Predicting Coupon Redemption:

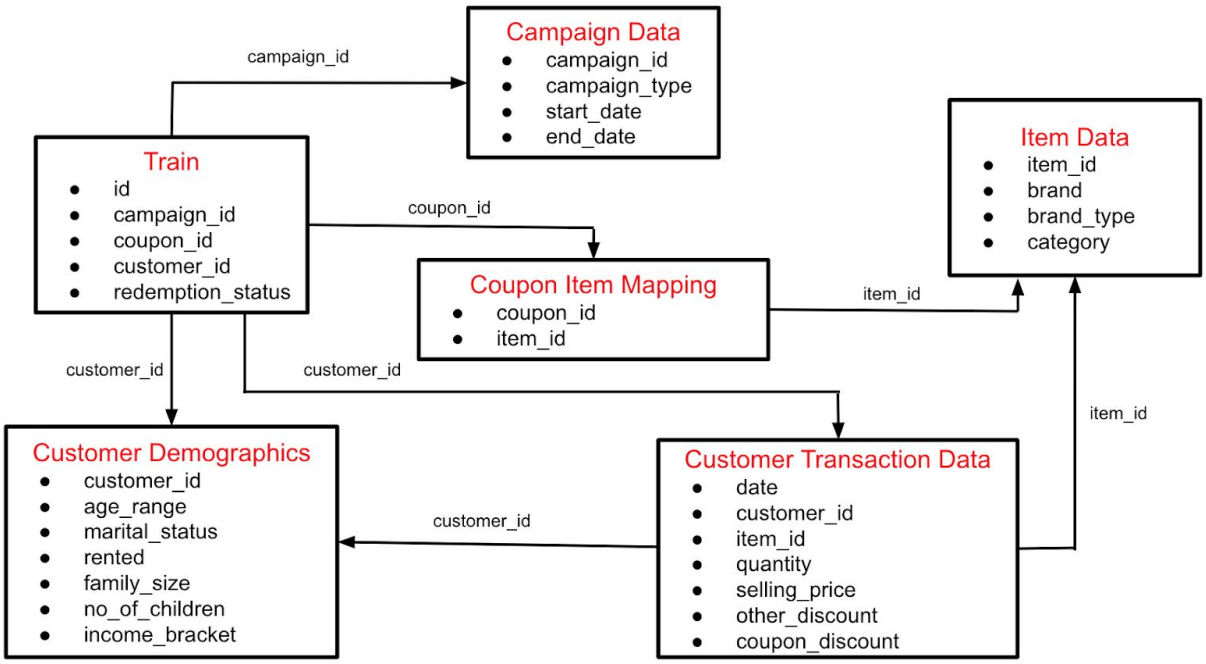
The data available contains the following information, including the details of a sample of campaigns and coupons used in previous campaigns -

User Demographic Details

Campaign and coupon Details

Product details

Previous transactions



On the available data, aggregation, joining the data, adding new columns to get much meaningful results and other operations were performed.

To work on data aggregation, indexing the data for faster retrieval, etc, XSV, Trifacta and Pandas were used.

XSV:

Using XSV, each of the files were indexed and based on the matching columns, joins were performed on the data.

Indexing on the datafile using XSV:

```
C:\DigitalMarketing\Assignment2>xsv index C:\DigitalMarketing\Assignment2\campaign_data.csv

C:\DigitalMarketing\Assignment2>xsv index C:\DigitalMarketing\Assignment2\coupon_item_mapping.csv

C:\DigitalMarketing\Assignment2>xsv index C:\DigitalMarketing\Assignment2\customer_demographics.csv

C:\DigitalMarketing\Assignment2>xsv index C:\DigitalMarketing\Assignment2\customer_transaction_data.csv

C:\DigitalMarketing\Assignment2>xsv index C:\DigitalMarketing\Assignment2\item_data.csv

C:\DigitalMarketing\Assignment2>xsv index C:\DigitalMarketing\Assignment2\test.csv

C:\DigitalMarketing\Assignment2>xsv index C:\DigitalMarketing\Assignment2\train.csv

C:\DigitalMarketing\Assignment2>
```

