

به نام خدا

دانشگاه تبریز

دانشکده علوم کامپیوتر

گروه علوم کامپیوتر

پروژه کارشناسی:

Market Basket Optimization

استاد راهنما:

دکتر فرناز ماهان

دانشجو:

فاطمه محمدی

نیم سال دوم تحصیلی 1401

فهرست مطالب

صفحه	عنوان
3	مقدمه
5	Association Rule Learning
6	Types of Association Rule Learning
7	Apriori Algorithm
8	Support
9	Confidence
10	Lift
11	Main Code
15	Dataset

مقدمه

با پیشرفت روزافزون تکنولوژی، نیاز مبرم انسان به الگوریتم ها و برنامه هایی که به منظور راحتی بیشتر و صرفه جویی در زمان و انرژی برای کار با داده های بزرگ بدون دخالت مستقیم اپراتور، مورد استفاده قرار میگیرند بیشتر و بیشتر شد و مباحث اولیه و بنیادین یادگیری ماشین به وجود آمد.

یک تعریف غیر رسمی برای این مفاهیم در سال 1959 توسط آرتور ساموئل، دانشمند علوم کامپیوتر، به فرم زیر ارائه شد:

"The field of study that gives computers the ability to learn without being explicitly programmed"

Arthur Samuel

بعد ها تعریفی رسمی تر و آکادمیک تر و مورد قبول تر برای مفهوم یادگیری ماشین که تمامی جوانب آن را بیان میکند به فرم زیر بیان شد:

"A computer program is said to learn from experience E with respects to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P improves with experience E"

Machine Learning definition

در واقع یادگیری ماشین زیرمجموعه ای از یک مفهوم بسیار گسترده تر به نام هوش مصنوعی AI می باشد که با تحلیل داده ها در مقیاس گسترده و ساختن و پیدا کردن الگوها به منظور پیشبینی وقایع به کار میرود.

متداول ترین تقسیم بندی در مباحث مختلف یادگیری ماشین براساس نوع داده ها و نحوه در اختیار الگوریتم قرار دادن آن ها توسط انسان انجام می گیرد که با توجه به این موضوع 3 تقسیم بندی رایج داریم:

1. یادگیری بانظارت (Supervised Learning)
2. یادگیری بدون نظارت (Unsupervised Learning)
3. یادگیری تقویتی (Reinforcement Learning)
4. سیستم های پیشنهاد دهنده (Recommender Systems)

تفاوت در این الگوریتم ها در نوع و مقدار داده هایی است که در اختیار الگوریتم توسط برنامه نویس قرار داده میشود و همچنین نوع خروجی مورد نظر در این پروژه تمرکز ما بر روی سیستم های پیشنهاد دهنده می باشد.

Association Rule Learning

تمرکز این نوع یادگیری بر روی یافتن ارتباط و قوانین بین محصولات می باشد.

این نوع یادگیری از زیرمجموعه های سیستم های پیشنهاد دهنده می باشد که در ابعاد وسیع تر در پلتفرم هایی همچون IMDB, Netflix, Spotify, Apple music و.... کاربرد دارد.

هدف از این نوع یادگیری این است که ماشین بتواند متوجه ساختار و قوانین بین کالاها بشود تا بتواند بهترین و مرتبط ترین کالاها و محصولات را به کاربران پیشنهاد دهد.

این نوع یادگیری با استفاده از دو نوع الگوریتم مختلف پیاده سازی می شود که در ادامه بررسی خواهیم کرد.



People who bought also bought

Types of Association Rule Learning

به طور کلی دو الگوریتم مختلف برای پیاده سازی مدل گفته شده وجود دارد که مبنای اصلی این الگوریتم‌ها بر اساس سه مفهوم کلی به نام های:

1. پشتیبانی (Support)
2. نرخ اطمینان (Confidence)
3. نرخ پیشرفت (Lift)

می باشد.

این الگوریتم‌ها عبارتند از:

1. Apriori Model

2. Eclat Model

که در این پروژه از مدل Apriori استفاده شده است.

مدل Apriori

این الگوریتم به صورت تکرار شونده (Iterative) می‌باشد که در هر مرحله مقدار Support, Confidence و Lift برای هر زیرمجموعه از کالاها محاسبه شده و با مقادیر قبلی این متغیرها مقایسه می‌شود و در صورتی که مقدار بیشتری داشته باشد جایگزین می‌شود.

