36106: Machine Learning Algorithms & Applications

AT3: Group Project

Group 27:
Michael Yaputra
Somayeh Amraee
Monali Patil
Tahmidul Islam

1. Unsupervised Learning (Michael Yaputra)

1.1 Business Understanding

The importance of data science in the financial industry is hard to be overstated and it is disrupting the sector like never before (Qualetics, 2019). Banks have realized the potential of big data technologies in helping them make smarter decisions, improve performances, utilize and focus their resources efficiently to keep up with the ever growing competition (ActiveWizards).

Banks are sitting on piles of data including their customers' information, purchasing behaviors, transactions, loans and savings (USDSI). Those who are able to harness the power of data will give them a significant edge over their competitors.

Australia is home to around 95 banks and the industry employs over 200,000 people to serve 22.9 million customers (Australian Banking Association, 2023). They play an important role in the financial system. In 2016 alone, the financial sector contributed $140 billion to the nation's GDP, making it the largest sector to contribute to the Australian economy (Australian Government Treasury, 2016).

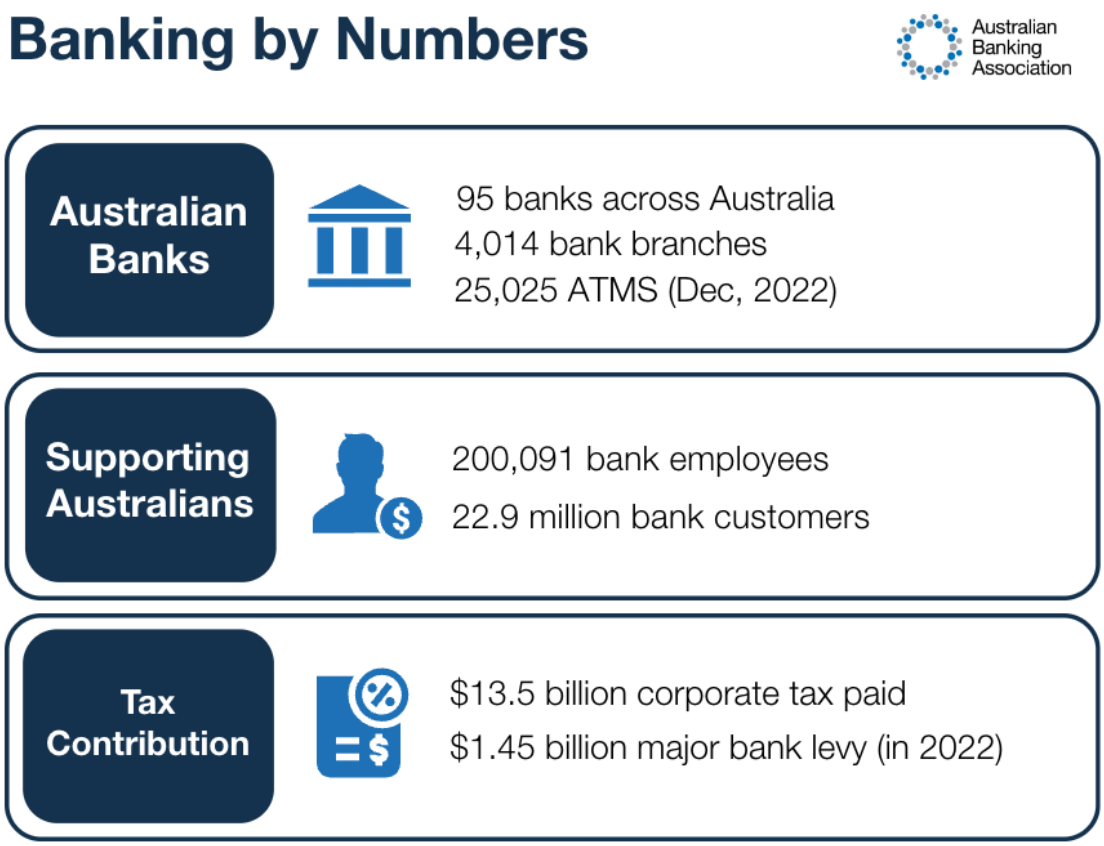


Figure 1: Summary of Australian banking industry (Australian Banking Association, 2023)

They play an important role in the financial system. The 'Big 4' banks account for more than 70% of the market share in retail and business lending (myNZTE, 2022)

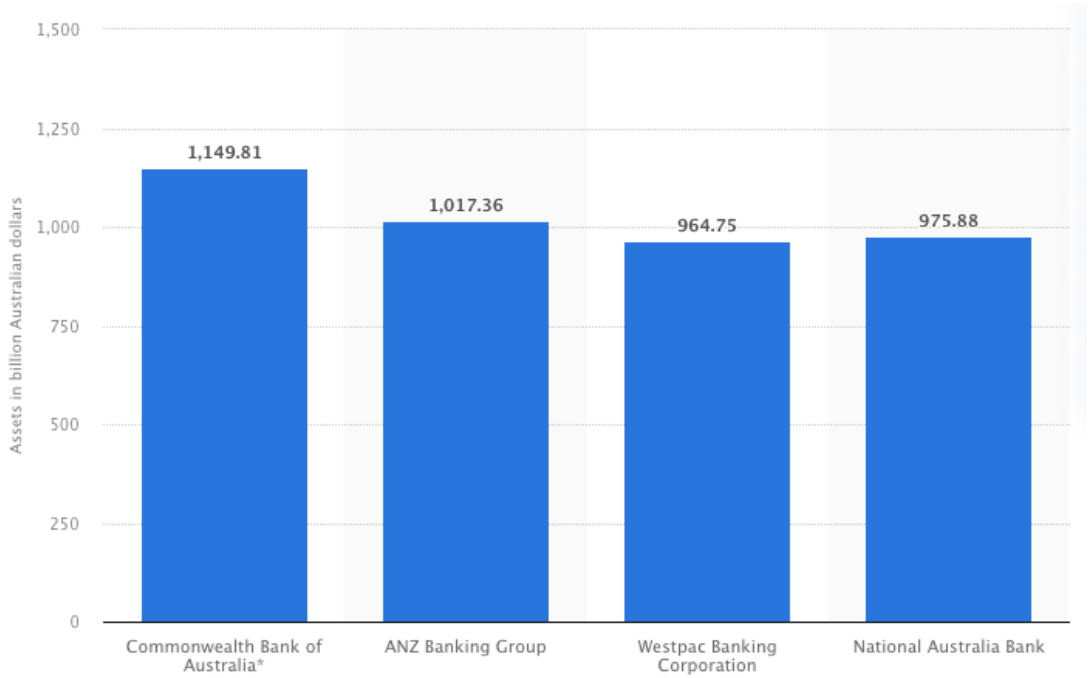


Figure 1: Largest bank in Australia by assets, 2022 (Statista, 2023)

There are several data science use cases in the banking sector including fraud detection, risk modeling, predictive analysis, marketing/sales campaign optimization and process automation. In this project, we will utilize unsupervised technique to help automate 2 processes:

1. **Customer segmentation:** By understanding customers' spending behaviors, the marketing/sales team can segment the customers accordingly. Not only will this help tailor their campaigns to specific segments and improve personalization, but also better engagement and more effective cross/upselling, which will ultimately translate to higher customer satisfaction and retention.

2. **Anomaly detection:** machine learning algorithms can be very useful to help detect anomalies in banking activities and help relevant stakeholders do their jobs, examples include:

   -Intrusion detection or cybersecurity threat detection can help cyber security teams manage and response to these threats in a timely manner

   -Fraud or money laundering activities detection can help compliance teams detect these activities earlier and more accurately to protect the business against the threats posed by these schemes.

Utilizing these machine learning algorithms and techniques can help banks save a lot of time by automating the majority of the processes and detect previously unseen patterns due to the sheer size of the data.

1.2 Data Understanding

SUMMARY

The data provided by the bank consists of 2 categories:

1. Customer dataset containing customer information
2. Transaction datasets containing the list of customers' transactions from 2019 to 2022

CUSTOMER DATASET

This dataset contain 1,000 customer information (1,000 rows) and 15 columns in total:

	ssn	cc_num	first	last	gender	street
0	115-04-4507	4218196001337	Jonathan	Johnson	M	863 Lawrence Valleys
1	715-55-5575	4351161559407816183	Elaine	Fuller	F	310 Kendra Common Apt. 164
2	167-48-5821	4192832764832	Michael	Cameron	M	05641 Robin Port
3	406-83-7518	4238849696532874	Brandon	Williams	M	26916 Carlson Mountain
4	697-93-1877	4514627048281480	Lisa	Hernandez	F	809 Burns Creek
...	...	...	...	...	...	...
995	392-96-7670	30125158904184	Sarah	Martin	F	1666 Jenna Unions
996	594-17-7993	180047909863618	Erin	Wells	F	444 Alexis Estate Suite 824
997	196-93-1156	4371450311809	Michael	James	M	914 Cassandra Gateway Suite 061
998	895-65-9304	3519925692476886	Michael	Lewis	M	34141 Katelyn Path
999	107-40-0160	343251790447085	Isaac	Smith	M	67148 Rose Cliff Apt. 314

1000 rows × 15 columns

Data format: Majority of the columns are integer and float with 1 datetime column

#	Column	Dtype
0	cc_num	int64
1	unix_time	datetime64[ns]
2	category	object
3	amt	float64
4	is_fraud	int64
5	merchant	object
6	merch_lat	float64
7	merch_long	float64
8	year	int64
9	month	int64
10	day	int64
11	hour	int64
12	minute	int64

TRANSACTION DATASETS

The transaction data comes in the form of more than 130 csv files, which we combined into 1 data set for easier analysis. The combined dataset contains 4,260,904 rows and 10 columns

	Unnamed: 0	cc_num	acct_num	trans_num
0	0	675925178633	585807126672	63431ec72fe61163125634f521cb88ca
1	1	675925178633	585807126672	7e8813f38be8c77be3e6e2d09d53538b
2	2	675925178633	585807126672	841f8c81ce9d32263ca3a897bc42a8a0
3	3	675925178633	585807126672	8ef81d22ea82b2cbb40b3e712f84bb42
4	4	675925178633	585807126672	c6b4413112caf11acef957b8faaa6a3b
...	...	...	...	...
4260899	4260899	6520575639836526	665775741850	0a1da5c0324c90f57c9378b106878461
4260900	4260900	6520575639836526	665775741850	32b6f6f07e6139a47e5255d0e22bc508
4260901	4260901	6520575639836526	665775741850	25d9c7d3bee03804f02bc31c2e517d8f
4260902	4260902	6520575639836526	665775741850	718b34b992d8123b3689a9b864d61747
4260903	4260903	6520575639836526	665775741850	39bb05b49878b5fadd75f8d7cdfd07d7

4260904 rows × 11 columns

Data format: Majority are object, float and integer data types

```
 #   Column    Non-Null Count  Dtype
---  ------    --------------  -----
 0   cc_num    1000 non-null   int64
 1   city      1000 non-null   object
 2   state     1000 non-null   object
 3   zip       1000 non-null   int64
 4   lat       1000 non-null   float64
 5   long      1000 non-null   float64
 6   city_pop  1000 non-null   int64
 7   job       1000 non-null   object
 8   dob       1000 non-null   object
```

1.3 Data Preparation

DROPPING IDENTIFIER COLUMNS AND PERSONAL INFORMATION TO MINIMIZE BIAS

There are several features that are too specific, so we dropped them. Also, we did this to prevent bias towards gender and minimize the risk of personal information being exposed.

Dropped columns:

-Customers = 'ssn', 'first', 'last', 'gender', 'street' and 'acct_num'
-Transactions = ‘acct_num’ and ‘trans_num’

MISSING VALUES

There are no missing values in both datasets:

-Customer:

```
cc_num      0
city        0
state       0
zip         0
lat         0
long        0
city_pop    0
job         0
dob         0
dtype: int64
```

-Transaction:

```
cc_num        0
unix_time     0
category      0
amt           0
is_fraud      0
merchant      0
merch_lat     0
merch_long    0
dtype: int64
```

DUPLICATE VALUES

There are no duplicate values in both datasets:

-Customer:

```
customer_df['cc_num'].duplicated().sum()
✓ 0.0s

0
```

-Transaction:

```
transaction_df['trans_num'].duplicated().sum()
✓ 1.0s

0
```

OUTLIERS

For unsupervised techniques: considering we will be performing anomaly detection, we will not be dropping any outliers in the dataset

```
#Transaction
transaction_df.describe()
```

✓ 0.4s

	cc_num	unix_time	amt	is_fraud	merch_lat	merch_long
count	4.260904e+06	4.260904e+06	4.260904e+06	4.260904e+06	4.260904e+06	4.260904e+06
mean	3.916811e+17	1.620228e+09	6.898790e+01	1.181439e-03	3.735276e+01	-9.247610e+01
std	1.267805e+18	3.795228e+07	1.618467e+02	3.435177e-02	5.504630e+00	1.741603e+01
min	6.040027e+10	1.546261e+09	1.000000e+00	0.000000e+00	1.859001e+01	-1.603677e+02
25%	1.800618e+14	1.587482e+09	9.100000e+00	0.000000e+00	3.370138e+01	-9.903072e+01
50%	3.524238e+15	1.626431e+09	4.449000e+01	0.000000e+00	3.819365e+01	-8.727445e+01
75%	4.604409e+15	1.654918e+09	8.158000e+01	0.000000e+00	4.120567e+01	-7.966098e+01
max	4.986227e+18	1.672492e+09	4.130053e+04	1.000000e+00	6.577610e+01	-6.724632e+01

```
#Customer
customer_df.describe()
```

✓ 0.0s

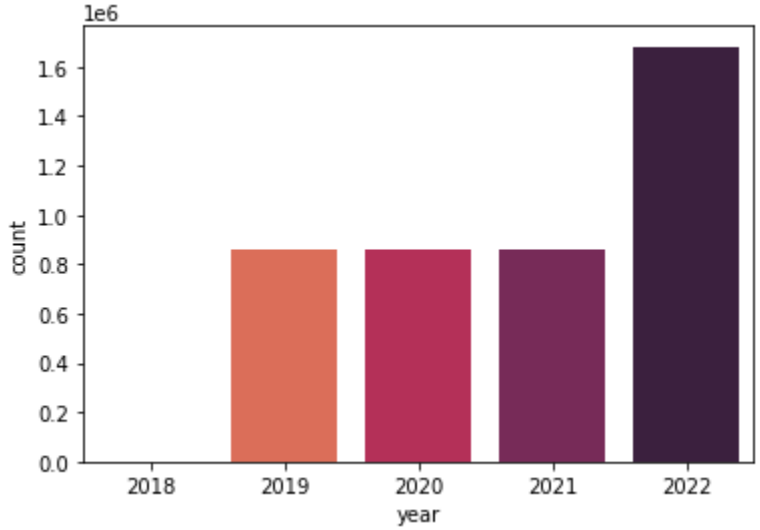
	cc_num	zip	lat	long	city_pop
count	1.000000e+03	1000.00000	1000.000000	1000.000000	1.000000e+03
mean	3.656964e+17	51786.20900	37.422795	-92.505277	2.867058e+05
std	1.227337e+18	29933.13195	5.574397	17.510134	5.329632e+05
min	6.040027e+10	1571.00000	19.589300	-159.368300	1.050000e+02
25%	1.800353e+14	27528.25000	33.712500	-99.159475	2.022950e+04
50%	3.517359e+15	49016.00000	38.431750	-87.156600	6.728250e+04
75%	4.538733e+15	78599.00000	41.220300	-80.018850	2.499788e+05
max	4.986227e+18	99705.00000	64.780500	-68.244800	2.906700e+06

TRANSFORMING 'UNIX_TIME' TO READABLE FORMAT

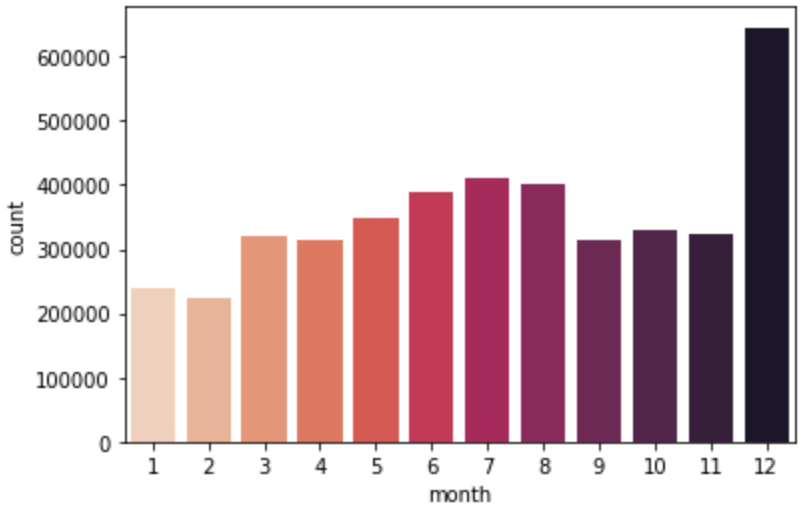
We transformed 'unix_time' column into year, month, day, hour and minute columns:

year	month	day	hour	minute
2022	2	15	17	1
2022	3	12	16	46
2022	3	31	13	1
2022	6	23	21	15
2022	1	13	18	46
...	...	...	...	...
2021	1	30	8	22
2020	5	11	2	51
2022	11	19	8	2
2021	2	10	10	21
2022	2	4	4	23

