

Pivotlama

Ornek olarak film isimleri ve o filmlere verilmiş beğeni notlarını taşıyan bir veri tabanını işleyeceğiz. Verimiz üç ayrı dosyaya yayılmış halde. Üç tabloyu alttaki şekilde, merge komutu ile birleştiriyoruz - Pandas otomatik olarak ortak kolon ismini bulacak ve onun üzerinden birleştirimi yapacak.

```
import pandas as pd
unames = ['user_id', 'gender', 'age', 'occupation', 'zip']
users = pd.read_table('users.dat', sep='::', header=None, names=unames)
rnames = ['user_id', 'movie_id', 'rating', 'timestamp']
ratings = pd.read_table('ratings.dat', sep='::', header=None, names=rnames)
mnames = ['movie_id', 'title', 'genres']
movies = pd.read_table('movies.dat', sep='::', header=None, names=mnames)
data = pd.merge(pd.merge(ratings, users), movies)
```

data

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 1000209 entries, 0 to 1000208
Data columns:
user_id      1000209  non-null values
movie_id     1000209  non-null values
rating       1000209  non-null values
timestamp    1000209  non-null values
gender       1000209  non-null values
age          1000209  non-null values
occupation   1000209  non-null values
zip          1000209  non-null values
title        1000209  non-null values
genres       1000209  non-null values
dtypes: int64(6), object(4)
```

Eğer erkeklerin en çok sevdiği ama kadınların en az sevdiği (ve hanımlar için tam tersi olan) filmleri bulmak istiyorsak, bu işlemi nasıl yaparız? Bu işlemi Pandas ile yapmak için ilginç bir takla atacagız. “Bir grubun en çok diğerinin en az” sorusu, onların bir filme verdiği ortalama notun farkının en büyük olması demektir. Bunu düşünebilmek önemli.

İkinci olarak bu işlemin kodlaması için ne gerekir? Bir çıkartma işlemi lazım. İdeal olarak bir kolonu (ya da satiri) diğerinden çıkartmak - bu tür toptan işlemler zaten Pandas ile çok hızlı.

Fakat verimiz halen o formatta değil. Her satır, tek bir film, tek bir kişi (cinsiyet) ve tek bir not için kaydedilmiş. Bizim ilgilendığımız analiz için biz film bazında için cinsiyet verisini *yanyana, değişik kolonlarda* gormeliyiz.

Peki nasıl? Cevap pivotlamak.

Pivotlamak bir kolonu (hatta birkaç kolonu) alıp onu x eksenini yapmak, aynı şekilde bir (veya birkaç) kolonu y eksenini yapmak anlamına gelir. Yani bir kolon üzerindeki tüm değerler okunur, ve kordinatmış gibi o eksene yayılır. Aynı şekilde diğer kordinat halledilir. Daha sonra bu iki

Bizim pivot için cinsiyet kolona yayılacak, film ismi satıra yayılacak. Kesisim ise not ortalaması (rating mean) olacak.

| gender | F | M |
|-------------------------------|----------|----------|
| title | | |
| \$1,000,000 Duck (1971) | 3.375000 | 2.761905 |
| 'Night Mother (1986) | 3.388889 | 3.352941 |
| 'Til There Was You (1997) | 2.675676 | 2.733333 |
| 'burbs, The (1989) | 2.793478 | 2.962085 |
| ...And Justice for All (1979) | 3.828571 | 3.689024 |

```
Index(['burbs, The (1989)', '10 Things I Hate About You (1999)', '101 Dalmatians (1961)', '101 Da
```

Devam edelim, simdi ortalama notlari ustteki yeni Index'e gore azaltalim (ve .ix kullanacagiz, cunku Index objesi satirlar uzerinde islem yapar ve .ix cagrisi satirlara erismek icin kullanilir), ve hanimlarin en cok sevdigi filmlere bakalim,

| gender | F | M \ |
|--|----------|----------|
| title | | |
| Close Shave, A (1995) | 4.644444 | 4.473795 |
| Wrong Trousers, The (1993) | 4.588235 | 4.478261 |
| Sunset Blvd. (a.k.a. Sunset Boulevard) (1950) | 4.572650 | 4.464589 |
| Wallace & Gromit: The Best of Aardman Animation (1996) | 4.563107 | 4.385075 |
| gender | diff | |

