |
| 꽃 (Feat. 김태우)       | 10.31~11.06 | 2016 | 10        | 2016        | 20161031  | 305.0     | 봄      | 꽃       | 겨울         |
| 꽃 (Feat. 타블로 애예박하이) | 03.02~03.08 | 2015 | 03        | 2015        | 20150302  | 217.0     | 봄      | 꽃       | 봄          |
| 꽃길                  | 03.12~03.18 | 2018 | 03        | 2018        | 20180312  | 376.0     | 봄      | 꽃       | 봄          |
| 꽃, 바람 그리고 너         | 08.29~09.04 | 2016 | 08        | 2016        | 20160829  | 296.0     | 봄      | 꽃       | 여름         |

|            |                   |
|------------|-------------------|
| 봄, 꽃, 봄꽃   | 여름, 썸머, Summer    |
| 가을, 낙엽, 단풍 | 첫눈, 눈, 크리스마스, 눈사람 |

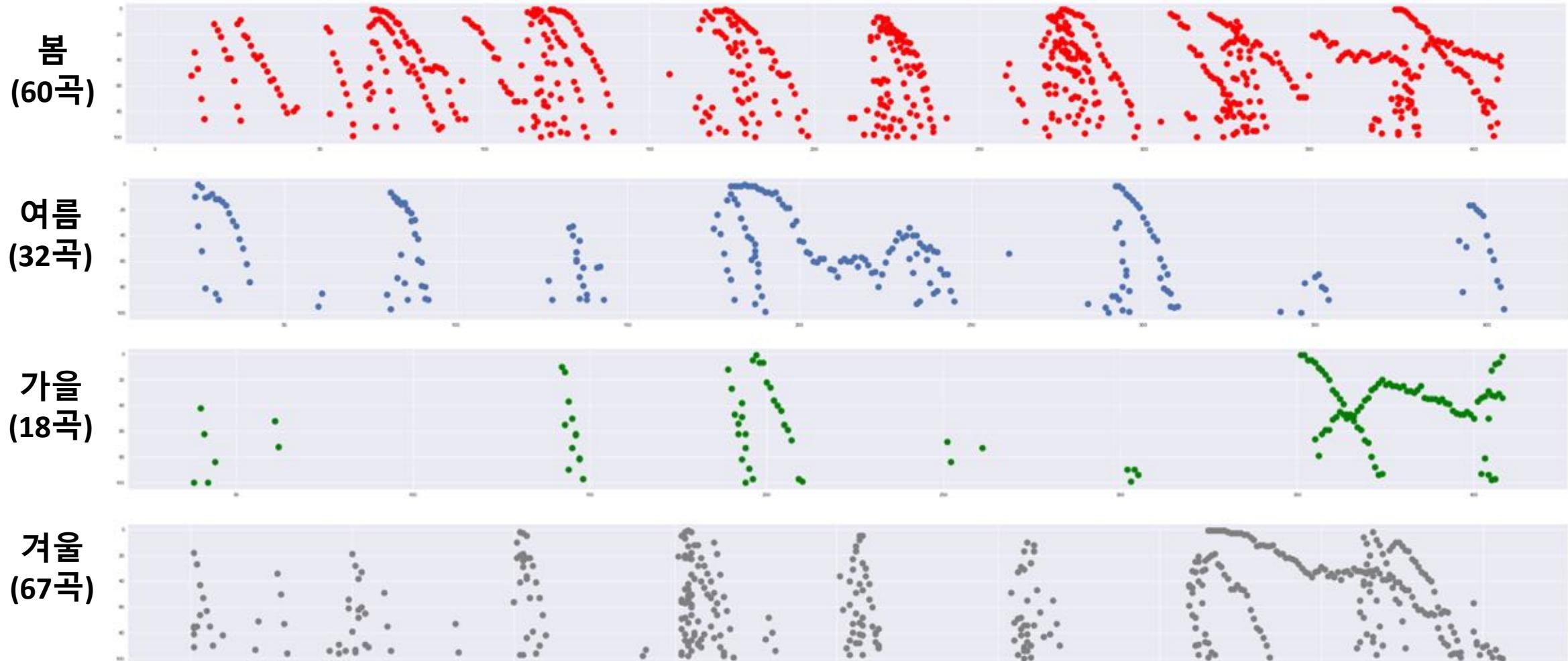
## Part 2 : 멜론 음원 차트와 역주행

### 2.4 계절성과 역주행

멜론 차트 역주행 분석 및 음원 추천 알고리즘

#### 2) 계절별 분포 확인

▼ 비교적 유의미한 패턴이 나타남. 그러나 진정한 Seasonality가 되려면 반복성을 지녀야 함.



## Part 2 : 멜론 음원 차트와 역주행

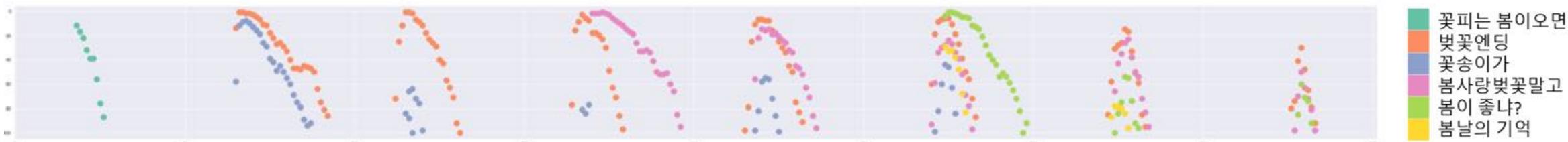
### 2.4 계절성과 역주행

멜론 차트 역주행 분석 및 음원 추천 알고리즘

#### 3) 세 가지 조건 설정

- ① 각 계절별 평균 순위가 해당 계절에 가장 높을 것
- ② 각 계절별 랭킹 진입 빈도수가 해당 계절에 가장 높을 것
- ③ 발매년도 이후 년도에도 랭킹 진입을 할 것

#### -봄 (6곡)



#### -겨울 (18곡)



▲ 봄이 3~5월에 걸쳐서 흥행한 반면, 겨울은 크리스마스 전후로 반짝 흥행함을 유추할 수 있음.

#### 4) 필터링 결과

|             | 봄  | 여름 | 가을 | 겨울 |
|-------------|----|----|----|----|
| 전체          | 60 | 32 | 18 | 67 |
| ①,②<br>총족   | 33 | 25 | 0  | 58 |
| ①,②,③<br>총족 | 6  | 0  | 0  | 18 |

## Part 3: 자연어처리를 통한 추천 시스템

- 3.1 모델링을 위한 EDA
- 3.2 클러스터링을 통한 음악 추천
- 3.3 Word2Vec을 통한 음악 추천

# Part 3 : 자연어처리를 통한 추천 시스템

멜론 차트 역주행 분석 및 음원 추천 알고리즘

## 3.1 모델링을 위한 EDA

크롤링한 시계열 데이터의 음원차트 순위정보 제거한 후 unique한 id를 기준으로 6037개의 case로 정리

| 구분 | 변수명          | 변수설명  |                     |
|----|--------------|---|---------------------|
| 변수 | times_appear | 10년의 멜론 주간 차트에 등장하는 횟수<br>:Duration의 개념을 가지지만 한번 등장 후 차트에서 사라질 때까지가 아닌 총 등장 횟수라는 점에서 다름 |                     |
|    | top_artist   | 역주행 곡의 artist (binary)  | positive / negative |
|    | rank_3       | 차트에서 1,2,3위 기록이 있는 artist (binary)  | anger               |
|    | genre        | 곡의 장르 (9 levels)  | anticipation        |
|    | label        | 역주행 곡 (binary)  | fear                |
|    | year         | 음원 발매 년도 (24 levels)  | joy                 |
|    | month        | 음원 발매 달 (12 levels)   | sadness             |
|    | feat         | 피쳐링 여부 (binary)   | surprise            |
|    | onair (*)    | 방송 출연 여부 (binary)   | trust               |
|    | season (*)   | 타이틀에 계절 정보 등장 여부 (binary)   | num_words           |
|    | weather (*)  | 타이틀에 날씨 정보 등장 여부 (binary)   | lexical_density     |