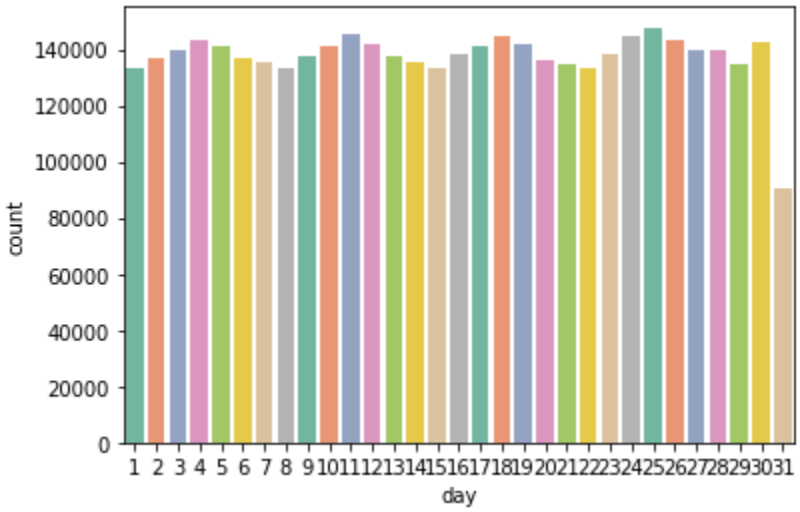
'Year' column: majority of transactions happened in 2022



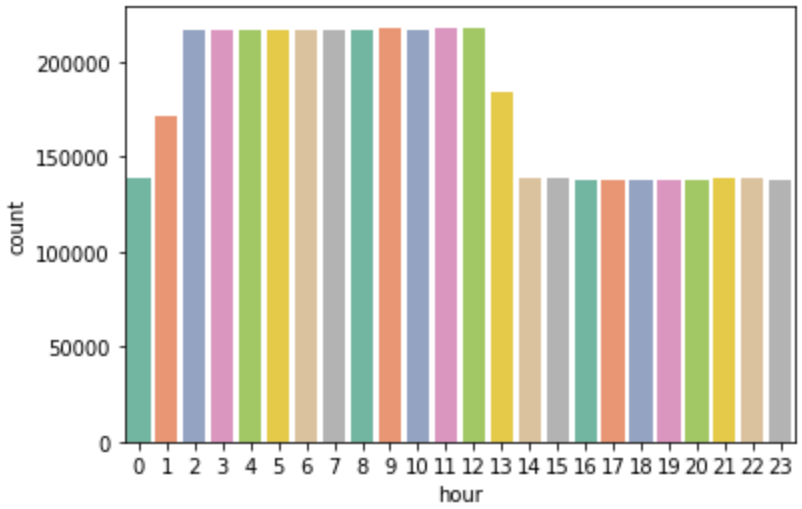
'Month' column: majority of transaction happened in December, could be due to holiday/christmas period



‘Day’ column: Almost evenly distributed except for the 31st which makes sense since not all months have 31st



‘Hour’ column: Majority of transactions happened at and before noon



MERGING CUSTOMER DATASET WITH THEIR TOTAL TRANSACTION AMOUNT

1. Aggregated total transactions and combined them with the customer dataset (all four years 2019-2022) to see how much each customer spent during these periods:

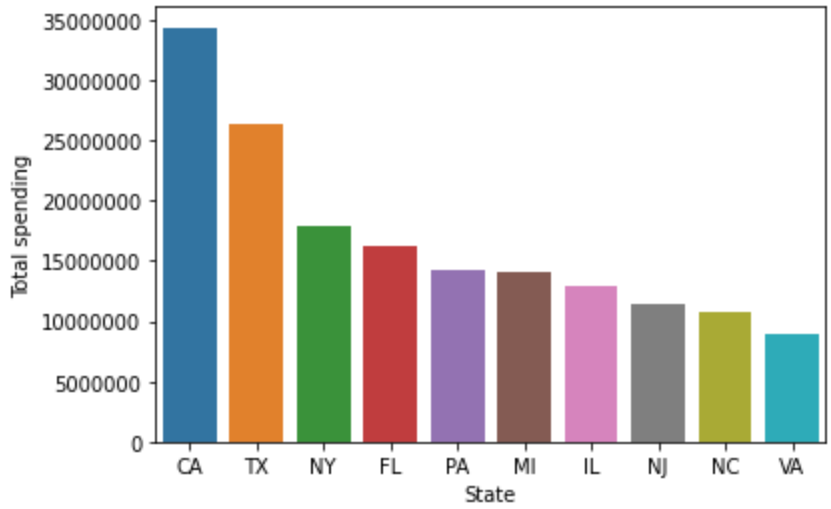
	cc_num	city	state	zip	lat	long	city_pop	job	dob	amt
0	4218196001337	Ambler	PA	19002	40.1809	-75.2156	32412	Accounting technician	1959-10-03	206986.20
1	4351161559407816183	Leland	NC	28451	34.2680	-78.0578	27112	Professor Emeritus	1963-06-07	105741.56
2	4192832764832	Cordova	SC	29039	33.4275	-80.8857	4215	International aid/development worker	1973-05-30	329105.71
3	4238849696532874	Birmingham	AL	35242	33.3813	-86.7046	493806	Seismic interpreter	1942-12-26	180341.52
4	4514627048281480	Fargo	GA	31631	30.7166	-82.5801	559	Medical laboratory scientific officer	1939-05-22	395370.93
...	...	...	...	...	...	...	...	...	...	...
995	30125158904184	Denver	CO	80236	39.6535	-105.0376	990452	Colour technologist	1993-08-31	551053.30
996	180047909863618	Wasco	CA	93280	35.6480	-119.4487	27152	Software engineer	1982-05-27	584600.74
997	4371450311809	Escondido	CA	92026	33.1605	-117.0978	171802	Agricultural consultant	1999-05-14	254961.37
998	3519925692476886	Mattapan	MA	2126	42.2739	-71.0939	25562	Civil Service administrator	1994-02-23	129984.35
999	343251790447085	Saint Petersburg	FL	33710	27.7898	-82.7243	341043	Manufacturing systems engineer	1942-07-21	350440.28

1000 rows × 10 columns

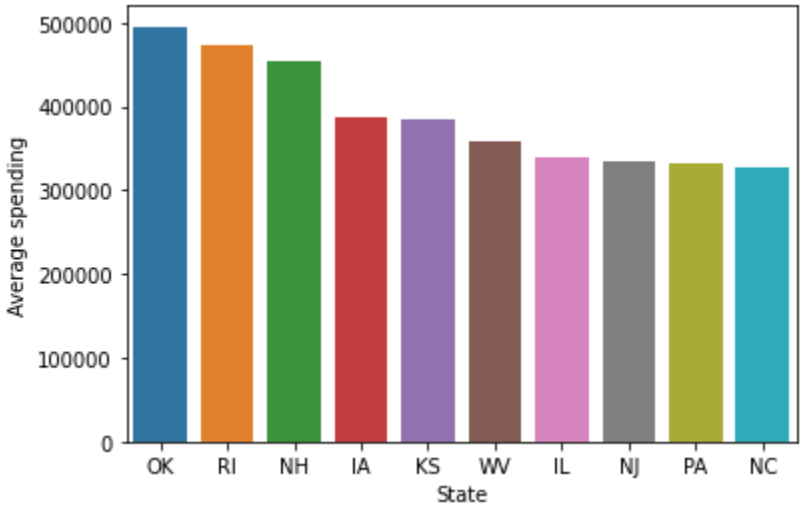
2. Dropped customers with no transaction and now we have 983 customers left:

```
customer_df_combined.shape
✓ 0.0s
(983, 10)
```

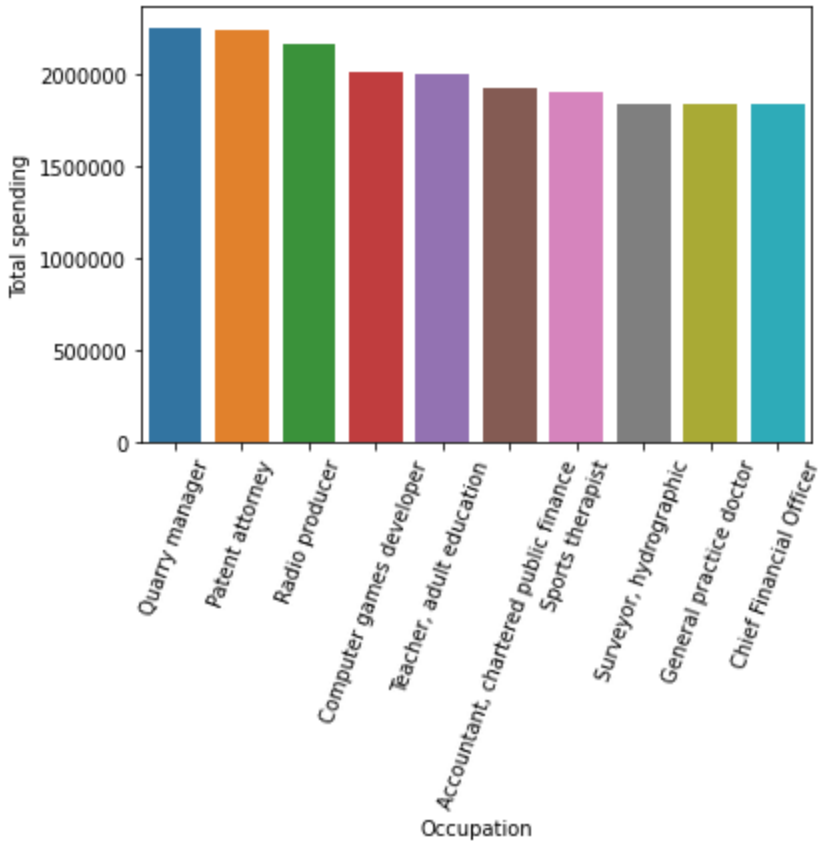
3. Top 10 states with highest total spending:



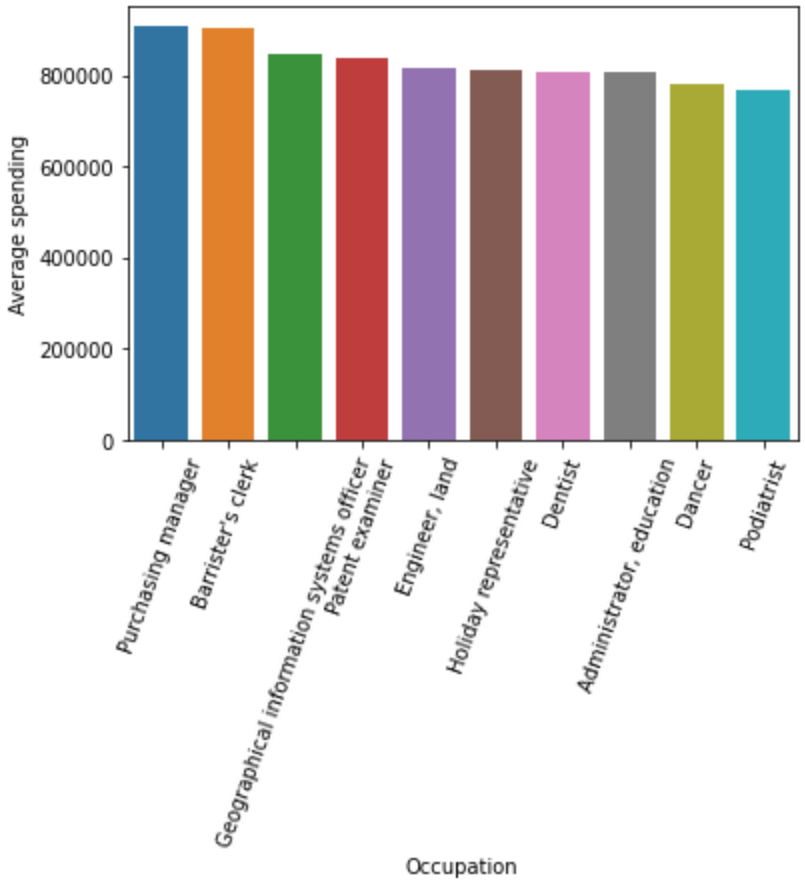
4. Top 10 states with highest average spending



5. Top 10 occupations with highest total spending



6. Top 10 occupations with highest average spending



CALCULATING AGE BASED 'DOB' COLUMN

1. 'dob' column was used to calculate the age of each customer:

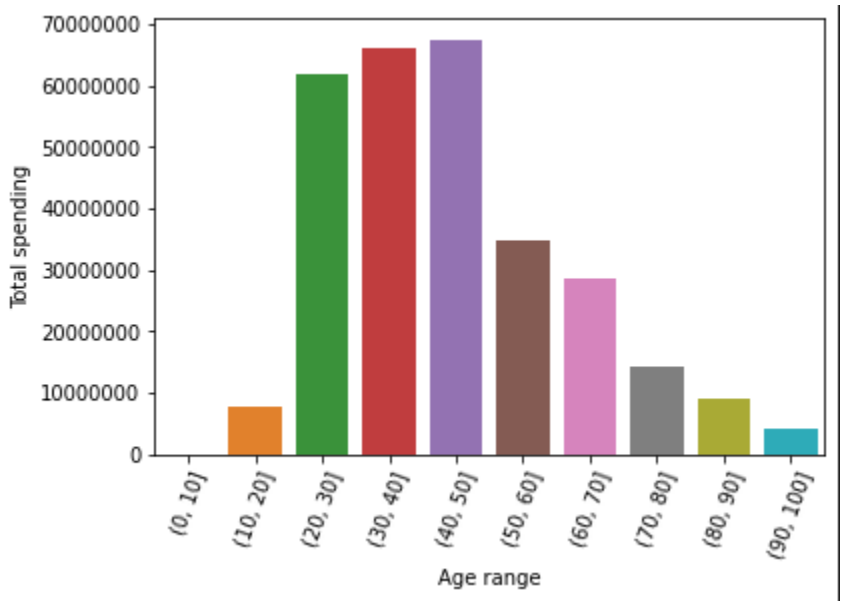
	cc_num	city	state	zip	lat	long	city_pop	job	dob	amt	age
0	4218196001337	Ambler	PA	19002	40.1809	-75.2156	32412	Accounting technician	1959-10-03	206986.20	63.0
1	4351161559407816183	Leland	NC	28451	34.2680	-78.0578	27112	Professor Emeritus	1963-06-07	105741.56	59.0
2	4192832764832	Cordova	SC	29039	33.4275	-80.8857	4215	International aid/development worker	1973-05-30	329105.71	49.0
3	4238849696532874	Birmingham	AL	35242	33.3813	-86.7046	493806	Seismic interpreter	1942-12-26	180341.52	80.0
4	4514627048281480	Fargo	GA	31631	30.7166	-82.5801	559	Medical laboratory scientific officer	1939-05-22	395370.93	84.0
...	...	...	...	...	...	...	...	...	...	...	...
995	30125158904184	Denver	CO	80236	39.6535	-105.0376	990452	Colour technologist	1993-08-31	551053.30	29.0
996	180047909863618	Wasco	CA	93280	35.6480	-119.4487	27152	Software engineer	1982-05-27	584600.74	40.0
997	4371450311809	Escondido	CA	92026	33.1605	-117.0978	171802	Agricultural consultant	1999-05-14	254961.37	24.0
998	3519925692476886	Mattapan	MA	2126	42.2739	-71.0939	25562	Civil Service administrator	1994-02-23	129984.35	29.0
999	343251790447085	Saint Petersburg	FL	33710	27.7898	-82.7243	341043	Manufacturing systems engineer	1942-07-21	350440.28	80.0

983 rows × 11 columns

2. Check for outlier in age column

	cc_num	zip	lat	long	city_pop	amt	age
count	9.830000e+02	983.000000	983.000000	983.000000	9.830000e+02	983.000000	983.000000
mean	3.719789e+17	51852.734486	37.381733	-92.529232	2.867738e+05	299034.387091	49.408952
std	1.236976e+18	29908.399119	5.595588	17.528035	5.322249e+05	196933.226161	18.705173
min	6.040027e+10	1571.000000	19.589300	-159.368300	1.050000e+02	3300.030000	15.000000
25%	1.800375e+14	27547.500000	33.626000	-99.189850	2.027400e+04	160667.735000	34.000000
50%	3.517537e+15	49017.000000	38.317000	-87.167800	6.867700e+04	270474.300000	49.000000
75%	4.548747e+15	78621.000000	41.169500	-80.014400	2.492575e+05	380931.435000	62.000000
max	4.986227e+18	99705.000000	64.780500	-68.244800	2.906700e+06	973811.600000	95.000000

3. Top spenders based on age: Majority of big spenders fall under 20-50 years old bracket



1.4 Modeling

I. CUSTOMER/MARKET SEGMENTATION

To help sales/marketing teams improve their campaign efficiency, we segment the customers using KMeans and hierarchical clustering based on their spending behavior using customer dataset merged with their total transaction amount.

