GENERATING BEST COMBINATION OF PASSENGERS FOR RIDE SHARING SYSTEM

Problem Statement

Problem statement

Everyday hundreds of traveller's travels on the same route with different destination points in a particular interval of time and use private sharing transportation. Let's assume that you have to go from Bangalore to Mysore thrice in a week for office work and you are going alone in your vehicle, and now, the option has been given to you that you can share your ride with some other travellers who wants to travel the same route. And you as a vehicle owner agrees to share your ride with some other travellers. The question that arises now is that how will you decide who are the two or three other travellers that you should choose? Our project starts from here onwards as through our project we will give the vehicle owner the best frequent travellers who are more likely to go with each other. We are using Apriori algorithm for solving this problem and forming association rules that will give the top 5 combination of passengers who are travelling with each other.

Business Need Assessment & Target Specification -

World is manoeuvring towards adopting sustainable environmental friendly technologies. Globally we have started harnessing renewable energies and adopting innovative methodologies that will reduce carbon footprints. Our project is also in the same line as through our project we are contributing towards minimizing carbon footprints and making India a better place. One of the main reasons for carbon footprints are transport sector. Creating decentralised ride sharing system which will be operated by mango people of India and no need of special drivers will be there. The ride sharing services carries huge potential especially in urban cities as it provides high flexibility, low cost, trips of short medium and even interstate distances. According to studies, vehicles annually contribute about 290 gigagrams (Gg) of PM2.5. At the same time, around 8% of total Greenhouse Gas (GHG) Emissions in India are from the transport sector, and in Delhi, it exceeds 30%.

Here are some statistics which made our solution more important, considering todays scenario:

- The transport sector accounts for a quarter of total emissions, out of which road transport accounts for three-quarters of transport emissions (and 15% of total global CO2 emissions).
- Passenger vehicles are the largest chunk of this, releasing about 45% of CO2.
- If the conditions prevail, annual GHG emissions in 2050 will be 90% higher than those of 2020.

Our concept of ride sharing: To tackle the pollution problem we are aiming to build the ecosystem where vehicle owner can share their rides with other passengers who are travelling in the same route as vehicle owner. Here vehicle owner is not special taxi drivers rather here anyone having vehicle and travelling to any route can share his/her ride. Ride sharing will be done through a mobile app where the vehicle owner enters its destination, and he will get top 5 best set of passengers who want to travel along the same route. Best set of passengers here means the combination of passengers who are most likely to travel together.

For finding best set of passengers, we will be using Apriori algorithm. Apriori algorithm is a machine learning algorithm which uses certain metric to find most frequent itemset. Through metrics of Apriori algorithm we are generating rules of association and these rules of association combinedly gives best combination of items from the entire data set

Here are some statistics which made our solution more important, considering todays scenario:

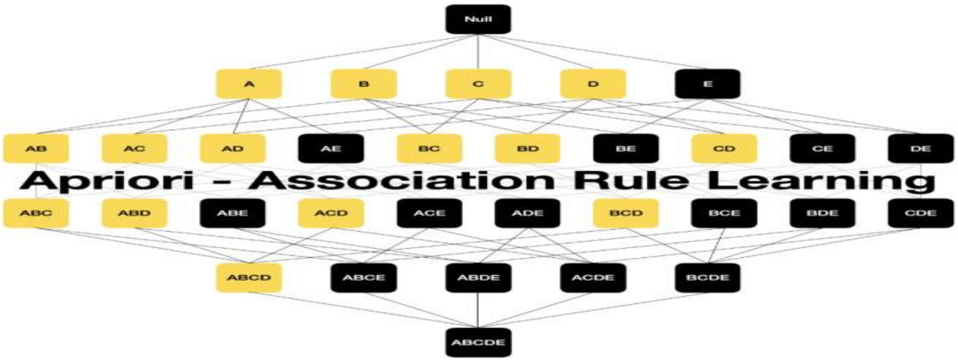
- The transport sector accounts for a quarter of total emissions, out of which road transport accounts for three-quarters of transport emissions (and 15% of total global CO2 emissions).
- Passenger vehicles are the largest chunk of this, releasing about 45% of CO2.
- If the conditions prevail, annual GHG emissions in 2050 will be 90% higher than those of 2020.