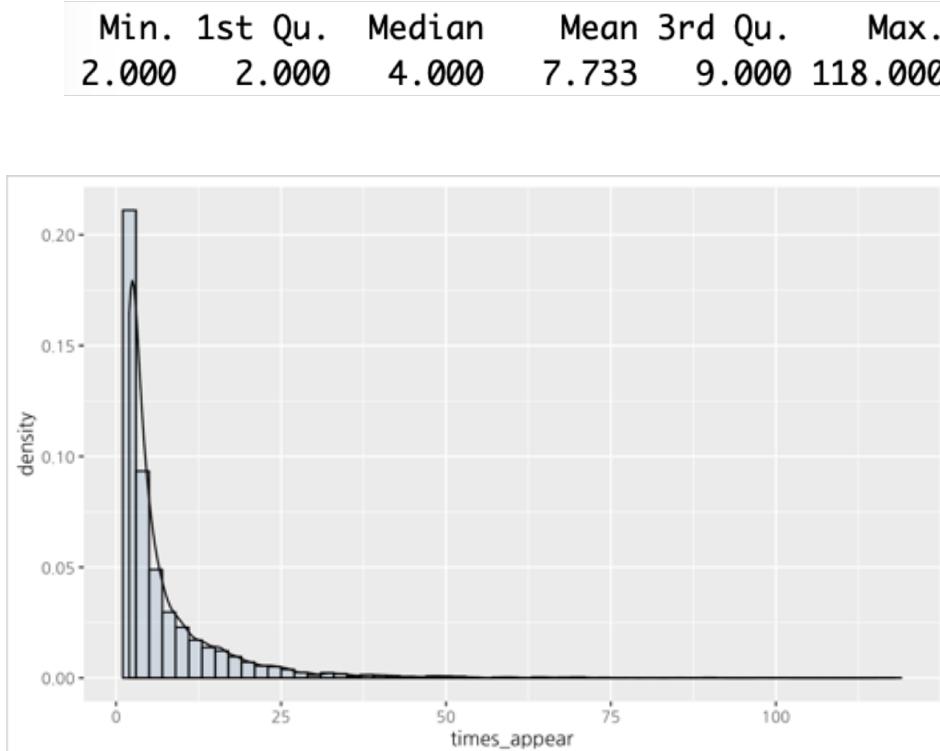
\* onair, season, weather 변수 생성은 별첨을 참고

# Part 3 : 자연어처리를 통한 추천 시스템

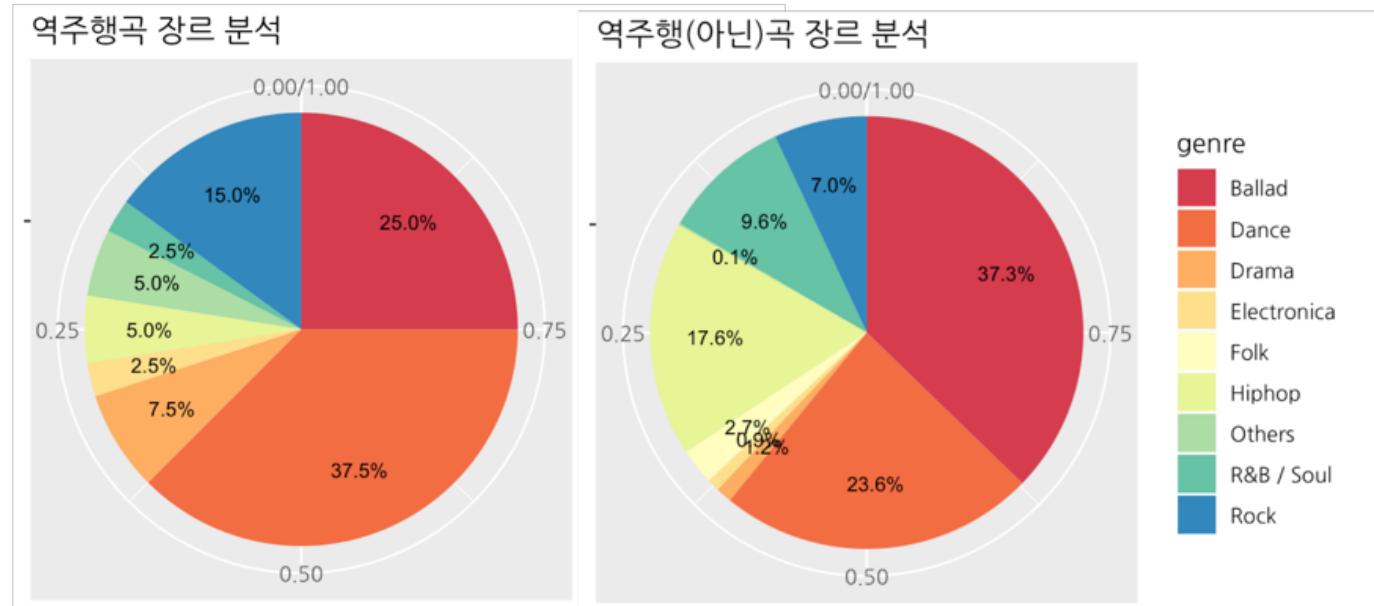
## 3.1 모델링을 위한 EDA

멜론 차트 역주행 분석 및 음원 추천 알고리즘

차트에 등장하는 횟수 변수: times\_appear & 크롤링으로 얻은 변수 EDA: genre



▲ 10년치 차트에 등장하는 횟수의 density



▲ 장르분석 (좌: 역주행 곡, 우: 역주행 아닌 곡)

전체 곡에는 발라드가 가장 많은 비율을 차지하지만 역주행으로 라벨링 된 곡에는 댄스곡의 비율(37.5%)이 가장 큰 것을 확인. 그 뒤로 발라드(25.9%), 락(15.0%), 일렉트로닉 (7.5%) 장르가 많음.

# Part 3 : 자연어처리를 통한 추천 시스템

멜론 차트 역주행 분석 및 음원 추천 알고리즘

## 3.1 모델링을 위한 EDA

가사 분석을 통한 파생변수 생성: positive, negative, anger, anticipation, fear, joy, sadness, surprise, trust



| title   | label | token_lyric  | token_lyric_twitter                                      | positive | negative | anger | anticipation | fear | joy | sadness | surprise | trust |
|---|-------|--|--|----------|----------|-------|--------------|------|-----|---------|----------|-------|
| 브레이킹베드<br>(Feat. 기리보<br>이) (Prod.<br>Cosmic Boy,<br>기리보이) | 0     | [날, 학교, 다니,<br>고, 때, 왕따, 시키,<br>ㄴ, 애, 들, 은,<br>PC, 방, 예...]  | [날, 학교, 다닐, 때, 왕따,<br>시킨, 애, 들, 은, PC, 방,<br>에서, 아르바...  | 10.0     | 21.0     | 10.0  | 5.0          | 6.0  | 7.0 | 15.0    | 0.0      | 12.0  |
| XXL (Feat. 딥<br>플로우,<br>Dok2)                             | 0     | [나이, 만, 치, 어,<br>먹, 은, 래퍼, 들,<br>이, 진지, 해이, 어<br>리, ㄴ, 놈...] | [나이, 만, 치, 먹은, 래퍼,<br>들, 이, 진지해, 이, 어린,<br>놈, 의, 침대,...] | 7.0      | 26.0     | 13.0  | 8.0          | 6.0  | 5.0 | 10.0    | 3.0      | 6.0   |
| 빌어먹을 인연<br>(Feat. 식케이)                                    | 0     | [i, need, new,<br>whipi, need,<br>new, crib, 이제,<br>노리,...]  | [i, need, new, whipi,<br>need, new, crib, 이제,<br>노리,...] | 10.0     | 26.0     | 6.0   | 1.0          | 2.0  | 1.0 | 7.0     | 1.0      | 3.0   |

▶ twitter 형태소 분석기 사용:  
Komoran보다 NRC 감성사전의  
형태와 비슷

▶ 각 감성별 매칭되는 단어를 차후  
분석에서 비율로 환산하여 활용

▲ komoran과 twitter 형태소 분석기를 사용하여 토큰화

# Part 3 : 자연어처리를 통한 추천 시스템

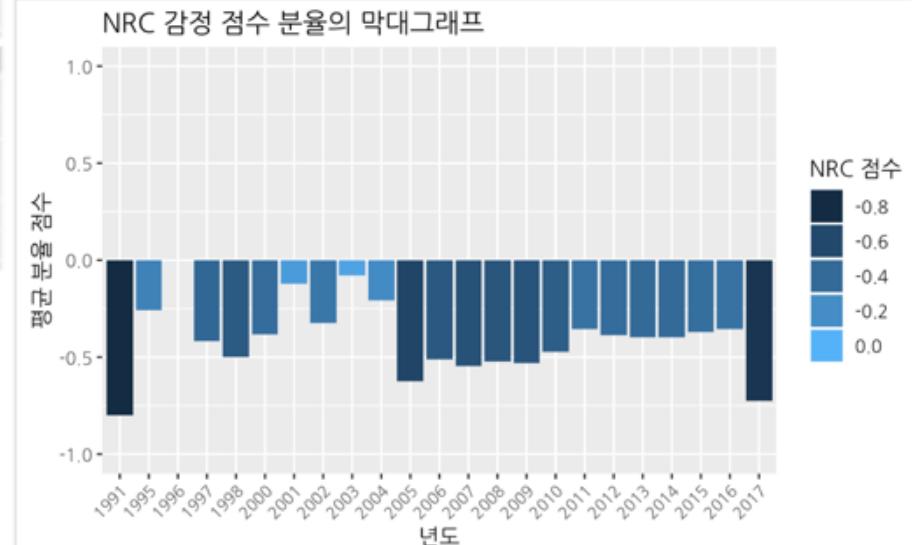
## 3.1 모델링을 위한 EDA

멜론 차트 역주행 분석 및 음원 추천 알고리즘

가사 분석을 통한 파생변수 생성: positive, negative, anger, anticipation, fear, joy, sadness, surprise, trust