KMeans

We primarily used KMeans to find clusters due to KMeans' speed and efficiency. We segment the customers based on their spending, age, state and occupation:

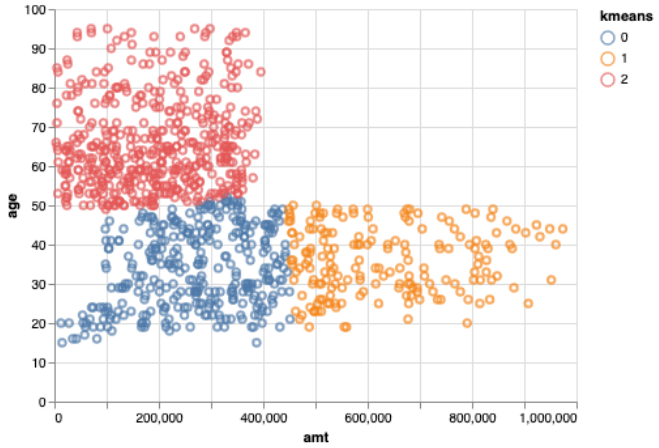
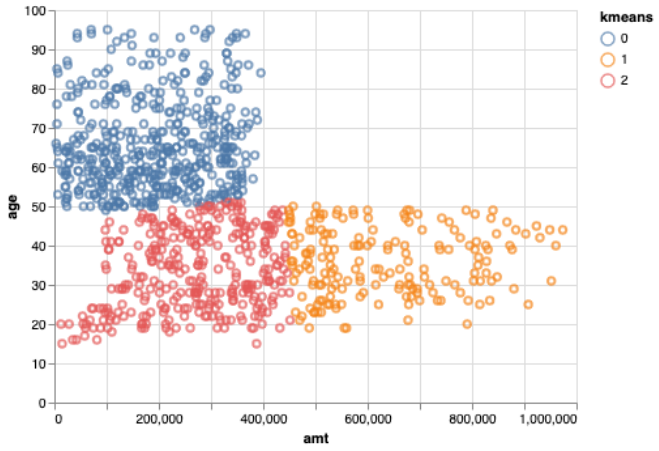
Experiments:

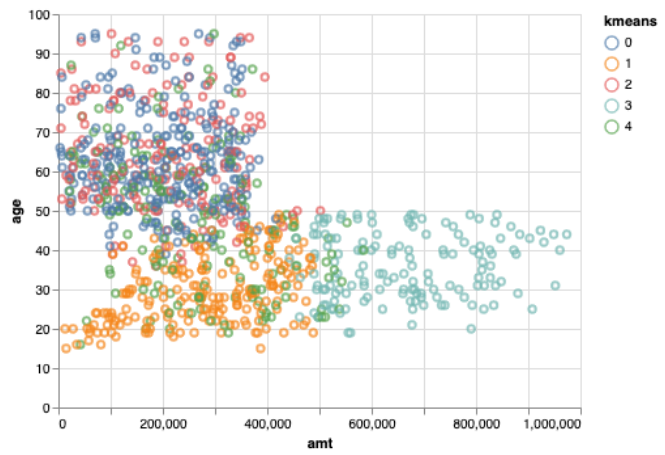
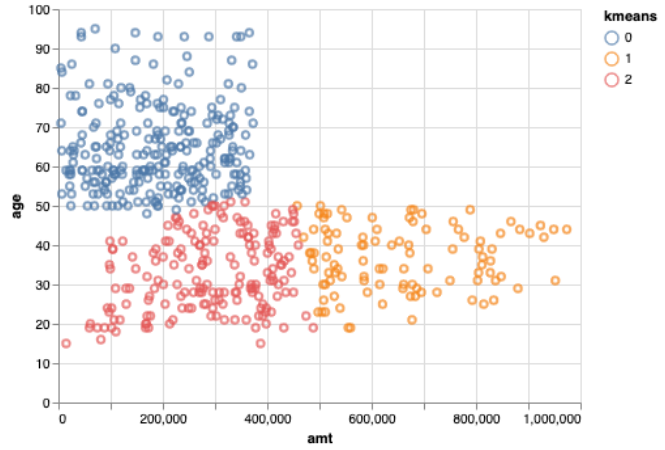
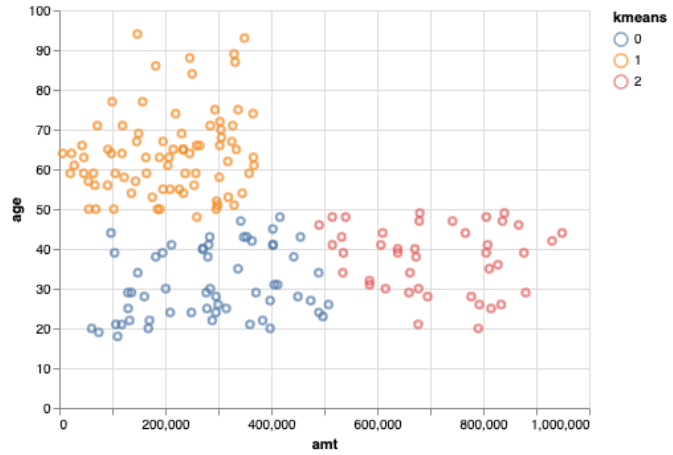
Customer segmentation based on:

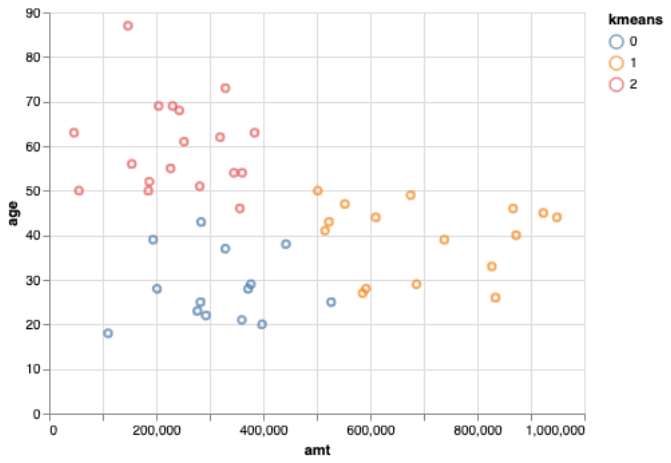
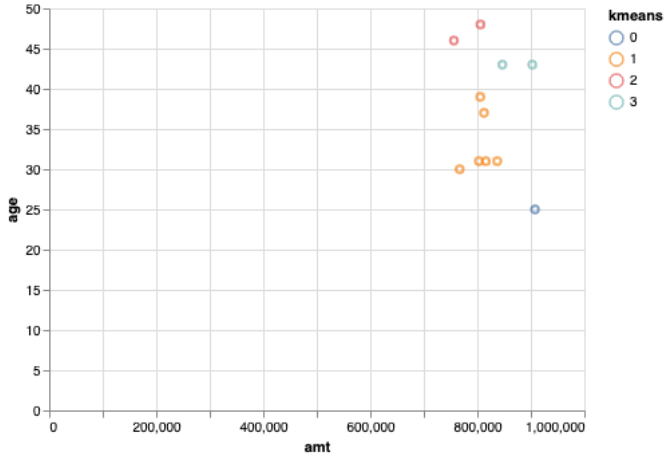
1. Spending and age using StandardScaler

2. Spending and age using MinMaxScaler
3. Amount, age, latitude and longitude
4. Spending and age in top 10 states with highest total spending
5. Spending and age in top 10 states with highest average spending
6. Spending and age in top 10 jobs with highest total spending
7. Spending and age in top 10 jobs with highest average spending

Results:

	Features	Scaler	Result
1	-Spending -Age	StandardScaler	
2	-Spending -Age	MinMaxScaler	

3	-Spending -Age -Latitude -Longitude	StandardScaler	
4	-Spending -Age -Top 10 states with highest total spending	StandardScaler	
5	-Spending -Age -Top 10 states with highest average spending	StandardScaler	

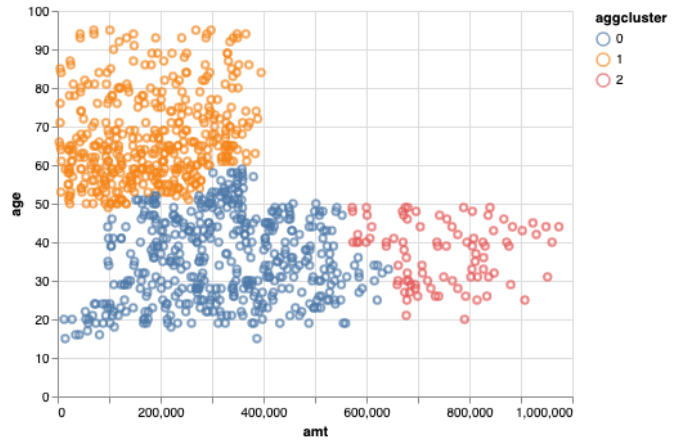
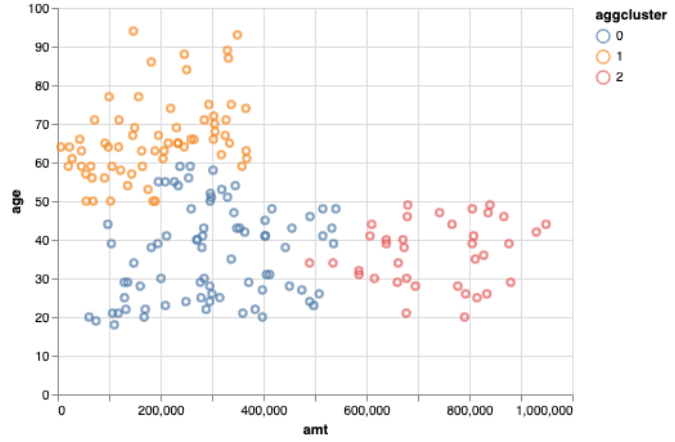
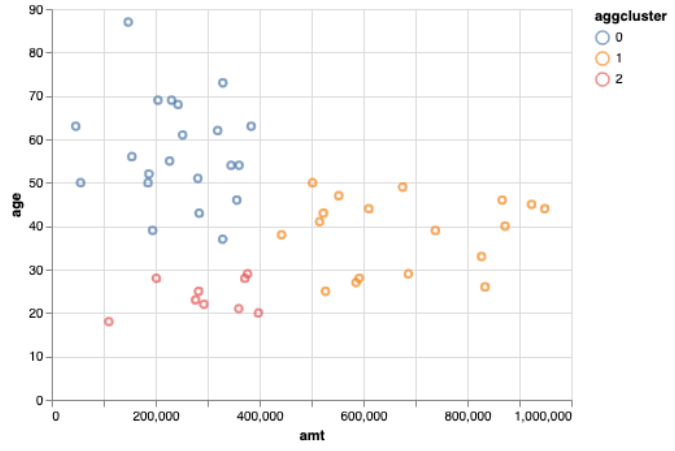
6	-Spending -Age -Top 10 jobs with highest total spending	StandardScaler	
7	-Spending -Age -Top 10 jobs with highest average spending	StandardScaler	

Hierarchical Clustering
To cross checks KMeans' clustering results, we also ran 3 experiments using hierarchical clustering to compare the results with KMeans

Experiments:
Customer segmentation based on:

1. Spending and age using StandardScaler
2. Top 10 states with highest average spending
3. Top 10 jobs with highest total spending

Results:

	Features	Scaler	Result
1	-Spending -Age	StandardScaler	
2	-Spending -Age -Top 10 states with highest average spending	StandardScaler	
3	-Spending -Age -Top 10 jobs with highest total spending	StandardScaler	

II. Anomaly Detection

We can use anomaly detection for many use cases such as intrusion detection or fraud. In the transaction dataset, each of the transactions has been labeled fraud and not fraud, which we can use to assess the accuracy of our model in detecting anomalies.

We will utilize the widely used anomaly detection algorithm for fraud detection called Local Outlier Factor (LOF), this algorithm is an unsupervised learning algorithm meaning it does not require labeled training data. We will run the data through LOF and compare the result with the labeled transaction data to see how good the model is in detecting anomaly (fraud). The target is to be able to achieve 50% accuracy in detecting if the transaction is fraud or not.

We are using 3 metrics to define the model's performance:

1. True negative (target > 50%): how accurate the model is in detecting irregular transaction
2. Precision (target > 80%): how accurate the model is in detecting normal transaction
3. Accuracy (target > 80%): how accurate the model is in predicting both normal and irregular transactions

In experiment 1-7, we used merchants' latitude and longitude as one of the features among other features:

	with latitude and longitude						
Experiment	1	2	3	4	5	6	7
Scaling	None	StandardScaler	MinMax	MinMax	MinMax	MinMax	MinMax
Features used	-amt -merch_lat -merch_long -year -month -day -hour -minute	-amt -merch_lat -merch_long -year -month -day -hour -minute	-amt -merch_lat -merch_long -year -month -day -hour -minute	-amt -merch_lat -merch_long -year -month -day -hour	-amt -merch_lat -merch_long -year -month -day	-amt -merch_lat -merch_long -year -month	-amt -merch_lat -merch_long -year
Accuracy	66%	65%	69%	71%	76%	**84%**	71%
Precision	66%	66%	68%	70%	74%	**81%**	70%
True negative	0.02%	6%	7.60%	16%	34%	**57%**	16%

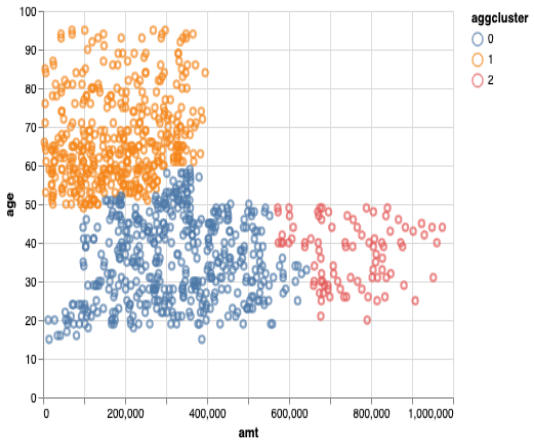
In experiment 8-13, we calculated the distance between the merchant and the customer based on their latitudes and longitudes and used it as a feature replacing 'merch_lat' and 'merch_long'

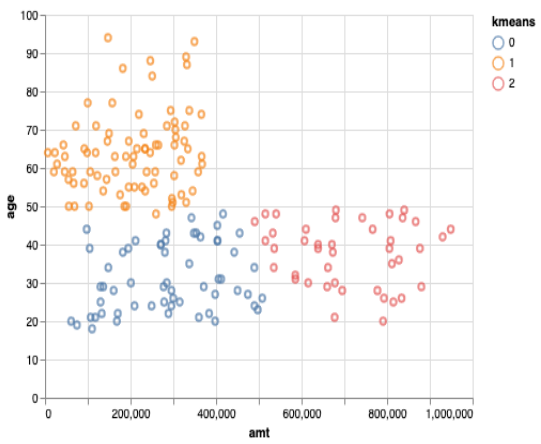
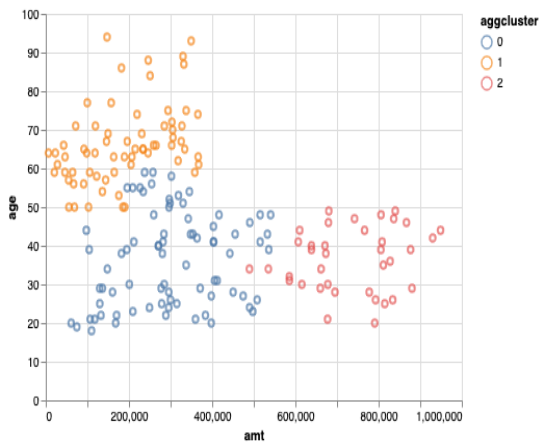
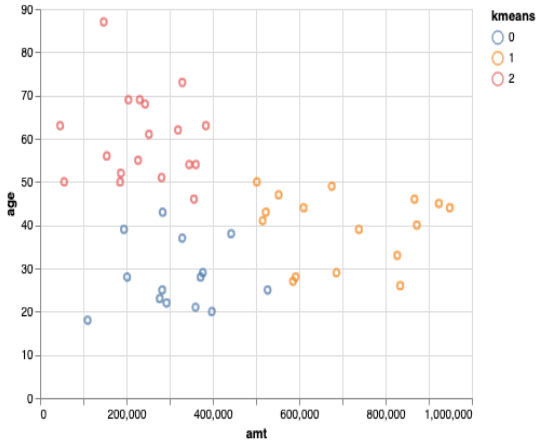
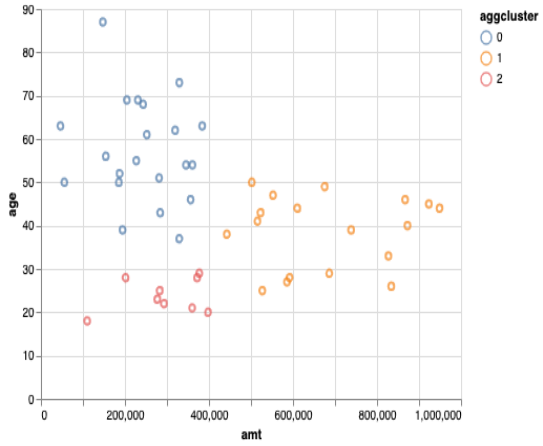
	with distance					
Experiment	8	9	10	11	12	13
Scaling	MinMax	MinMax	MinMax	MinMax	MinMax	MinMax
Features used	-amt -merch_lat -merch_long -year -month -day -hour -minute -distance	-amt -year -month -day -hour -minute -distance	-amt -year -month -day -hour -distance	-amt -year -month -day -distance	-amt -year -month -distance	-amt -merch_lat -merch_long -year -month -distance
Accuracy	68%	69%	73%	**87%**	75%	76%
Precision	67%	68%	71%	**85%**	73%	74%
True negative	4.80%	10%	22%	**69%**	28%	32%