در نهایت محصولات و قوانین بین آنها بر اساس مقادیر Lift محاسبه شده، به صورت نزولی محاسبه می‌شود.

فلوچارت الگوریتم Apriori:

Apriori - Algorithm

Step 1: Set a minimum support and confidence



Step 2: Take all the subsets in transactions having higher support than minimum support



Step 3: Take all the rules of these subsets having higher confidence than minimum confidence



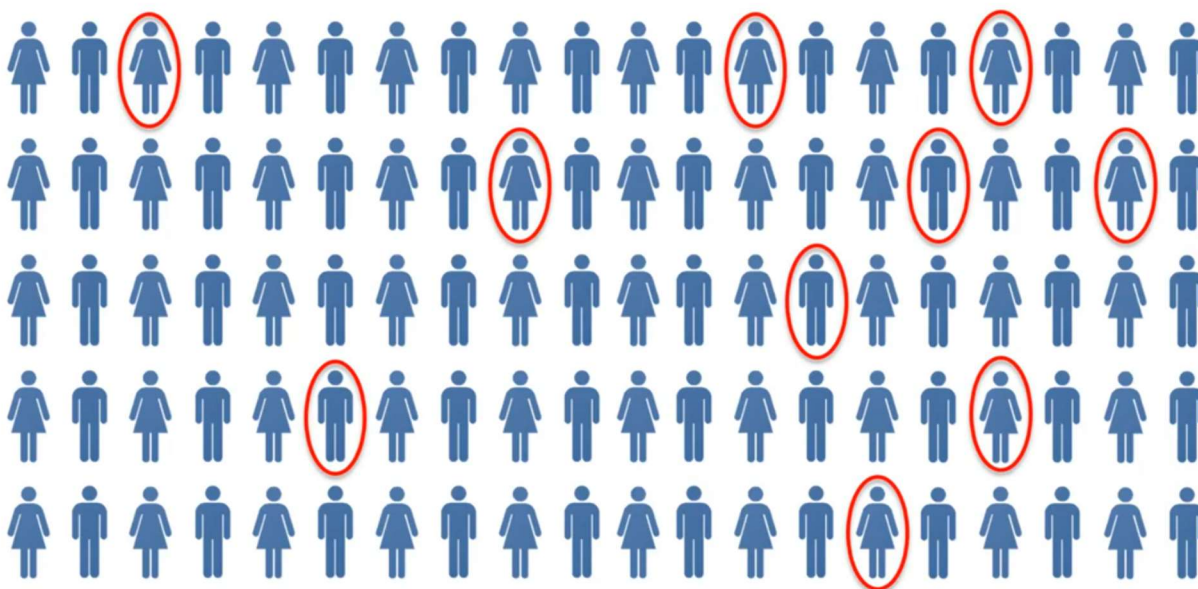
Step 4: Sort the rules by decreasing lift

نرخ پشتیبانی (Support)

نرخ پشتیبانی به ازای هر محصول یا کالا محاسبه می‌شود.

نرخ پشتیبانی برای هر محصول برابر است با نسبت تعداد دفعات خریداری شده‌ی آن محصول توسط کاربران مختلف به تعداد کل کاربران.

$$\text{Market Basket Optimisation: } \text{support}(I) = \frac{\# \text{ transactions containing } I}{\# \text{ transactions}}$$



اگر از مجموعه‌ی بالا کاربرانی که دور آنها دایره قرمز کشیده شده است، کاربرانی باشند

که محصول I را خریداری کرده باشند، آنگاه نرخ پشتیبانی محصول I برابر است با :