- ◀ 음원 발매년도에 따른 가사의 감성 분석  
: anger, sadness, trust 등의 감성에서 연도별 차이를 관측 가능.
- ▼ 음원 발매년도에 따른 가사의 긍부정 분석  
: 대체적으로 부정의 score가 높은 것은 많은 음원이 사랑 중 이별에 대한 것이기 때문이라 추측



# Part 3 : 자연어처리를 통한 추천 시스템

## 3.1 모델링을 위한 EDA

멜론 차트 역주행 분석 및 음원 추천 알고리즘

가사 분석을 통한 파생변수 생성: **lexical\_density, num\_words**

### Lexical Density

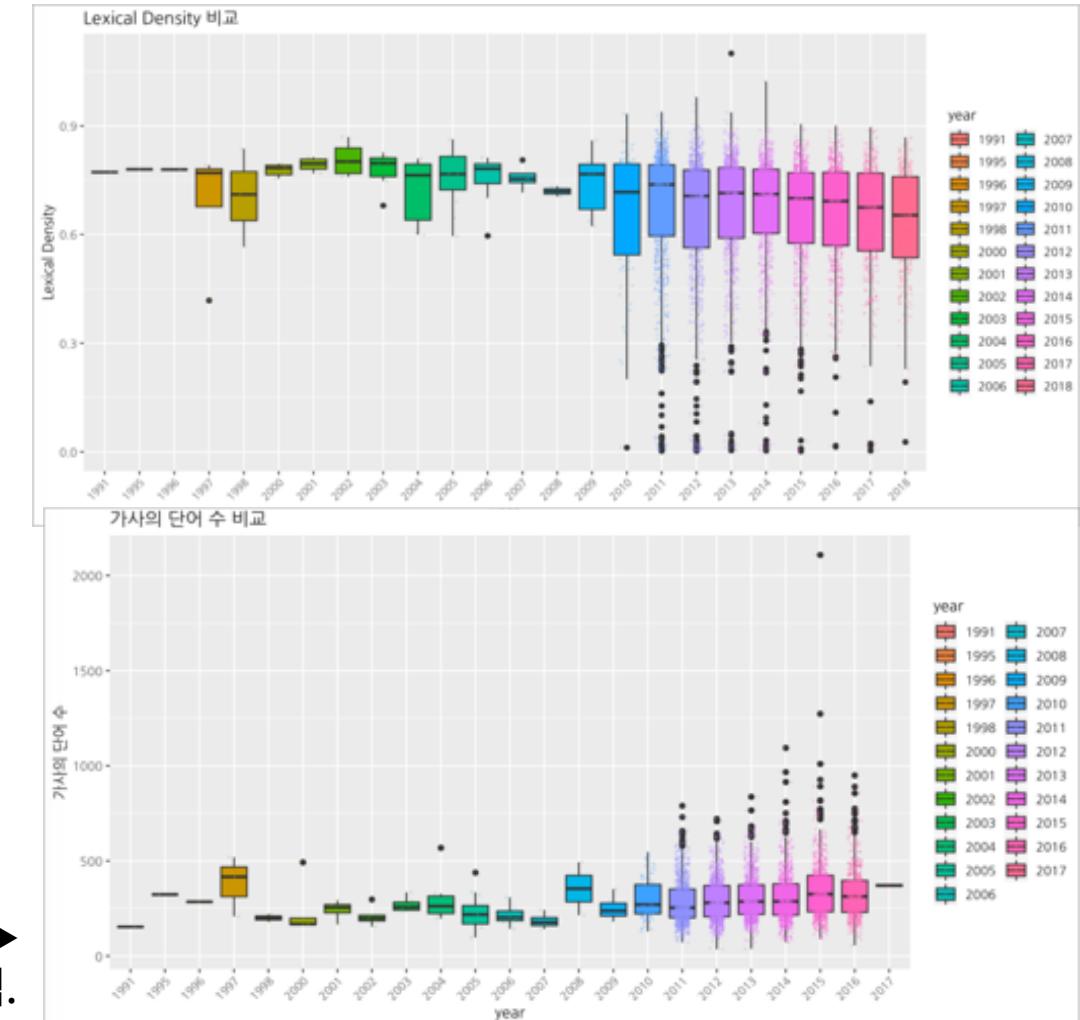
$$L_d = \left( \frac{L_{lex}}{N} \right) * 100$$

$N_{lex}$ : the number of lexical word tokens  
(nouns, adjective, verbs, adverbs)  
 $N$ : the number of all tokens

lexical density가 높을 수록 정보를 많이 함축하는 혹은 문어체에 가까운 가사를 의미. 연도별 lexical density를 보았을 때, 최근에 가까울 수록 variance가 큰 것을 통해 다양한 형태의 작사 작업이 이루어지는 것을 확인 가능.

\* ref: wikipedia

연도별 lexical density와 가사의 token 수 비교 박스 플랏 ►  
: 최근으로 갈수록 작사 방식에 variation이 커짐.

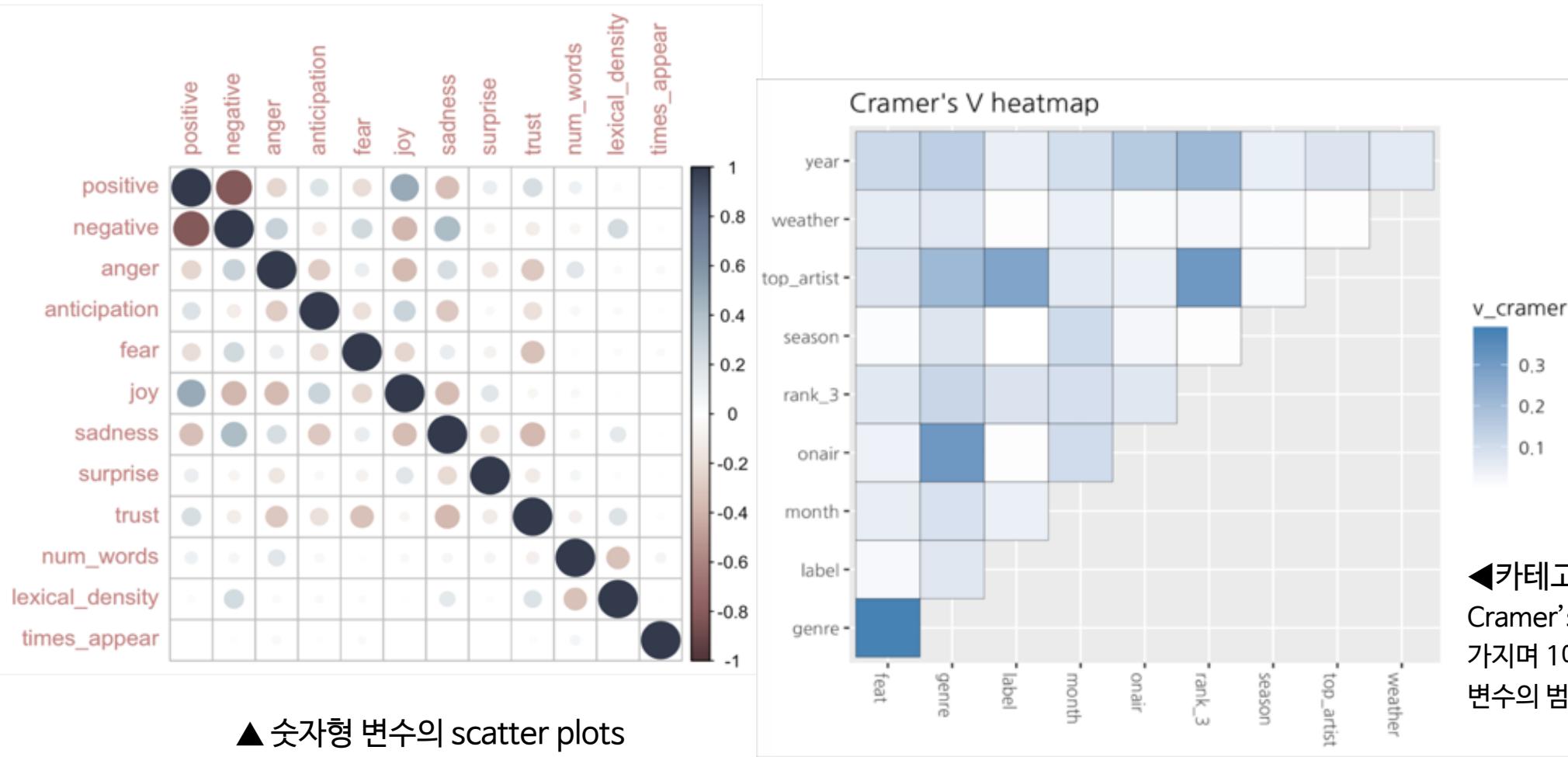


# Part 3 : 자연어처리를 통한 추천 시스템

## 3.1 모델링을 위한 EDA

멜론 차트 역주행 분석 및 음원 추천 알고리즘

### 변수들 간의 상관관계 분석



#### K-Prototype Clustering 개념 소개

- Motivation

기존의 K-means 클러스터링은 numeric 케이스 간의 Euclidean 거리를 활용. 따라서 numeric 변수와 categorical 변수가 함께 있는 경우 새로운 형태의 distance가 정의되어야 함.