1.5 Evaluation

I. CUSTOMER/MARKET SEGMENTATION

Using KMeans and hierarchical clustering, we were able to segment the customers based on their age, spending, state and occupation as seen below, both algorithms produced similar results which further warrant the segmentation results:

	Features	KMeans	Hierarchical Clustering
1	-Spending -Age		

2	-Spending -Age -Top 10 states with highest average spending		
3	-Spending -Age -Top 10 jobs with highest total spending		

Looking closer, we can also see the average age and amount of each clusters, both KMeans and hierarchical clustering have similar clusters when compared:

1. Customer segmentation based on spending and age using StandardScaler

KMeans

kmeans	cc_num	zip	lat	long	city_pop	amt	age
0	4.081273e+17	53988.624625	37.407802	-93.698709	286446.408408	271594.179339	33.921922
1	4.024248e+17	52159.983333	37.312416	-92.343196	344854.144444	631675.233556	35.872222
2	3.347072e+17	50221.763830	37.389810	-91.771892	264762.148936	191081.699468	65.565957

Hierarchical clustering

aggcluster	cc_num	zip	lat	long	city_pop	amt	age
0	3.993399e+17	54390.479570	37.155597	-93.845331	305396.627957	320210.209505	35.825806
1	3.572125e+17	50021.992991	37.451992	-91.776663	267876.063084	181635.703645	66.735981
2	3.008357e+17	47447.244444	38.215982	-89.308270	280424.700000	747921.932556	37.188889

2. Customer segmentation based on top 10 states with highest average spending

KMeans

kmeans	cc_num	zip	lat	long	city_pop	amt	age
0	8.473080e+17	31657.758621	39.578948	-81.959891	210061.982759	288351.018793	31.137931
1	5.421679e+17	32979.802469	39.481686	-81.609320	127396.543210	205624.318272	63.950617
2	5.419520e+17	36172.428571	39.147505	-83.421702	187885.404762	708987.215000	37.666667

Hierarchical clustering

aggcluster	cc_num	zip	lat	long	city_pop	amt	age
0	8.582549e+17	32470.525641	39.680358	-82.047029	195000.705128	300362.721026	36.500000
1	4.061322e+17	32922.893939	39.237914	-81.696542	138278.878788	190791.920909	66.212121
2	5.965633e+17	35706.594595	39.270827	-83.137838	163714.621622	733427.915676	36.513514

3. Customer segmentation based on top 10 jobs with highest total spending

KMeans

kmeans	cc_num	zip	lat	long	city_pop	amt	age
0	6.591859e+17	53147.785714	37.677493	-92.792957	121227.571429	317619.010000	28.285714
1	8.126471e+17	50909.125000	36.392944	-91.225387	68655.562500	703317.345000	39.437500
2	2.426828e+17	36900.000000	37.937539	-83.235067	481990.833333	239451.120556	60.166667

Hierarchical clustering

aggcluster	cc_num	zip	lat	long	city_pop	amt	age
0	2.085659e+17	39894.047619	37.964338	-84.538919	424208.142857	243684.570476	57.238095
1	7.228826e+17	48142.666667	37.071972	-90.246378	62363.166667	679012.307778	38.555556
2	1.023053e+18	57608.000000	36.455678	-95.242967	160066.888889	296696.256667	23.777778

KMeans and hierarchical clustering provide a lot of benefits for the marketing/sales team by segmenting customers based on their spending and age. With clustering, the bank can:

1. Segment customers based on age and spending nationwide
2. Segment customers based on age and spending in top 10 states with highest total spending
3. Segment customers based on age and spending in top 10 states with highest average spending
4. Segment customers based on age and spending in top 10 jobs with highest total spending
5. Segment customers based on age and spending in top 10 jobs with highest average spending

If we were given the opportunity and more time, these are the other potential segmentations we can do based on their spending behavior:

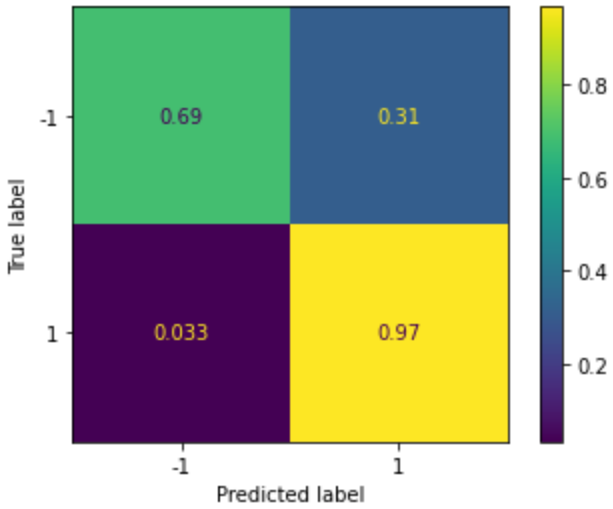
1. Segment based on city
2. Segment based on zip
3. Segment based on east or west coast
4. Segment based on clusters of state
5. Segment based on 2020, 2021, 2022

II. Anomaly Detection

We ran 13 experiments using different features and only 2 experiments achieved target scores:

Experiment	6	11
Scaling	MinMax	MinMax
Features used	-amt -merch_lat -merch_long -year -month	-amt -year -month -day -distance
Accuracy	**84%**	**87%**
Precision	**81%**	**85%**
True negative	**57%**	**69%**

Experiment 11's confusion matrix:



With only 5 features, experiment #11 yields the best result, it was able to detect 69% of the anomalies in the transaction dataset and over 85% in detecting non-anomalies transactions. This model's performance is promising in helping the bank automatically detect anomalies in their customer's transactions and can potentially save the bank a lot of time, resources and manpower.

ETHICAL CONSIDERATION

The datasets contain many sensitive information such as names, gender, addresses, social security numbers and credit card numbers and can raise significant ethical concerns, some of the key issues include:

1. **Privacy:** PII (Personally identifiable information) in the dataset must be treated with utmost care to protect the customers' privacy rights. Access, use and disclosure of data must be regulated and stay within customer's privacy rights and laws.
2. **Data Security:** Strong security measures must be in place to safeguard data from unauthorized access or data breaches to prevent identity theft or fraud.
3. **Bias and discrimination:** datasets with personal information should be used with awareness and understanding of potential biases and discriminatory effects.

To minimize privacy and bias issues, we dropped several identifier columns such as names, gender, address and social security number. We also kept the data in a secure local folder to protect against unauthorized access.

1.6 Deployment

Both KMeans and LOF models have promising results and we think they are ready for trial run using real time transactions in the real world to ensure the readiness of the model before deploying it to a larger audience.

Next step is to develop a robust and comprehensive plan which includes close monitoring of the model's performance, careful assessment of potential impacts such as bias, gathering feedback from users/stakeholders and fine tuning the model along the way. This approach will ensure the models' effectiveness and suitability for a larger audience.

2. Regression analysis (Somayeh Amraee)

2.1 Business Understanding

The project's business use case is to assist clients in creating better financial budgets by forecasting their monthly spending totals. Individuals struggle to adequately anticipate their future spending and manage their budgets in today's fast-paced and changeable financial environment. The bank promises to give consumers accurate spending forecasts so they may make wise financial decisions by utilising machine learning algorithms and the accessible transactional data. Accurate spending predictions hold significant relevance to the bank's customers as they offer several benefits:

Improved Financial Planning: Customers can proactively arrange their budgets based on the anticipated spending amounts, which leads to improved financial planning. Customers can better utilise their resources, alleviate financial stress, and make sure that basic necessities are satisfied by foreseeing future expenses like bills, rent, or loan payments.

The bank has used a number of machine learning methods, including Linear Regression, Gradient Boosting, and Random Forest, to achieve this. These algorithms produce estimates for the overall spending amount based on historical transaction data and related factors like category, merchant, time, and location.

It is crucial to take into account potential risks or difficulties brought on by incorrect predictions, as well as how they can affect customers:

Customers may experience financial instability if the machine learning algorithms' expenditure estimates are off. When making budgeting decisions, relying too heavily on these estimates could result in unforeseen costs or overspending, which would put extra strain on your finances.

The bank can employe cross-validation methods, such as K-fold cross-validation, to evaluate the performance of the models and guarantee their dependability in an effort to reduce these risks. The bank can measure the generalisation ability of the models and spot any overfitting or underfitting concerns by dividing the data into training and validation sets and testing the models on several folds.

The performance of the models is also evaluated using evaluation measures like R-squared, Mean Absolute Error (MAE), and Mean Squared Error (MSE). These metrics shed light on the degree of model fit to the data as well as the size of prediction errors.

It is crucial to remember that the reliability of spending forecasts depends on the quality and relevance of the available data. To ensure the data's integrity and improve the performance of the models, data preprocessing approaches such as data cleaning, feature engineering, and normalisation have been used.

The bank wants to boost client satisfaction and their ability to manage their finances by giving customers accurate expenditure estimates. These forecasts can be used by customers to help them use their resources effectively, make timely financial decisions, and move towards their financial objectives.

In conclusion, the business use case of assisting customers in creating better financial budgets by projecting their total monthly spending has enormous value for both the bank and its clients. By using machine learning algorithms, the bank can provide its clients with insightful information that will help them plan their budgets wisely, reduce financial risks, and enjoy more financial stability and satisfaction.

2.2 Data Understanding

Please refer to section 1.2

2.3 Data Preparation

For preparing the data, the following steps are taken:

To ensure accurate and reliable machine learning models for predicting customers' total spending amount, the provided dataset underwent thorough data preparation steps.
1. Data Cleaning:
• Missing Values: Techniques such as imputation or deletion of missing records were applied to handle missing values in the dataset.
• Duplicate Records: Duplicate records were identified and eliminated from the dataset.
2. Feature Selection:
• Relevant Features: A subset of features with potential impact on spending predictions was selected, including 'category', 'is_fraud', 'merchant', 'merch_lat', 'merch_long', 'lat', 'long', 'job', 'year', 'month', and 'day'.
• Ethical Considerations: Features containing sensitive identification information, such as 'trans_num', 'last', 'first', 'ssn', 'acct_num_x', 'acct_num_y', and 'cc_num', were removed to uphold privacy and ethical considerations.
3. Outlier Removal:
• Outliers in numerical features were addressed using methods like the Interquartile Range (IQR) or Z-score. Specific codes were used to identify and remove outliers, ensuring robust model performance.
4. Data Transformation:
• Scaling: Numerical features were scaled to a standard range to prevent dominance of certain features. Techniques like Min-Max scaling or Standard scaling were employed for data normalization.
5. Train-Validation-Test Split:
• The dataset was divided into training, validation, and testing sets. The training set was used for model training, the validation set aided in hyperparameter tuning and performance evaluation, while the testing set assessed the final model's performance on unseen data.

Through these data preparation steps, the dataset was refined, reducing noise, handling missing values, removing outliers, and transforming features. This process enhances the accuracy of predictions and ensures reliable models. Additionally, the removal of sensitive identification features like trans_num','last','first','ssn','acct_num_x','acct_num_y','cc_num' upholds privacy and ethical considerations.

It's important to note that data preparation is an iterative process, involving exploration, pre-processing, and evaluation. Continuous refinement of the data preparation pipeline contributes to improved model performance and addresses potential data-related challenges.

2.4 Modeling

Using the supplied dataset, we train and assess machine learning models throughout the modeling step. Finding a model that can correctly forecast clients' monthly spending totals is the objective.

Our main modeling strategies for this project include univariate and multivariate linear regression, gradient boosting regression, and random forest methods.

K-fold Cross-Validation was also used by us during the training procedure. The performance and generalizability of the model are evaluated using the cross-validation technique. The dataset is divided into various subsets (folds), the model is trained on some of the folds, and it is then tested on the final fold. The average performance is calculated after this procedure has been repeated several times. We can get a more reliable estimate of the model's performance and lower the chance of overfitting by utilising cross-validation.

These methods aid in the development of a more precise and trustworthy predictive model.

2.5 Evaluation

Gradient Boosting Regression

The Gradient Boosting model achieved a relatively low R-squared value every time we used it with different features, indicating that only a small portion of the variance in the target variable could be explained by the selected features. The model also exhibited a high Mean Absolute Error (MAE), representing the average difference between the predicted and actual values. Additionally, the Mean Squared Errors (MSE) were very high, reflecting the average squared difference between the predicted and actual values. These results suggest that the model's predictive performance may be limited, and further improvements or alternative modeling approaches may be necessary to enhance the accuracy of predicting customers' total spending amounts. The table below showcases the results of Gradient Boosting Regression in 6 different experiments using different sets of features.

	category	Is_fraud	merchant
R-squared	0.010	0.010	0.010
MAE	50.03	50.03	50.03
MSE	5830	5830	5830
Mean Cross-Validation R-squared	0.00097		

	Merch_lat	Merch_long
R-squared	0	0
MAE	50.40	50.40
MSE	5878	5878
Mean Cross-Validation R-squared	-0.0099	

	lat
R-squared	0.03
MAE	49.96
MSE	5716

	job
R-squared	0
MAE	50.28
MSE	5848

	year	month	day
R-squared	0.01	0.01	0.015
MAE	50.19	50.19	50.19
MSE	5804	5804	5804
Mean Cross-Validation R-squared	-0.00725		

And the result of modelling with some more features at once: 'category', 'is_fraud', 'merchant', 'merch_lat', 'merch_long', 'lat', 'long', 'job', 'year', 'month', 'day' is:

R-squared	0.02
MAE	49.5
MSE	5726

Random Forest

The results of the Random Forest Regression experiments with different features indicate a low R-squared value and relatively high MAE and MSE. This suggests that these features alone have limited predictive power in determining customers' total spending amount for the next month. Consequently, relying solely on these features may not provide accurate predictions for the business use case of financial management and budgeting capabilities.

	lat
R-squared	0.030
MAE	49.96
MSE	5716

	job
R-squared	0.01
MAE	50.19
MSE	5786

	year	month	day
R-squared	0.03	0.03	0.031
MAE	49.95	49.95	49.95
MSE	5710	5710	5710
Mean Cross-Validation R-squared	-0.02		

	lat	job	year	month	day
R-squared	0.030	0.01	0.03	0.03	0.031
MAE	49.96	50.19	49.95	49.95	49.95
MSE	5716	5786	5710	5710	5710
Mean Cross-Validation R-squared			-0.02		

Linear Regression

Linear regression, as you can see the result of different experiments in tables, cannot accurately predict determining customers' total spending amount for the next month because it has very low R square and very high MAE and MSE.

	job
R-squared	0
MAE	50.43
MSE	5880.65

	category
R-squared	0
MAE	50.38
MSE	5878.01

	merchant
R-squared	0
MAE	50.43
MSE	5880.82

	Merch_lat
R-squared	0
MAE	50.42
MSE	5880.30

	Merch_long
R-squared	0
MAE	50.43
MSE	5880.77

	year	month	day
R-squared	0	0	0
MAE	50.41	50.41	50.41

MSE	5879.78	5879.78	5879.78

2.6 Deployment

We carefully considered the usability and influence on the business use case of the regression models in light of their limitations. Despite the fact that the results were not ideal for precise estimation of clients' overall spending amounts, we were nonetheless able to draw important conclusions from the study.

C. Conclusion

Regression algorithms may not be suitable for accurately predicting the customer's next month spending. Therefore, it is highly recommended to explore other algorithms such as classification to improve the predictive performance.

3. Classification (Monali Patil)

3.1 Business Understanding

The bank has been collecting transactional data from its customers for the past three years. This dataset includes various features such as transaction amount, transaction type (e.g., online purchase, grocery, health-related expenses), merchant information, customer details, and other relevant features.

As part of the technical solutions team, our group needs to identify the business problem and develop a machine learning model for each use case namely Regression, Classification and Unsupervised learning and share our results achieved with the bank’s compliance team to develop solutions for which they rely on our team’s expertise.

BUSINESS PROBLEM

Within the scope of one of the Classification use cases, our objective is to develop a robust machine learning model that can effectively analyze transactional data and accurately classify transactions as either fraudulent or legitimate.

The resulting model could empower the bank to detect, prevent, and mitigate fraudulent activities, reduce financial losses, protect customers' interests, comply with regulatory requirements, strengthen its security measures, and enhance operational efficiency.

The model's adverse effects could result in monetary losses for the bank and impact customer confidence in the security of their accounts if undetected fraudulent transactions occur. Additionally, there is a possibility of misidentifying legitimate transactions as fraudulent, which could lead to customer dissatisfaction and a decline in trust.

3.2 Data Understanding

DATA COLLECTION

The dataset was acquired from the canvas portal in CSV format, and there were no copyright or privacy issues associated with it since it was obtained as a student from the University portal.

