COMS E6111- Advanced Database Systems

Project 3 README

A. Team Members

a. Aishwarya Sivakumar - as6418
b. Sairam Haribabu - sh4188

B. Files

FILE	USAGE
main.py	main file which takes user inputS, finds all the candidate pairs from the dataset, the association rules from candidate pairs and writes required output to output.txt
INTEGRATED-DATASET.csv	File containing the integrated dataset
output.txt	File containing sample run output

C. Instructions for running the program

python3 main.py <*dataset filename*> <minimum support> <*minimum confidence*>

D. CSV Description

a. Data Set Used - NYPD arrest data for the year 2021 from NYC Open Data Set

https://data.cityofnewyork.us/Public-Safety/NYPD-Arrest-Data-Year-to-Date-/uip8-fykc

b. High Level Procedure

- The given dataset contains 19 attributes some of which are arrest_key, jurisdiction code, arrest_date, offense, arrest_boro, arrest_precinct, perp information, coordinates, latitude, longitude etc..
- We chose only interesting columns from the original dataset. The INTEGRATED-DATASET.csv contains the following attributes

[arrest_date, ofns_desc, arrest_boro, arrest_precinct, age_group, perp_sex, perp_race]

- The arrest_date column contains date, month and year details. We modified this and extracted only the month and added that as a new attribute.
- The arrest_boro contained values [Q, M, B, K, S]. We changed this to [Queens, Manhattan, Bronx, Brooklyn and Staten Island] to not conflict with the M value in perp_sex.
- The dataset was pretty huge and had over 155000 rows. So we removed rows which have 'UNKNOWN' or '' values for any of the chosen 7 attributes.
- Final dataset contains 7 attributes = [arrest_month, ofns_desc, arrest_boro, arrest_precinct, age_group, perp_sex, perp_race] and ~ 154000 rows

c. Justification of data set choice -

- The dataset is a breakdown of every arrest in NYC by the NYPD during the current year. Each row includes wholesome information on the arrest including offense description, specific new york borough, perp details such as age, race, sex and the precinct which arrested.
- The dataset contains over 150K market baskets with each having 7 items.
- We chose this dataset knowing that NYC is a major hub for crime and arrests. We wanted to dive deeper into these statistics and obtain association rules which gives us a more granular level of information about these crimes, specifically around the types of crimes and the boroughs where they most occur, the perp's demographic, etc.
- The offense description contains all major categories of over 60 crimes. Some of them are burglary, assault 3 & related offenses, robbery, sex crimes, vehicle and traffic laws etc, felony, murder, non-legal manslaughter etc.
- Association rules revealing hidden information about boroughs would be more interesting than the latitude and longitude. Hence the former was chosen as the prime location attribute. The dataset includes all 5 boroughs of New York.
- The precinct information could also reveal valuable information as to which precinct area within the borough has the most crimes.

E. Internal Design

Main.py is the main logic of the whole program. The file takes care of processing user inputs, executing apriori, finding rules and writing all required outputs to the output file.

The file contains several methods and helper methods. Their functionalities are as follows.

Major Methods

- main - Takes user input, Calls methods to read and load csv file, aid in execution of apriori algorithm, get association rules and write results
- apriori_algorithm - This method takes the support threshold value as input. The while loop calls get_frequent_itemsets over multiple passes with previous pass result as input. The identified item sets are added to the result set. The method terminates when no new itemset is identified in a said iteration.
- get_frequent_itemsets - The main method which traverses previous pass itemsets, data set, identifies itemsets and their supports. The method has three blocks. If size of itemset is 1, it is just a direct pass over whole data counting support values. If size of itemset is 2, then all ordered combinations of previous pass itemsets are traversed to find pairs with above threshold support values. For all other sizes of itemsets to be found, the apriori algorithm defined in the paper requires following rules to be satisfied.
    - Candidate pair of size k can be formed from two (k-1) pairs with same values upto k-2
    - The k-1 th item in pairs being combined should be in ascending order.
    - Current pass itemsets should be pruned based on (k-1) size subset existence in previous pass itemsets.
- prune_itemsets - It checks whether all k-1 size subsets of every new itemset is already found before. If not it deletes the new itemset.
- get_association_rules - This method makes use of rule helper methods to get all plausible association rules and finalize on ones which have above threshold confidence. If the association rule has already been found, the itemset having highest support is retained.

Helper Methods

- read_load_csv() - Reads the csv data file and returns list of headers and data rows.
- write_rules() - Writes the association rules found to the output.txt file with the confidence and support
- write_itemsets() - Writes the itemsets found with their support
- get_all_combinations() - Returns all possible subsets of a set

- get_rule_pairs() - Returns the different combinations of rules from an itemset with the condition that RHS has one item and LHS has one or more and intersection of RHS and LHS is none.

F. Sample Run and Compelling Results

python3 main.py INTEGRATED-DATASET.csv 0.08 0.2

We set the minimum support to 8% and minimum confidence to 20%. This sample run gives us 64 high frequency candidate sets and 79 association rules with high confidence.

Insights: Disclaimer needed?

- The candidate set with the highest support of 82.88% is [‘M’] indicating that major arrests are of male perpetrators.
- The top 5 candidate sets are ['M'], ['25-44'], ['BLACK'], ['25-44', 'M'], ['BLACK', 'M'] with support values ranging from 82% to 41%.
- Months from Feb to May do not occur in the high frequency itemsets whereas the other months do have frequency over 8%.

- A major chunk of rules identified contain ‘M’ as the RHS indicating most arrested gender was that of Male.
- One of the rules with the highest confidence is : ['25-44', 'Brooklyn'] => ['M'] with confidence : 83% and support 12%. This implies that the highest percentage of crimes occur in Brooklyn with the perpetrators being Male and in the 25-44 age group. Comparing with the similar association rules like ['25-44', 'Queens'] => ['M'] (Conf: 82.8378 %, Supp: 10.1139 %), and ['25-44', 'Manhattan'] => ['M'] (Conf: 82.0852 %, Supp: 11.8947 %) we see that these arrests are more common in Brooklyn over Queens over Manhattan.
- ['Brooklyn'] => ['M'] (Conf: 83.3877 %, Supp: 22.2973 %), ['Manhattan'] => ['M'] (Conf: 83.1606 %, Supp: 21.1999 %) , ['Queens'] => ['M'] (Conf: 83.0092 %, Supp: 17.6434 %). From these 3 rules, we can infer that a higher proportion of crimes by males happen in Brooklyn than any other borough.
- Our dataset contains the precinct column. But, no rules were mined based on the precinct. This could mean that crime is widespread throughout the city and there is no particular precinct with an abnormally high proportion of crimes.
- No rules pertain to patterns of arrest related to month indication existence of crime year around.

- The occurrence of age categories <18 and [‘18-24’] in the rules on either side is rare and comforting.
- A good number of rules also indicate ‘ASSAULT 3 & RELATED OFFENSES’ as the offense description. These rules also derive ‘M’, ‘25-44’ and ‘BLACK’ as the perp description.
- A number of rules like ['M', 'WHITE HISPANIC'] => ['25-44'] , ['ASSAULT 3 & RELATED OFFENSES'] => ['25-44'] tell us that when most of these crimes occur, the perpetrator happens to be in the 25-44 Age group. This is also true for ['F'] => ['25-44'], which tells us that when a crime is committed by a female, it is most often from this age group.