- 기존의 클러스터링 방식과의 차이점

K-Prototype 클러스터링은 K-means와 유사한 패러다임을 공유하지만 categorical 변수에 대한 update에서는 K-modes 클러스터링 방식을 사용.

- 새로 정의하는 dissimilarity measure

Define  $n$  to be Euclidean distance between numerical variables and  $c$  to be the dissimilarity measure between categorical variables.  $n+r*c$  becomes the dissimilarity between two cases where  $r$  is the weight to balance the two parts. ( $r$ 은 R에서 lambda로 표현 됨)

# Part 3 : 자연어처리를 통한 추천 시스템

## 3.2 클러스터링을 통한 음악 추천

멜론 차트 역주행 분석 및 음원 추천 알고리즘

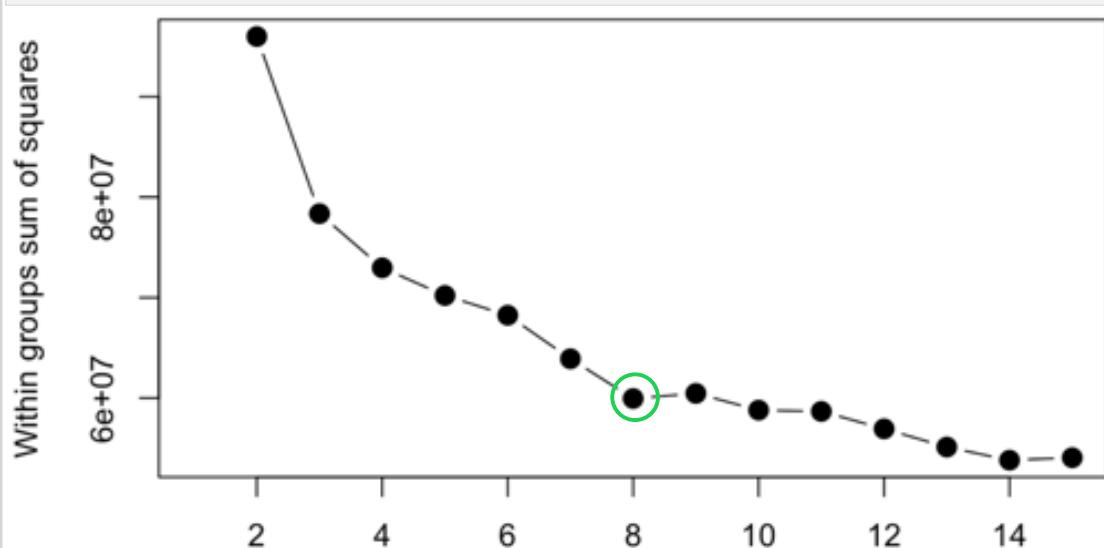
### K-Prototype Clustering 분석 결과

\* 추가적인 분석결과 시각화는 별첨을 참고

```
par(mfrow=c(1,2))

for(i in 1: 1:6){
  plot(pokemon[,c(5,5+i)], col=df_eda$cluster, main="K-prototypes")
}

lambdaest(df_eda)
res <- kproto(df_eda, 8, lambda = lambdaest(x))
clprofiles(res, df_eda)
df_eda$cluster = res$cluster
```



▲ elbow method를 통해 8개 클러스터 생성

| title  | artist           |
|--|------------------|
| 전 여자친구에게 (Feat. 선우정아)                          | San E            |
| 역주행 (Feat. Dok2, 천재노창)                         | 스윙스              |
| Gravity (Feat. Mad Clown, 기리보이)                | 스윙스              |
| BORN HATER (Feat. 빈지노, 베벌진트, B.I, MINO, BOBBY) | 에픽하이 (EPIK HIGH) |
| The Time Goes On                               | BewhY (비와이)      |
| Hood   | Tablo            |
| 야유회 (Feat. 지코(ZICO))                           | 다이나믹 듀오          |
| 니가 알던 내가 아냐 (Prod. By GRAY)                    | 사이먼 도미닉          |
| 공중도덕 (Air DoTheQ)                              | The Quiett       |
| N분의1 (Feat. 다이나믹듀오)                            | 넉살               |
| 도박 (Life Is a Gamble) (Feat. 박재범, Dok2)        | Ja Mezz          |
| 노땡큐 (Feat. MINO, 사이먼 도미닉, 더콰이엇)                | 에픽하이 (EPIK HIGH) |

▲ 클러스터 7에 포함된 곡

힙합 위주의 곡으로 묶인 것을 확인할 수 있다. 클러스터 7이외의 클러스터에서도 발라드, 밴드음악, 가수별로 클러스터간 유사도를  
직관적으로 확인 가능