DESCRIBE DATA

Sr. No.	Features Names	Missing Values	Data Type	Description
1	ssn	0	object	Customer's social security number
2	acct_num	0	object/int64	Customer's bank account number
3	cc_num	0	object/int64	Customer's credit card number
4	trans_num	0	object	Transaction number
5	amt	0	float64	Amount of the transaction
6	**is_fraud**	**0**	**object**	**If transaction is fraud or not**
7	unix_time	0	object	Transaction time recorded in Unix format
8	category	0	object	Category of transaction
9	merchant	0	object	Merchant's name for transaction occurred
10	merch_lat	0	float64	Latitude of merchant's location
11	merch_long	0	float64	Longitude of merchant's location
12	first	0	object	Customer's first name
13	last	0	object	Customer's last name
14	gender	0	object	Customer's gender
15	street	0	object	Customer's residential street name
16	city	0	object	Customer's residential city name
17	state	0	object	Customer's residential state name
18	zip	0	int64	Customer's residential zip code
19	lat	0	float64	Latitude of customer's resident
20	long	0	float64	Longitude of customer's resident
21	city_pop	0	int64	Customer's residential city population
22	job	0	object	Customer's job name
23	dob	0	object	Customer's date of birth

Table 1: Combined Bank's Transactional and its Customers dataset

- Identifiers
- Target Variable

The two datasets are combined based on the 'acct_num' account number feature as the key. The dataset contains multiple features that provide information related to the transactions and its customers.

All the features are of a numerical (integer and float) and categorical data type. Table 1, provides a summary of the dataset, its features data types, names, and descriptions.

Below are some of the key limitations of the dataset.

1) Despite the presence of numerous features information, Table 1 shows that many of them are unique identifiers. These identifiers do not contribute to the model's ability to learn generalized patterns for detecting fraudulent behavior.
2) It is essential to consider potential biases and fairness implications associated with using a customer's gender as a feature.
3) Transaction dates are recorded in the raw Unix format which needs further processing to convert it into readable format.

EXPLORE DATA

Datasets	Rows	Columns
Transactions	4260904	10
Customers	1000	15

Table 2: Datasets size

Datasets	Missing Values	Duplicate values
Transactions	0	0
Customers	0	0

Table 3: Datasets related detail

The transactional and customer data collected for the past 3 years is of below size and do not contain any missing or duplicate values as shown in table 2 and 3.

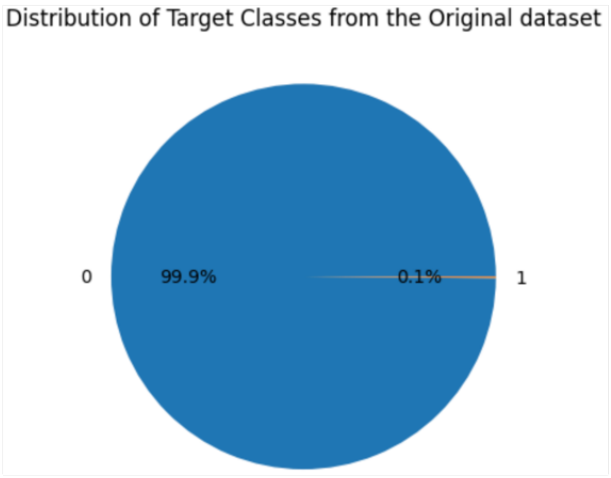


Figure 1: Distribution of target classes

Figure 1 illustrates that a significant portion of the total observations belongs to one target class, labeled as non-fraudulent transactions represented by the value 0. So, the bank's transaction dataset shows a high level of class imbalance.

Note: For detailed data exploration and its insights please refer to the 2.2] Exploring Data section from the Python notebook.

3.3 Data Preparation

Machine learning algorithms cannot take non-numerical inputs so it is necessary to transform categorical data into numerical. The following activities are performed to prepare and process the data for developing the classification models.

Feature Engineering: Extracted new information related to transaction date, customer's age, and distance in km i.e., how far the transaction occurred from the customer's residence that can help in capturing patterns and trends associated with fraudulent transactions.

Features Selection: Including non-generalised and irrelevant informative features in the model can introduce noise and unnecessary complexity, potentially leading to overfitting or reduced model performance. Therefore, it is important to select features that are likely to have a meaningful relationship with the target variable and can effectively distinguish between fraudulent and non-fraudulent transactions.

Transforming Categorical Data into Numerical: To develop machine learning models that require numerical inputs, the categorical features are encoded as numerical values before training.

Splitting Data into Different Sets: Considering the high imbalance nature of the dataset, the sampling is performed and the dataset is divided into training (80%), validation (20%), and testing (20%) subsets.

Features Scaling: Scaling is applied to all the features to standardize their values to a uniform level. This allows the model to effectively use all the information from the features to learn generalized patterns, identify fraudulent behavior and make accurate classifications.

For a comprehensive understanding, please refer to the 3] Data Preparation section from the Python notebook.

3.4 Modeling

Given that the target variable comprises two classes, with 0, indicating a non-fraudulent transaction, and 1, indicating a fraudulent transaction, employing a classification model is the suitable technique as our objective is to classify transactions within these two specific classes.

The following experiments from Table 4, are performed with XGBoost (eXtreme Gradient Boosting), Random Forest, and MLPClassifier Neural Network models to assess their performance and identify the best model that can most effectively classify fraudulent and non-fraudulent transactions.

Exp. No	Models	Hyperparameters Used	Techniques Used
1	XGBoost	colsample_bytree=0.73, gamma=0.16, learning_rate=0.11, max_depth=13, min_child_weight=3, reg_alpha=0.19, reg_lambda=0.05, subsample=0.67	Cross-Validation and Random Search
2	Random Forest	n_estimators=91, max_depth=9, min_samples_leaf=13, class_weight='balanced', criterion='entropy', max_features='log2', random_state=19	
3	Random Forest with Important Features	n_estimators=169, max_depth=15, min_samples_leaf=5, class_weight='balanced', criterion='entropy', max_features='log2', random_state=19	Feature Importance and Fine-tuned Hyperparameters
4.1	MLPClassifier Neural Network	hidden_layer_sizes=256, batch_size=32, activation='relu', solver='adam', learning_rate = 'constant', learning_rate_init=0.001, max_iter=10(epochs), random_state=19	
4.2	MLPClassifier Neural Network	hidden_layer_sizes=256, batch_size=32, activation='relu', solver='adam', learning_rate = 'constant', **learning_rate_init=0.01**, max_iter=10(epochs), random_state=19	
4.3	MLPClassifier Neural Network	**hidden_layer_sizes=512**, batch_size=32, activation='relu', solver='adam', learning_rate = 'constant', learning_rate_init=0.01, max_iter=10(epochs), random_state=19	
4.4	MLPClassifier Neural Network	hidden_layer_sizes=512, batch_size=32, activation='relu', solver='adam', learning_rate = 'constant', **learning_rate_init=0.0001,** max_iter=10(epochs), random_state=19	

Table 4: Models trained, its hyperparameters and techniques applied for the classification use case.

3.5 Evaluation

To evaluate the model's performance using the following performance metrics.

- Precision
- Recall
- Weighted F1 Score
- Binary F1 Score
- Confusion Matrix

Precision, Recall, and F1 Score are employed because they provide different insights into the model's performance, enabling a comprehensive understanding of various aspects of the classification use case. The binary F1 score is used to evaluate the performance specifically for classes that have positive labels.

The aim of the model is to not only achieve high performance but also prioritize minimizing false negatives i.e., incorrectly classifying a fraudulent transaction as legitimate to ensure that potentially fraudulent activities are not overlooked.

Therefore, Confusion Matrix is used which informs valuable false negatives error along with how well the model is performing in terms of its ability to correctly predict the positive and negative classes.

Following are the performance metrics of the classification models.

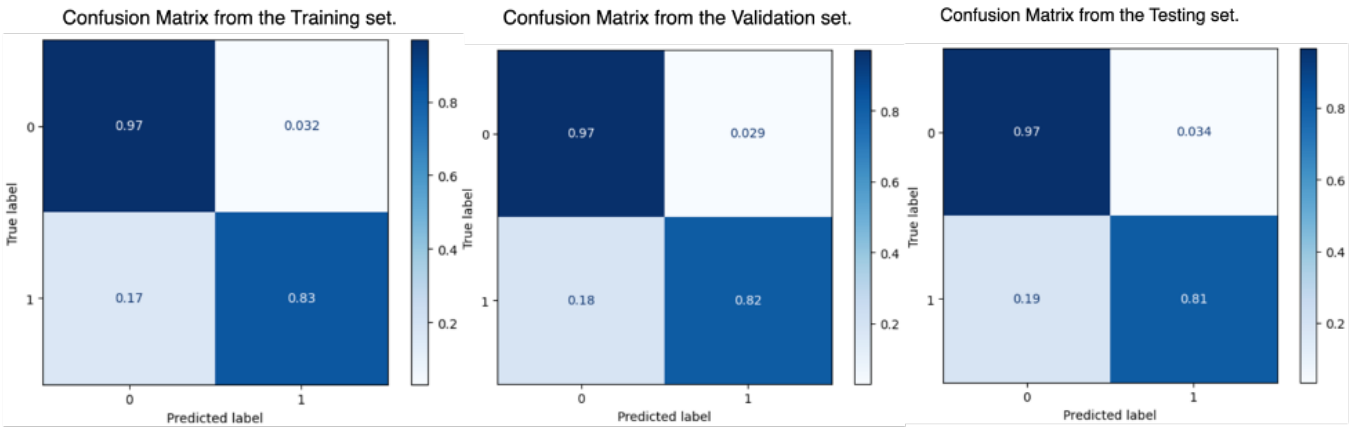
Experiment 1]

Excluding the performance score for the XGBoost classification model as it resulted in lower scores compared to other trained models.

Experiment 2]

Random Forest

Dataset	Precision	Recall	Weighted F1 Score	Binary F1 Score	False Negative Error	False Positive Error Error
Training	0.9632	0.8326	0.8999	0.8931	0.17	0.032
Validation	0.9653	0.8156	0.8925	0.8931	0.18	0.029
Testing	0.0257	0.8097	0.9816	0.0498	0.19	0.034



- The weighted F1 score on the training of 89% and validation of 89% indicates that the model with Random Forest is performing consistently, while on the unseen testing data with the same high imbalanced target classes distribution as the original data, it results in 98% indicating that the model is capturing important patterns and generalized enough to identify potential fraudulent and legitimate transactions from unseen data.

- The binary F1 score denoting the classifications that belong to the positive class is identical at 89% for the training and validation set, while for testing its 0.0498% which might be due to the presence of a huge imbalance of target classes.

- The Recall score of 83% on training, 81% on validation and 80% on testing sets are decent, indicating that the model correctly predicted fraudulent transactions i.e., above 80% across all the sets, out of all the actual fraudulent transactions. A high recall is important as the cost of false negatives, such as failing to identify fraudulent transactions, is significant, and it is crucial to detect and prevent as many fraudulent activities as possible.

- The rate of False Negative errors, such as flagging fraudulent transactions as legitimate from the confusion matrix shows 17% for the training, 18% for the validation and 19% for the testing set which is relatively consistent.

- The rate of False Positive errors, which involves classifying legitimate transactions as fraudulent, is consistently low and remains stable throughout the different datasets with 0.032% for the training set, 0.029% for the validation set, and 0.034% for the testing set.

FEATURE IMPORTANCE

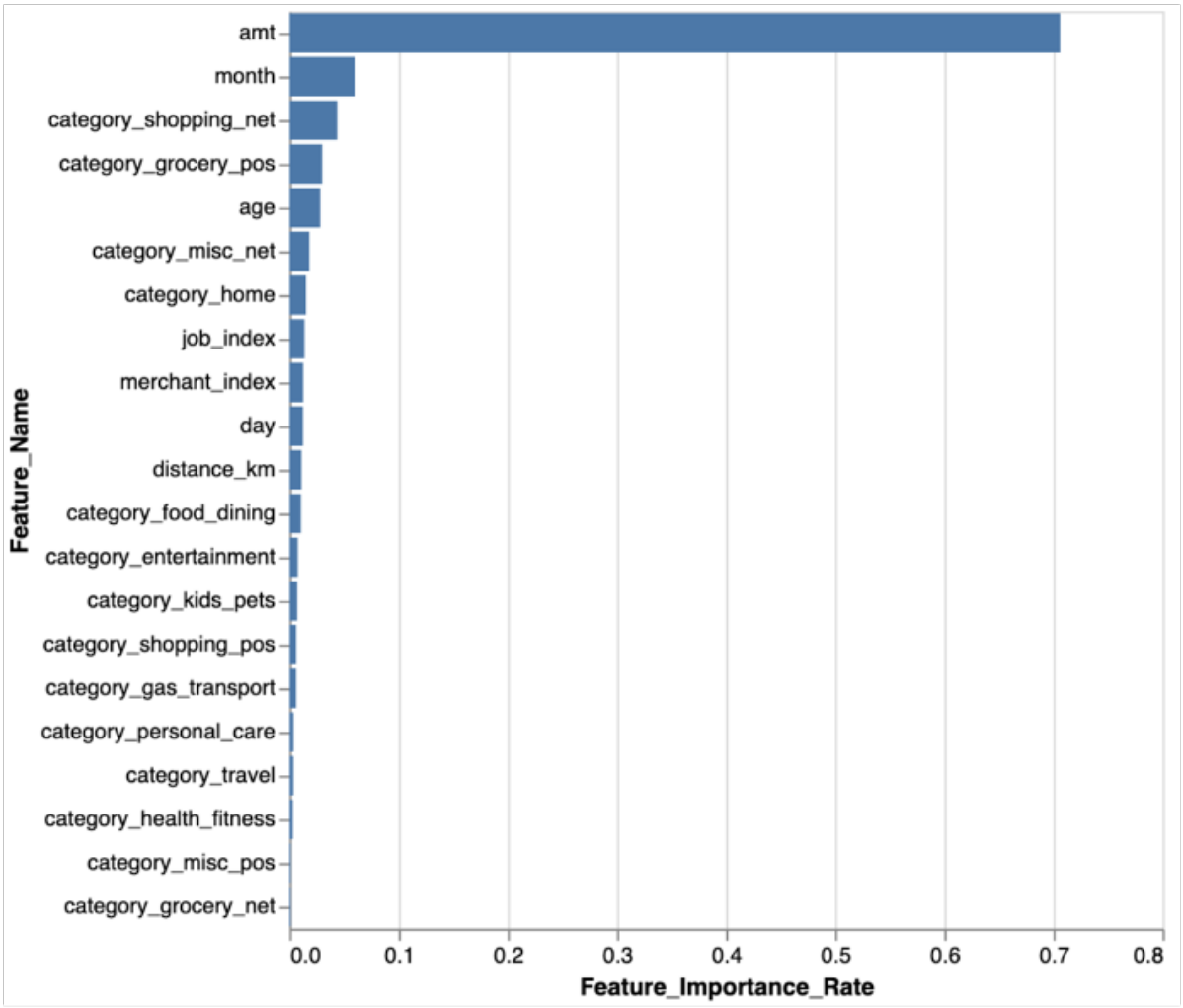


Figure 2: Important Features

- The above Figure 2, informs the feature importance measure, which provides valuable insights into the features and helps in understanding which features are making the most significant contributions to the model's predictions.

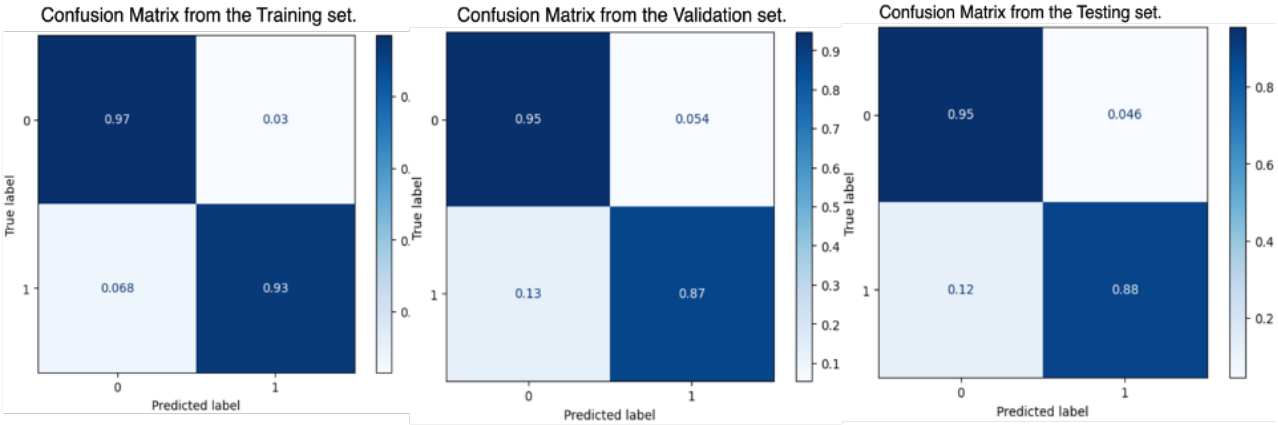
- Looking at the graph, it seems that the least contributing features are related to some of the values of the transaction's category namely 'category_travel', 'category_health_fitness', 'category_misc_pos', and 'category_grocery_net'.

- The rest of the features are contributing the most to the prediction of target classes and their feature's importance rate ranges between 0.73% to 0.01%.