Headers to find the common field to make a join:

```
C:\DigitalMarketing\Assignment2>xsv headers C:\DigitalMarketing\Assignment2\campaign_data.csv
1   campaign_id
2   campaign_type
3   start_date
4   end_date

C:\DigitalMarketing\Assignment2>xsv headers C:\DigitalMarketing\Assignment2\coupon_item_mapping.csv
1   coupon_id
2   item_id

C:\DigitalMarketing\Assignment2>xsv headers C:\DigitalMarketing\Assignment2\item_data.csv
1   item_id
2   brand
3   brand_type
4   category

C:\DigitalMarketing\Assignment2>xsv headers C:\DigitalMarketing\Assignment2\customer_transaction_data.csv
1   date
2   customer_id
3   item_id
4   quantity
5   selling_price
6   other_discount
7   coupon_discount

C:\DigitalMarketing\Assignment2>xsv headers C:\DigitalMarketing\Assignment2\customer_demographics.csv
1   customer_id
2   age_range
3   marital_status
4   rented
5   family_size
6   no_of_children
7   income_bracket

C:\DigitalMarketing\Assignment2>xsv headers C:\DigitalMarketing\Assignment2\test.csv
```

Stats of each datafile:

```
C:\DigitalMarketing\Assignment2>xsv stats C:\DigitalMarketing\Assignment2\test.csv | xsv table
field        type     sum         min  max     min_length  max_length  mean              stddev
id           Integer  3225514822  3    128594  1           6           64220.02194082747  37115.7632492176
campaign_id  Integer  974970      16   25      2           2           19.41165929996414  2.382041898786121
coupon_id    Integer  29616125    28   1116    2           4           589.6572492334645  312.2395985274364
customer_id  Integer  40803961    1    1582    1           4           812.4071397284281  456.7206191488561

C:\DigitalMarketing\Assignment2>xsv stats C:\DigitalMarketing\Assignment2\train.csv | xsv table
field              type     sum         min  max     min_length  max_length  mean                stddev
id                 Integer  5042886488  1    128595  1           6           64347.97544947614   37126.20398485845
campaign_id        Integer  1095163     1    30      1           2           13.97444142454287   8.019163370045167
coupon_id          Integer  44385321    1    1115    1           4           566.3632431190913   329.963948652818
customer_id        Integer  61711817    1    1582    1           4           787.4518878638236   456.8084244449382
redemption_status  Integer  729         0    1       1           1           0.00930214753282548  0.09599800822987432

C:\DigitalMarketing\Assignment2>xsv stats C:\DigitalMarketing\Assignment2\item_data.csv | xsv table
field       type     sum         min          max              min_length  max_length  mean              stddev
item_id     Integer  2742923211  1            74066            1           5           37033.5           21381.012516950643
brand       Integer  110029491   1            5528             1           4           1485.5600545459456  1537.3752940545194
brand_type  Unicode              Established  Local            5           11
category    Unicode              Alcohol      Vegetables (cut)  4          22

C:\DigitalMarketing\Assignment2>xsv stats C:\DigitalMarketing\Assignment2\coupon_item_mapping.csv | xsv table
field      type     sum         min  max   min_length  max_length  mean               stddev
coupon_id  Integer  14452406    1    1116  1           4           155.96738719877396  282.9901933773678
item_id    Integer  3382997613  1    74061 1           5           36508.61307102087   21131.198693177314
```

Joining the data using XSV:

```
C:\DigitalMarketing\Assignment2>xsv join --left item_id C:\DigitalMarketing\Assignment2\item_data.csv item_id C
:\DigitalMarketing\Assignment2\coupon_item_mapping.csv > coupon_item_final.csv

C:\DigitalMarketing\Assignment2>xsv join --left customer_id C:\DigitalMarketing\Assignment2\coupon_item_final.c
sv item_id C:\DigitalMarketing\Assignment2\customer_demographics.csv > customer_coupon_item.csv
```

Advantages of XSV tool:

- The tool is useful in creating indexes on large datasets to make them easily accessible
- XSV tool is effective in joining large datasets and saving them

The shortcomings of using the XSV tool :

- The tool cannot handle data cleaning process like handling null values, formatting dates etc

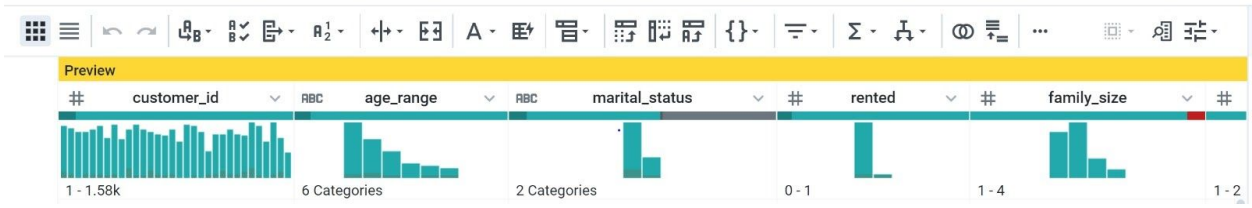
Trifacta

Advantages-

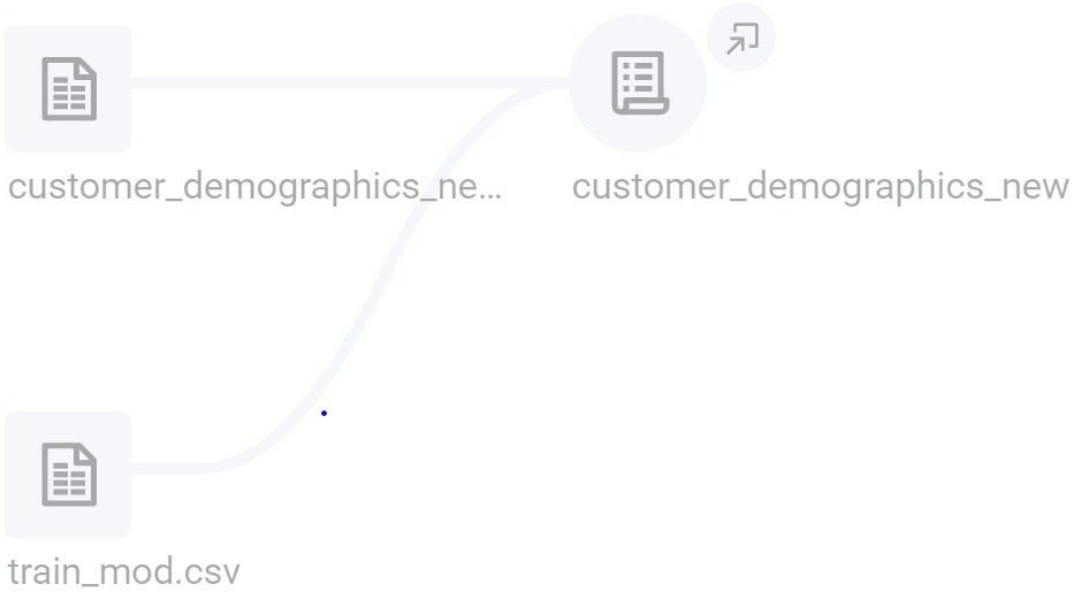
Easy UI and best tool for Data Wrangling