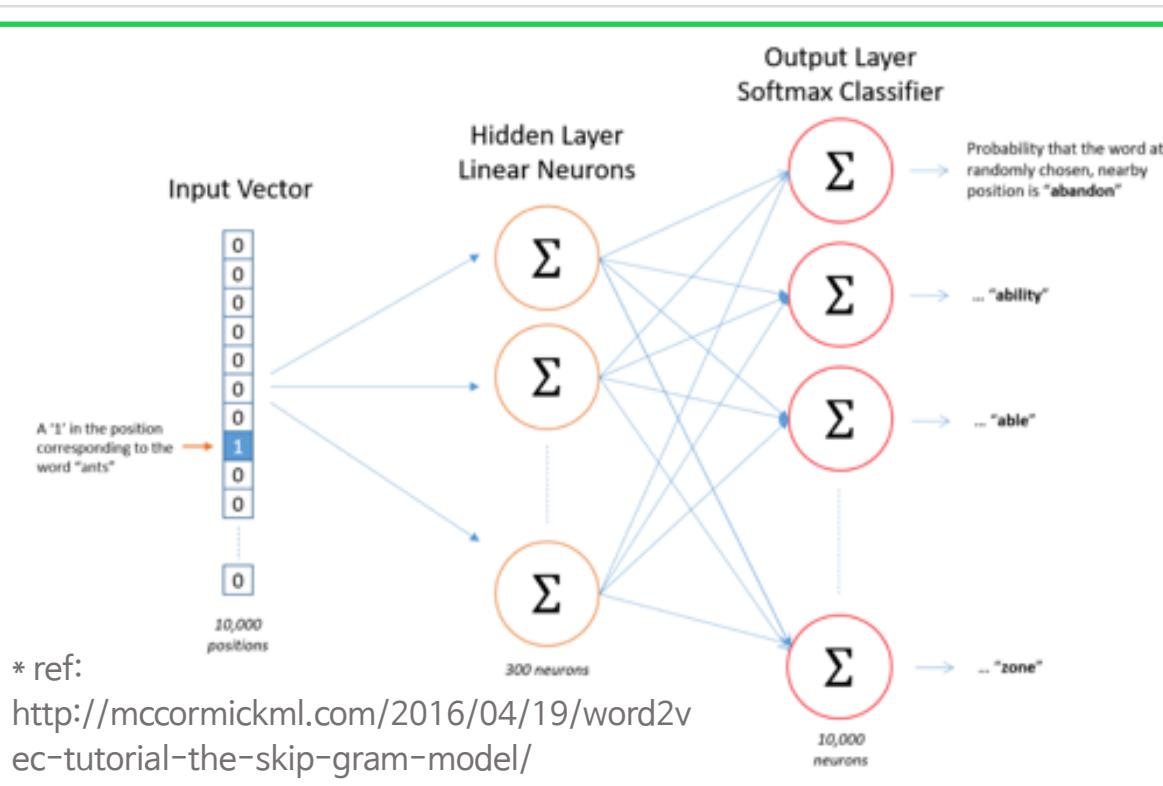
# Part 3 : 자연어처리를 통한 추천 시스템

## 3.3 Word2Vec 통한 음악 추천

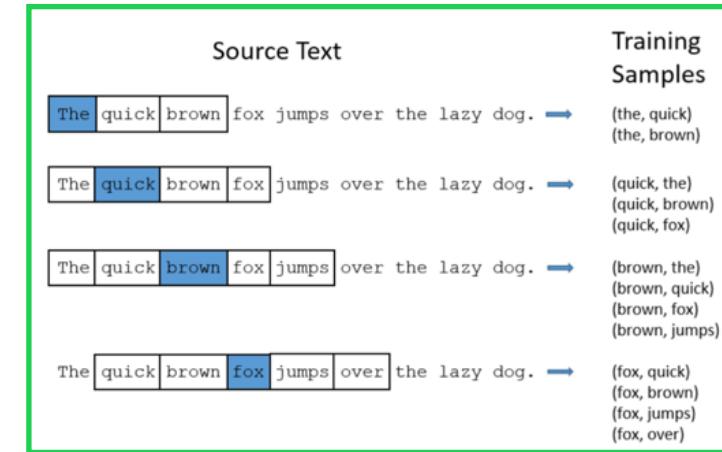
멜론 차트 역주행 분석 및 음원 추천 알고리즘

### Word2Vec 모델을 통해 weight, 즉 hidden layer를 학습

- **CBOW(Continuous Bag of Words)**: 주변에 있는 단어로 중심에 있는 단어를 맞추도록 학습하는 방식
- **Skip-Gram**: 중심에 있는 단어를 중심으로 주변의 단어를 맞추도록 학습하는 방식



Skip-Gram의 input  
◀ Skip-Gram 아키텍쳐



- **Input Layer**  
: one-hot encoded word vector
- **Hidden Layer**  
: weight matrix (# of words \* # of features)
- **Output Layer**  
: input으로 들어온 단어가 다른 단어 근처에 있을 확률 (softmax를 통한 0~1의 값으로 반환)

# Part 3 : 자연어처리를 통한 추천 시스템

## 3.3 Word2Vec 통한 음악 추천

멜론 차트 역주행 분석 및 음원 추천 알고리즘

### Word2Vec 모델을 통해 얻은 word vector (weight)와 결과

```
embedding_model = word2vec.Word2Vec(data, size=200, window = 5, min_count=50,  
                                     workers=2, iter=100, sg=1)
```

- **size**: input이 변환되는 벡터의 차원 (hidden layer의 neuron 수)
- **window**: 앞 뒤로 동시에 고려하는 단어의 수
- **min\_count**: 코퍼스 내 출현 빈도가 min\_count 미만인 것은 제외
- **workers**: 사용하는 코어 갯수
- **iter**: 학습 횟수
- **sg**: CBOW(0)와 skip-gram(1) 중 선택

◀ Gensim의 Word2Vec function으로 Skip-Gram 뉴럴넷을 학습

```
embedding_model.wv.most_similar(positive=[ "봄" ], topn=10)  
  
[ ('겨울', 0.4736689329147339),  
  ('꽃', 0.4298463463783264),  
  ('피다', 0.3984295725822449),  
  ('계절', 0.38550102710723877),  
  ('벚꽃', 0.3690962791442871),  
  ('가을', 0.36882591247558594),  
  ('따뜻하다', 0.3416163921356201),  
  ('여름', 0.33199405670166016),  
  ('설레다', 0.3315635919570923),  
  ('핀', 0.3310455083847046)]
```

‘사랑’과 가까운  
단어들 10개 ►

◀ ‘봄’과 가까운  
단어들 10개

(considers cosine  
similarity)

```
embedding_model.wv.most_similar(positive=[ "사랑" ], topn=10)  
  
[ ('이별', 0.49479424953460693),  
  ('아프다', 0.4455459713935852),  
  ('하다', 0.44501277804374695),  
  ('말', 0.43216997385025024),  
  ('아끼다', 0.37557464838027954),  
  ('영원하다', 0.3679729104042053),  
  ('헤어지다', 0.35831543803215027),  
  ('미워하다', 0.35714852809906006),  
  ('정말', 0.3563750982284546),  
  ('후회', 0.35318320989608765)]
```

# Part 3 : 자연어처리를 통한 추천 시스템

## 3.3 Word2Vec 통한 음악 추천

멜론 차트 역주행 분석 및 음원 추천 알고리즘

### Word2Vec 모델을 통해 얻은 가중치 매트릭스와 결과

Word2Vec 워드 백터 (1952\*201)

|     | V1  | V2  | V3  | ... | V199 | V200 |
|-----|-----|-----|-----|-----|------|------|
| 가을  | 0.2 | 0.1 | 0.3 | 0.1 | 0.1  | 0.1  |
| ... | ... | ... | ... | ... | ...  | ...  |
| 힙합  | 0.1 | 0.2 | 0.2 | 0.2 | 0.2  | 0.1  |

$$W_{ij} = \exp\left(-\frac{d(x_i, x_j)^2}{2}\right)$$

가중치 매트릭스(1952\*1952)

|     | 가을   | 가끔   | 눈물   | ... | 하트   | 힙합   |
|-----|------|------|------|-----|------|------|
| 가을  | 1.00 | 0.01 | 0.03 | ..  | 0.01 | 0.02 |
| ... | ...  | ...  | ...  | ... | ...  | ...  |
| 힙합  | 0.01 | 0.02 | 0.02 | ..  | 0.02 | 1.00 |

Document Term Matrix(DTM) (6038\*1952)

|      | 가을  | 가끔  | 눈물  | ... | 하트  | 힙합  |
|------|-----|-----|-----|-----|-----|-----|
| doc1 | 0   | 0   | 1   | ..  | 1   | 0   |
| ...  | ... | ... | ... | ... | ... | ... |
| 6038 | 1   | 1   | 0   | ... | 0   | 0   |

X

4개의 토픽만 추출한 가중치 매트릭스 (4\*1952)

|     | 가을   | 가끔   | 눈물   | ... | 하트   | 힙합   |
|-----|------|------|------|-----|------|------|
| 사랑  | 1.00 | 0.02 | 0.03 | ..  | 0.01 | 0.02 |
| 겨울  | 0.01 | 0.02 | 0.02 | ..  | 0.02 | 0.01 |
| ... | ...  | ...  | ...  | ... | ...  | ...  |

토픽 별 음원의 score (6038\*4)

|      | 사랑   | 겨울   | 크리스마스 | 가족   |
|------|------|------|-------|------|
| doc1 | 0.04 | 0.43 | 0.02  | 0.01 |
| ...  | ...  | ...  | ...   | ...  |
| 6038 | 0.01 | 0.35 | 0.14  | 0.02 |

\*구현:

<https://github.com/SeoHyeong/MelonChart/tree/master/Word2Vec>

\* ref:

<https://ratsgo.github.io/natural%20language%20processing/2017/03/08/word2vec/>

선택한 토픽에 따라 score가 높은 음원 추출 및 추천 가능

# Part 3 : 자연어처리를 통한 추천 시스템

## 3.3 Word2Vec 통한 음악 추천

멜론 차트 역주행 분석 및 음원 추천 알고리즘

### 스코어 매트릭스를 통해 얻은 토픽 별 추천 음원

| title         | artist  | genre       | lyric   |
|---------------|---------|-------------|---|
| Sweetest Name | 클래지콰이   | Electronica | Don't you wanna be free 이런 시간들도 Don't wanna...    |
| 다르다는 것        | 어반자카파   | Ballad      | 이제야 모두 널 알았다 생각했을 때또 다른 네 모습 보는 나모든 걸 알아야 하는 나... |
| 10~Jan        | 브로콜리너마저 | Rock        | 우리가 함께 했던 날들의 옆에 하나만 기억해줄래 우리가 아파했던 날은 모두 나 혼...  |
| 십년이 지나도       | 권진아     | Ballad      | 그래 난 괜찮아너만 그 사람이 좋다면 나 웃으며 널 보내줄 수가 있어하지만 왜 자꾸... |
| Love Me       | 배다해     | Ballad      | 잊어달란 그 말도무지 믿을 수 없어사랑한다 내게 말했던 너였잖아그 시간들 추억이 날... |

◀ '사랑' 토픽에서 가장 높은 스코어를 얻은 5곡

▼ '겨울' 토픽에서 가장 높은 스토어를 얻은 5곡

- 선택한 토픽(단어)가 학습한 단어에 포함되어 있어야 함
- GloVe, FastText 등의 다른 임베딩으로 더 좋은 분류를 기대할 수 있음

| title          | artist   | genre       | lyric   |
|----------------|----------|-------------|---|
| 사랑하오 (김현철&윤상)  | 김범수      | Ballad      | 그대 사랑하오 아직도 사랑을 알지 못하지만 이 나이 되도록그대 사랑하오 그대...     |
| Only One       | 거미       | Electronica | 처음 봤던 너의 모습 이상하게 좀 진했고 두 번 보는 너의 모습에 내 가슴은 뛰어댔... |
| 그대라서           | 신용재 (포맨) | Ballad      | 눈 내리는 밤거리를 함께 걷는 이 순간눈 내리는 하얀 밤을 함께 있는 이 순간따뜻하... |
| Crying         | 허영생      | Dance       | Cuz I can't see you no more Cryin' cryin' cryi... |
| 실화             | 케이윌      | Ballad      | 눈을 감아도 또 니가 보이고 길을 걸어도 또 너만 생각나 오늘 하루도 난 너를 빼고... |
| Christmas Time | 성시경      | Ballad      | 혹시 기억하니 우리 처음 본 날 그때처럼 하얀 눈이 내려너에게 가는 길 내 맘과 같... |