Experiment 3]

Random Forest with Important Features

Dataset	Precision	Recall	Weighted F1 Score	Binary F1 Score	False Negative Error	False Positive Error Error
Training	0.9688	0.9321	0.9511	0.9501	0.068	0.03
Validation	0.9419	0.8717	0.9089	0.9055	0.13	0.054
Testing	0.0206	0.8799	0.9752	0.0403	0.12	0.046



- With a weighted F1 score of 95% on the training, 90% on the validation and 97% on the testing set, the Random Forest model trained with important features demonstrates relatively consistent performance suggesting that the model has a strong ability to generalize and effectively identify potentially fraudulent activities on unseen data.

- The binary F1 score for classifying positive class instances is 95% for training and 90% for validation. However, the score drops significantly to 0.0403% for testing, potentially due to a significant imbalance in the target classes.

- The Recall score of 93% on training is slightly higher than the 87% on validation and testing sets, which indicates that the model correctly identified a higher proportion of fraudulent transactions compared to the classification model from the 2nd experiment. This performance was consistent across all sets, with the model correctly predicting over 87% of actual fraudulent transactions.

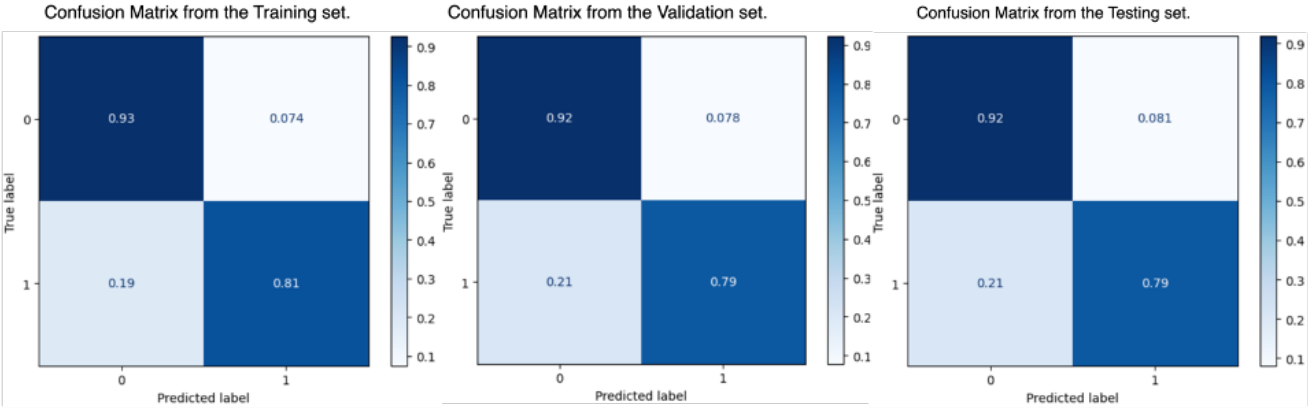
- However, from the confusion matrix, the rate of False Negative errors, with 0.068% for training, 13% for validation and 12% for testing illustrates the overfitting nature of the model indicating that the model is misclassifying a significant amount of fraudulent transactions as legitimate which might lead to financial losses and loss in customer's trust.

- The rate of False Positive errors, which resulted in 0.03% for training, 0.054% for validation and 0.046% for the testing set is not consistent which represents misclassifying legitimate transactions as fraudulent.

Experiment 4]

MLPClassifier Neural Networks

Dataset	Precision	Recall	Weighted F1 Score	Binary F1 Score	False Negative Error	False Positive Error Error
Training	0.9164	0.8145	0.8697	0.8625	0.19	0.074
Validation	0.9101	0.7912	0.8559	0.8465	0.21	0.078
Testing	0.0106	0.7885	0.9565	0.0209	0.21	0.081



- The MLPClassifier Neural Network model trained with important features exhibits varying performance with a weighted F1 score of 86% on the training set, 85% on the validation set, and 95% on the testing set. This suggests that the model's performance is not consistent, indicating that it may not have generalized well enough to accurately detect potentially fraudulent activities in unseen data.

- The binary F1 score for classifying positive class instances is 86% for training and 84% for validation. However, there is a significant drop in the score to 0.0209% for testing, indicating a potential issue of overfitting (also from the training and validation scores) and highlighting the impact of class imbalance in the target classes.

- The Recall score of 81% on training is slightly higher than the 79% on validation and 78% on the testing set, indicating that the model quite consistently identifies a higher proportion of fraudulent transactions but this score is lower as compared to the classification model from the 2nd experiment.

- Additionally, from the confusion matrix, the rate of False Negative errors, with 19% for training, 21% for validation and 21% is higher as compared to the False negative error rate of the model from experiment 2nd indicating that the MLPClassifier model is misclassifying a significant amount of fraudulent transactions as legitimate which might lead to financial losses and loss in customer's trust.

- The rate of False Positive errors that denotes misclassifying legitimate transactions as fraudulent, resulted in 0.074% for training, 0.078% for validation and 0.081% is also higher in comparison to the model from 2nd experiment.

ETHICAL CONSIDERATION

While developing a classification model to classify fraudulent and non-fraudulent transactions using the provided dataset, there are potential ethical issues to consider. Some of these issues include.

Privacy: The dataset contains sensitive customer information, such as social security numbers, bank account numbers, credit card numbers, and residential addresses. Ensuring the privacy and security of this data is crucial to protect customer confidentiality.

Bias: The inclusion of features like gender, and residential location might introduce biases into the model. Care must be taken to prevent any unfair discrimination based on these features.

Fairness: The model should be evaluated to ensure it does not disproportionately misclassify or target specific groups of customers, such as certain age groups, gender, or demographics.

Transparency: The model's decision-making process should be transparent and explainable. Customers have the right to understand why their transactions are flagged as fraudulent or non-fraudulent.

Data quality: Ensuring the accuracy, reliability, and completeness of the dataset is essential to prevent incorrect or misleading classifications.

Informed consent: If the data was collected from customers, obtaining their informed consent, and clearly communicating the purpose and use of their data is important.

Compliance: Adhering to relevant legal and regulatory frameworks, such as data protection laws and regulations, is crucial to maintain ethical standards.

It is essential to address these potential ethical issues throughout the development and deployment of the classification model to ensure fair and responsible use of the data and protect the interests of the customers.

3.6 Deployment

Based on the business objective of accurately classifying transactions as fraudulent and legitimate, and considering the performance scores from the model, it can be concluded that the Random Forest Model from the second experiment demonstrates superior performance.

With the capabilities of this model, combined with additional monitoring methods such as auditing or scrutiny by domain experts, the team could efficiently detect, manage, and mitigate instances of fraudulent transactions.

With this model deployed to identify fraudulent transactions, the bank can take appropriate actions in real-time, such as blocking the transaction, notifying the customer, or launching an investigation. The compliance team can then use the classifications generated by the model as an additional tool to aid their decision-making process and improve the bank's overall fraud detection capabilities.

Thus, the classification problem is crucial for preventing financial losses and maintaining the trust and security of the bank's customers.

4. Unsupervised learning (clustering) and Regression - Tahmidul Islam

4.1 Business Understanding

Sales is one of the front liners of business expansion. In order for an organization, such as a bank, to grow it is imperative that the sales team is successful in bringing in new customers to its service circle and scale accordingly. However, a sales campaign is often to go to waste in terms of time, energy and money if the approach is not effective enough. The response out of the effort would go nowhere since it can not capture the trends of the market and attract the attention of service-seekers. Thus a focused and strategized approach is necessary to ensure optimality.

The purpose of finding the frequency of customers is one of the approach to understand consumer behavior. Over a period of time, how often they buy products, what influences them to do so and how much they end up spending are some important things to consider when defining any sales objective. In this project we are going to aid the sales team target customers better.

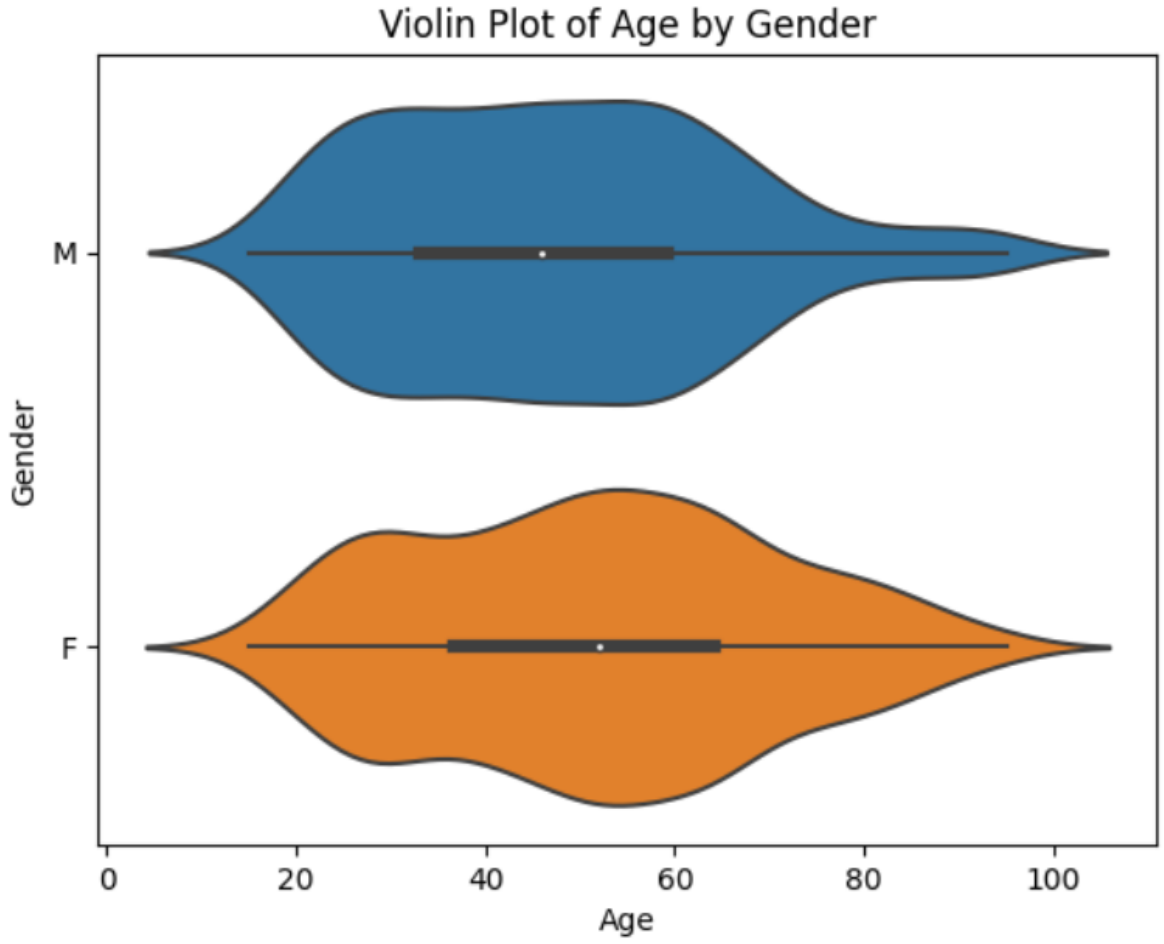
The sources available to us are customer data of 1000 users, and their personal and demographic information. In addition, we also have almost 4 million cases of transaction data that are produced by the customers everytime they purchase something. These are actually somethings that the project requires. The data we have is sufficient to run experiments on our Machine Learning models and get a deeper understanding of purchase behavior. The risk in this project is that the trends and predictions we find from our experiments may not 100 percent hold in the future, given that the global situation is ever so volatile. But the study will definitely provide some insights and suggestions that, if applied, may produce a significant change. Thus the study is worthwhile doing.

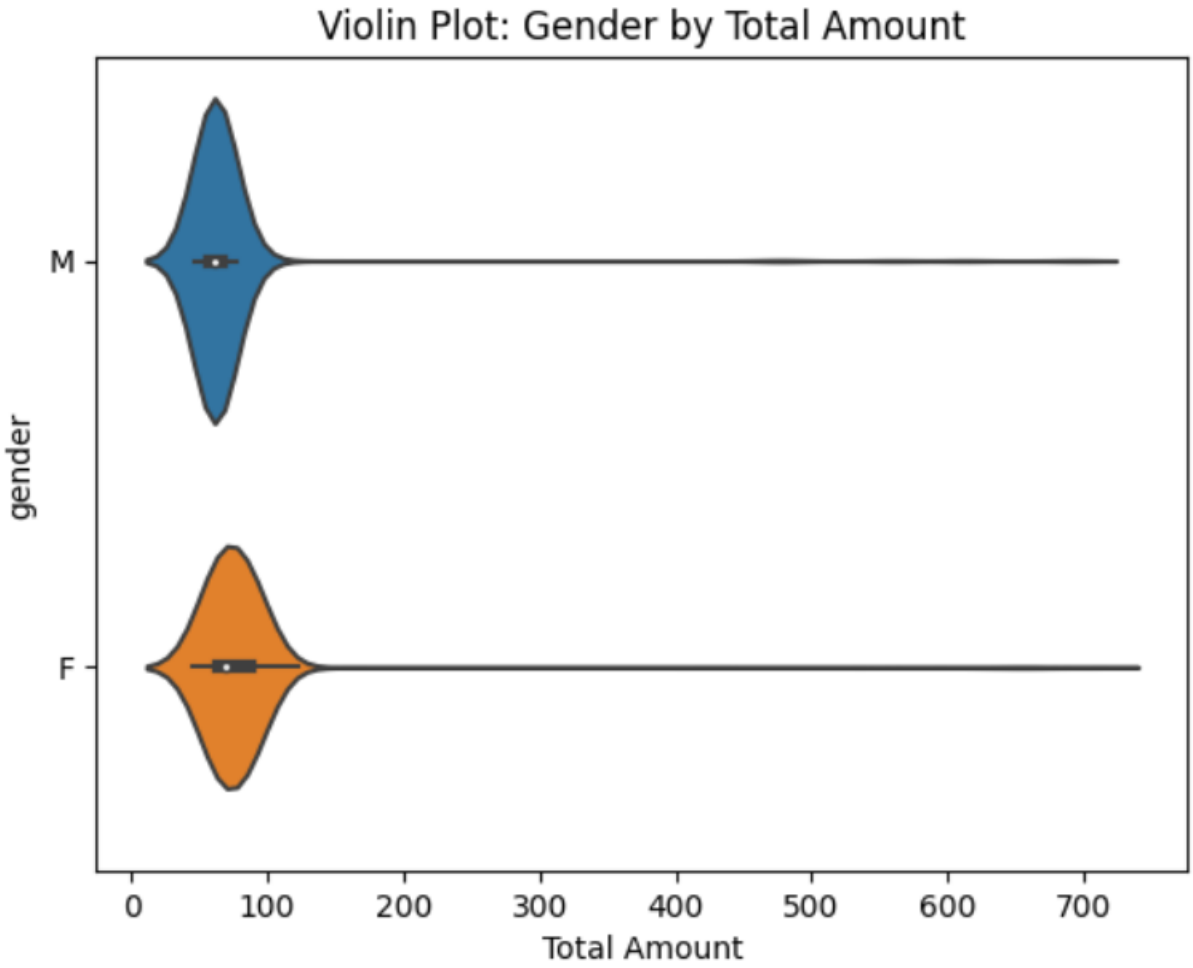
From the technical point of view, success would mean that our accuracy scores are high, we are getting numeric values and graphical contents that are not ambiguous or vague and it is providing some clear pictures to us.

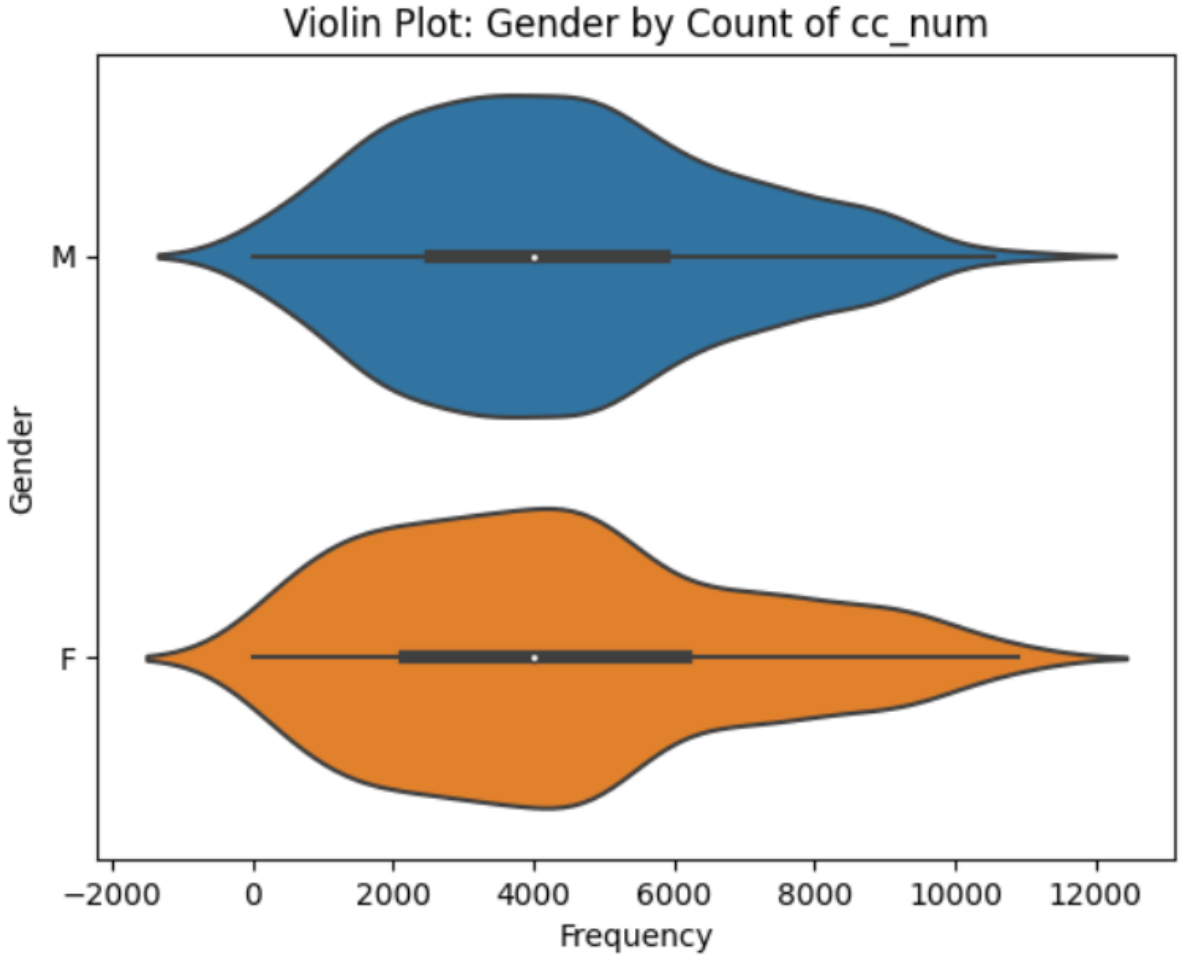
The plan of the project will be to use Python and certain machine learning libraries to clean the data as the initial step and then move forward to do feature engineering and eventually apply machine learning models on the data to run experiments and draw conclusions.

