|  |
| --- |
| به نام خداوند جان و خرد  پروژه شماره 1 (فصل 4) - قوانین همبستگی  نام درس: مبانی داده کاوی  استاد درس: دکتر رضا رمضانی  حل تمرین‌ها: محمدرضا توکلیان، نگین آبادانی، علی عابدزاده، سپیده صالح  مهلت تحویل: اعلام خواهد شد  سامانه تحویل: lms.ui.ac.ir |

توجه:

* لطفا تمرین‌ها را قبل از ددلاین‌ها تحویل دهید. تاخیر ارسال تمرین بعد از پایان یافتن مهلت تحویل، به ازای هر ۲۴ ساعت تاخیر ۱۰ درصد کاهش نمره از نمره‌ی تمرین را دارد.
* در صورت دیده شدن تقلب و کپی، نمره‌ی هر دو طرف صفر در نظر گرفته خواهد شد.

در این پروژه به شما مجموعه‌ داده‌ای از اطلاعات فروش یک فروشگاه داده شده است. هدف پروژه، پیاده‌سازی الگوریتم Apriori است. برای این منظور، مجموعه داده یک فروشگاه آنلاین در اختیار شما قرار گرفته است. این دیتاست حاوی حدود ۹۵۰۰ تراکنش است که هر سطر این دیتاست نشان دهنده محتویات یک خرید توسط مشتریان است.

۱)‌ در گام اول این پروژه، شما باید یک Exploratory Analysis از دیتاست داده شده نمایش دهید. برای اینکار طبق تحلیل خود می‌توانید نمودارهایی که نشان دهنده اطلاعاتی درباره دیتاست است را رسم کنید. (مانند پرفروش‌ترین اجناس فروشگاه و ...)‌

۲) در دومین گام این پروژه شما باید الگوریتم Apriori را پیاده‌سازی کنید. دقت کنید که الگوریتم Apriori از مرتبه حافظه بالایی برخوردار است و شما در پیاده سازی خود باید حتما به گونه‌ای از ساختمان داده و ابزارهایی استفاده کنید که با مشکل دچار نشوید. در اینجا شما باید یک کلاس به نام Arules بنویسید که بدنه‌ی این کتابخانه در پایین آمده است.

class Arules:  
 def *\_\_init\_\_*(*self*):  
 ...  
  
 def get\_frequent\_item\_sets(*self*,  
 *transactions*,  
 *min\_support*,  
 *min\_confidence*):  
 ...  
  
 def get\_arules(*self*,*min\_support*=None,  
 *min\_confidence*=None,  
 *min\_lift*=None,  
 *sort\_by*='lift'):  
 *# sort\_by: lift , confidence, support* ...

متد get\_frequent\_item\_sets باید بتواند دیتاست داده شده را گرفته و با اعمال الگوریتم Apriori انتخابی، لیستی از مجموعه‌های پرتکرار را پیدا کند. متد get\_arules نیز نتایج بدست آمده در متد قبلی را به قانون تبدیل کرده و بر اساس support , confidence و lift این قوانین را مرتب‌سازی می‌کند. خروجی متد get\_frequent\_item\_sets و get\_arules جواب گام ۳ و ۴ شما خواهد بود.

۳) در گام سوم شما باید با اعمال الگوریتم پیاده‌سازی شده روی این دیتاست، با min\_support= 0.005 و min\_confidence= 0.2، ۱۰ مجموعه‌ی پرتکرار را نمایش دهید.

۴) از مجموعه‌های بدست آمده در مرحله سوم قوانین مورد نیاز را تولید کنید و معیار lift را برای آنها محاسبه کرده و به صورت نزولی بر اساس این معیار مرتب کنید.

۵) مرحله سوم و چهارم برای ارزیابی درستی پیاده‌سازی شما در نظر گرفته شده بود. اما در گام پنجم شما به عنوان یک تحلیلگر با تغییر مقدار min support و min confidence و محاسبه دوباره مقادیر lift , support , confidence در قوانین بدست آمده هرگونه تحلیل جدید و ابتکاری را بر روی این تراکنش‌ها انجام دهید.

نکات مهم

* شما می‌توانید از هر زبان برنامه‌نویسی‌ای برای پیاده‌سازی این پروژه استفاده کنید. اما پیشنهاد ما استفاده از زبان پایتون است که بتوانیم ارزیابی بهتری روی کار شما داشته باشیم.
* سعی کنید نتایج خود را تحلیل دقیق کنید. با تغییر مقادیر support و conf ممکن است به نتایج متفاوتی برسید.
* مصورسازی‌های خود را به صورت زیبا و شکیل ارائه دهید و هرگونه اطلاعاتی که می‌توان از دیتاست بدست آورد را استخراج کنید.
* در صورت تمایل می‌توانید از زبان R برای مصورسازی قوانین پایانی خود استفاده کنید (نمی‌توانید از آن برای محاسبه و اجرای الگوریتم Apriori استفاده کنید)
* به سریع‌ترین اجرای کد در میان تمامی کدها در زبان پایتون نمره اضافه دوم تعلق خواهد گرفت. (دقت کنید که شما می‌توانید کد را به هرزبانی پیاده‌سازی کنید و در نهایت با نوشتن یک wrapper کتابخانه نوشته شده توسط خودتان را با پایتون اجرا کنید)
* لطفا در داکیومنت نهایی خود نتایج تحلیل خود را بنویسید. توضیح کد و پیاده‌سازی به صورت شفاهی خواهد بود.