## Part 4: 프로젝트 결론

4.1 Part 2 프로젝트 분석 결론

4.2 Part 3 프로젝트 분석 결론

## Part 4 : 프로젝트 결론

### 4.1 Part 2 프로젝트 분석 결론

멜론 차트 역주행 분석 및 음원 추천 알고리즘

#### Part 2 역주행 곡 분석에 대한 결론 및 한계점

##### - 역주행 곡 분석 결론

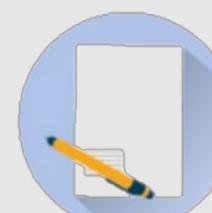
✓ 음원의 차트 순위는 계절, 날씨, 입소문 등의 영향을 받음

✓ 음원의 역주행 패턴은 크게 5개로 분류 가능하며, 분포에 따라 어떠한 원인에 의한 역주행인지 추론 가능

##### - 분석의 한계점



음원 외생적 정보의 부족 (가수의 인기도 등의 변수)



음원 내생적 정보의 부족 (음원의 길이, 템포, 키의 변화)

##### - 분석의 활용방안



음원의 역주행 분포를 이미지로 CNN (Convolutional Neural Networks)를 통해 학습. 차후 순위에 대한 대략적인 예측 가능.

# Part 4 : 프로젝트 결론

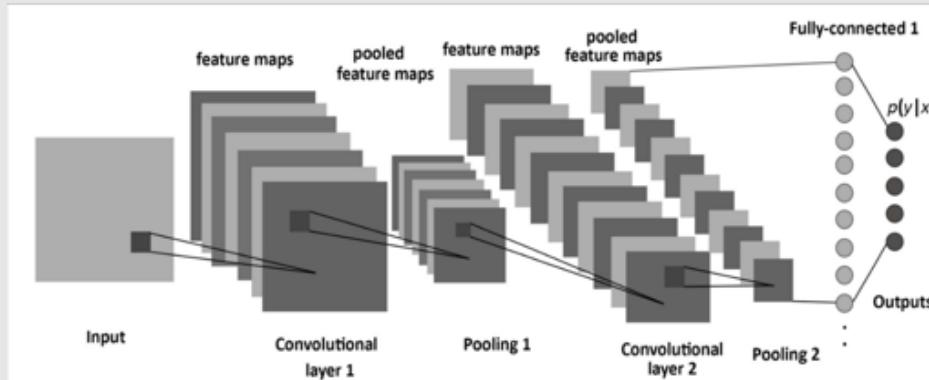
## 4.1 Part 2 프로젝트 분석 결론

멜론 차트 역주행 분석 및 음원 추천 알고리즘

CNN(Convolutional Neural Networks)를 활용한 차트 순위 예측

### 1. 데이터 형태를 변환하기

- 음원의 차트 순위를 이미지로 변환
- CNN 아키텍처를 통해 학습



Ref: Zhiguang Wang and Tim Oates. 2015. Encoding Time Series as Image for Visual Inspection and Classification Using Tiled Convolutional Neural Networks. 2015 AAAI Workshop

### 2. WaveNet을 통해 학습

- 구글의 DeepMind에서 개발한 WaveNet은 text and speech recognition을 위한 아키텍처
- WaveNet은 LSTM등의 모델처럼 과거의 값을 기억하도록 하는 모델 구조를 가지고 있음.

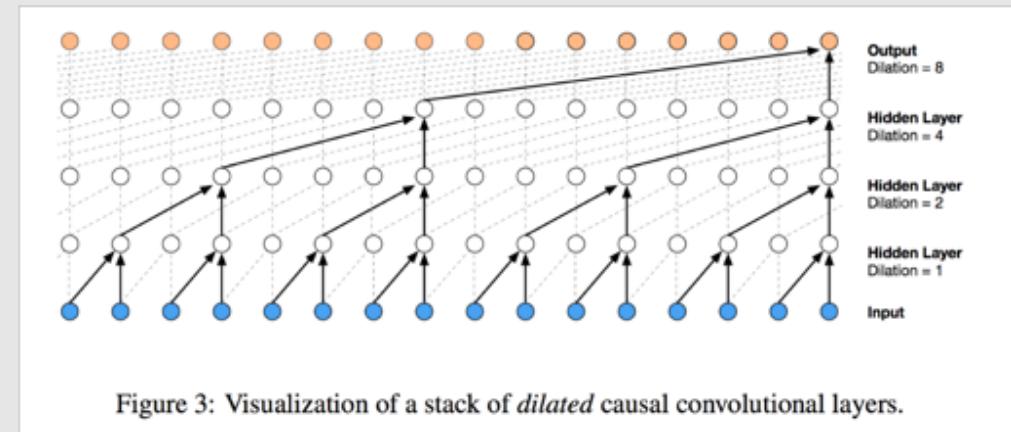


Figure 3: Visualization of a stack of *dilated* causal convolutional layers.

Ref: arXiv: 1609.03499v2 [sc.SD] 19 Sep 2016

## Part 4 : 프로젝트 결론

### 4.2 Part 3 프로젝트 분석 결론

멜론 차트 역주행 분석 및 음원 추천 알고리즘

#### Part 3 추천 시스템 분석의 활용 방안 및 한계점 요약

- 추천 시스템 분석 결론 및 활용방안



NRC Lexicon을  
사용하여 단어 중심의  
추천 시스템



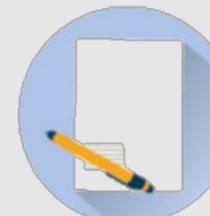
Word2Vec으로 가사를  
학습한 문맥 중심의  
추천 시스템

- 분석의 한계점 및 아쉬운 점



##### NRC Lexicon 활용의 한계

: 형태소 분석기의 불일치, 동음이의어  
및 비유적 표현을 미반영



##### 다양한 임베딩 시도

: Glove, FastText

감사합니다

## Part 5: 별첨: 코드 분석 및 참고자료

최신음악 > 국내 | 해외

## 이외의 파생변수: onair, season, weather

```
lst <- c("나는 가수다", "위대한 탄생", "무한도전", "명불허전", "나는  
작사가다", "슈퍼스타", "불후의 명곡", "SBS", "보이스 코리아",  
"보이스코리아", "MBC", "Voice Korea", "Show Me The Money", "쇼미더머니",  
"K팝 스타", "OST", "복면가왕", "언니들의 슬램덩크", "언프리티 랩스타",  
"고등래퍼")
```

