

به نام خدا

گزارش پروژه ۱

اعضا گروه

امیر سرتیپی

مهدی مالوردی

## Contents

۳	..... فایل های برنامه
۳	..... sql_manager.py
۳	..... excel_manager.py
۳	..... q2.py
۴	..... q1.py
۴	..... q3.py
۵	..... نتایج بدست آمده

## sql\_manager.py

به منظور افزایش سرعت در مراحل بعدی تصمیم بر تبدیل فایل Excel به یک فایل SQL گرفته شد.  
در این فایل متدهای

۱. excel\_to\_SQL

این متد وظیفه تبدیل فایل excel را به فایل SQL دارد.

که خوب دیتاهای outlier مانند مقدار فیلد Description که مقادیری همچون Manual, Discount, Damaged و یا خالی دارند را به table های پایگاه داده اضافه نمی‌کند.

۲. create\_database

وظیفه ساختن دیتابیس با ستون‌هایی که در فایل اکسل Project1-groceries هست را برعهده دارد.

## excel\_manager.py

برای مدیریت فایل‌های اکسل ساخته شده است.

۱. delet\_excel

برای حذف یک فایل اکسل به کار می‌رود.

۲. get\_create\_excel

برای ساخت یک فایل اکسل استفاده می‌شود.

۳. create\_sheet

برای ایجاد شیت‌های متفاوت در یک فایل اکسل استفاده می‌شود.

۴. add\_row

برای ایجاد سطر در فایل اکسل موجود استفاده می‌شود.

۵. add\_rows

برای افزودن چندین سطر به یک فایل اکسل موجود استفاده می‌شود.

۶. read\_rows

برای خواندن سطرهای یک فایل اکسل موجود استفاده می‌شود.

## q2.py

پیاده سازی الگوریتم Apriori که از طریق بدست آوردن مجموعه کاندید و سپس بدست آوردن مجموع بزرگ از طریق مقدار کمینه باعث پیدا کردن آیتم‌های پرتکرار می‌شود. این کلاس برای برآورده کردن خواسته سوال ۲ نوشته شده است

a. Apriori: تابع اصلی که در ابتدا  $L[1]$  تولید میشود و  $C[k]$  ها را به بررسی می‌کند و  $L[k]$  را تولید می‌کند.

b. apriori\_gen: مسئولیت تولید  $L[k]$  را بر عهده دارد.

c. has\_infrequent\_subset: مسئولیت بررسی پرتکرار بودن زیر مجموعه‌ها را بر عهده دارد.

خروجی این فایل یک فایل اکسل است که هر شیت آن شامل اسم با min\_sup مربوطه و در آن زمان و large item ها ذخیره شده است.

q1.py

a. find\_transactions

این متد خواسته سوال ۱ را برآورده می کند و در انتها خروجی را در یک فایل اکسل در پوشه Out به اسم می نویسد.

b. find\_items\_count

این متد خواسته سوال ۲ را برآورده می کند و از اقلام داده های موجود در مجموعه داده را به همراه تعداد تکرار آن به دست می آورد و در قالب عکس هایی به صورت نمودار که در فایل Out قرار دارد آن را نمایش می دهد.

خروجی این فایل برنامه موارد زیر است:

• k:q1\_item\_frequency\_10 پرفروش ترین اقلام بر روی نمودار ( sort شده)

q3.py

برای استخراج قوانین به کار می رود.

۱. make\_subset

برای ساختن زیر مجموعه چپ و راست قوانین به کار می رود و در possible\_rules لیست هایی را اضافه می کند که شامل دو عضو است که با این کار سمت راست و چپ قوانین ساخته می شود.

۲. Write\_strongs

این تابع برای ساختن خروجی اکسل از قوانین استفاده می شود که شامل ۵ ستون است.

- ستون ۱: سمت چپ قانون ها
- ستون ۲: سمت راست قانون ها
- ستون ۳: مقدار lift قوانین
- ستون ۴: مقدار confidence قانون
- ستون ۵: خود قانون

۳. که همه ی موارد ذکر شده از تابع generate\_strongs به دست آورده شده است.

۴. read\_frequent

برای بدست آوردن frequent item set ها استفاده می شود. به این صورت که از فایل اکسلی که در frequent ها با min\_sup مشخص و جود دارند، سطرهای آن را استخراج کرده و در صورتی که ستونی از آن None بود آن را اضافه نمی کند.

۵. make\_rule

در این تابع به تولید قوانین و محاسبه مقدار Confidence و Lift می پردازیم

۶. prepare\_data

براساس make\_subset خروجی این تابع اطلاعات مورد نیاز برای محاسبه قوانین را به دست می آوریم.

برای بدست آوردن قواعد بهتر و قوی تر لازم است تا مقدار **confidence** را بالا ببریم و هرچه این مقدار بیشتر شود قوانین کمتری برای ما تولید خواهد شد. قوانین با **confidence** های مختلف در فایل **excel** قابل مشاهده می باشد. با توجه به رخداد فروش متوجه می شویم که لابیات، سبزیجات از اقلام اصلی سبد خرید بوده است چون به مقدار بسیار زیادی از آن ها خریداری شده است. قواعدی بدست آمد که برای مثال کسی که همبرگر می خرد، ژامبون نیز خریداری می کند.

[لینک گیت هاب پروژه](#)