```

title
Close Shave, A (1995) -0.170650
Wrong Trousers, The (1993) -0.109974
Sunset Blvd. (a.k.a. Sunset Boulevard) (1950) -0.108060
Wallace & Gromit: The Best of Aardman Animation (1996) -0.178032

```

Baylara pek tanidik gelmeyen bir liste. Simdi erkekler ve hanimlar begeni farkini hesaplayalim ve en buyuk farklar en ustte olacak sekilde siralama (sort) yapalim,

```

mean_ratings['diff'] = mean_ratings['M'] - mean_ratings['F']
sorted_by_diff = mean_ratings.sort_index(by='diff')
sorted_by_diff[:6]

```

| gender | F | M | diff |
|---------------------------|----------|----------|-----------|
| title | | | |
| Dirty Dancing (1987) | 3.790378 | 2.959596 | -0.830782 |
| Jumpin' Jack Flash (1986) | 3.254717 | 2.578358 | -0.676359 |
| Grease (1978) | 3.975265 | 3.367041 | -0.608224 |
| Little Women (1994) | 3.870588 | 3.321739 | -0.548849 |
| Steel Magnolias (1989) | 3.901734 | 3.365957 | -0.535777 |
| Anastasia (1997) | 3.800000 | 3.281609 | -0.518391 |

Dirty Dancing, *Grease* gibi romantik filmler ustte cikti. Simdi listeyi ters cevirelim ve en alta bakalim, orada baylarin en cok hanimlarin en az sevdigi filmler olmalı,

```

sorted_by_diff[::-1][:15]

```

| gender | F | M | diff |
|--|----------|----------|----------|
| title | | | |
| Good, The Bad and The Ugly, The (1966) | 3.494949 | 4.221300 | 0.726351 |
| Kentucky Fried Movie, The (1977) | 2.878788 | 3.555147 | 0.676359 |
| Dumb & Dumber (1994) | 2.697987 | 3.336595 | 0.638608 |
| Longest Day, The (1962) | 3.411765 | 4.031447 | 0.619682 |
| Cable Guy, The (1996) | 2.250000 | 2.863787 | 0.613787 |
| Evil Dead II (Dead By Dawn) (1987) | 3.297297 | 3.909283 | 0.611985 |
| Hidden, The (1987) | 3.137931 | 3.745098 | 0.607167 |
| Rocky III (1982) | 2.361702 | 2.943503 | 0.581801 |
| Caddyshack (1980) | 3.396135 | 3.969737 | 0.573602 |
| For a Few Dollars More (1965) | 3.409091 | 3.953795 | 0.544704 |
| Porky's (1981) | 2.296875 | 2.836364 | 0.539489 |
| Animal House (1978) | 3.628906 | 4.167192 | 0.538286 |
| Exorcist, The (1973) | 3.537634 | 4.067239 | 0.529605 |
| Fright Night (1985) | 2.973684 | 3.500000 | 0.526316 |
| Barb Wire (1996) | 1.585366 | 2.100386 | 0.515020 |

Burada da *Good, The Bad and The Ugly* gibi kovboy filmleri, ve buna benzer vurdulu kirdili filmler ya da enseye tokat turunden *Aptal ve Daha Aptal (Dumb & Dumber)* gibi filmler cikti. Ilginc bir analiz oldu. :)

Burada takip edilen mantiga, ve onun nasil Pandas islemlerina cevirdigine dikkat. “X grubunun en cok ama Y grubunun en az” turunde bir sorgu bir aritmetik fark hesabina cevirdi ve bir grup icin onemli olan kalemelerin en ustte, digeri icin en onemli olanin en altta olacagi akil edildi (en altta eksi degerler vardi tabii ki, bunun sebebini iyi dusunelim) ve sonuca varildi.

Yapay Ogrenim engin bir alandir, ama regresyon, siniflama gibi islemlerden once hala yapilabilecek ilginç ve onemli, ustteki gibi veri analizler var.

Kaynak

McKinney, W., Python for Data Analysis