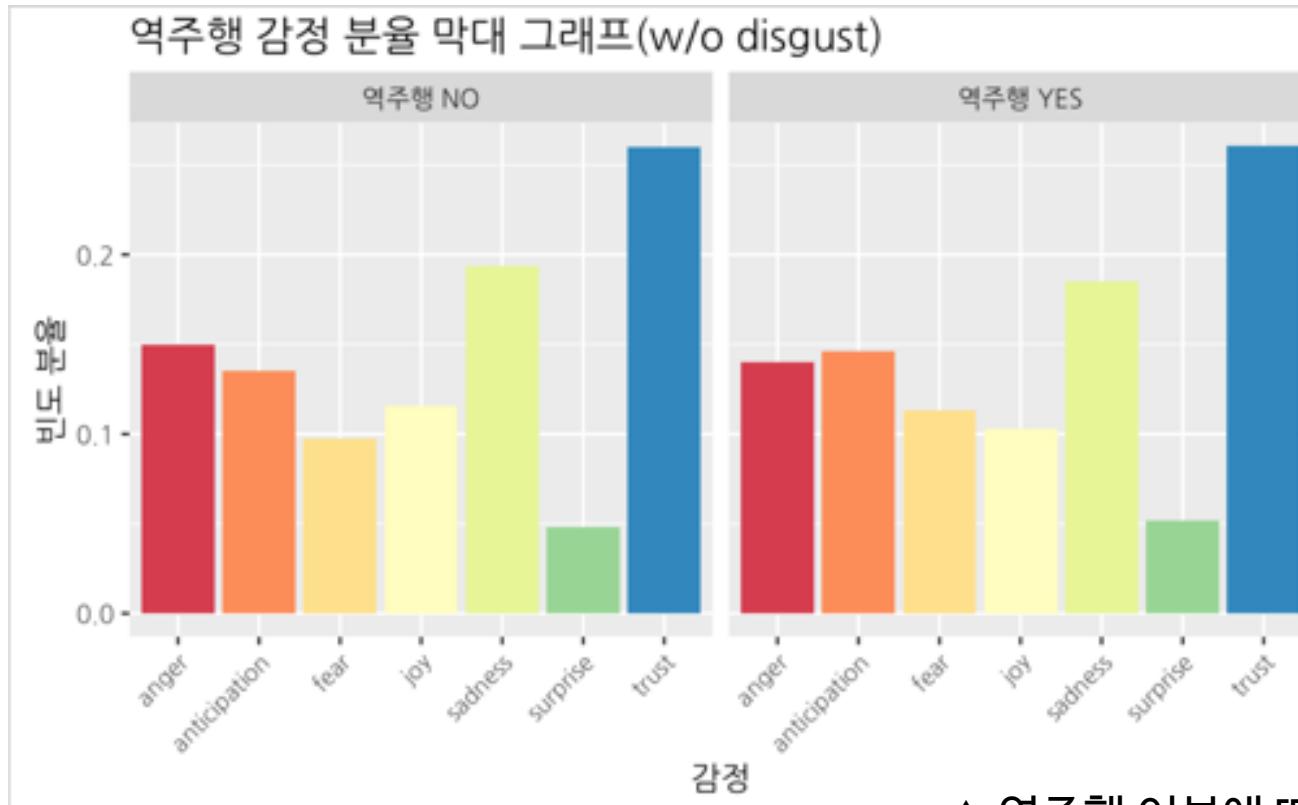
```
lst <- c("봄", "여름", "가을", "겨울", "크리스마스", "christmas",  
"Christmas", "벚꽃", "Summer", "summer", "Valentine", "valentine",  
"발렌타인")
```

```
lst <- c("비", "Rainy", "rain", "Rain", "rainy", "눈", "소나기")  
for (i in 1:6048){  
  for (item in lst){  
    if (grepl(item, df$title[i])){  
      df$weather[i] <- 1  
    }  
  }  
}
```

```
> summary(df_eda$onair)  
 0   1  
5318 719  
> summary(df_eda$season)  
 0   1  
5889 148  
> summary(df_eda$weather)  
 0   1  
5817 220
```

◀ 다음의 단어가 title에서 발견되면  
순서대로 onair, season, weather  
변수에 1을 기입하여 binary 변수 3개  
(onair, season, weather)를 생성

### 가사 자연어처리를 통한 변수 생성: 추가적인 EDA



▲ 역주행 여부에 따른 가사의 감정 분율 막대 그래프 (disgust 감정 미포함)



◀ 역주행 곡 가사에 대한 word cloud

▼ 비역주행 곡 가사에 대한 word cloud



대표적인 역주행 곡에 대한 TF-IDF (Term Frequency - Inverse Document Frequency) 시각화

**TF-IDF:** 여러 문서로 이루어진 문서군이 있을 때 어떤 단어가 특정 문서 내에서 얼마나 중요한지를 나타내는 통계적 수치

\* ref: wikipedia



▲ 총 40개의 역주행 선정 곡 중 10개 곡에 대한 tf-idf

선형 회귀분석을 통해 times\_appear 변수 설명 시도

가설: 앞서 소개된 변수(음원의 내생적 정보)와 인기도를 나타내는 times\_appear 변수 간의 **선형적 관계**가 존재

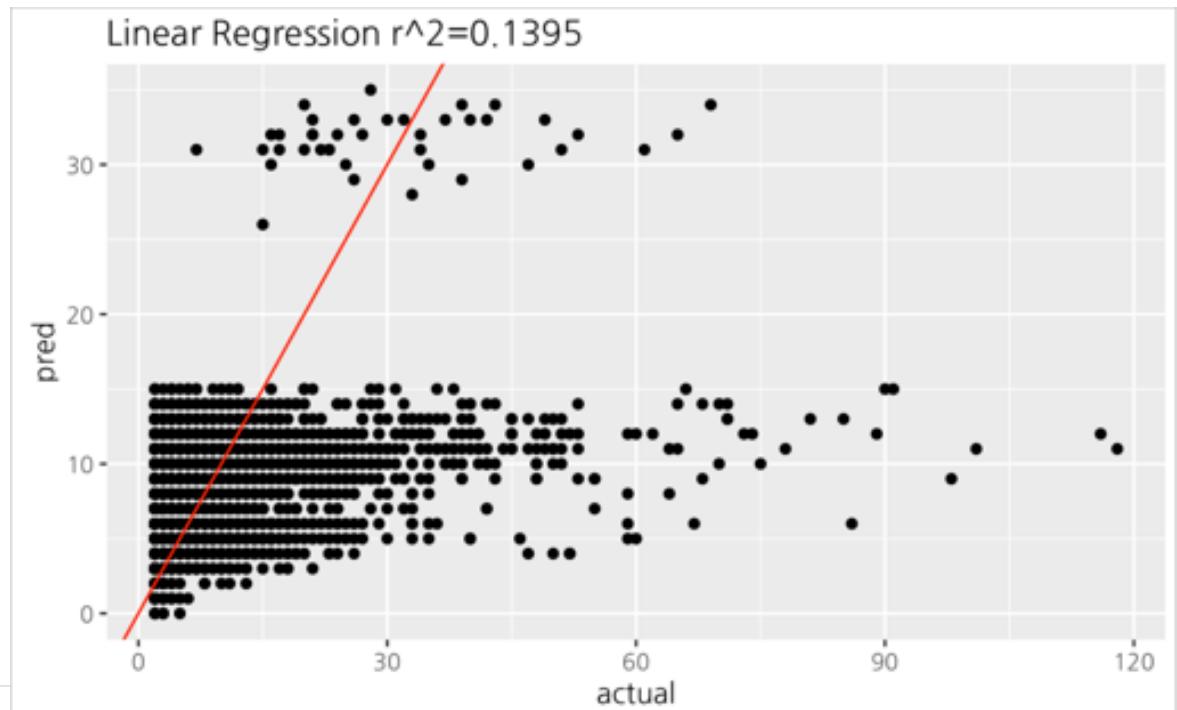
```

Call:
lm(formula = times_appear ~ ., data = df_eda)

Residuals:
    Min      1Q  Median      3Q     Max 
-23.895 -4.441 -1.810  1.570 107.235 

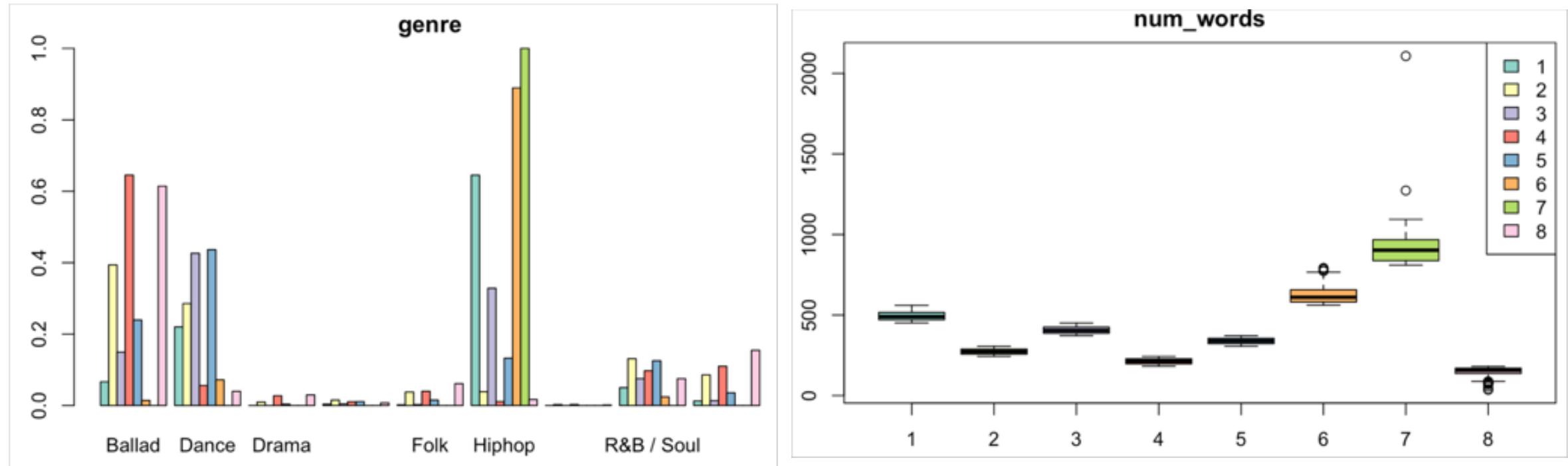
Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) -1.527457  9.024772 -0.169 0.865605  
genreDance   1.347566  0.355346  3.792 0.000151 *** 
...           ...
weather1     0.779977  0.618010  1.262 0.206970  
rank_31      4.947294  0.252128 19.622 < 2e-16 *** 
Residual standard error: 8.914 on 5976 degrees of freedom
(11 observations deleted due to missingness)
Multiple R-squared:  0.1395,    Adjusted R-squared:  0.1309 
F-statistic: 16.15 on 60 and 5976 DF,  p-value: < 2.2e-16