Our concept of ride sharing: To tackle the pollution problem we are aiming to build the ecosystem where vehicle owner can share their rides with other passengers who are travelling in the same route as vehicle owner. Here vehicle owner is not special taxi drivers rather here anyone having vehicle and travelling to any route can share his/her ride. Ride sharing will be done through a mobile app where the vehicle owner enters its destination, and he will get top 5 best set of passengers who want to travel along the same route. Best set of passengers here means the combination of passengers who are most likely to travel together.

For finding best set of passengers, we will be using Apriori algorithm. Apriori algorithm is a machine learning algorithm which uses certain metric to find most frequent itemset. Through metrics of Apriori algorithm we are generating rules of association and these rules of association combinedly gives best combination of items from the entire data set

MAIN IDEA: -

Our main idea is to provide the right set of passengers who have travelled with each other more frequently than any other combination of passengers. This will help the vehicle owner to choose the right set of passengers to travel with him/her. As per our problem statement we had done market basket analysis the passengers following the same route for the cost-effective solution by using right combination of the passengers.

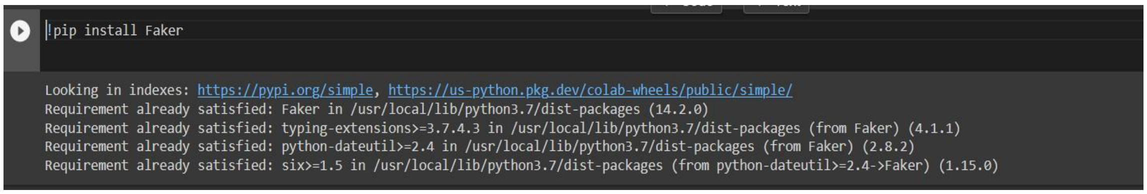


1. PROCESS: -

1.1. Data generation

We generated the dummy data for this project because this is the new concept therefore no past data available for us. We have brainstormed about the type of data that we need to generate which will be exact replica of the originally generated dataset. After lots of trials we came to conclusion that dataset should consist of 3 columns. Each row of all three columns will have unique passengers Ids. It is not necessary to have each row filled with set of three passengers and some rows can also have only two passengers.

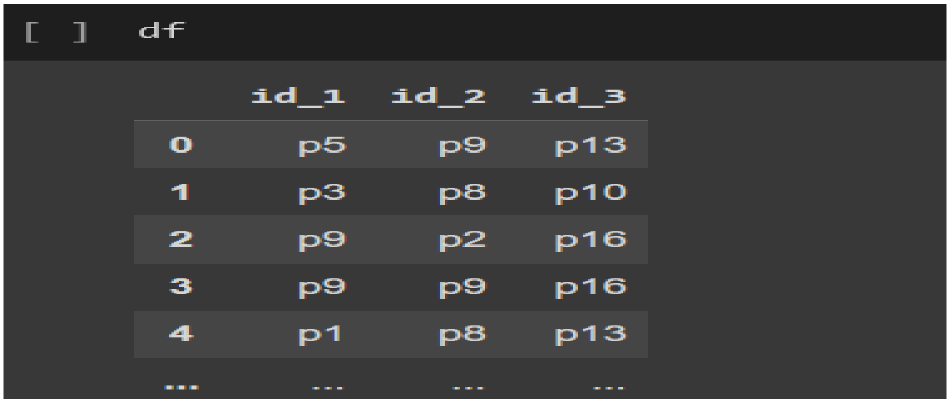
We have generated dataset showing 6 months of data using faker library in Python and subjected that data to the machine learning algorithm.



```
!pip install Faker
```

```
Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-wheels/public/simple/
Requirement already satisfied: Faker in /usr/local/lib/python3.7/dist-packages (14.2.0)
Requirement already satisfied: typing-extensions>=3.7.4.3 in /usr/local/lib/python3.7/dist-packages (from Faker) (4.1.1)
Requirement already satisfied: python-dateutil>=2.4 in /usr/local/lib/python3.7/dist-packages (from Faker) (2.8.2)
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.7/dist-packages (from python-dateutil>=2.4->Faker) (1.15.0)
```

Generated Dataset:



```
[ ]  df

         id_1  id_2  id_3
     0    p5    p9   p13
     1    p3    p8   p10
     2    p9    p2   p16
     3    p9    p9   p16
     4    p1    p8   p13
    ...   ...   ...   ...
```

Algorithm used

Apriori algorithm is used to create most frequently occurring passengers set. Apriori algorithm is used for market basket analysis. Let us look at the common example of grocery store to understand it. In grocery store we have items like bread, butter, milk, eggs, oats, coffee and so on. Each customer comes and purchases some of them and go. Now if we want to find which of the two grocery items are purchased together more frequently then we will use Apriori algorithm to figure out the top combinations of grocery items which are bought by the customers in combination.