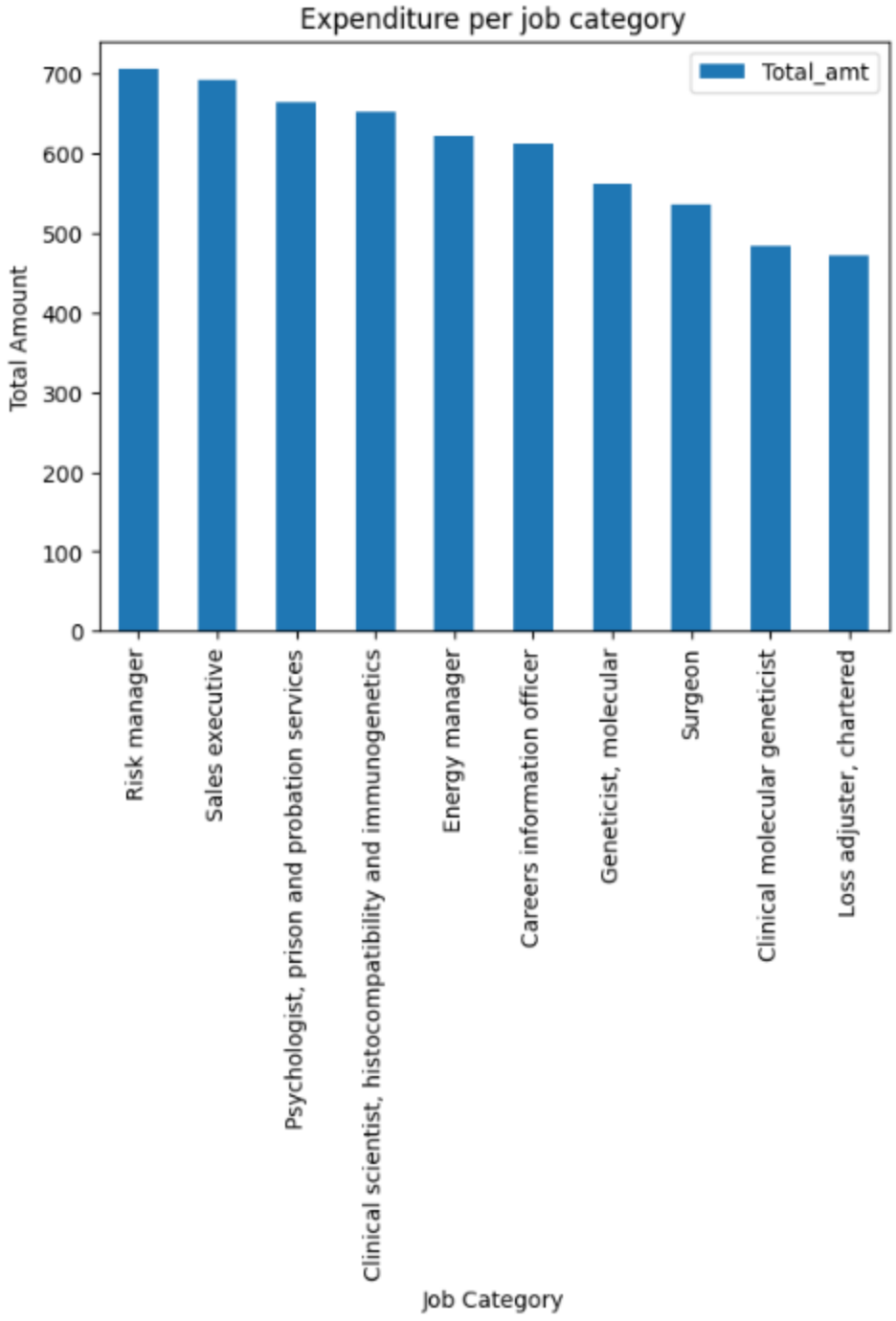
4.2 Data Understanding

At first we collect data. Fortunately for us, the data are provided in two excel sheets – one containing personal information of 1000 customers, and the other containing 4 million rows of transactional data produced by the 1000 customers. We are going to load them onto python.




Violin Plot: Gender by Total Amount
gender
M
F
0
100
200
300
400
500
600
700
Total Amount


Violin Plot: Gender by Count of cc_num
Gender
M
F
−2000
0
2000
4000
6000
8000
10000
12000
Frequency



4.3 Data Preparation

We are going to merge the two files given to us, but drop a lot of the columns. The dropped columns will be mostly categorical data that have a huge range of unique values which will be too cumbersome to encode to numeric values. Not only will this save a lot of computation time, but will also prevent a lot of over fitting when we run our models.

As part of data construction, we are going to create a column called ‘frequency’ which will count the presence of Credit Card Numbers that have appeared on the transaction file. This means that each time a certain person makes a purchase, the frequency count profiling him will increase by 1.

We are also creating two other variables called Recency and Monetary. The first one is a measure of how recently a transaction was made. This will be calculated by subtracting the time of that transaction from the latest instant of the transaction in the file.

Then data will be formatted by ensuring the data ranges and mean values are similar for all. This is called scaling the data.

4.4 Modeling

The first part of the project will have Regression Algorithms. We will use Decision Tree Regression, Random forest Regression and Univariate Linear Regression. In the Second half, we will apply Kmeans clustering.

For the regression part, the dataset will be splitted to train and test in 80:20 ratio. Following this, the Regression models will be built by fitting on the training dataset.

Next, we are going to analyse the performance of regression by predicting values and comparing with the test values.

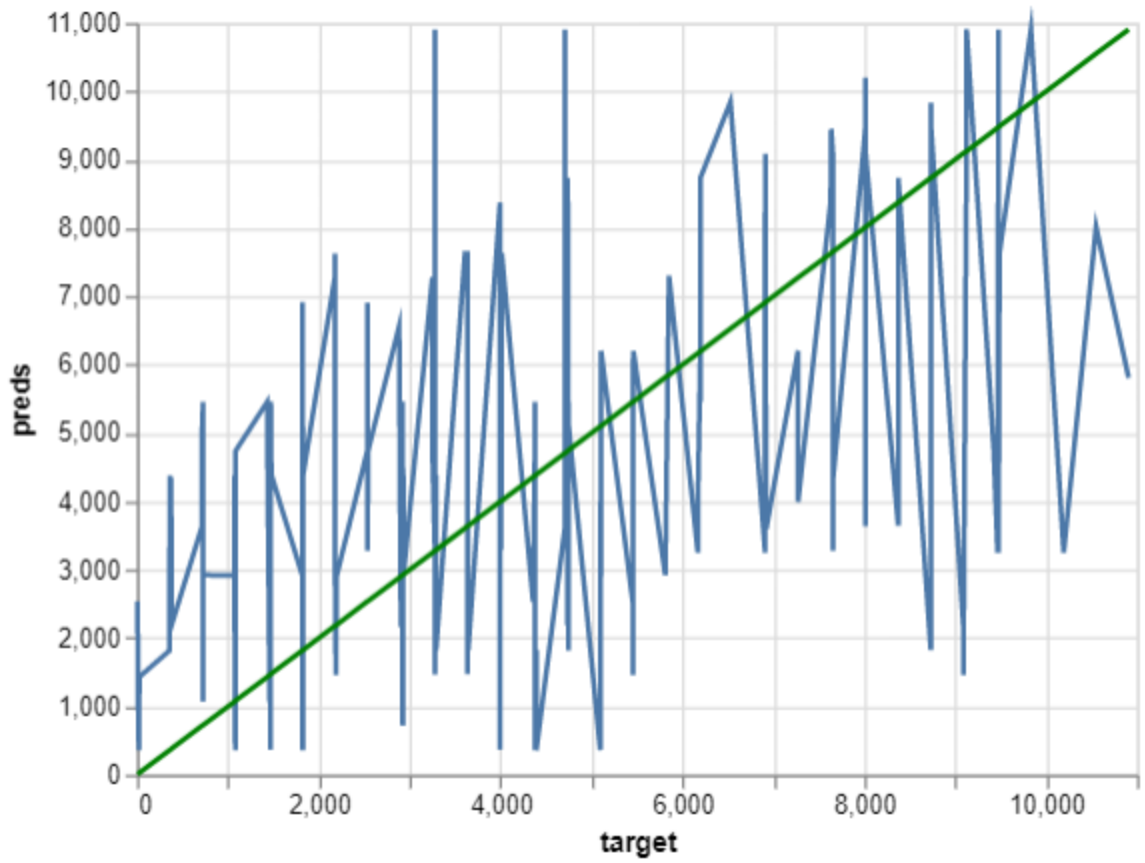
For Clustering algorithms, it will be a few steps of building the Kmeans model, find the optimum number of clusters, and apply those clusters to our dataset.

4.5 Evaluation

In the linear regression model, Decision Tree Regression gives us an accuracy score that gets improved by Random Forest Regression. Following this, we find the parameters that most closely signify the target variable. We have found that age was that parameter. Having found this, we did a univariate linear regression keeping with frequency on Target, and age as the independent variable. The scores obtained were very similar to that from Random Forest, indicating that age was indeed the most relevant parameter.

```
print(mse(y_test, y_pred, squared=False))
print(mae(y_test, y_pred))
```

```
2280.275276242628
1946.3928980021135
```



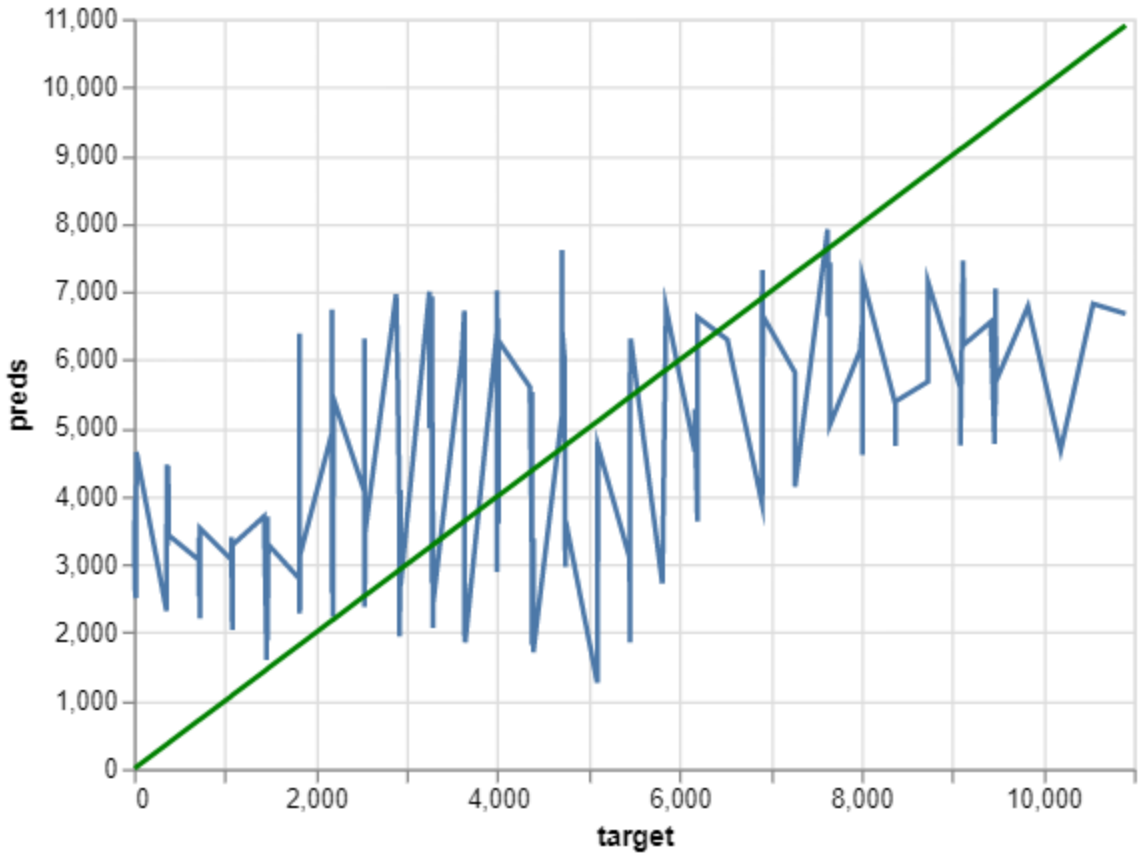
	Feature	Importance
0	gender	0.015108
1	lat	0.266169
2	long	0.229042
3	age	0.489681

Random Forest

```
print(mse(y_test, y_pred, squared=False))
print(mae(y_test, y_pred))
```

```
2298.579173978004
1966.572098138748
```

|:

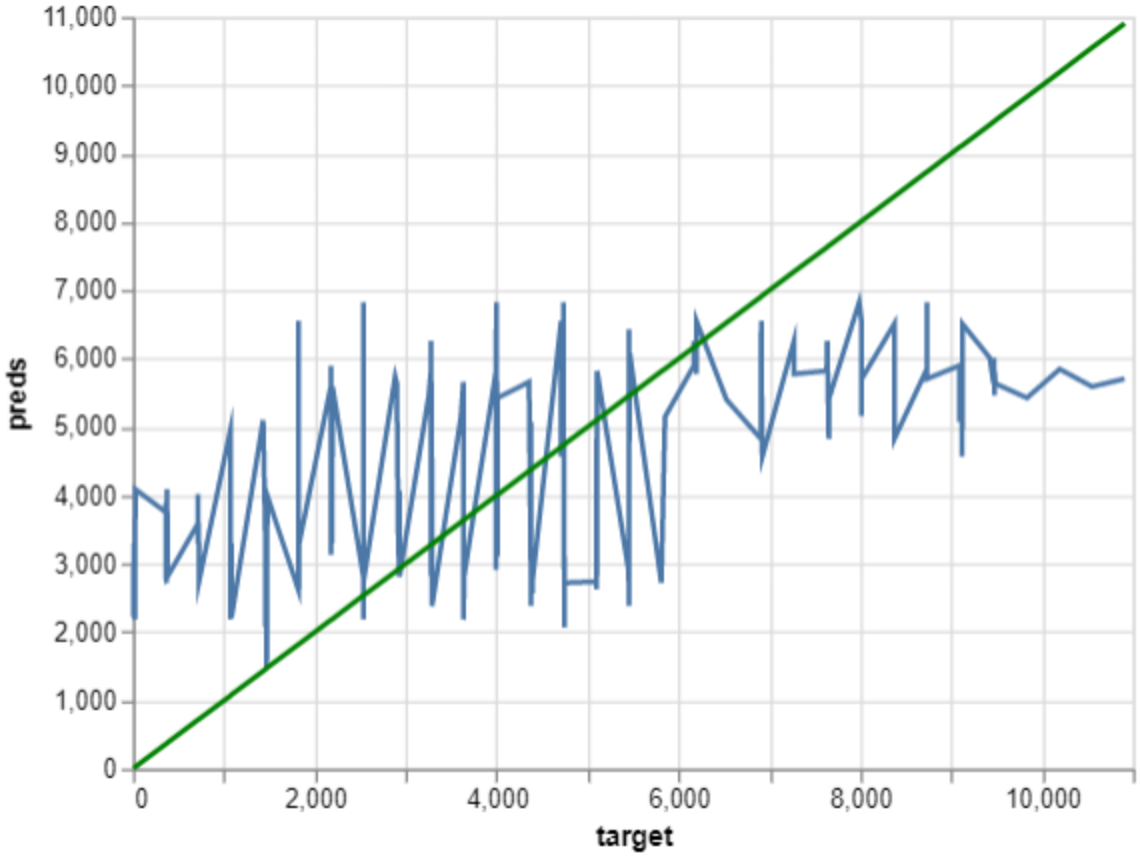


	Feature	Importance
0	gender	0.027430
1	lat	0.257104
2	long	0.242625
3	age	0.472840