We can perform operations like

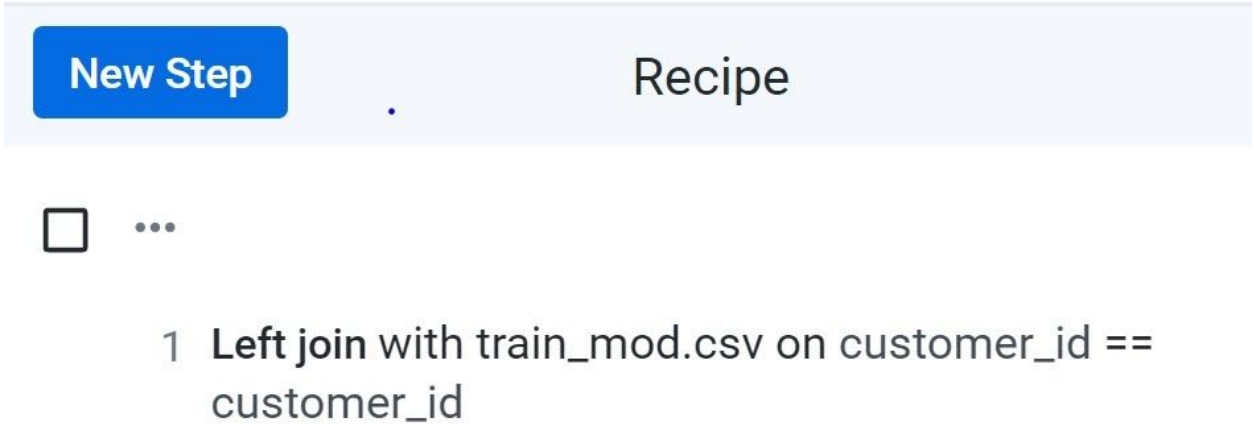
We can view the data as shown below and the missing values are indicated in a different color.



Data Flow- Performing joins



A recipe looks as shown below



In a recipe the joins and other operations performed are mentioned.

So we can apply the recipe to the dataset and Run a job.

Manual errors are less and pretty intuitive. It has a graphical interface and we get a good sense of the data.

Pandas

A package in python used for manipulating data and analyzing it.

Pandas are used

1) When multiple data sets have to be merged with each other, or a reshaping/reordering of data has to be done
2) Import data from or export data to a specific file format like Excel, HDF5 or SQL. Pandas comes with convenient functions for this

For the above data set, we performed these functions using Pandas

1) **Imported the data**
2) **Checked for null values in every dataset**

Only customer demographics had missing values.

```
customer_demographics.isnull().sum()
```

```
customer_id          0
age_range            0
marital_status     329
rented               0
family_size          0
no_of_children     538
income_bracket       0
dtype: int64
```

3) **Can apply formulas and generate meaningful values using lambda**

```
customer_transaction_data['day'] = customer_transaction_data["date"].apply(lambda x: x.day)
customer_transaction_data['dow'] = customer_transaction_data["date"].apply(lambda x: x.weekday())
customer_transaction_data['month'] = customer_transaction_data["date"].apply(lambda x: x.month)
```

```
customer_transaction_data.head()
```

	date	customer_id	item_id	quantity	selling_price	other_discount	coupon_discount	day	dow	month
0	2012-01-02	1501	26830	1	35.26	-10.69	0.0	2	0	1
1	2012-01-02	464	20697	1	92.26	-21.37	-35.62	2	0	1
2	2012-01-02	464	20717	2	28.5	-27.78	0.0	2	0	1
3	2012-01-02	464	21008	1	35.26	-17.81	0.0	2	0	1
4	2012-01-02	464	22243	2	118.97	-22.8	0.0	2	0	1

4) **Can perform all kinds of joins**

```
coupons_items = pd.merge(coupon_item_mapping, item_data, on="item_id", how="left")
```

```
coupons_items.head()
```

	coupon_id	item_id	brand	brand_type	category
0	105	37	56	1	6
1	107	75	56	1	6
2	494	76	209	0	6
3	522	77	278	0	6
4	518	77	278	0	6

Aggregate the customer transaction by 'item_id'

```
In [332]: customer_transaction_data.head()
```

Out[332]:

	date	customer_id	item_id	quantity	selling_price	other_discount	coupon_discount	day	dow	month	coupon_used
0	2012-01-02	1501	26830	1	45.950	-10.69	0.00	2	0	1	0
1	2012-01-02	464	20697	1	113.630	-21.37	-35.62	2	0	1	1
2	2012-01-02	464	20717	2	28.140	-13.89	0.00	2	0	1	0
3	2012-01-02	464	21008	1	53.070	-17.81	0.00	2	0	1	0
4	2012-01-02	464	22243	2	70.885	-11.40	0.00	2	0	1	0