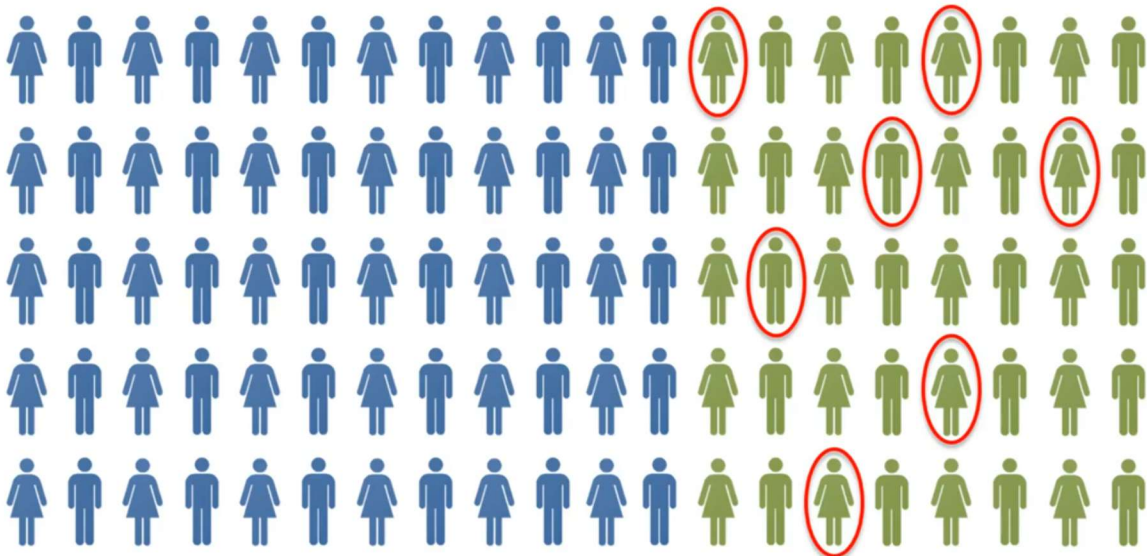
$$\frac{10}{100} = 0.10 = 10\%$$

نرخ اطمینان (Confidence)

نرخ اطمینان به ازای هر زیرمجموعه از مجموعه کالاها و محصولات محاسبه می‌شود و هدف از محاسبه این نرخ، کاهش مجموعه اصلی (کاربران) به منظور افزایش میزان پشتیبانی هر محصول است تا با محاسبه نرخ اطمینان، به ارتباطی بین محصولات پی ببریم و با افزایش این نرخ به ارتباطات و قوانین مشخصی بین محصولات برسیم.

$$\text{Market Basket Optimisation: } \text{confidence}(l_1 \rightarrow l_2) = \frac{\# \text{ transactions containing } l_1 \text{ and } l_2}{\# \text{ transactions containing } l_1}$$

نرخ اطمینان بین محصول l_1 و l_2 برابرست با: $\frac{(I_1 \cap I_2)}{I_1}$



نرخ اطمینان بین l_1 و l_2 برابرست با: $\frac{7}{40} = 17.5\%$

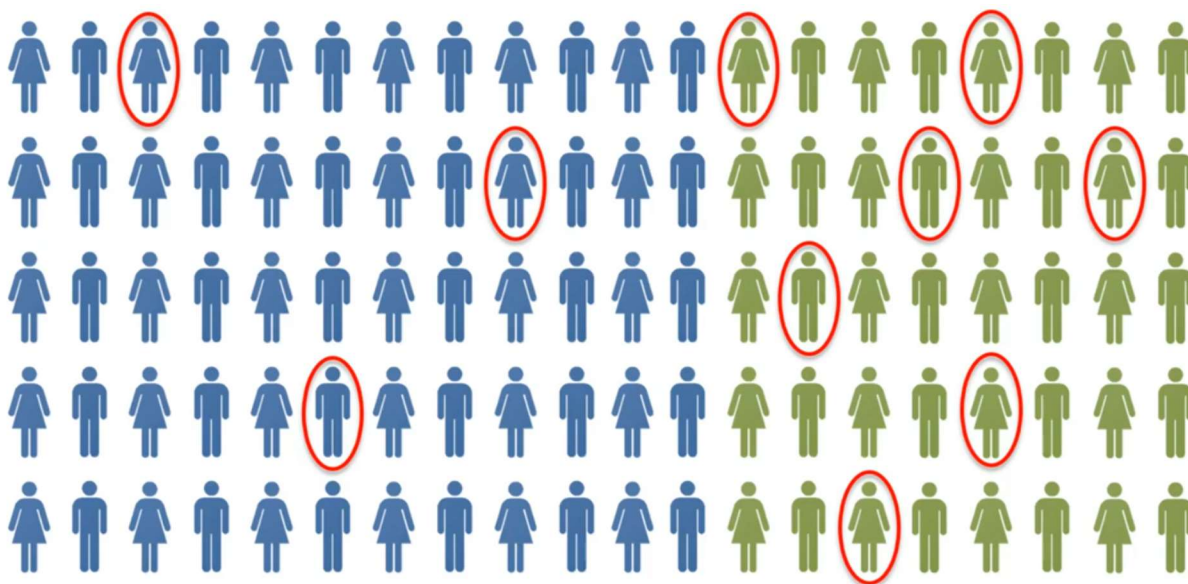
نرخ پیشرفت (Lift)