```



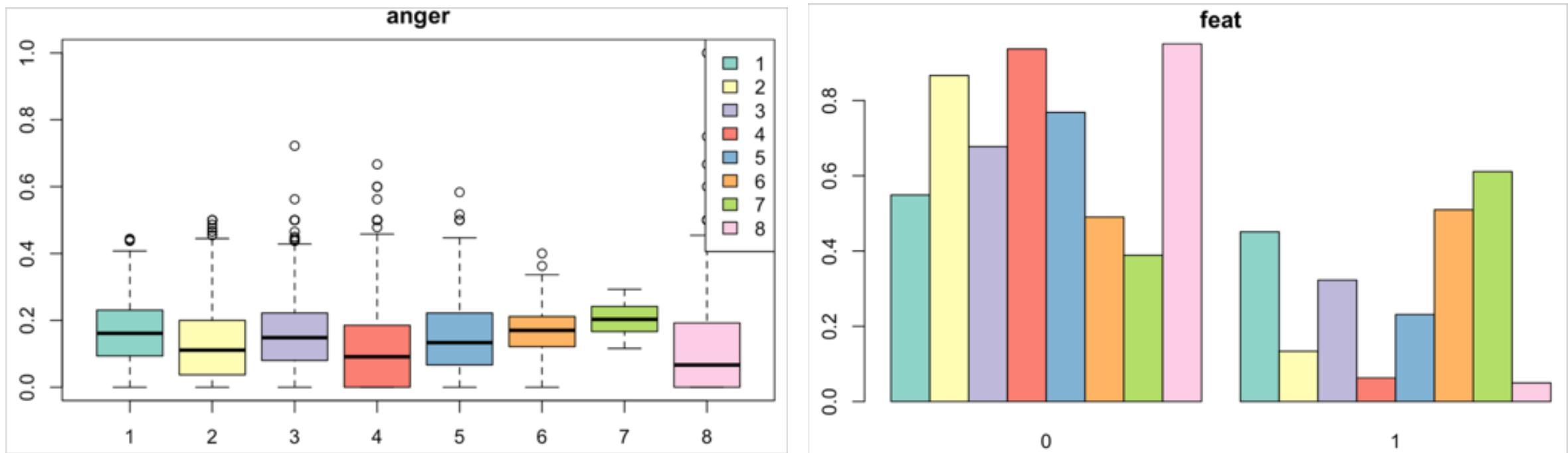
음원의 내생적 정보로 음원의 인기도를 측정하기에 한계가 있으며 이는 가수의 tv 출연 여부, 도덕성 등의 외생적 정보가 인기도에 더 많은 영향을 미치기 때문이라 판단

### K-Prototype Clustering 분석 결과 변수 별 시각화



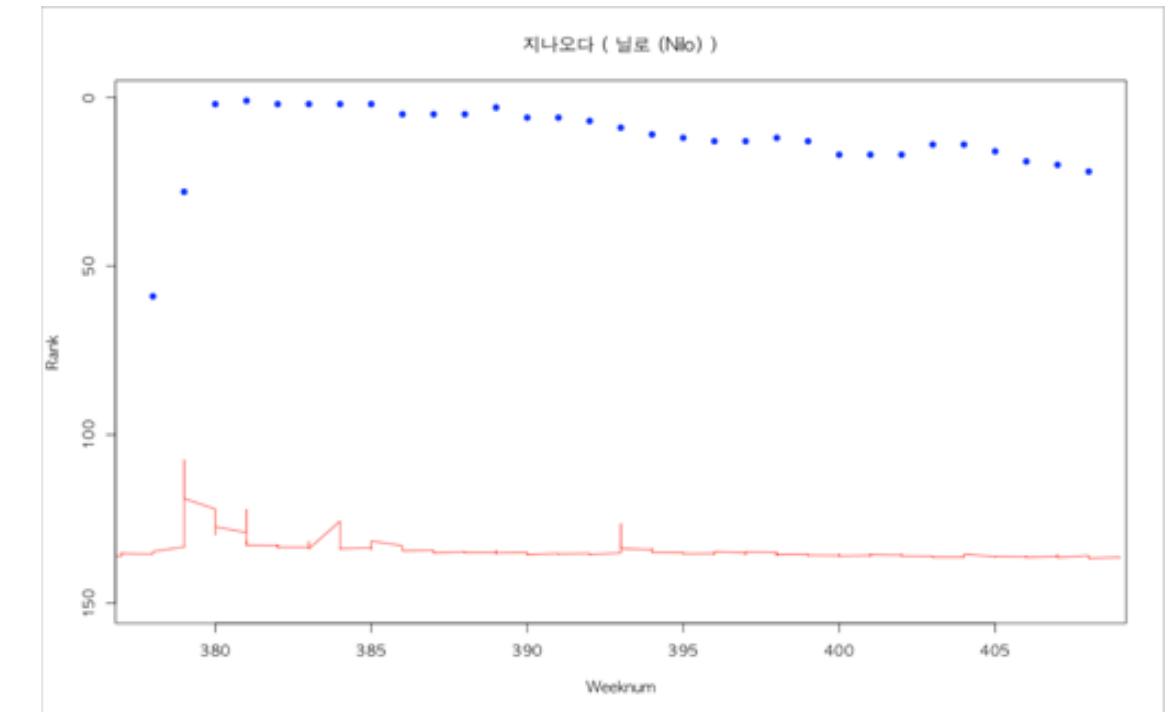
▲ R의 clustMixType 라이브러리를 활용한 K-Prototype Clustering 결과 시각화  
 : 클러스터 별 장르의 뮤음(좌), 클러스터 별 num\_words의 뮤음(우). 힙합 곡이 많은  
 클러스터 7(연한녹색)에서는 rap 음악인 만큼 num\_words가 다른 클러스터에 비해 높음

### K-Prototype Clustering 분석 결과 변수 별 시각화



▲ R의 clustMixType 라이브러리를 활용한 K-Prototype Clustering 결과 시각화  
: 클러스터 별 anger 감정 묶음(좌), 클러스터 별 피처링 여부 묶음(우). 힙합 곡이 많은  
클러스터 7(연한녹색)에서는 anger와 피처링 곡이 다른 클러스터에 비해 많음.

### 네이버 데이터 랩 검색어 트랜드와 역주행 곡 '지나오다(닐로)' 음원의 차트 변동



▲지나오다  
: 검색트렌드 상승(**빨간색**)에 따라 음원의 랭킹(**파란색**)이 상승

- Lexical Density: [https://en.wikipedia.org/wiki/Lexical\\_density](https://en.wikipedia.org/wiki/Lexical_density)
- TF-IDF: <https://ko.wikipedia.org/wiki/Tf-idf>
- K-Prototype Clustering: ZHEXUE HUANG , *Data Min. Knowl. Discov.* 2, 283-304 (1998)
- K-Prototype Clustering: Z. He, S. Deng, X. Xu, "Approximation Algorithms for K-Modes Clustering" in , Springer: Berlin/Heidelberg, vol. 4114, pp. 296-302, 2006.
- Code written for part 3: <https://github.com/SeoHyeong/MelonChart>
- Word2Vec: <https://ratsgo.github.io/natural%20language%20processing>
- CNN image: [https://res.mdpi.com/entropy/entropy-19-00242/article\\_deploy/html/images/entropy-19-00242-g001.png](https://res.mdpi.com/entropy/entropy-19-00242/article_deploy/html/images/entropy-19-00242-g001.png)

### 1.1 프로젝트의 목적

멜론 음원 차트에 나타난 역주행의 현황 및 원인 분석

Q. “역주행”이란 무엇인가?

A. 활동이 종료되는 등의 이유로 더 이상 크게 주목받지 못하던 곡이 재조명되어 음악 관련 차트나 가요프로 순위 상승이 다시 일어나는 것

Q. “역주행”을 추적할 수 있다면 어떨까?

A. 당장 인기를 끌지 못하더라도 어떤 곡이 차트에 빈번하게 등장하는지, 계절이나 날씨 등 어떤 주기로 등장하는지, 어떤 시점과 맞물려 역주행 현상이 발생하는지 예측할 수 있다.

### [프로젝트의 목적]

멜론 음원 차트에 나타난 역주행의 현황 및 원인을 분석함으로써,  
서비스 제공자인 멜론의 입장에서 사용자에게 음악을 추천하는 알고리즘 고안

# Part 1 : 멜론 음원 차트와 빅데이터

## 1.2 프로젝트의 흐름

멜론 차트 역주행 분석 및 음원 추천 알고리즘

전체적인 프로젝트의 흐름 소개