5) Aggregation

6) Cleaning

```
3]: campaign_data['start_date'] = pd.to_datetime(campaign_data['start_date'], format = '%d/%m/%y')
    campaign_data['end_date'] = pd.to_datetime(campaign_data['end_date'], format = '%d/%m/%y')
```

Getting the data in appropriate format

7) We can derive additional columns from existing columns

```
campaign_data["campaign_duration"] = campaign_data["end_date"] - campaign_data["start_date"]
campaign_data["campaign_duration"] = campaign_data["campaign_duration"].apply(lambda x: x.days)
```

```
campaign_data.head()
```

	campaign_id	campaign_type	start_date	end_date	campaign_duration
0	24	1	2013-10-21	2013-12-20	60
1	25	1	2013-10-21	2013-11-22	32
2	20	1	2013-09-07	2013-11-16	70
3	23	1	2013-10-08	2013-11-15	38
4	21	1	2013-09-16	2013-10-18	32

Customer Lifetime value

Retention is a lot cheaper than acquisition. Thus, successful marketers don't focus only on strategies for acquiring new customers. They also work out tactics to retain customers and stimulate them to make more purchases. CLV gives an understanding of your promotion spendings, based on which you can further optimize and plan your budget. What's more, CLV provides useful insights for how to encourage customers to spend more.

Calculated keeping in mind the next 5 years based on transactions past one year.

The ones with a higher CLV score (38 people)are in the Platinum Category and special effort will be made to give them a good shopping experience

Out[43]:

	customer_id	TotalPrice	Total_visits	Total_Profit	Customer_Lifetime_value	CLV_Category
0	1	51737.06	33	0.5	4268	Copper
1	2	24176.29	28	0.8	2707	Copper
2	3	49172.86	29	0.7	4991	Copper
3	4	28690.83	22	0.5	1577	Copper
4	5	56424.57	92	0.7	18168	Copper

```
Shows the number of customers in each Category
```

```
In [47]: ctd_merge['CLV_Category'].value_counts()
```

```
Out[47]: Copper      1375
         Silver       138
         Platinum      38
         Gold          30
         Name: CLV_Category, dtype: int64
```

For eg- The Platinum category can be

Offered special discounts on multiple purchases
Creating a loyalty program (punch or swipe cards are popular)
Offering rewards for new customer referrals
Providing special customer service
Offering preferential credit terms

RFM

1. Recency — number of days since the last purchase
2. Frequency — number of transactions made over a given period
3. Monetary — amount spent over a given period of time

The targeting decisions can be made by selecting a subset of segments from the RFM cube.

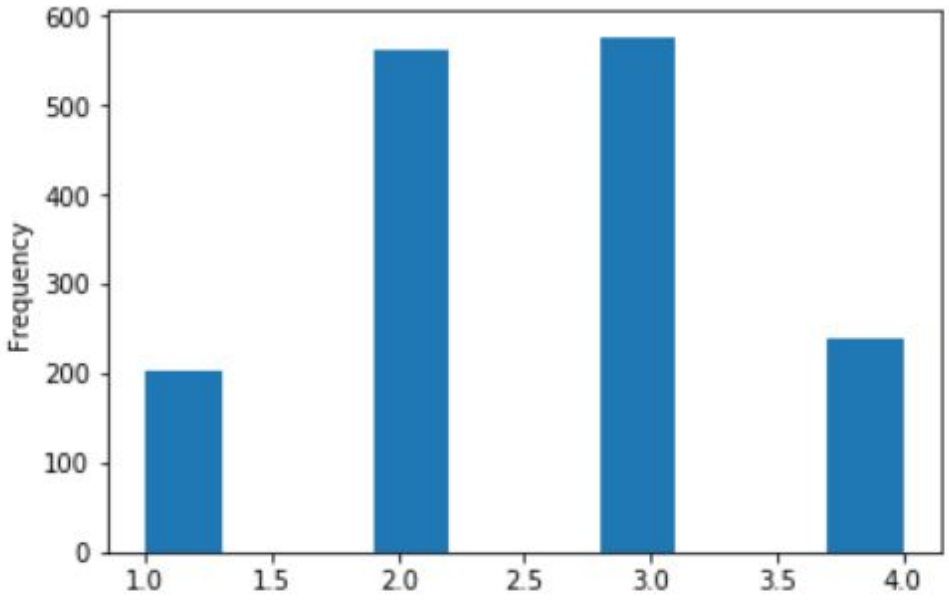
RFM analysis is based on the empirical observation that recency, frequency, and monetary metrics are often correlated with the probability to respond and the lifetime value