Metrics of Apriori

Apriori algorithm find the best combination of items using combination of 3 metrices. These three metrices are Support, Confidence and Lift. Let us understand them one by one.

1. **Support**: It states the frequency of a particular item or itemset occurs in the entire dataset.
   Support = Frequency of item / Total number of items in the dataset

2. **Confidence**: Confidence tells us the frequency of time the combination of items occurs from the number of times any one item of that item set occurs. For example, how many times a customer purchased bread and butter together out of total number of times bread is being purchased.
   Confidence = frequency of (Bread + Butter) / Frequency of (Bread)

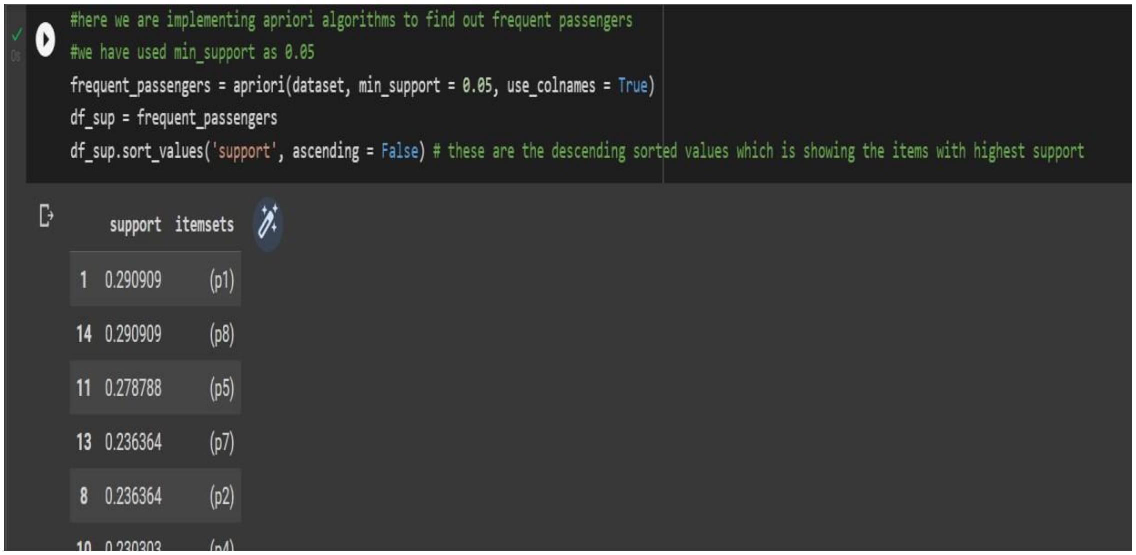
3. **Lift:** Lift metrics tells us about the degree of association between items and itemset. If the value of lift is less than 1 then it means substitute of the item is present in the market. If lift value is equal to 1 then it means no association between items in the itemset. If lift if greater than 1 then it means the items are associated and items in the itemset are more likely to be bought/occur together.

Generating rules of association

Combining support, confidence, and lift metrices we have generated the rules of association to figure out top five combinations of most frequently travelled passengers.

Thresholds of rules of association used in the algorithm are:

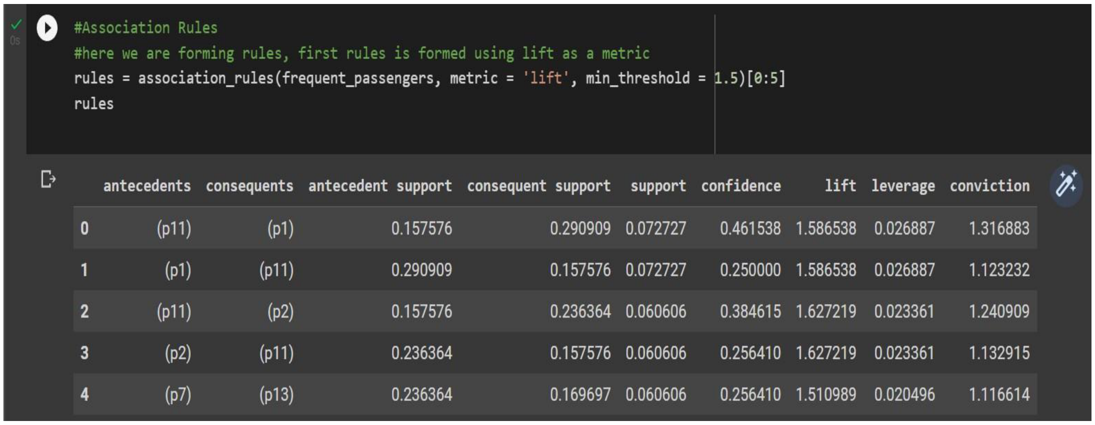
1. **Support**: Minimum support values is set to 5%. It means we are filtering out all the passengers and combination of passengers who have not travelled more than 9 times in 180 days.



```
#here we are implementing apriori algorithms to find out frequent passengers
#we have used min_support as 0.05
frequent_passengers = apriori(dataset, min_support = 0.05, use_colnames = True)
df_sup = frequent_passengers
df_sup.sort_values('support', ascending = False) # these are the descending sorted values which is showing the items with highest support
```

```
    support  itemsets
1   0.290909     (p1)
14  0.290909     (p8)
11  0.278788     (p5)
13  0.236364     (p7)
8   0.236364     (p2)
10  0.230303     (p4)
```

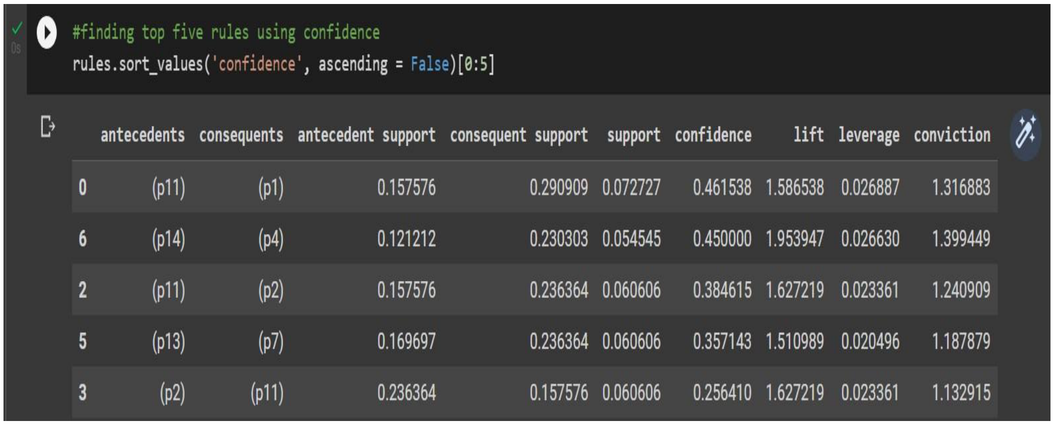
2. **Lift**: We have minimum threshold value for lift is 1.5. It means any combination of passengers who are showing value less than 1.5 will be filtered out.



```
#Association Rules
#here we are forming rules, first rules is formed using lift as a metric
rules = association_rules(frequent_passengers, metric = 'lift', min_threshold = 1.5)[0:5]
rules
```

```
   antecedents  consequents  antecedent support  consequent support   support  confidence      lift  leverage  conviction
0        (p11)         (p1)            0.157576            0.290909  0.072727    0.461538  1.586538  0.026887    1.316883
1         (p1)        (p11)            0.290909            0.157576  0.072727    0.250000  1.586538  0.026887    1.123232
2        (p11)         (p2)            0.157576            0.236364  0.060606    0.384615  1.627219  0.023361    1.240909
3         (p2)        (p11)            0.236364            0.157576  0.060606    0.256410  1.627219  0.023361    1.132915
4         (p7)        (p13)            0.236364            0.169697  0.060606    0.256410  1.510989  0.020496    1.116614
```

3. **Confidence**: We set the confidence threshold value as 20%. It means if 10 times passenger A is travelling then the passenger set (A+B) will be considered if (A+B) travels at least 2 times together.