Univariate Linear Rergession

```
print(mse(y_test, y_pred, squared=False))
print(mae(y_test, y_pred))
```

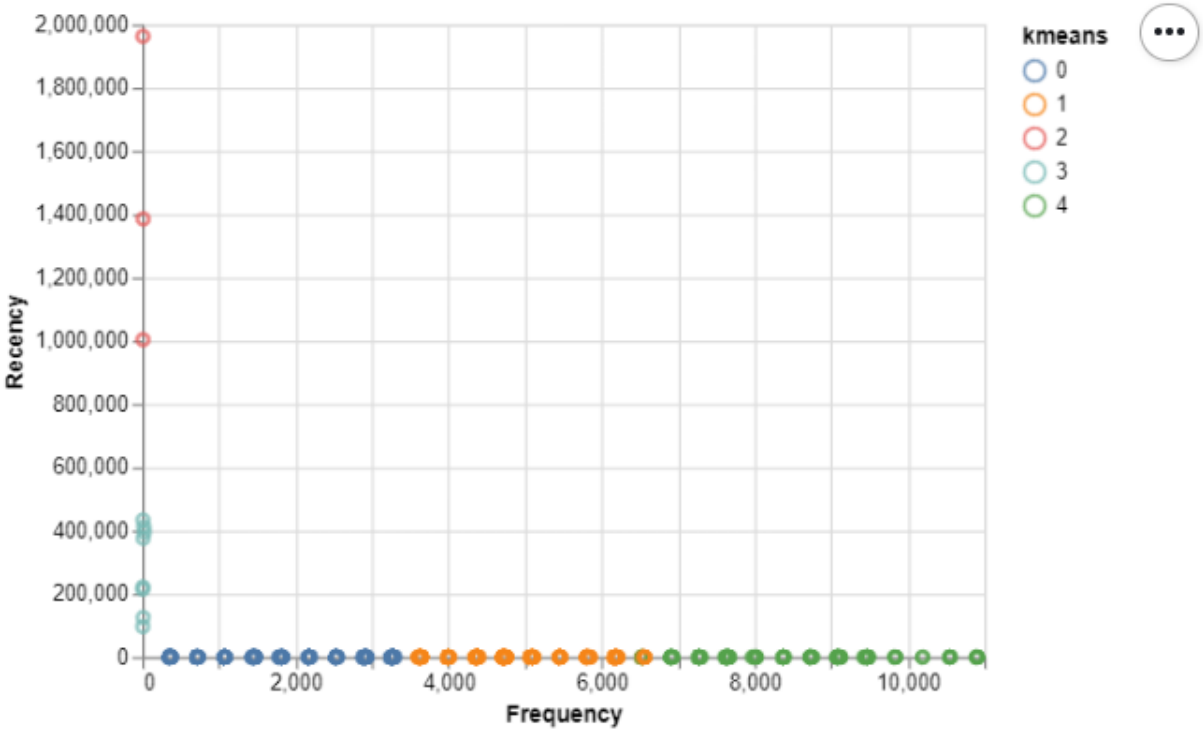
```
2278.6796873658463
1942.404087205746
```

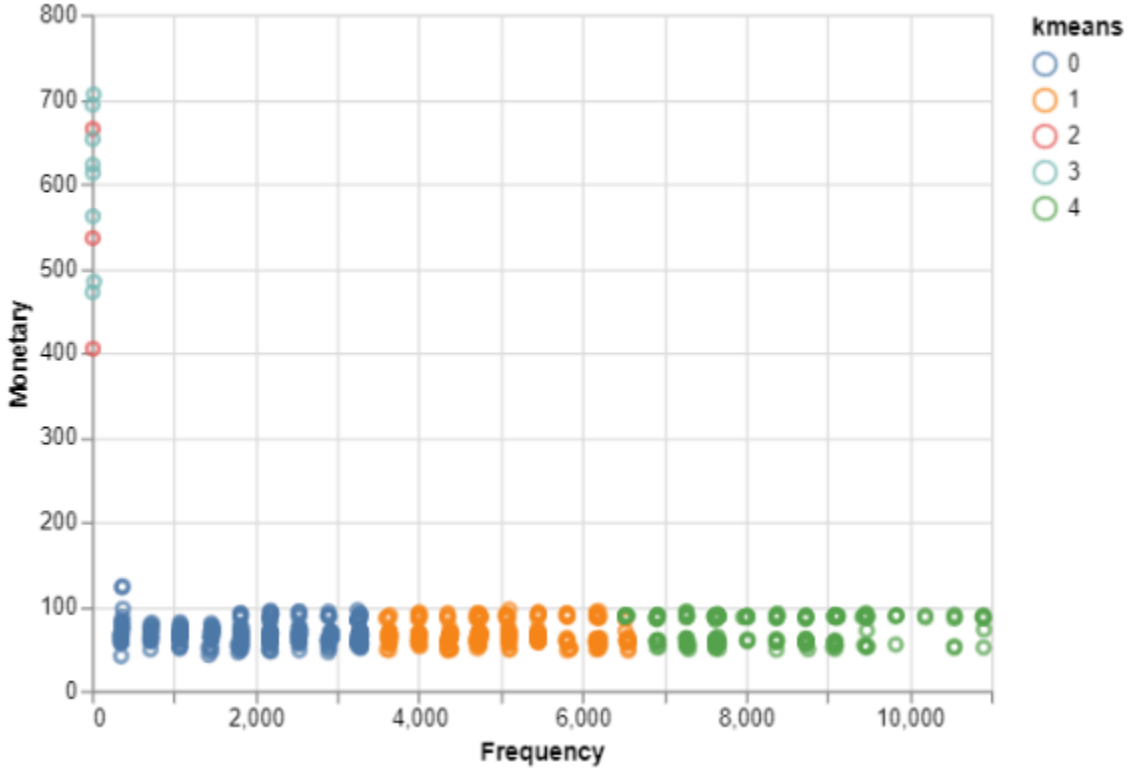


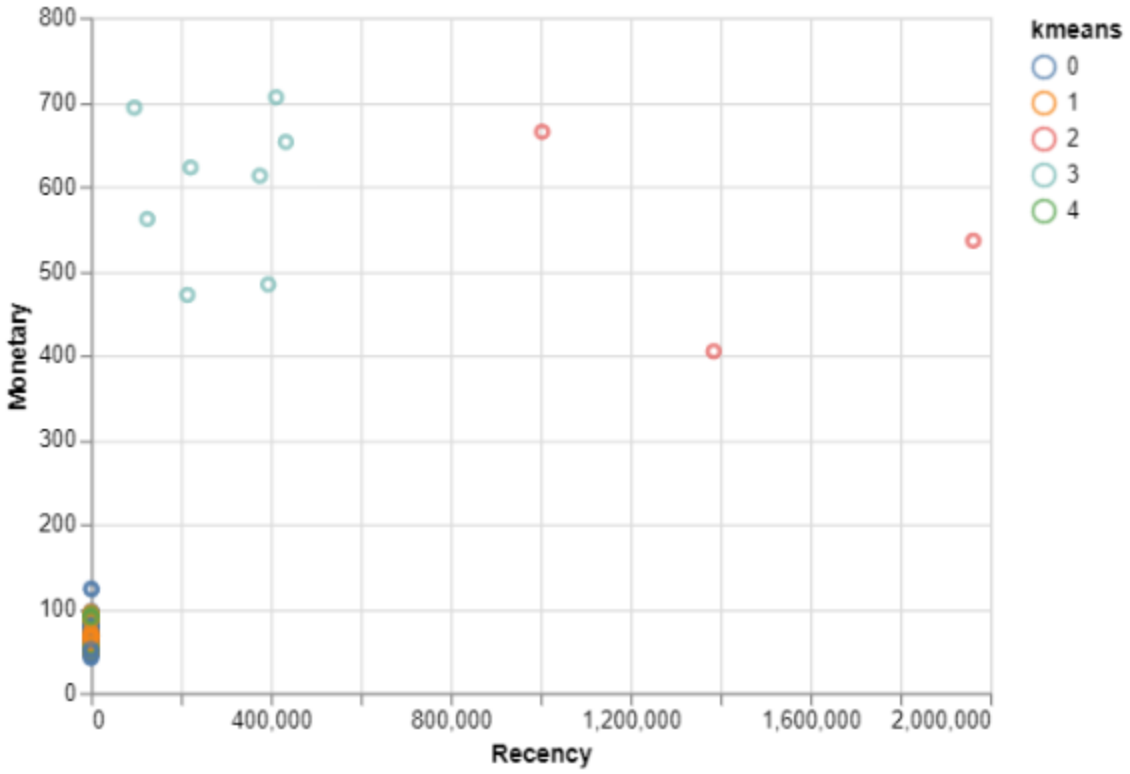
For our clustering model, we have obtained 4 neat sets of clusters with frequency and expense amount on the axes. This gives some information on whether the frequency of shopping can determine the amount of expenditure that will be done. The result is however, anticlimactic. It shows us that all the customers have a similar expense regardless of how frequent they are.

There are some outliers to this however. It shows that some customers have very low frequencies of shopping, but their expense is way higher than the average. This is actually for fraudulent transactions because someone was trying to steal money from the account.

The next step of our project is to decide whether we can deploy the models we have implemented.


kmeans
0
1
2
3
4
Recency
2,000,000
1,800,000
1,600,000
1,400,000
1,200,000
1,000,000
800,000
600,000
400,000
200,000
0
0
2,000
4,000
6,000
8,000
10,000
Frequency


kmeans
0
1
2
3
4
Monetary
800
700
600
500
400
300
200
100
0
0
2,000
4,000
6,000
8,000
10,000
Frequency

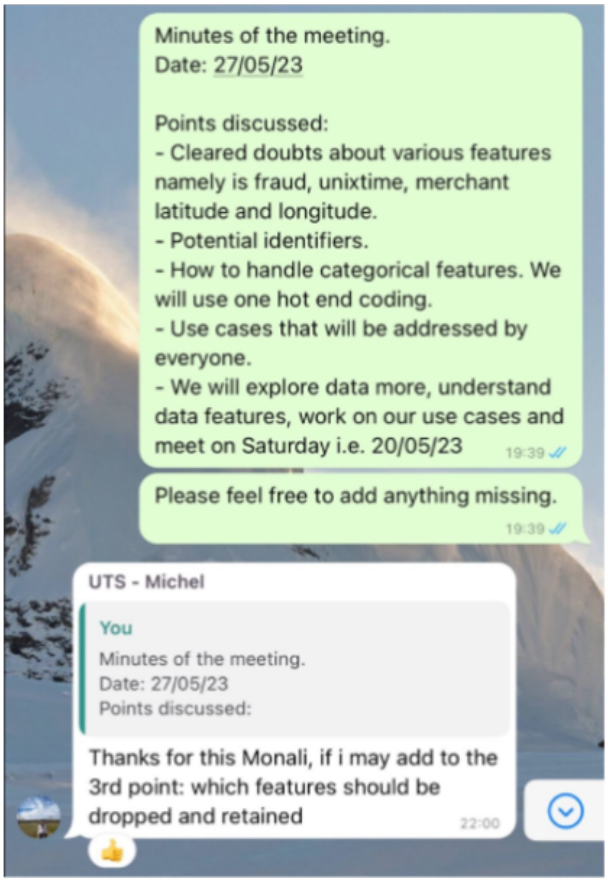


4.6 Deployment

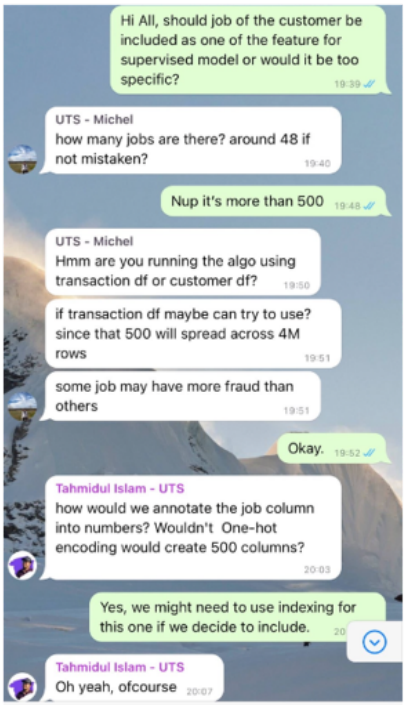
The models do meet the business requirement, albeit it does not meet the desired performance. However, it can be packaged into a deployable format and be uploaded into the cloud server. The model would need to be supervised constantly to ensure the performance is upto the level. This involves constant monitoring and testing for performance issues and coming up with new versions to keep the model updated.

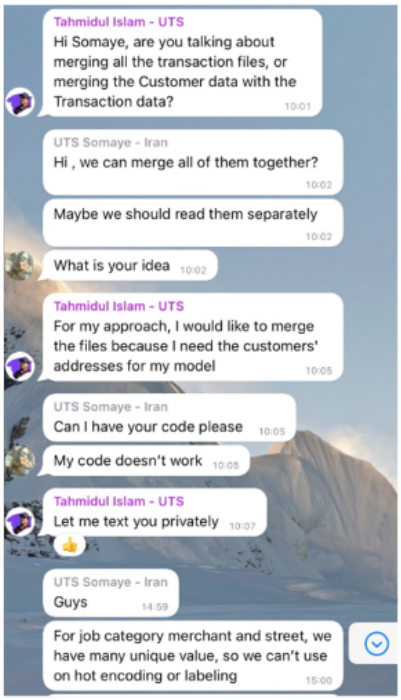
5. Minutes of meeting

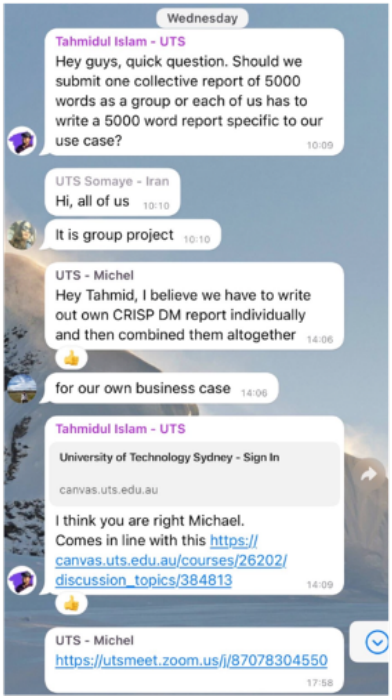
No	Date	Discussion
1	12 May	• Introduction and preliminary discussion on tasks allocation
2	17 May	• Tasks allocation - 1 machine learning technique for each member • Clarification on what each features mean • Ideas on how to handle categorical features • Deep dive into the data and come up with questions and ideas for next meeting
3	22 May	• Discussed progress and roadblocks • Shared model performances and potential ways to improve them
4	24 May	• Final checkpoints on progress and cleared any last minute roadblocks
5	26 May	• Combining our reports and finalizing the final version of the report



1. Queries and other discussions related to the project.







6. Contributions

Github Repository: https://github.com/mcyaputra/MLAA_AT3

Name	Use Cases Handled
Monali Patil	Classification ● XGBoost (eXtreme Gradient Boosting) ● Random Forest ● Random Forest with Important Features ● MLPClassifier Neural Networks
Tahmidul Islam	Unsupervised learning: ● Customer segmentation -KMeans ● Regression -Decision Tree Regression

	-Random Forest -Univariate Linear Regression
Michael Yaputra	Unsupervised learning: ● Customer segmentation: -KMeans -Hierarchical Clustering ● Anomaly detection: -Local Outlier Factor (LOF)
Somayeh Amraee	Regression analysis: ● Spending prediction

References:

1. Qualetics. (2019, October). *Data Science in Banking: 5 Use Cases for Banks.* https://qualetics.com/data-science-in-banking-5-use-cases-for-banks/
2. ActiveWizards. *Top 9 Data Science Use Cases in Banking.* https://activewizards.com/blog/top-9-data-science-use-cases-in-banking/
3. USDSI. *The Growing Role of Data Science Professionals in Banking.* https://www.usdsi.org/data-science-insights/the-growing-role-of-data-science-professionals-in-banking#:~:text=Instead%20of%20manually%20calculating%20the,risks%20and%20classify%20their%20defaulters.&text=Banks%20always%20require%20a%20360%2Ddegree%20analysis%20of%20customers.
4. Australian Banking Association. (2023). *Banking by Numbers.* https://www.ausbanking.org.au/insight/banking-by-numbers/
5. Statista. (2023, March). *Largest banks in Australia in the first half of 2022, by assets.* https://www.statista.com/statistics/434596/leading-banks-in-australia-assets/
6. myNZTE. (2022, Dec). *The banking sector in Australia: facts, figures and trends.* https://my.nzte.govt.nz/article2/the-banking-sector-in-australia-facts-figures-and-trends#:~:text=Types%20of%20banking,and%20employ%20about%20198%2C000%20people.
7. Australian Government Treasury. (2016, March). *The strength of Australia's financial sector.* https://treasury.gov.au/publication/backing-australian-fintech/the-strength-of-australias-financial-sector