Customer Transactions Table - 18 months of data.
We have sliced it to a year of data.

1) Calculate average purchase value- per customer
2) Calculate average purchase frequency rate
3) Calculate customer value- **avg purchase value * frequency**
4) Calculate average customer lifespan - 5 years
5) CLTV = **customer value * customer lifespan**

```
: ctd_merge['rfm_rank'].plot(kind='hist')
```

```
: <matplotlib.axes._subplots.AxesSubplot at 0x1966f96f518>
```



The categories are as below

```
]: ctd_merge['rfm_rank'].value_counts()
```

```
]: 3    577
   2    563
   4    239
   1    202
   Name: rfm_rank, dtype: int64
```

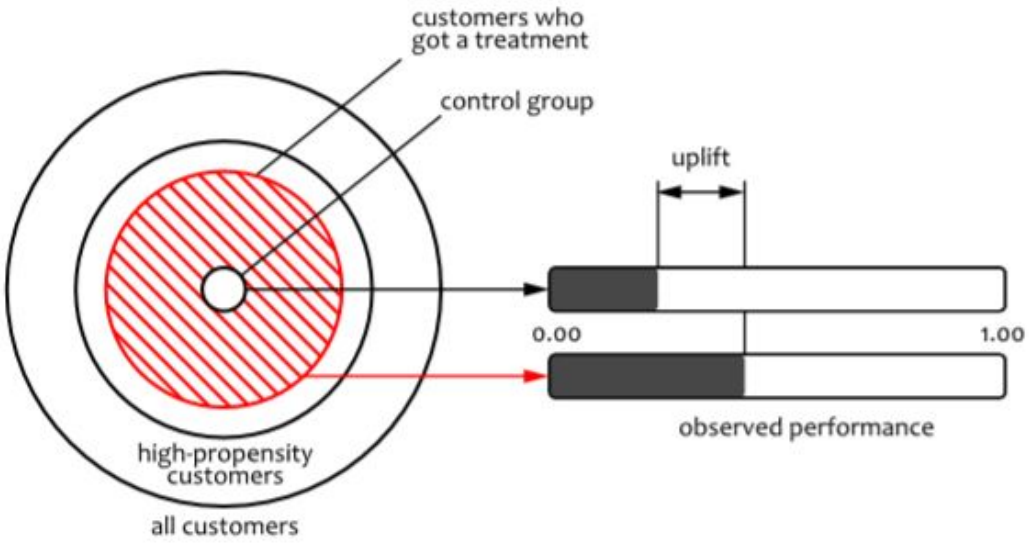
The ones in 4th segment are the best customers in terms of RFM score

Response measurement

The response modeling framework provides a basic tool for response prediction. The counterpart of this framework is a measurement framework that can be used to evaluate the results of a campaign –

Did it make existing customers spend more? - Yes

The standard approach to measure the incremental gains is to compare the performance of two groups of consumers: ones who received the promotion (test group) and ones who did not receive it (control group).



Test group- had 1428 customers who were given coupons

```
cust_coupon= t.customer_id.sort_values().unique().tolist()
print(len(cust_coupon))
```

```
1428
```

The total number of customers= 1582
Control group = 154.

```
cust_transactions= ctd.customer_id.sort_values().unique().tolist()
print(len(cust_transactions))
```

```
1582
```

The ones with flag 0 did not get a coupon.