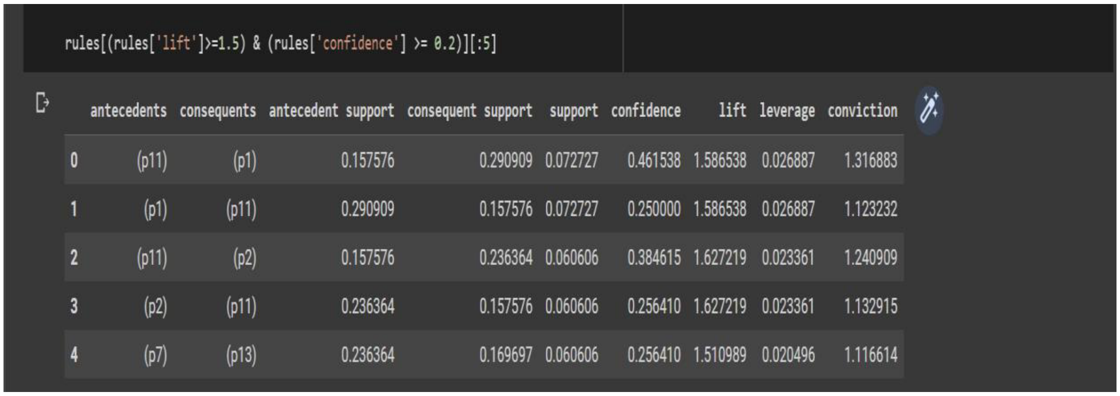
```
#finding top five rules using confidence
rules.sort_values('confidence', ascending = False)[0:5]
```

	antecedents	consequents	antecedent support	consequent support	support	confidence	lift	leverage	conviction
0	(p11)	(p1)	0.157576	0.290909	0.072727	0.461538	1.586538	0.026887	1.316883
6	(p14)	(p4)	0.121212	0.230303	0.054545	0.450000	1.953947	0.026630	1.399449
2	(p11)	(p2)	0.157576	0.236364	0.060606	0.384615	1.627219	0.023361	1.240909
5	(p13)	(p7)	0.169697	0.236364	0.060606	0.357143	1.510989	0.020496	1.187879
3	(p2)	(p11)	0.236364	0.157576	0.060606	0.256410	1.627219	0.023361	1.132915

2. OUTCOME:

We have successfully conducted market basket analysis using Apriori algorithm.Out of 180 (6 months of data) combinations of passengers who are travelling in same route we have found top five combinations of passengers whose probability of travelling with each other is highest using Apriori algorithm

Objective of this algorithm is successfully achieved as we have eliminated the effort which vehicle owner needs to put in selecting right set of passengers for ride sharing. We have displayed top 5 combinations of passengers for the vehicle owner. Now, vehicle owner can select anyone of them.



```
rules[(rules['lift']>=1.5) & (rules['confidence'] >= 0.2)][:5]
```

	antecedents	consequents	antecedent support	consequent support	support	confidence	lift	leverage	conviction
0	(p11)	(p1)	0.157576	0.290909	0.072727	0.461538	1.586538	0.026887	1.316883
1	(p1)	(p11)	0.290909	0.157576	0.072727	0.250000	1.586538	0.026887	1.123232
2	(p11)	(p2)	0.157576	0.236364	0.060606	0.384615	1.627219	0.023361	1.240909
3	(p2)	(p11)	0.236364	0.157576	0.060606	0.256410	1.627219	0.023361	1.132915
4	(p7)	(p13)	0.236364	0.169697	0.060606	0.256410	1.510989	0.020496	1.116614

References:

- Al-Maolegi, M., & Arkok, B. (2014). An improved Apriori algorithm for association rules. *arXiv preprint arXiv:1403.3948*.
- Yuan, X. (2017, March). An improved Apriori algorithm for mining association rules. In *AIP conference proceedings* (Vol. 1820, No. 1, p. 080005). AIP Publishing LLC.
- Perego, R., Orlando, S., & Palmerini, P. (2001, September). Enhancing the apriori algorithm for frequent set counting. In *international conference on data warehousing and knowledge discovery* (pp. 71-82). Springer, Berlin, Heidelberg.
- Ye, Y., & Chiang, C. C. (2006, August). A parallel apriori algorithm for frequent itemsets mining. In *Fourth International Conference on Software Engineering Research, Management and Applications (SERA'06)* (pp. 87-94). IEEE.
- Dongre, J., Prajapati, G. L., & Tokekar, S. V. (2014, February). The role of Apriori algorithm for finding the association rules in Data mining. In *2014 International Conference on Issues and Challenges in Intelligent Computing Techniques (ICICT)* (pp. 657-660). IEEE.