نرخ پیشرفت به ازای هر زیرمجموعه از مجموعه کالاها محاسبه می‌شود و هدف از محاسبه این مقدار، سنجش میزان قدرت و درستی قانون و ارتباط یافت شده بین دو یا چند محصول می‌باشد.

نرخ پیشرفت برای هر محصول برابرست با نسبت نرخ اطمینان بین آن محصول و محصولات دیگر (دیگر زیرمجموعه‌ها) به نرخ پشتیبانی آن محصول.

Market Basket Optimisation:

$$\text{lift}(I_1 \rightarrow I_2) = \frac{\text{confidence}(I_1 \rightarrow I_2)}{\text{support}(I_2)}$$



$$\frac{17.5}{10} = 1.75$$

نرخ پیشرفت برای محصول I_1 و I_2 برابرست با: 1.75

و این عبارت یعنی ما توانستیم پشتیبانی I_1 را با ایجاد قانونی بین I_1 و I_2 به اندازه 1.75 درصد افزایش بدهیم.

Main Code

کد اصلی برای پیاده سازی این الگوریتم از بخش های زیر تشکیل شده است که در ادامه توضیح داده خواهند شد.

1. فراخوانی کتابخانه ها
2. وارد کردن دیتاست
3. فراخوانی تابع Apriori
4. نمایش قوانین بدست آمده

فراخوانی کتابخانه ها

در این بخش 3 کتابخانه کلیدی و مهم در مباحث یادگیری ماشین را فراخوانی میکنیم که این سه کتابخانه عبارتند از:

1. Apyori
2. Numpy
3. Pandas
4. Matplotlib.pyplot

Importing The Libraries

```
[ ] pip install apyori
```

```
[ ] import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
```

وارد کردن دیتاست

در این بخش از کد دیتاست اصلی که مجموعه ای از تمامی خرید های کاربران در طول یک هفته است با استفاده از کتابخانه pandas در قالب یک فایل CSV به برنامه داده می شود تا مدل Apriori بر روی این دیتاست فیت شود.

Importing The Dataset

```
[ ] dataset = pd.read_csv('Market_Basket_Optimization', header=None)
    transactions = []
    for i in range(7501):
        transactions.append([str(dataset.values[i, j]) for j in range(20)])
```

فیت کردن مدل Apriori

در این بخش از کد، مدل Apriori بر روی دیتاست ما که لیستی از تمام تراکنش های انجام شده توسط کاربران در طول هفته است، اعمال می شود تا بیشترین مقادیر پشتیبانی (Support)، اطمینان (Confidence) و پیشرفت (Lift) محاسبه شود و براساس آن بهترین قوانین و ارتباط موجود بین محصولات شناسایی شود.

▼ Training The Apriori Model On The Dataset

```
[ ] from apyori import apriori
    rules = apriori(transactions=transactions, min_support=0.003, min_confidence=0.2,
                    min_lift=3, min_length=2, max_length=2)
```

مینیموم و ماکسیمم طول قوانین در این تابع توسط min_length و max_length مشخص می شوند که در این مثال مقدار 2 به آنها داده شده است تا قوانین مطلوب بین کالاها دو کلمه ای باشد.

به طور مثال : Buy one ... Get one for free

همچنین مقادیر کمینه نرخ پشتیبانی، نرخ اطمینان و نرخ پیشرفت به طور کاملاً شهودی با توجه به صورت مسئله انتخاب شده و توسط `min_support`، `min_confidence` و `min_lift` به تابع `apriori` از کتابخانه `apyori` داده شده‌اند.

به طور مثال، اگر یک کالا در روز در سه تراکنش متفاوت وجود داشته باشد، در طول یک هفته در 21 تراکنش مختلف حضور خواهد داشت لذا کمینه نرخ پشتیبانی برای آن کالا

$$\frac{21}{7500} \cong 0.003 \quad \text{برابر است با :}$$

دیگر مقادیر اولیه نیز تماماً به صورت شهودی با توجه به صورت مسئله انتخاب می‌شوند.

نمایش قوانین

در این بخش از کد، تمامی قوانین شناسایی شده بین محصولات در قالب یک `Pandas DataFrame` نمایش داده می‌شوند.

لازم به ذکر است که تکه کد مطلوب برای نمایش قوانین خارج از چهارچوب مسئله اصلی می‌باشد لذا به صورت آماده از سایت <https://stackoverflow.com> برداشته شده‌است.

▼ Putting The Results Well Organized Into A Pandas Dataframe

```
[ ] def inspect(results):
    lhs      = [tuple(result[2][0][0])[0] for result in results]
    rhs      = [tuple(result[2][0][1])[0] for result in results]

    supports  = [result[1] for result in results]
    confidences = [result[2][0][2] for result in results]
    lifts     = [result[2][0][3] for result in results]
    return list(zip(lhs, rhs, supports, confidences, lifts))
resultsinDataFrame = pd.DataFrame(inspect(results), columns = ['Left Hand Side', 'Right Hand Side', 'Support', 'Confidence', 'Lift'])
```

نمایش قوانین به صورت مرتب شده

در این بخش از کد تمامی قوانین یافت شده بین محصولات با توجه به مقدار نرخ پیشرفت (Lift) به صورت نزولی مرتب شده و نمایش داده می‌شوند.

▼ Displaying The Results Sorted By Descending Lifts

```
resultsinDataFrame.nlargest(n=10, columns='Lift')
```

	Left Hand Side	Right Hand Side	Support	Confidence	Lift
3	fromage blanc	honey	0.003333	0.245098	5.164271
0	light cream	chicken	0.004533	0.290598	4.843951
2	pasta	escalope	0.005866	0.372881	4.700812
8	pasta	shrimp	0.005066	0.322034	4.506672
7	whole wheat pasta	olive oil	0.007999	0.271493	4.122410
5	tomato sauce	ground beef	0.005333	0.377358	3.840659
1	mushroom cream sauce	escalope	0.005733	0.300699	3.790833
4	herb & pepper	ground beef	0.015998	0.323450	3.291994
6	light cream	olive oil	0.003200	0.205128	3.114710

همانطور که از جدول پیداست، بیشترین ارتباط بین دو محصول Fromage Blanc و Honey با نرخ پیشرفت 5.164271 وجود دارد که جزو ترکیبات بسیار محبوب فرانسوی در وعده های صبحانه و عصرانه و به ویژه به عنوان پیش غذا می‌باشد.

دیتاست

دیتاست این پروژه شامل 7500 تراکنش مختلف دارای محصولات متفاوت یک هایپرمارکت در جنوب فرانسه در طول یک هفته می باشد که برخی از این تراکنش ها به شکل زیر می باشند.

100	frozen vegetables	whole wheat pasta	ground beef	spaghetti	chocolate	green tea
101	mineral water	barbecue sauce	chocolate			
102	burgers	herb & pepper	energy bar	almonds	eggs	com
103	turkey	burgers	ground beef	chocolate	soup	almonds
104	ham	spaghetti	chocolate	eggs		
105	ground beef	energy bar	pet food	carrots	protein bar	
106	ground beef	tomato sauce	spaghetti	mineral water	almonds	eggs
107	mineral water	olive oil	gums	cooking oil		
108	shrimp	pasta	mineral water	soup	avocado	milk
109	shrimp	pasta	soup	cake	cooking oil	chicken
110	spaghetti	mineral water	chocolate	french fries	champagne	escalope
111	shrimp	pasta	mineral water	eggs		
112	burgers	oil	tomato juice	fresh bread		
113	french wine	eggs	chocolate	low fat yogurt		
114	chicken	eggs	french fries	pancakes		
115	turkey	herb & pepper	salmon	white wine	candy bars	
116	grated cheese	spaghetti	olive oil	eggs	french fries	green tea
117	mineral water	energy bar				
118	eggs	french fries				
119	burgers	tomatoes	mineral water	soup	milk	almonds