The ones with flag 1 got a coupon

	flag	No. of Customers	TotalPrice
0	0.0	154	6.929663e+06
1	1.0	1428	1.217963e+08

```
rm_final['Respone_measurement']=rm_final['TotalPrice']/rm_final['No. of Customers']
```

```
rm_final.head()
```

flag	No. of Customers	TotalPrice	Respone_measurement
0.0	154	6.929663e+06	44997.811558
1.0	1428	1.217963e+08	85291.526366

```
response_measurement = (85291.526366/44997.811558)
```

```
response_measurement
```

```
1.8954594326451488
```

Response measurement is calculated as shown above
1.89 is the value and the campaign was effective in bringing up the sales by 89 %

The ones with a coupon had purchased more. So the coupon/campaign helped.

Dashboard:

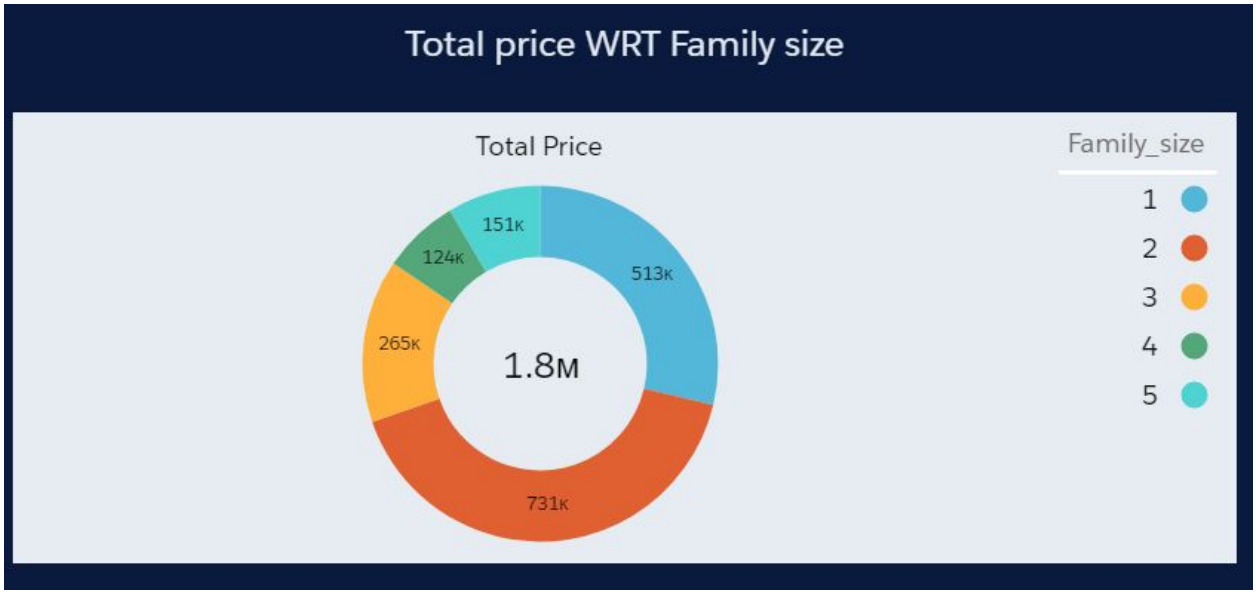
The following visualizations were made based on the data that was cleaned and processed using Pandas, Trifacta and XSV.

Big Spenders in less visit:
Based on the RFM calculation done, the following customers are the ones who have spent more money in lesser visit (freq is less than 50, monetary > 1M)

Number of customers who have recently visited:
Based on the recency information from the RFM data, there have been 272 customers who have visited
Recency factor of 1 indicates that the customer is the most recent and the customer with higher recency factor has not visited the store in the recent times

Total price WRT to family size:

- The total transaction price from the customer transaction amounts upto $1.8M. From the chart, it can be concluded that families with 1 and 2 members are the most contributors
- Larger families tend to spend less as they have other expenditures to spend on



Sales based on marital status:

Marital status of 0 indicates unmarried person and 1 is a married person

- The sales data indicates that unmarried people contribute to the sales more than married people
- The data shows that unmarried people spend up to twice more than married people

Purchase trend in each quarter:

- The number of customers who have purchased in each quarter, it can be seen that Q1 and Q2 are the most active quarters of the year
- More promotions need to be provided in the Q3 and Q4 of the year to increase sale

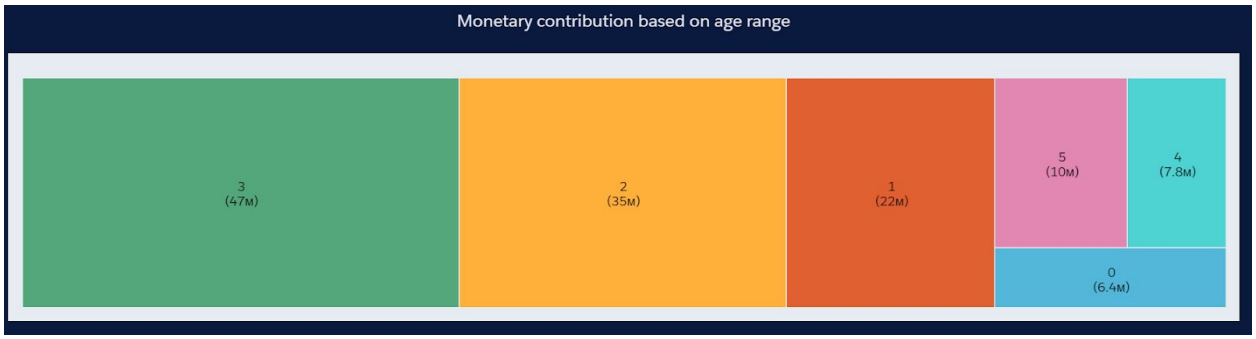
Almost lost customers:

- The almost lost customers are the number of customers who have monetarily contributed more but in the recent times, they have not made any visit to the store (Monetary > 1000,000, Recency >120)

Rarely contributing customers:

- The rarely contributing customers are the number of customers who visit the store very frequently but do not contribute the sales much (Frequency > 200, Monetary < 30k)

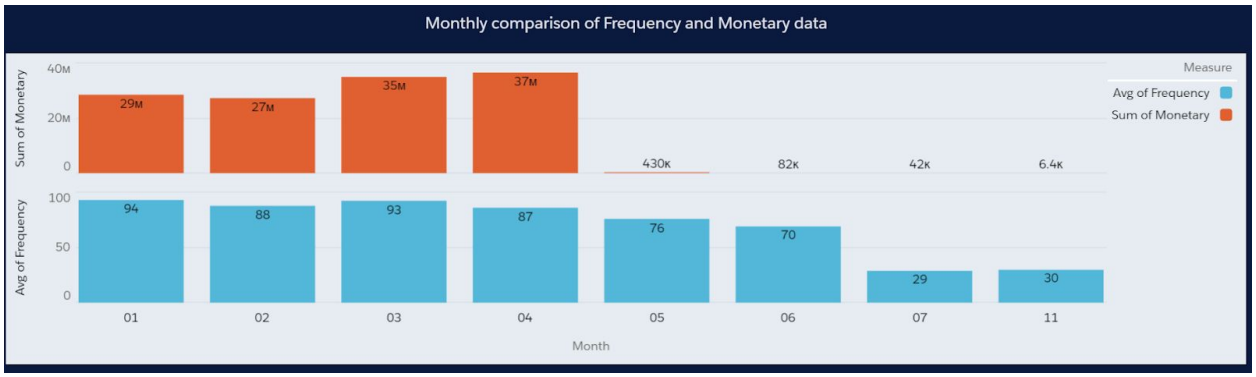
Monetary contribution based on age group:



Customers who need promotion:

Most loyal customers based on RFM score:

Monthly comparison of frequency and monetary data:



Discount amount based on discount type:

Top 20 customers based on coupon discount: