EDA No.2

AAA Project - Martin George mgeorgevienna@gmail.com

In [77]:
```
%matplotlib inline
import matplotlib.pyplot as plt
import numpy as np
import pandas as pd
df = pd.read_csv('member_sample_step_01.csv', index_col = 0)
```

Application of classification model on AAA data

In [78]:
```
df.head()
df.info()
df.columns
```

```
<class 'pandas.core.frame.DataFrame'>
Float64Index: 21344 entries, 10000003.0 to 28273400.0
Columns: 112 entries, Household Key to Date
dtypes: float64(34), int64(1), object(77)
memory usage: 18.4+ MB
```

Out[78]:
```
Index(['Household Key', 'Member Flag', 'City', 'State - Grouped', 'ZIP5',
       'ZIP9', 'FSV CMSI Flag', 'FSV Credit Card Flag',
       'FSV Deposit Program Flag', 'FSV Home Equity Flag',
       ...
       'SC Vehicle Model Name', 'SVC Facility Name', 'SVC Facility Type',
       'Total Cost', 'Tow Destination Latitude', 'Tow Destination Longitude',
       'Tow Destination Name', 'Was Duplicated', 'Was Towed To AAR Referral',
       'Date'],
      dtype='object', length=112)
```

In [79]:
```
df.head()
```

Out[79]:

Individual Key	Household Key	Member Flag	City	State - Grouped	ZIP5	ZIP9	FSV CMSI Flag	FSV Credit Card Flag	FSV Deposit Program Flag	FSV Home Equity Flag	...	SC Vehicle Model Name	SVC Facility Name	SVC Facility Type	To Co
10000003.0	10462590.0	Y	NEW HAVEN	CT	6511.0	65111349.0	0	N	N	N	...	NaN	NaN	NaN	N.
52211550.0	4500791.0	Y	WEST WARWICK	RI	2893.0	28933850.0	0	Y	N	N	...	CAMRY	ASTRO WRECKER SERVICE	independent repair	3:
52211550.0	4500791.0	Y	WEST WARWICK	RI	2893.0	28933850.0	0	Y	N	N	...	CAMRY	Astro Wrecker Service	independent repair	3(
52211550.0	4500791.0	Y	WEST WARWICK	RI	2893.0	28933850.0	0	Y	N	N	...	CAMRY	ASTRO WRECKER SERVICE	independent repair	3:
52211550.0	4500791.0	Y	WEST WARWICK	RI	2893.0	28933850.0	0	Y	N	N	...	CAMRY	ASTRO WRECKER SERVICE	independent repair	3(

5 rows × 112 columns

In [80]:

```
df['City'].value_counts()[:3].sum()
df['City'].value_counts()
```

```
Out[80]: WARWICK                 2815
         CRANSTON                2742
         PROVIDENCE              1690
         NORTH PROVIDENCE         948
         COVENTRY                 778
         JOHNSTON                 703
         EAST GREENWICH           641
         PAWTUCKET                640
         NORTH KINGSTOWN          637
         WEST WARWICK             606
         CUMBERLAND               560
         NARRAGANSETT             531
         WAKEFIELD                497
         RIVERSIDE                436
         LINCOLN                  402
         EAST PROVIDENCE          372
         PORTSMOUTH               328
         MANVILLE                 322
         NEWPORT                  321
         BARRINGTON               313
         GREENVILLE               288
         SMITHFIELD               284
         BRISTOL                  279
         MIDDLETOWN               272
         RUMFORD                  268
         WOONSOCKET               262
         TIVERTON                 253
         NORTH SCITUATE           222
         JAMESTOWN                218
         NORTH SMITHFIELD         179
                                 ...
         BRIDGEPORT                 2
         RICHMOND                   2
         NORTH GROSVENORDALE        2
         QUINEBAUG                  2
         NEW MILFORD                2
         EAST LYME                  1
         HADDAM                     1
         HAMDEN                     1
         BRANFORD                   1
         PLAINFIELD                 1
         WETHERSFIELD               1
         WILTON                     1
         VERNON ROCKVILLE           1
         ANSONIA                    1
         COLLINSVILLE               1
         FARMINGTON                 1
```

```
SALEM                    1
SLOCUM                   1
NO SMITHFIELD            1
DEEP RIVER               1
WEST SIMSBURY            1
ALBION                   1
AVON                     1
CLINTON                  1
PORTLAND                 1
NAUGATUCK                1
WEST HAVEN               1
SHELTON                  1
GLASTONBURY              1
STAMFORD                 1
Name: City, Length: 167, dtype: int64
```

In [81]:
```
df['State - Grouped'].value_counts()
```

Out[81]:
```
RI    20937
CT      407
Name: State - Grouped, dtype: int64
```

In [82]:
```
df['FSV Mortgage Flag'].value_counts()
```

Out[82]:
```
N    21317
Y       27
Name: FSV Mortgage Flag, dtype: int64
```

A function to digitise the column

In [83]:
```
def y_n_to_binary(x):
    if x == 'Y':
        return 1
    else:
        return 0
```

In [84]:
```
df['FSV Mortgage Flag'] = df['FSV Mortgage Flag'].apply(y_n_to_binary)
```

In [85]:
```
df['FSV Mortgage Flag'].value_counts()
```

Out[85]:
```
0    21317
1       27
Name: FSV Mortgage Flag, dtype: int64
```

In [86]:
```
#[for i in df.columns in i.startswith('Children')]
```

In [87]:
```
df['Children'].value_counts()
```

Out[87]:
```
Yes    6631
No     5379
Name: Children, dtype: int64
```

In [88]:
```
df.groupby("Children")["Number of Children"]
```

Out[88]:
```
<pandas.core.groupby.groupby.SeriesGroupBy object at 0x000001D53A474320>
```

In [89]:

```
df["Number of Children"]
```

Out[89]:
```
Individual Key
10000003.0                 NaN
52211550.0           One Child
52211550.0           One Child
52211550.0           One Child
52211550.0           One Child
52211550.0           One Child
52211550.0           One Child
52211550.0           One Child
52211550.0           One Child
52211550.0           One Child
1606764.0                  NaN
2766867.0          No children
2766867.0          No children
2766867.0          No children
2766869.0          No children
2766868.0          No children
2766868.0          No children
2766868.0          No children
2766868.0          No children
2766868.0          No children
13746947.0         No children
1788453.0            One Child
1788452.0            One Child
1788452.0            One Child
1788452.0            One Child
1788452.0            One Child
1788455.0            One Child
14243585.0       Four Children
14243587.0      Three Children
14243587.0      Three Children
                           ...
4458026.0                  NaN
12849942.0           One Child
12849942.0           One Child
12849942.0           One Child
12849942.0           One Child
12849942.0           One Child
12849942.0           One Child
12849941.0                 NaN
22426406.0      Three Children
22426405.0      Three Children
19764804.0                 NaN
19764802.0                 NaN
19764801.0                 NaN
19764793.0           One Child
16521338.0        Two Children
```

```
16521338.0      Two Children
16521338.0      Two Children
16521338.0      Two Children
16521338.0      Two Children
16521336.0      Two Children
1619870.0       Two Children
1619868.0                NaN
1619869.0                NaN
54745437.0      Two Children
25797262.0               NaN
25797262.0               NaN
25797262.0               NaN
28273400.0      Two Children
28273400.0      Two Children
28273400.0      Two Children
Name: Number of Children, Length: 21344, dtype: object
```

In [90]:
```
df.groupby('Children')['Number of Children'].value_counts()
```

Out[90]:
```
Children  Number of Children
No        No children           5379
Yes       One Child             3871
          Two Children          1582
          Three Children         750
          Four Children          276
          No children            103
          Five Children           45
          Six Children             4
Name: Number of Children, dtype: int64
```

Conclusion: Validity of data is required. With the flag "Y" for children, still there are 103 records with "No Children" description.

In [91]:
```
df['Number of Children'].value_counts()
```

Out[91]:
```
No children       5482
One Child         3871
Two Children      1582
Three Children     750
Four Children      276
Five Children       45
Six Children         4
Name: Number of Children, dtype: int64
```

In [92]:
```
vals = df['Number of Children'].value_counts().index
print(vals)
nums = list(range(7))
nums
```

```
Index(['No children', 'One Child', 'Two Children', 'Three Children',
       'Four Children', 'Five Children', 'Six Children'],
      dtype='object')
```

Out[92]:
```
[0, 1, 2, 3, 4, 5, 6]
```

Converting string description of "no of children" to an equivalent number value.

In [93]:
```
df['Number of Children'] = df['Number of Children'].map({v:n for v,n in zip(vals,nums)})
```

In [94]:

```
df[['Children','Number of Children']]
```

Out[94]:

Individual Key	Children	Number of Children
10000003.0	NaN	NaN
52211550.0	Yes	1.0
52211550.0	Yes	1.0
52211550.0	Yes	1.0
52211550.0	Yes	1.0
52211550.0	Yes	1.0
52211550.0	Yes	1.0
52211550.0	Yes	1.0
52211550.0	Yes	1.0
52211550.0	Yes	1.0
1606764.0	NaN	NaN
2766867.0	No	0.0
2766867.0	No	0.0
2766867.0	No	0.0
2766869.0	No	0.0
2766868.0	No	0.0
2766868.0	No	0.0
2766868.0	No	0.0
2766868.0	No	0.0
2766868.0	No	0.0
13746947.0	No	0.0
1788453.0	Yes	1.0
1788452.0	Yes	1.0
1788452.0	Yes	1.0
1788452.0	Yes	1.0
1788452.0	Yes	1.0
1788455.0	Yes	1.0
14243585.0	Yes	4.0

	Children	Number of Children
Individual Key		
14243587.0	Yes	3.0
14243587.0	Yes	3.0
...	...	...
4458026.0	NaN	NaN
12849942.0	Yes	1.0
12849942.0	Yes	1.0
12849942.0	Yes	1.0
12849942.0	Yes	1.0
12849942.0	Yes	1.0
12849942.0	Yes	1.0
12849941.0	NaN	NaN
22426406.0	Yes	3.0
22426405.0	Yes	3.0
19764804.0	NaN	NaN
19764802.0	NaN	NaN
19764801.0	NaN	NaN
19764793.0	Yes	1.0
16521338.0	Yes	2.0
16521338.0	Yes	2.0
16521338.0	Yes	2.0
16521338.0	Yes	2.0
16521338.0	Yes	2.0
16521336.0	Yes	2.0
1619870.0	Yes	2.0
1619868.0	NaN	NaN
1619869.0	NaN	NaN
54745437.0	Yes	2.0
25797262.0	NaN	NaN
25797262.0	NaN	NaN

	Children	Number of Children
Individual Key		
25797262.0	NaN	NaN
28273400.0	Yes	2.0
28273400.0	Yes	2.0
28273400.0	Yes	2.0

21344 rows × 2 columns

In [95]:

```
df.loc[(df['Number of Children'] == 0) & (df['Children'] == 'Yes')]
```

Out[95]:

Individual Key	Household Key	Member Flag	City	State - Grouped	ZIP5	ZIP9	FSV CMSI Flag	FSV Credit Card Flag	FSV Deposit Program Flag	FSV Home Equity Flag	...	SC Vehicle Model Name	SVC Facility Name	Fa
6286759.0	10625447.0	Y	CHARLESTOWN	RI	2813.0	28133901.0	0	N	N	N	...	NaN	NaN	
30250103.0	5550046.0	Y	CUMBERLAND	RI	2864.0	28643545.0	0	N	N	N	...	NaN	NaN	
30250103.0	5550046.0	Y	CUMBERLAND	RI	2864.0	28643545.0	0	N	N	N	...	NaN	NaN	
10266781.0	10473088.0	Y	WARWICK	RI	2889.0	28896214.0	0	N	N	N	...	NaN	NaN	
10125674.0	49275813.0	Y	CRANSTON	RI	2910.0	29103541.0	0	N	N	N	...	CR-V	Davis Towing	tov f
10125674.0	49275813.0	Y	CRANSTON	RI	2910.0	29103541.0	0	N	N	N	...	CRV	Pat's Towing	tov f
10125674.0	49275813.0	Y	CRANSTON	RI	2910.0	29103541.0	0	N	N	N	...	VIBE	AAA SNE RI FLEET FULL SERVICE	
10125674.0	49275813.0	Y	CRANSTON	RI	2910.0	29103541.0	0	N	N	N	...	VIBE	AAA SNE RI FLEET FULL SERVICE	
24070437.0	20249145.0	Y	WARWICK	RI	2886.0	28867529.0	0	N	N	N	...	NaN	NaN	
26306028.0	63157220.0	Y	PROVIDENCE	RI	2906.0	29065300.0	0	N	N	N	...	NaN	NaN	
13285893.0	3399486.0	Y	COVENTRY	RI	2816.0	28168456.0	0	N	N	N	...	NaN	NaN	
26208448.0	5349292.0	Y	LITTLE COMPTON	RI	2837.0	28371933.0	0	N	N	N	...	NaN	NaN	
33371516.0	14490039.0	Y	MIDDLETOWN	RI	2842.0	28427506.0	0	N	N	N	...	NaN	NaN	
33371518.0	14490039.0	Y	MIDDLETOWN	RI	2842.0	28427506.0	0	N	N	N	...	NaN	NaN	
17494924.0	687773.0	Y	PROVIDENCE	RI	2906.0	29065182.0	0	N	N	N	...	NaN	NaN	
5960222.0	252739.0	Y	WARWICK	RI	2886.0	28867910.0	0	N	N	N	...	NaN	NaN	
12909537.0	7658612.0	Y	LINCOLN	RI	2865.0	28651629.0	0	N	N	N	...	NaN	NaN	
25510097.0	17196894.0	Y	PROVIDENCE	RI	2906.0	29064807.0	0	N	N	N	...	NaN	NaN	
25510099.0	17196894.0	Y	PROVIDENCE	RI	2906.0	29064807.0	0	N	N	N	...	ES300	Aaa Sne Ri Light Service	n b se

	Household Key	Member Flag	City	State - Grouped	ZIP5	ZIP9	FSV CMSI Flag	FSV Credit Card Flag	FSV Deposit Program Flag	FSV Home Equity Flag	...	SC Vehicle Model Name	SVC Facility Name	Fa
Individual Key														
25510099.0	17196894.0	Y	PROVIDENCE	RI	2906.0	29064807.0	0	N	N	N	...	ES300	Aaa Sne Ri Light Service	n b se
25510098.0	17196894.0	Y	PROVIDENCE	RI	2906.0	29064807.0	0	N	N	N	...	NaN	NaN	
25510098.0	17196894.0	Y	PROVIDENCE	RI	2906.0	29064807.0	0	N	N	N	...	NaN	NaN	
102130.0	14500942.0	Y	LINCOLN	RI	2865.0	28653607.0	0	N	N	N	...	NaN	NaN	
10024932.0	6692943.0	Y	CRANSTON	RI	2921.0	29212747.0	0	N	N	N	...	NaN	NaN	
17436941.0	7865970.0	Y	CRANSTON	RI	2920.0	29201912.0	0	N	N	N	...	GRAND CARAVAN	Aaa Sne Ri Light Service	n b se
184280197.0	1663584.0	Y	WARWICK	RI	2889.0	28898532.0	0	N	N	N	...	NaN	NaN	
32355478.0	19055457.0	Y	GREENVILLE	RI	2828.0	28281469.0	0	N	N	N	...	NaN	NaN	
1171756.0	9940580.0	Y	MIDDLETOWN	RI	2842.0	28425620.0	0	N	N	N	...	XL-7	RAY'S TOWING	indepe
1171756.0	9940580.0	Y	MIDDLETOWN	RI	2842.0	28425620.0	0	N	N	N	...	VUE	RAY'S TOWING	indepe
1171756.0	9940580.0	Y	MIDDLETOWN	RI	2842.0	28425620.0	0	N	N	N	...	XL-7	RHODE ISLAND TOWING	indepe
...	...	...	...	...	...	...	...	...	...	...	...	...	...	
23321408.0	9573428.0	Y	PROVIDENCE	RI	2907.0	29072150.0	1	N	N	N	...	CORSICA	AAA SNE RI LIGHT SERVICE	
23321408.0	9573428.0	Y	PROVIDENCE	RI	2907.0	29072150.0	1	N	N	N	...	A4 QUATTRO	Aaa Sne Ri Fleet Full Service	n b se
21610725.0	10961810.0	Y	WOONSOCKET	RI	2895.0	28951946.0	0	N	N	N	...	NaN	NaN	
8610352.0	3202564.0	Y	JOHNSTON	RI	2919.0	29193322.0	0	N	N	N	...	NaN	NaN	
8971323.0	16430276.0	Y	WESTERLY	RI	2891.0	28912807.0	0	N	N	N	...	NaN	NaN	
15071042.0	10675058.0	Y	BRISTOL	RI	2809.0	28091501.0	0	N	N	N	...	NaN	NaN	
6850986.0	72111681.0	Y	COVENTRY	RI	2816.0	28165015.0	0	N	N	N	...	NaN	NaN	

Individual Key	Household Key	Member Flag	City	State - Grouped	ZIP5	ZIP9	FSV CMSI Flag	FSV Credit Card Flag	FSV Deposit Program Flag	FSV Home Equity Flag	...	SC Vehicle Model Name	SVC Facility Name	Fa
12675474.0	24921077.0	Y	NORTH PROVIDENCE	RI	2904.0	29043303.0	0	N	N	N	...	NaN	NaN	
31845691.0	5615630.0	Y	COVENTRY	RI	2816.0	28166945.0	0	N	N	N	...	CAMRY	Aaa Sne Ri Fleet Full Service	n b se
31845691.0	5615630.0	Y	COVENTRY	RI	2816.0	28166945.0	0	N	N	N	...	330XI	King's Service Center	se s
31845691.0	5615630.0	Y	COVENTRY	RI	2816.0	28166945.0	0	N	N	N	...	330XI	King's Service Center	se s
31845691.0	5615630.0	Y	COVENTRY	RI	2816.0	28166945.0	0	N	N	N	...	CAMRY	ACHIN'S GARAGE	indepe I
195266817.0	12145070.0	Y	RIVERSIDE	RI	2915.0	29152446.0	0	N	N	N	...	MONTE CARLO	AAA SNE RI FLEET FULL SERVICE	n b se
195266817.0	12145070.0	Y	RIVERSIDE	RI	2915.0	29152446.0	0	N	N	N	...	MONTE CARLO	AAA SNE RI LIGHT SERVICE	n b se
10044179.0	19593345.0	Y	PORTSMOUTH	RI	2871.0	28713930.0	0	N	N	N	...	NaN	NaN	
22971029.0	15535491.0	Y	CRANSTON	RI	2921.0	29212333.0	0	N	N	N	...	NaN	NaN	
6987889.0	4560832.0	Y	LITTLE COMPTON	RI	2837.0	28371714.0	0	N	N	N	...	NaN	NaN	
8411744.0	37560807.0	Y	WARWICK	RI	2889.0	28896727.0	0	N	N	N	...	ES300	Aaa Sne Ri Light Service	n b se
8411744.0	37560807.0	Y	WARWICK	RI	2889.0	28896727.0	0	N	N	N	...	ACCORD	Aaa Sne Ri Light Service	n b se
8411744.0	37560807.0	Y	WARWICK	RI	2889.0	28896727.0	0	N	N	N	...	ES300	AAA SNE RI LIGHT SERVICE	n b se
8411744.0	37560807.0	Y	WARWICK	RI	2889.0	28896727.0	0	N	N	N	...	ES300	Aaa Sne Ri Light Service	n b se

Individual Key	Household Key	Member Flag	City	State - Grouped	ZIP5	ZIP9	FSV CMSI Flag	FSV Credit Card Flag	FSV Deposit Program Flag	FSV Home Equity Flag	...	SC Vehicle Model Name	SVC Facility Name	Fa
8411744.0	37560807.0	Y	WARWICK	RI	2889.0	28896727.0	0	N	N	N	...	ES300	AAA SNE RI FLEET FULL SERVICE	
8411744.0	37560807.0	Y	WARWICK	RI	2889.0	28896727.0	0	N	N	N	...	ACCORD	AAA SNE RI FLEET FULL SERVICE	n b se
8411744.0	37560807.0	Y	WARWICK	RI	2889.0	28896727.0	0	N	N	N	...	ES300	Aaa Sne Ri Light Service	n b se
8411744.0	37560807.0	Y	WARWICK	RI	2889.0	28896727.0	0	N	N	N	...	ES300	Aaa Sne Ri Light Service	n b se
1086728.0	4866654.0	Y	CRANSTON	RI	2920.0	29203822.0	0	N	N	N	...	ACCORD	PHIL HOWE'S TOWING	tov f
20247872.0	7984132.0	Y	WESTBROOK	CT	6498.0	64981690.0	0	N	N	N	...	NaN	NaN	
20247873.0	7984132.0	Y	WESTBROOK	CT	6498.0	64981690.0	0	N	N	N	...	FORESTER	New Beverly Auto Clinic Inc	indepe
20247873.0	7984132.0	Y	WESTBROOK	CT	6498.0	64981690.0	0	N	N	N	...	FORESTER	Aaa Mv Fleet	n b se
20247874.0	7984132.0	Y	WESTBROOK	CT	6498.0	64981690.0	0	N	N	N	...	NaN	NaN	

103 rows × 112 columns

In [96]:

```
for col in df.columns:
    print(col)
```

Household Key
Member Flag
City
State - Grouped
ZIP5
ZIP9
FSV CMSI Flag
FSV Credit Card Flag
FSV Deposit Program Flag
FSV Home Equity Flag
FSV ID Theft Flag
FSV Mortgage Flag
INS Client Flag
TRV Globalware Flag
Number of Children
Responded to Catalog
Race
Length Of Residence
Mail Responder
Home Owner
Income
Date Of Birth
Children
Education
Dwelling Type
Credit Ranges
Language
Gender
Active Expiration Date
Address Change Date
Bad Address Flag
Billing Code Description
Birth Date MMDDYYYY
Branch Name
Cancel Date
Cancel Reason
County
Do Not Direct Mail Solicit
Email Available
Email Status
ERS ENT Count Year 1
ERS ENT Count Year 2
ERS ENT Count Year 3
ERS Member Cost Year 1
ERS Member Cost Year 2
ERS Member Cost Year 3
Right_Gender

Right_Individual Key
Join AAA Date
Join Club Date
Member Key
Member Map Location
Member Number Associate ID
Member Phone Type
Member Status
Member Tenure Years
Member Type
Membership ID
Months from Join to Cancel
Opt-Out - Publication
Reason Joined
Reinstate Date
Renew Method
ZIP
Mosaic Household
Mosaic Global Household
kcl_B_IND_MosaicsGrouping
New Mover Flag
Occupation Code
Occupation Group
Right_Dwelling Type
Move Distance
Occupant Type
Breakdown Map Location
Breakdown City
Breakdown State
Basic Cost
Calculated Tow Miles
Call Canceled
Call Killed
Call Status Recv Date
Cash Call
Clearing Code Last Description
Dispatch Code1 Description
Dispatch Code2Description
DTL Prob1 Code Description
Fleet Indicator
Is Duplicate
Is NSR
Member Match Flag
Member Number and Associate ID
Motorcycle Indicator
Plus Cost
Plus Indicator Description
Premier Cost

Prob1 Code Description
Prob2 Code Description
SC Call Club Code Description
SC Date
Rec ID
SC STS RSN Code Description
SC Vehicle Manufacturer Name
SC Vehicle Model Name
SVC Facility Name
SVC Facility Type
Total Cost
Tow Destination Latitude
Tow Destination Longitude
Tow Destination Name
Was Duplicated
Was Towed To AAR Referral
Date

In [97]:

```
df['Birth Date MMDDYYYY']
```

Out[97]:
```
Individual Key
10000003.0                     NaN
52211550.0     1922-02-05 00:00:00
52211550.0     1922-02-05 00:00:00
52211550.0     1922-02-05 00:00:00
52211550.0     1922-02-05 00:00:00
52211550.0     1922-02-05 00:00:00
52211550.0     1922-02-05 00:00:00
52211550.0     1922-02-05 00:00:00
52211550.0     1922-02-05 00:00:00
52211550.0     1922-02-05 00:00:00
1606764.0                      NaN
2766867.0      1956-02-02 00:00:00
2766867.0      1956-02-02 00:00:00
2766867.0      1956-02-02 00:00:00
2766869.0      1924-05-05 00:00:00
2766868.0      1989-12-28 00:00:00
2766868.0      1989-12-28 00:00:00
2766868.0      1989-12-28 00:00:00
2766868.0      1989-12-28 00:00:00
2766868.0      1989-12-28 00:00:00
13746947.0     1935-11-16 00:00:00
1788453.0      1937-04-16 00:00:00
1788452.0      1965-08-15 00:00:00
1788452.0      1965-08-15 00:00:00
1788452.0      1965-08-15 00:00:00
1788452.0      1965-08-15 00:00:00
1788455.0      1938-06-23 00:00:00
14243585.0     1937-11-29 00:00:00
14243587.0     1949-10-31 00:00:00
14243587.0     1949-10-31 00:00:00
                       ...
4458026.0                      NaN
12849942.0     1930-01-01 00:00:00
12849942.0     1930-01-01 00:00:00
12849942.0     1930-01-01 00:00:00
12849942.0     1930-01-01 00:00:00
12849942.0     1930-01-01 00:00:00
12849942.0     1930-01-01 00:00:00
12849941.0                     NaN
22426406.0     1949-08-02 00:00:00
22426405.0     1947-09-27 00:00:00
19764804.0     1935-03-07 00:00:00
19764802.0                     NaN
19764801.0                     NaN
19764793.0     1944-10-19 00:00:00
16521338.0     1965-01-01 00:00:00
```

```
16521338.0    1965-01-01 00:00:00
16521338.0    1965-01-01 00:00:00
16521338.0    1965-01-01 00:00:00
16521338.0    1965-01-01 00:00:00
16521336.0    1964-02-20 00:00:00
1619870.0     1975-06-23 00:00:00
1619868.0                     NaN
1619869.0                     NaN
54745437.0    1979-07-16 00:00:00
25797262.0    1922-12-13 00:00:00
25797262.0    1922-12-13 00:00:00
25797262.0    1922-12-13 00:00:00
28273400.0    1941-06-10 00:00:00
28273400.0    1941-06-10 00:00:00
28273400.0    1941-06-10 00:00:00
Name: Birth Date MMDDYYYY, Length: 21344, dtype: object
```

In [98]:
```
import datetime
```

In [99]:
```
df['Birth Date MMDDYYYY'] = pd.to_datetime(df['Birth Date MMDDYYYY'])
```

In [ ]:
```

```

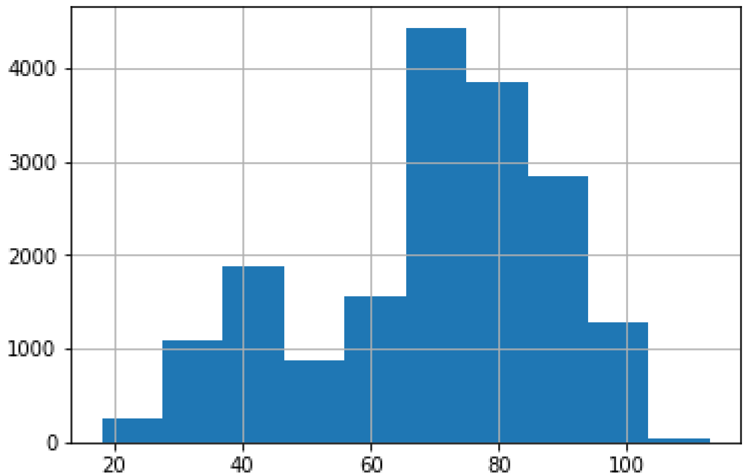
In [100]:
```
now = datetime.datetime.now().year
```

In [101]:
```
now
```

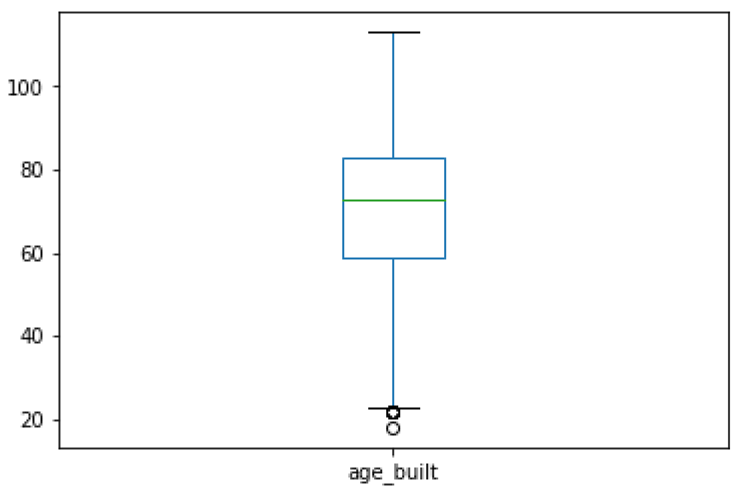
Out[101]:
```
2020
```

In [102]:
```
df['age_built'] = 2020 - df['Birth Date MMDDYYYY'].dt.year
```

In [103]:
```
df['age_built'].hist()
```
Out[103]: <matplotlib.axes._subplots.AxesSubplot at 0x1d53a467d68>



In [104]:
```
df['age_built'].plot(kind='box')
```
Out[104]: <matplotlib.axes._subplots.AxesSubplot at 0x1d53a4b5ef0>



In [105]:
```
df['age_built'].describe()
```

Out[105]:
```
count    18094.000000
mean        69.462861
std         18.923710
min         18.000000
25%         59.000000
50%         73.000000
75%         83.000000
max        113.000000
Name: age_built, dtype: float64
```

In [106]:
```
df['Household Key'].describe()
```

Out[106]:
```
count    2.134400e+04
mean     1.597040e+07
std      2.138079e+07
min      8.750000e+02
25%      4.350387e+06
50%      9.811235e+06
75%      1.654012e+07
max      1.000791e+08
Name: Household Key, dtype: float64
```

In [107]:
```
household_groupped = df.groupby('Household Key')[['age_built','Total Cost']].mean()
```

In [108]:

```
household_groupped
```

Out[108]:

Household Key	age_built	Total Cost
875.0	51.058824	75.942857
969.0	63.000000	56.525000
3338.0	90.000000	NaN
8718.0	68.000000	NaN
11524.0	63.500000	58.850000
13422.0	64.000000	39.616667
19747.0	95.000000	NaN
20469.0	76.230769	44.770833
20850.0	95.500000	NaN
25365.0	81.500000	NaN
30007.0	61.333333	34.000000
37468.0	99.000000	NaN
38093.0	62.428571	55.585000
41756.0	49.583333	47.122727
43381.0	53.250000	51.000000
49578.0	76.000000	15.000000
55047.0	72.750000	30.000000
55295.0	73.000000	NaN
73421.0	72.000000	NaN
93896.0	79.000000	65.000000
94927.0	97.000000	NaN
103545.0	71.000000	43.372222
106487.0	72.666667	44.675000
115289.0	92.000000	NaN
115306.0	78.500000	NaN
115346.0	86.000000	NaN
115351.0	78.250000	38.000000
115430.0	98.000000	53.000000

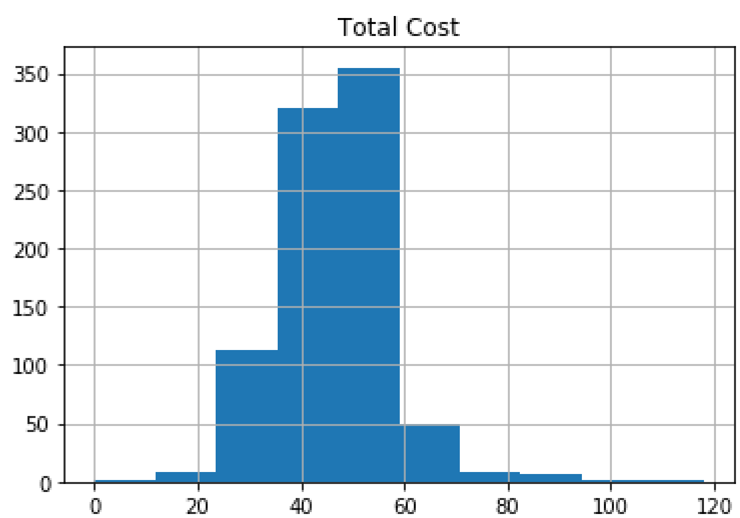
Household Key	age_built	Total Cost
116806.0	56.980000	43.792000
117430.0	77.333333	27.500000
...	...	...
99800577.0	72.083333	41.350000
99817387.0	80.857143	45.141667
99817390.0	101.000000	NaN
99839301.0	79.000000	NaN
99843098.0	56.166667	58.850000
99851820.0	53.333333	29.500000
99873114.0	79.500000	63.770000
99881116.0	51.800000	53.000000
99953012.0	90.000000	58.850000
99987696.0	32.000000	26.500000
99991498.0	NaN	NaN
99992624.0	65.857143	46.000000
99992663.0	58.600000	30.500000
99993288.0	93.200000	57.387500
99996562.0	57.333333	53.000000
100004477.0	93.000000	NaN
100016608.0	71.166667	64.678000
100020029.0	92.000000	53.000000
100022741.0	NaN	NaN
100023243.0	92.000000	NaN
100035899.0	93.000000	NaN
100053546.0	93.000000	53.000000
100064720.0	68.000000	27.000000
100065197.0	79.000000	49.558333
100067809.0	67.666667	53.000000
100069201.0	87.000000	53.000000

	age_built	Total Cost
Household Key		
100070004.0	89.000000	30.000000
100071861.0	95.000000	55.925000
100071870.0	61.571429	35.166667
100079136.0	71.000000	45.397059

5241 rows × 2 columns

In [109]:
```
household_groupped.groupby('age_built').mean().hist()
```

Out[109]:
```
array([[<matplotlib.axes._subplots.AxesSubplot object at 0x000001D53A51F4A8>]],
      dtype=object)
```



In [110]:
```
household_groupped.describe()
```

Out[110]:

	age_built	Total Cost
count	4963.000000	3408.000000
mean	75.834483	46.614296
std	14.637376	21.781499
min	18.000000	0.000000
25%	67.000000	34.833333
50%	76.500000	49.066389
75%	87.225000	53.000000
max	108.000000	588.000000

In [111]:
```
individual_groupped = df.groupby('Individual Key')[['age_built','Total Cost']].mean()
```

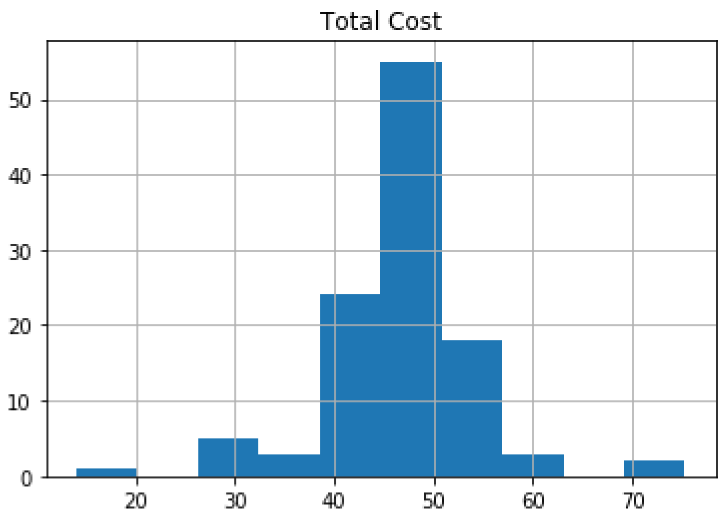
In [112]:
```
individual_groupped.describe()
```

Out[112]:

	age_built	Total Cost
count	9194.000000	4956.000000
mean	71.358934	46.662158
std	18.653233	21.695904
min	18.000000	0.000000
25%	63.000000	33.250000
50%	74.000000	51.000000
75%	85.000000	53.000000
max	113.000000	588.000000

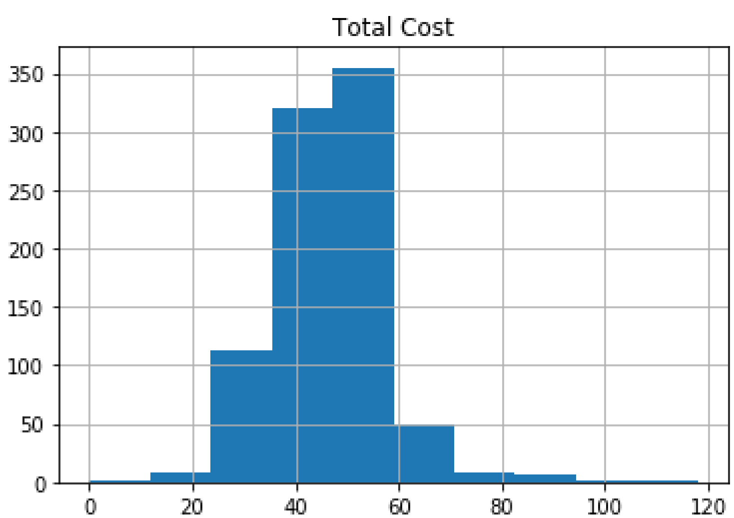
In [113]:
```
individual_groupped.groupby('age_built').mean().hist()
```

Out[113]:
```
array([[<matplotlib.axes._subplots.AxesSubplot object at 0x000001D53A5A9278>]],
      dtype=object)
```



In [114]:
```
household_groupped.groupby('age_built').mean().hist()
```

Out[114]:
```
array([[<matplotlib.axes._subplots.AxesSubplot object at 0x000001D53A5A99B0>]],
      dtype=object)
```



In [115]:

```
individual_groupped
```

Out[115]:

	age_built	Total Cost
Individual Key		
17293.0	92.0	NaN
19897.0	54.0	55.750000
19918.0	NaN	NaN
19943.0	77.0	58.850000
19959.0	76.0	NaN
40801.0	76.0	NaN
64060.0	73.0	NaN
64091.0	72.0	NaN
80381.0	90.0	NaN
80405.0	NaN	NaN
81223.0	58.0	53.000000
82646.0	77.0	NaN
97951.0	NaN	NaN
97960.0	92.0	0.000000
98730.0	40.0	NaN
102108.0	87.0	32.500000
102130.0	49.0	NaN
107776.0	83.0	53.000000
107831.0	84.0	53.000000
109884.0	NaN	NaN
109891.0	88.0	NaN
112857.0	70.0	29.000000
116711.0	102.0	NaN
116714.0	NaN	NaN
119409.0	78.0	NaN
119432.0	66.0	40.500000
123909.0	41.0	57.528571
126592.0	74.0	32.500000

	age_built	Total Cost
Individual Key		
126611.0	74.0	NaN
131688.0	48.0	44.557500
...	...	...
212175225.0	32.0	33.000000
212176106.0	39.0	30.571429
212177294.0	65.0	NaN
212177295.0	30.0	41.500000
212186756.0	70.0	46.540000
212196842.0	28.0	NaN
212196843.0	NaN	NaN
212204623.0	73.0	NaN
212211578.0	66.0	41.500000
212216450.0	59.0	68.175000
212231332.0	74.0	NaN
212236220.0	NaN	NaN
212240294.0	79.0	NaN
212251176.0	63.0	53.000000
212262614.0	56.0	58.850000
212264471.0	70.0	58.850000
212264868.0	63.0	53.000000
212270342.0	49.0	165.000000
212294716.0	57.0	55.925000
212302836.0	80.0	NaN
212303147.0	92.0	53.000000
212306183.0	71.0	50.141667
212308372.0	43.0	NaN
212308805.0	69.0	NaN
212310962.0	71.0	53.000000
212310963.0	55.0	NaN

	age_built	Total Cost
Individual Key		
212312283.0	84.0	30.000000
212317919.0	72.0	28.000000
212320920.0	93.0	NaN
212321856.0	94.0	57.387500

11903 rows × 2 columns

Conclusion: While considering the total cost of grouping based on household key and individual key, we can see that young customers makes less incidents and so less total cost in histogram. When the average household age considers, the most cost comes from more aged customers.

In [116]:

```
df['Credit Ranges'].value_counts()
```

Out[116]:

```
750-799      4101
700-749      2542
650-699      1911
800+         1785
600-649       883
Unknown       414
550-599       197
500-549       165
499 & Less     12
Name: Credit Ranges, dtype: int64
```

In [117]:
```
df.groupby('Credit Ranges')['Total Cost'].mean()
```

Out[117]:
```
Credit Ranges
499 & Less    47.812500
500-549       50.222340
550-599       54.188418
600-649       46.334104
650-699       46.927304
700-749       47.592103
750-799       47.013935
800+          49.582599
Unknown       48.444103
Name: Total Cost, dtype: float64
```

Conclusion: Based on the credit ranges of customers, we cn see that most customers falls in the group of 750-799. But is it doesn't change the average cost of other groups, and it seems to be same for all most all groups. Only the group 550-599 has some significant average cost.

In [118]:
```
df['kcl_B_IND_MosaicsGrouping'].value_counts()
```

Out[118]:
```
Autumn Years                2934
Booming with Confidence     2412
Golden Year Guardians       2099
Thriving Boomers            1663
Blue Sky Boomers            1001
Surburban Style              931
Power Elite                  916
Flourishing Families         752
Singles and Starters         737
Family Union                 591
Middle Class Melting Pot     549
Promising Families           356
Significant Singles          349
Young City Solos             177
Cultural Connections         162
Economic Challenges          116
Aspirational Fusion           90
Families in Motion            25
Pastoral Pride                21
Name: kcl_B_IND_MosaicsGrouping, dtype: int64
```

In [119]:
```
df.groupby('kcl_B_IND_MosaicsGrouping')['Total Cost'].mean()
```

Out[119]:
```
kcl_B_IND_MosaicsGrouping
Aspirational Fusion          51.519138
Autumn Years                 47.261838
Blue Sky Boomers             47.647917
Booming with Confidence      48.366323
Cultural Connections         44.453689
Economic Challenges          50.013043
Families in Motion           49.487059
Family Union                 45.428541
Flourishing Families         48.845640
Golden Year Guardians        46.265855
Middle Class Melting Pot     50.484960
Pastoral Pride               71.812500
Power Elite                  49.779392
Promising Families           43.094527
Significant Singles          48.080170
Singles and Starters         45.544367
Surburban Style              46.423783
Thriving Boomers             47.424082
Young City Solos             50.169626
Name: Total Cost, dtype: float64
```

Similary the mosaic grouping " Pastoral Pride" is significantly involved in incidents and cases and so generated high average value like 71.

Convert all columns starts with FSV to 0 and 1 and do get_dummies action on these columns

The purpose of below exercise is to see whehter the mosaic features influce the FSV Credit Card Flag. We will prepare the data and do logistics regression to see the effect.

In [120]:
```
[ i for i in df.columns if i.startswith('FSV')]
```

Out[120]:
```
['FSV CMSI Flag',
 'FSV Credit Card Flag',
 'FSV Deposit Program Flag',
 'FSV Home Equity Flag',
 'FSV ID Theft Flag',
 'FSV Mortgage Flag']
```

In [121]:
```
for col in ['FSV CMSI Flag',
 'FSV Credit Card Flag',
 'FSV Deposit Program Flag',
 'FSV Home Equity Flag',
 'FSV ID Theft Flag',
 'FSV Mortgage Flag']:
    df[col] = df[col].apply(y_n_to_binary)
```

In [122]:
```
df_exp = pd.get_dummies(df, columns = ['kcl_B_IND_MosaicsGrouping'])
```

In [123]:
```
df_exp
```

Out[123]:

Individual Key	Household Key	Member Flag	City	State - Grouped	ZIP5	ZIP9	FSV CMSI Flag	FSV Credit Card Flag	FSV Deposit Program Flag	FSV Home Equity Flag	...	kcl_B_IND_MosaicsGrouping_Gol Year Guardi
10000003.0	10462590.0	Y	NEW HAVEN	CT	6511.0	65111349.0	0	0	0	0	...	
52211550.0	4500791.0	Y	WEST WARWICK	RI	2893.0	28933850.0	0	1	0	0	...	
52211550.0	4500791.0	Y	WEST WARWICK	RI	2893.0	28933850.0	0	1	0	0	...	
52211550.0	4500791.0	Y	WEST WARWICK	RI	2893.0	28933850.0	0	1	0	0	...	
52211550.0	4500791.0	Y	WEST WARWICK	RI	2893.0	28933850.0	0	1	0	0	...	
52211550.0	4500791.0	Y	WEST WARWICK	RI	2893.0	28933850.0	0	1	0	0	...	
52211550.0	4500791.0	Y	WEST WARWICK	RI	2893.0	28933850.0	0	1	0	0	...	
52211550.0	4500791.0	Y	WEST WARWICK	RI	2893.0	28933850.0	0	1	0	0	...	
52211550.0	4500791.0	Y	WEST WARWICK	RI	2893.0	28933850.0	0	1	0	0	...	
52211550.0	4500791.0	Y	WEST WARWICK	RI	2893.0	28933850.0	0	1	0	0	...	
1606764.0	4317516.0	Y	TIVERTON	RI	2878.0	28781026.0	0	0	0	0	...	
2766867.0	11622991.0	Y	WARWICK	RI	2889.0	28892920.0	0	0	0	0	...	
2766867.0	11622991.0	Y	WARWICK	RI	2889.0	28892920.0	0	0	0	0	...	
2766867.0	11622991.0	Y	WARWICK	RI	2889.0	28892920.0	0	0	0	0	...	
2766869.0	11622991.0	Y	WARWICK	RI	2889.0	28892920.0	0	0	0	0	...	
2766868.0	11622991.0	Y	WARWICK	RI	2889.0	28892920.0	0	0	0	0	...	
2766868.0	11622991.0	Y	WARWICK	RI	2889.0	28892920.0	0	0	0	0	...	
2766868.0	11622991.0	Y	WARWICK	RI	2889.0	28892920.0	0	0	0	0	...	
2766868.0	11622991.0	Y	WARWICK	RI	2889.0	28892920.0	0	0	0	0	...	
2766868.0	11622991.0	Y	WARWICK	RI	2889.0	28892920.0	0	0	0	0	...	
13746947.0	579810.0	Y	CENTRAL FALLS	RI	2863.0	28631322.0	0	0	0	0	...	

Individual Key	Household Key	Member Flag	City	State - Grouped	ZIP5	ZIP9	FSV CMSI Flag	FSV Credit Card Flag	FSV Deposit Program Flag	FSV Home Equity Flag	...	kcl_B_IND_MosaicsGrouping_Gol Year Guardi
1788453.0	7187017.0	Y	WARWICK	RI	2888.0	28882811.0	0	0	0	0	...	
1788452.0	7187017.0	Y	WARWICK	RI	2888.0	28882811.0	0	0	0	0	...	
1788452.0	7187017.0	Y	WARWICK	RI	2888.0	28882811.0	0	0	0	0	...	
1788452.0	7187017.0	Y	WARWICK	RI	2888.0	28882811.0	0	0	0	0	...	
1788452.0	7187017.0	Y	WARWICK	RI	2888.0	28882811.0	0	0	0	0	...	
1788455.0	7187017.0	Y	WARWICK	RI	2888.0	28882811.0	0	0	0	0	...	
14243585.0	7728088.0	Y	BARRINGTON	RI	2806.0	28065003.0	0	0	0	0	...	
14243587.0	7728088.0	Y	BARRINGTON	RI	2806.0	28065003.0	0	0	0	0	...	
14243587.0	7728088.0	Y	BARRINGTON	RI	2806.0	28065003.0	0	0	0	0	...	
...	...	...	...	...	...	...	...	...	...	...	...	
4458026.0	1588987.0	Y	WARWICK	RI	2886.0	28861711.0	0	0	0	0	...	
12849942.0	16604128.0	Y	PROVIDENCE	RI	2906.0	29063709.0	0	0	0	0	...	
12849942.0	16604128.0	Y	PROVIDENCE	RI	2906.0	29063709.0	0	0	0	0	...	
12849942.0	16604128.0	Y	PROVIDENCE	RI	2906.0	29063709.0	0	0	0	0	...	
12849942.0	16604128.0	Y	PROVIDENCE	RI	2906.0	29063709.0	0	0	0	0	...	
12849942.0	16604128.0	Y	PROVIDENCE	RI	2906.0	29063709.0	0	0	0	0	...	
12849942.0	16604128.0	Y	PROVIDENCE	RI	2906.0	29063709.0	0	0	0	0	...	
12849941.0	16604128.0	Y	PROVIDENCE	RI	2906.0	29063709.0	0	0	0	0	...	
22426406.0	45466286.0	Y	BRISTOL	RI	2809.0	28092304.0	0	0	0	0	...	
22426405.0	45466286.0	Y	BRISTOL	RI	2809.0	28092304.0	0	0	0	0	...	
19764804.0	15397653.0	Y	PORTSMOUTH	RI	2871.0	28712143.0	0	0	0	0	...	
19764802.0	15397653.0	Y	PORTSMOUTH	RI	2871.0	28712143.0	0	0	0	0	...	
19764801.0	15397653.0	Y	PORTSMOUTH	RI	2871.0	28712143.0	0	0	0	0	...	
19764793.0	15397653.0	Y	PORTSMOUTH	RI	2871.0	28712143.0	0	0	0	0	...	
16521338.0	13735475.0	Y	BRISTOL	RI	2809.0	28091350.0	0	0	0	0	...	
16521338.0	13735475.0	Y	BRISTOL	RI	2809.0	28091350.0	0	0	0	0	...	
16521338.0	13735475.0	Y	BRISTOL	RI	2809.0	28091350.0	0	0	0	0	...	

Individual Key	Household Key	Member Flag	City	State - Grouped	ZIP5	ZIP9	FSV CMSI Flag	FSV Credit Card Flag	FSV Deposit Program Flag	FSV Home Equity Flag	...	kcl_B_IND_MosaicsGrouping_Gol Year Guardi
16521338.0	13735475.0	Y	BRISTOL	RI	2809.0	28091350.0	0	0	0	0	...	
16521338.0	13735475.0	Y	BRISTOL	RI	2809.0	28091350.0	0	0	0	0	...	
16521336.0	13735475.0	Y	BRISTOL	RI	2809.0	28091350.0	0	1	0	0	...	
1619870.0	5462399.0	Y	WAKEFIELD	RI	2879.0	28791421.0	0	0	0	0	...	
1619868.0	5462399.0	Y	WAKEFIELD	RI	2879.0	28791421.0	0	0	0	0	...	
1619869.0	5462399.0	Y	WAKEFIELD	RI	2879.0	28791421.0	0	0	0	0	...	
54745437.0	5462399.0	Y	COVENTRY	RI	2816.0	28167132.0	0	0	0	0	...	
25797262.0	20330346.0	Y	WARWICK	RI	2886.0	28867552.0	0	0	0	0	...	
25797262.0	20330346.0	Y	WARWICK	RI	2886.0	28867552.0	0	0	0	0	...	
25797262.0	20330346.0	Y	WARWICK	RI	2886.0	28867552.0	0	0	0	0	...	
28273400.0	8325571.0	Y	WARWICK	RI	2886.0	28868235.0	0	0	0	0	...	
28273400.0	8325571.0	Y	WARWICK	RI	2886.0	28868235.0	0	0	0	0	...	
28273400.0	8325571.0	Y	WARWICK	RI	2886.0	28868235.0	0	0	0	0	...	

21344 rows × 131 columns

In [124]:
```
household_grouped = df_exp.groupby('Household Key').mean()
```

In [125]:
```
def buy_or_not(x):
    if x > 0:
        return 1
    else:
        return 0
```

In [126]:
```
fsvs = [col for col in household_grouped.columns if col.startswith('FSV')]
```

In [127]:
```
for col in fsvs:
    household_grouped[col] = household_grouped[col].apply(buy_or_not)
```

In [128]:
```
household_grouped[fsvs].describe()
```

Out[128]:

	FSV CMSI Flag	FSV Credit Card Flag	FSV Deposit Program Flag	FSV Home Equity Flag	FSV ID Theft Flag	FSV Mortgage Flag
count	5241.0	5241.000000	5241.000000	5241.000000	5241.000000	5241.0
mean	0.0	0.110284	0.005342	0.001145	0.044839	0.0
std	0.0	0.313274	0.072904	0.033819	0.206970	0.0
min	0.0	0.000000	0.000000	0.000000	0.000000	0.0
25%	0.0	0.000000	0.000000	0.000000	0.000000	0.0
50%	0.0	0.000000	0.000000	0.000000	0.000000	0.0
75%	0.0	0.000000	0.000000	0.000000	0.000000	0.0
max	0.0	1.000000	1.000000	1.000000	1.000000	0.0

In [129]:
```
household_grouped['FSV Credit Card Flag'].value_counts()
```

Out[129]:
```
0    4663
1     578
Name: FSV Credit Card Flag, dtype: int64
```

In [130]:
```
x = household_grouped.drop(fsvs, axis =1)
```

In [131]:
```
y = household_grouped['FSV Credit Card Flag']
```

```
In [132]: x.columns
```

```
Out[132]: Index(['ZIP5', 'ZIP9', 'Number of Children', 'Length Of Residence',
       'Do Not Direct Mail Solicit', 'Email Available', 'ERS ENT Count Year 1',
       'ERS ENT Count Year 2', 'ERS ENT Count Year 3',
       'ERS Member Cost Year 1', 'ERS Member Cost Year 2',
       'ERS Member Cost Year 3', 'Right_Individual Key', 'Member Key',
       'Member Number Associate ID', 'Member Tenure Years', 'Membership ID',
       'Months from Join to Cancel', 'ZIP', 'Basic Cost',
       'Calculated Tow Miles', 'Dispatch Code2Description', 'Is Duplicate',
       'Is NSR', 'Member Match Flag', 'Member Number and Associate ID',
       'Plus Cost', 'Premier Cost', 'Rec ID', 'Total Cost',
       'Tow Destination Latitude', 'Tow Destination Longitude',
       'Was Duplicated', 'Was Towed To AAR Referral', 'age_built',
       'kcl_B_IND_MosaicsGrouping_Aspirational Fusion',
       'kcl_B_IND_MosaicsGrouping_Autumn Years',
       'kcl_B_IND_MosaicsGrouping_Blue Sky Boomers',
       'kcl_B_IND_MosaicsGrouping_Booming with Confidence',
       'kcl_B_IND_MosaicsGrouping_Cultural Connections',
       'kcl_B_IND_MosaicsGrouping_Economic Challenges',
       'kcl_B_IND_MosaicsGrouping_Families in Motion',
       'kcl_B_IND_MosaicsGrouping_Family Union',
       'kcl_B_IND_MosaicsGrouping_Flourishing Families',
       'kcl_B_IND_MosaicsGrouping_Golden Year Guardians',
       'kcl_B_IND_MosaicsGrouping_Middle Class Melting Pot',
       'kcl_B_IND_MosaicsGrouping_Pastoral Pride',
       'kcl_B_IND_MosaicsGrouping_Power Elite',
       'kcl_B_IND_MosaicsGrouping_Promising Families',
       'kcl_B_IND_MosaicsGrouping_Significant Singles',
       'kcl_B_IND_MosaicsGrouping_Singles and Starters',
       'kcl_B_IND_MosaicsGrouping_Surburban Style',
       'kcl_B_IND_MosaicsGrouping_Thriving Boomers',
       'kcl_B_IND_MosaicsGrouping_Young City Solos'],
      dtype='object')
```

```
In [133]: y.describe()
```

```
Out[133]: count    5241.000000
mean        0.110284
std         0.313274
min         0.000000
25%         0.000000
50%         0.000000
75%         0.000000
max         1.000000
Name: FSV Credit Card Flag, dtype: float64
```

In [134]:
```
y
```

```
Out[134]: Household Key
875.0         1
969.0         0
3338.0        0
8718.0        0
11524.0       0
13422.0       0
19747.0       0
20469.0       0
20850.0       0
25365.0       0
30007.0       0
37468.0       0
38093.0       0
41756.0       0
43381.0       0
49578.0       0
55047.0       0
55295.0       0
73421.0       0
93896.0       1
94927.0       0
103545.0      1
106487.0      0
115289.0      0
115306.0      0
115346.0      0
115351.0      1
115430.0      0
116806.0      1
117430.0      0
             ..
99800577.0    0
99817387.0    0
99817390.0    0
99839301.0    0
99843098.0    0
99851820.0    0
99873114.0    0
99881116.0    0
99953012.0    0
99987696.0    0
99991498.0    0
99992624.0    0
99992663.0    0
99993288.0    0
99996562.0    0
```

```
100004477.0    0
100016608.0    0
100020029.0    0
100022741.0    0
100023243.0    0
100035899.0    0
100053546.0    0
100064720.0    1
100065197.0    0
100067809.0    0
100069201.0    0
100070004.0    0
100071861.0    0
100071870.0    0
100079136.0    0
Name: FSV Credit Card Flag, Length: 5241, dtype: int64
```

In [135]:

```
x
```

Out[135]:

Household Key	ZIP5	ZIP9	Number of Children	Length Of Residence	Do Not Direct Mail Solicit	Email Available	ERS ENT Count Year 1	ERS ENT Count Year 2	ERS ENT Count Year 3	ERS Member Cost Year 1	...	kcl_B_IND_MosaicsG
875.0	2920.777778	2.920950e+07	NaN	NaN	0.00	0.333333	0.785714	0.428571	0.714286	46.239286	...	
969.0	2919.000000	2.919102e+07	0.000000	14.000000	0.00	0.833333	0.000000	0.666667	1.500000	0.000000	...	
3338.0	2919.000000	2.919492e+07	1.000000	5.000000	0.00	0.000000	0.000000	0.000000	0.000000	0.000000	...	
8718.0	2910.000000	2.910341e+07	0.000000	15.000000	0.00	0.500000	0.000000	0.000000	0.000000	0.000000	...	
11524.0	2919.000000	2.919507e+07	1.142857	12.142857	0.00	0.857143	0.333333	2.166667	1.000000	19.616667	...	
13422.0	2816.000000	2.816776e+07	0.000000	15.000000	0.00	0.500000	0.250000	0.000000	0.000000	14.712500	...	
19747.0	2919.000000	2.919363e+07	0.000000	15.000000	0.00	0.000000	0.000000	0.000000	0.000000	0.000000	...	
20469.0	2916.000000	2.916301e+07	0.000000	15.000000	0.00	0.000000	3.692308	0.923077	0.923077	217.292308	...	
20850.0	2910.000000	2.910193e+07	NaN	NaN	0.00	0.000000	0.000000	0.000000	0.000000	0.000000	...	
25365.0	2913.200000	2.913424e+07	2.000000	2.000000	0.00	0.200000	NaN	NaN	NaN	NaN	...	
30007.0	2842.000000	2.842407e+07	1.000000	11.666667	0.00	0.500000	0.000000	0.333333	0.000000	0.000000	...	
37468.0	2852.000000	2.852167e+07	NaN	NaN	1.00	0.000000	NaN	NaN	NaN	NaN	...	
38093.0	2888.083333	2.888645e+07	0.625000	2.125000	0.00	0.666667	0.000000	1.636364	0.454545	0.000000	...	
41756.0	2840.666667	2.840883e+07	NaN	NaN	0.00	0.916667	1.818182	1.636364	1.909091	89.409091	...	
43381.0	2904.750000	2.905122e+07	0.333333	10.333333	0.00	1.000000	0.000000	0.000000	0.333333	0.000000	...	
49578.0	2816.000000	2.816714e+07	0.000000	15.000000	0.00	0.333333	0.000000	0.000000	0.000000	0.000000	...	
55047.0	2816.000000	2.816526e+07	NaN	NaN	0.00	0.250000	0.000000	1.333333	1.333333	0.000000	...	
55295.0	2852.000000	2.852277e+07	0.000000	5.000000	0.00	0.000000	0.000000	0.000000	0.000000	0.000000	...	
73421.0	2898.000000	2.898112e+07	0.000000	13.000000	0.00	1.000000	0.000000	0.000000	0.000000	0.000000	...	
93896.0	2886.000000	2.886941e+07	0.000000	9.000000	0.00	0.000000	0.000000	0.000000	0.500000	0.000000	...	
94927.0	2886.000000	2.886176e+07	1.000000	15.000000	1.00	0.000000	NaN	NaN	NaN	NaN	...	
103545.0	2901.000000	2.901035e+07	NaN	NaN	0.00	1.000000	2.000000	2.000000	2.000000	103.990000	...	
106487.0	2885.000000	2.885170e+07	0.600000	9.600000	0.00	0.428571	0.800000	0.200000	0.000000	47.080000	...	
115289.0	2919.000000	2.919316e+07	0.000000	0.000000	0.00	1.000000	NaN	NaN	NaN	NaN	...	
115306.0	2919.000000	2.919314e+07	0.500000	7.500000	0.00	0.500000	0.000000	0.000000	0.000000	0.000000	...	

Household Key	ZIP5	ZIP9	Number of Children	Length Of Residence	Do Not Direct Mail Solicit	Email Available	ERS ENT Count Year 1	ERS ENT Count Year 2	ERS ENT Count Year 3	ERS Member Cost Year 1	...	kcl_B_IND_MosaicsG
115346.0	2919.000000	2.919314e+07	NaN	NaN	0.00	0.000000	NaN	NaN	NaN	NaN	...	
115351.0	2920.000000	2.920311e+07	0.500000	10.500000	0.20	0.400000	0.000000	0.000000	0.000000	0.000000	...	
115430.0	2919.000000	2.919347e+07	0.000000	8.000000	0.00	0.000000	0.000000	0.000000	2.000000	0.000000	...	
116806.0	2912.181818	2.912478e+07	1.928571	2.785714	0.00	0.763636	0.596154	2.692308	1.653846	29.126923	...	
117430.0	2857.000000	2.857028e+07	0.000000	13.333333	0.00	0.000000	0.000000	0.833333	0.000000	0.000000	...	
...	...	...	...	...	...	...	...	...	...	...	...	
99800577.0	2893.000000	2.893601e+07	2.000000	15.000000	0.25	0.000000	0.000000	1.181818	1.636364	0.000000	...	
99817387.0	2906.000000	2.906275e+07	1.000000	2.000000	0.00	0.142857	0.857143	0.000000	0.000000	50.442857	...	
99817390.0	2828.000000	2.828144e+07	NaN	NaN	0.00	0.000000	NaN	NaN	NaN	NaN	...	
99839301.0	2892.000000	2.892181e+07	NaN	NaN	0.00	0.250000	NaN	NaN	NaN	NaN	...	
99843098.0	2940.000000	2.940017e+07	0.500000	8.000000	0.00	0.333333	1.000000	0.500000	0.000000	58.850000	...	
99851820.0	2920.000000	2.920801e+07	NaN	NaN	0.00	1.000000	0.500000	0.333333	0.666667	16.250000	...	
99873114.0	2889.000000	2.889472e+07	1.000000	15.000000	0.00	0.000000	0.000000	0.833333	2.500000	0.000000	...	
99881116.0	2817.000000	2.817600e+07	NaN	NaN	0.00	0.500000	0.000000	0.000000	0.000000	0.000000	...	
99953012.0	2906.000000	2.906436e+07	2.000000	13.000000	0.00	0.000000	0.000000	1.000000	0.000000	0.000000	...	
99987696.0	2915.000000	2.915311e+07	NaN	NaN	0.00	1.000000	0.000000	0.000000	0.000000	0.000000	...	
99991498.0	2852.000000	2.852442e+07	NaN	NaN	1.00	1.000000	NaN	NaN	NaN	NaN	...	
99992624.0	2886.000000	2.886088e+07	NaN	NaN	0.00	0.285714	0.000000	0.666667	0.000000	0.000000	...	
99992663.0	2891.000000	2.891253e+07	1.000000	1.000000	1.00	1.000000	0.000000	0.000000	0.000000	0.000000	...	
99993288.0	2886.000000	2.886611e+07	NaN	NaN	0.00	0.800000	0.000000	3.000000	0.000000	0.000000	...	
99996562.0	2911.000000	2.911221e+07	1.000000	15.000000	0.00	0.285714	0.000000	0.333333	0.166667	0.000000	...	
100004477.0	2816.000000	2.816802e+07	1.000000	15.000000	0.00	0.000000	0.000000	1.000000	0.000000	0.000000	...	
100016608.0	2865.000000	2.865496e+07	0.000000	1.000000	0.00	1.000000	0.833333	0.000000	2.500000	27.083333	...	
100020029.0	2806.000000	2.806137e+07	NaN	NaN	0.00	0.000000	0.000000	1.000000	1.000000	0.000000	...	
100022741.0	2864.000000	2.864190e+07	NaN	NaN	0.00	0.000000	NaN	NaN	NaN	NaN	...	
100023243.0	2920.000000	2.920112e+07	NaN	NaN	0.00	0.000000	NaN	NaN	NaN	NaN	...	

Household Key	ZIP5	ZIP9	Number of Children	Length Of Residence	Do Not Direct Mail Solicit	Email Available	ERS ENT Count Year 1	ERS ENT Count Year 2	ERS ENT Count Year 3	ERS Member Cost Year 1	...	kcl_B_IND_MosaicsG
100035899.0	2816.000000	2.816281e+07	NaN	NaN	0.00	0.000000	0.000000	0.000000	0.000000	0.000000	...	
100053546.0	2909.000000	2.909241e+07	NaN	NaN	0.00	0.500000	0.000000	1.000000	0.000000	0.000000	...	
100064720.0	2813.000000	2.813275e+07	0.333333	15.000000	0.00	0.666667	0.000000	0.000000	0.000000	0.000000	...	
100065197.0	2823.000000	2.823014e+07	NaN	NaN	0.00	0.000000	1.000000	1.000000	1.000000	58.850000	...	
100067809.0	2910.833333	2.911165e+07	1.000000	11.500000	0.00	0.833333	0.000000	0.000000	0.800000	0.000000	...	
100069201.0	2904.000000	2.904110e+07	0.000000	13.000000	0.00	0.000000	0.000000	0.000000	1.000000	0.000000	...	
100070004.0	2889.000000	2.889593e+07	NaN	NaN	0.00	0.500000	0.000000	0.000000	2.000000	0.000000	...	
100071861.0	2920.000000	2.920295e+07	0.000000	8.000000	0.00	0.000000	0.000000	4.000000	1.000000	0.000000	...	
100071870.0	2893.857143	2.894205e+07	NaN	NaN	0.00	0.857143	5.000000	1.000000	1.000000	238.500000	...	
100079136.0	2879.000000	2.879446e+07	2.000000	27.000000	0.00	0.736842	1.529412	4.000000	3.176471	85.876471	...	

5241 rows × 54 columns

In [136]:
```
sub_df = household_grouped[['kcl_B_IND_MosaicsGrouping_Aspirational Fusion',
       'kcl_B_IND_MosaicsGrouping_Autumn Years',
       'kcl_B_IND_MosaicsGrouping_Blue Sky Boomers',
       'kcl_B_IND_MosaicsGrouping_Booming with Confidence',
       'kcl_B_IND_MosaicsGrouping_Cultural Connections',
       'kcl_B_IND_MosaicsGrouping_Economic Challenges',
       'kcl_B_IND_MosaicsGrouping_Families in Motion',
       'kcl_B_IND_MosaicsGrouping_Family Union',
       'kcl_B_IND_MosaicsGrouping_Flourishing Families',
       'kcl_B_IND_MosaicsGrouping_Golden Year Guardians',
       'kcl_B_IND_MosaicsGrouping_Middle Class Melting Pot',
       'kcl_B_IND_MosaicsGrouping_Pastoral Pride',
       'kcl_B_IND_MosaicsGrouping_Power Elite',
       'kcl_B_IND_MosaicsGrouping_Promising Families',
       'kcl_B_IND_MosaicsGrouping_Significant Singles',
       'kcl_B_IND_MosaicsGrouping_Singles and Starters',
       'kcl_B_IND_MosaicsGrouping_Surburban Style',
       'kcl_B_IND_MosaicsGrouping_Thriving Boomers',
       'kcl_B_IND_MosaicsGrouping_Young City Solos',
        'FSV Credit Card Flag']]
```

In [137]:
```
not_buyers = sub_df.loc[sub_df['FSV Credit Card Flag'] != 1]
```

In [138]:
```
buyers = sub_df.loc[sub_df['FSV Credit Card Flag'] == 1]
```

In [139]:
```
not_buyers.shape
```

Out[139]: (4663, 20)

In [140]:
```
buyers.shape
```

Out[140]: (578, 20)

As the number of records in the buyers and not_buyers groups are very different , we will make a new data set that contains equal number of buyers and not_buyers.

In [145]:
```
buyers.describe()
```

Out[145]:

	kcl_B_IND_MosaicsGrouping_Aspirational Fusion	kcl_B_IND_MosaicsGrouping_Autumn Years	kcl_B_IND_MosaicsGrouping_Blue Sky Boomers	kcl_B_IND_MosaicsGrouping with C
count	578.000000	578.000000	578.000000	5
mean	0.003460	0.187580	0.054354	
std	0.058773	0.384428	0.225341	
min	0.000000	0.000000	0.000000	
25%	0.000000	0.000000	0.000000	
50%	0.000000	0.000000	0.000000	
75%	0.000000	0.000000	0.000000	
max	1.000000	1.000000	1.000000	

In [146]:
```
not_buyers.describe()
```

Out[146]:

	kcl_B_IND_MosaicsGrouping_Aspirational Fusion	kcl_B_IND_MosaicsGrouping_Autumn Years	kcl_B_IND_MosaicsGrouping_Blue Sky Boomers	kcl_B_IND_MosaicsGrouping with C
count	4663.000000	4663.000000	4663.000000	46
mean	0.004122	0.130394	0.044878	
std	0.063201	0.335553	0.205811	
min	0.000000	0.000000	0.000000	
25%	0.000000	0.000000	0.000000	
50%	0.000000	0.000000	0.000000	
75%	0.000000	0.000000	0.000000	
max	1.000000	1.000000	1.000000	

In [147]:
```
X1 = not_buyers.sample(578)
```

In [148]:
```
X1.shape
```

Out[148]: (578, 20)

In [149]:
```
buyers.shape
```

Out[149]: (578, 20)

In [150]:
```
d_df = pd.concat([buyers,X1])
```

In [151]:
```
d_df.shape
```

Out[151]: (1156, 20)

In [152]:
```
X = d_df[['kcl_B_IND_MosaicsGrouping_Aspirational Fusion',
       'kcl_B_IND_MosaicsGrouping_Autumn Years',
       'kcl_B_IND_MosaicsGrouping_Blue Sky Boomers',
       'kcl_B_IND_MosaicsGrouping_Booming with Confidence',
       'kcl_B_IND_MosaicsGrouping_Cultural Connections',
       'kcl_B_IND_MosaicsGrouping_Economic Challenges',
       'kcl_B_IND_MosaicsGrouping_Families in Motion',
       'kcl_B_IND_MosaicsGrouping_Family Union',
       'kcl_B_IND_MosaicsGrouping_Flourishing Families',
       'kcl_B_IND_MosaicsGrouping_Golden Year Guardians',
       'kcl_B_IND_MosaicsGrouping_Middle Class Melting Pot',
       'kcl_B_IND_MosaicsGrouping_Pastoral Pride',
       'kcl_B_IND_MosaicsGrouping_Power Elite',
       'kcl_B_IND_MosaicsGrouping_Promising Families',
       'kcl_B_IND_MosaicsGrouping_Significant Singles',
       'kcl_B_IND_MosaicsGrouping_Singles and Starters',
       'kcl_B_IND_MosaicsGrouping_Surburban Style',
       'kcl_B_IND_MosaicsGrouping_Thriving Boomers',
       'kcl_B_IND_MosaicsGrouping_Young City Solos' ]]
```

In [153]:
```
y = d_df [['FSV Credit Card Flag']]
```

In [154]:
```
X.shape
```

Out[154]: (1156, 19)

In [155]:
```
y.shape
```

Out[155]: (1156, 1)

Applying Logistics Regression model

In [156]:
```
from sklearn.model_selection import train_test_split
```

In [157]:
```
X_train,X_test, y_train, y_test = train_test_split(X,y)
```

In [158]:
```
from sklearn.linear_model import LogisticRegression
```

In [159]:
```
from sklearn.neighbors import KNeighborsClassifier
```

In [160]:
```
lgr = LogisticRegression()
```

In [161]:
```
knn = KNeighborsClassifier()
```

In [162]:
```
knn.fit(X_train, y_train)
```

```
C:\Users\george\AppData\Local\Continuum\anaconda3\lib\site-packages\ipykernel_launcher.py:1: DataConversionWarning: A
column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples, ), for example us
ing ravel().
  """Entry point for launching an IPython kernel.
```

Out[162]:
```
KNeighborsClassifier(algorithm='auto', leaf_size=30, metric='minkowski',
           metric_params=None, n_jobs=None, n_neighbors=5, p=2,
           weights='uniform')
```

In [163]:
```
lgr.fit(X_train, y_train)
```

```
C:\Users\george\AppData\Local\Continuum\anaconda3\lib\site-packages\sklearn\linear_model\logistic.py:433: FutureWarnin
g: Default solver will be changed to 'lbfgs' in 0.22. Specify a solver to silence this warning.
  FutureWarning)
C:\Users\george\AppData\Local\Continuum\anaconda3\lib\site-packages\sklearn\utils\validation.py:761: DataConversionWar
ning: A column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples, ), for ex
ample using ravel().
  y = column_or_1d(y, warn=True)
```

Out[163]:
```
LogisticRegression(C=1.0, class_weight=None, dual=False, fit_intercept=True,
          intercept_scaling=1, max_iter=100, multi_class='warn',
          n_jobs=None, penalty='l2', random_state=None, solver='warn',
          tol=0.0001, verbose=0, warm_start=False)
```

In [164]:
```
lgr.score(X_test, y_test)
```

Out[164]:
```
0.5190311418685121
```

In [165]:
```
coef_df = pd.DataFrame( { 'coef': lgr.coef_[0], 'feature': X_test.columns})
```

```
In [167]: coef_df
```

Out[167]:

	coef	feature
0	-0.594611	kcl_B_IND_MosaicsGrouping_Aspirational Fusion
1	0.536667	kcl_B_IND_MosaicsGrouping_Autumn Years
2	-0.033830	kcl_B_IND_MosaicsGrouping_Blue Sky Boomers
3	0.319214	kcl_B_IND_MosaicsGrouping_Booming with Confidence
4	-0.249801	kcl_B_IND_MosaicsGrouping_Cultural Connections
5	-0.132287	kcl_B_IND_MosaicsGrouping_Economic Challenges
6	0.000000	kcl_B_IND_MosaicsGrouping_Families in Motion
7	0.637518	kcl_B_IND_MosaicsGrouping_Family Union
8	0.238953	kcl_B_IND_MosaicsGrouping_Flourishing Families
9	0.310161	kcl_B_IND_MosaicsGrouping_Golden Year Guardians
10	0.815685	kcl_B_IND_MosaicsGrouping_Middle Class Melting...
11	0.000000	kcl_B_IND_MosaicsGrouping_Pastoral Pride
12	-0.534926	kcl_B_IND_MosaicsGrouping_Power Elite
13	0.264591	kcl_B_IND_MosaicsGrouping_Promising Families
14	-0.393702	kcl_B_IND_MosaicsGrouping_Significant Singles
15	-0.436711	kcl_B_IND_MosaicsGrouping_Singles and Starters
16	0.233938	kcl_B_IND_MosaicsGrouping_Surburban Style
17	0.283676	kcl_B_IND_MosaicsGrouping_Thriving Boomers
18	-0.696195	kcl_B_IND_MosaicsGrouping_Young City Solos

```
In [166]: plt.figure(figsize = (25,15))
```

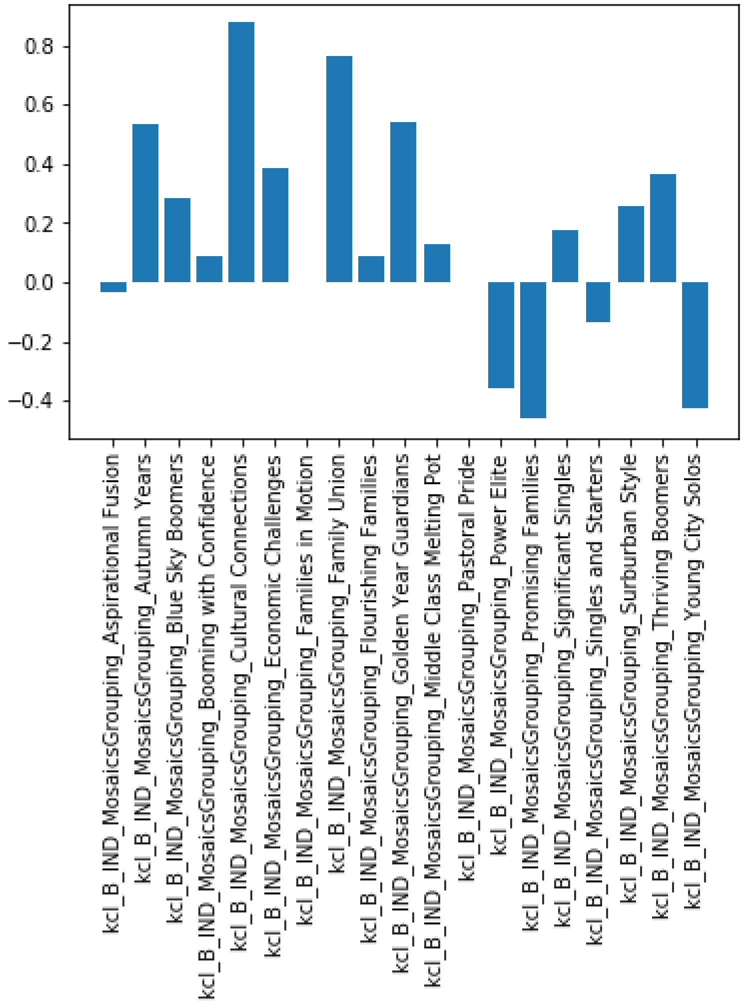
```
Out[166]: <Figure size 1800x1080 with 0 Axes>

<Figure size 1800x1080 with 0 Axes>
```

Evaluation features using their coefficient values in model

In [106]:
```
plt.bar( coef_df['feature'], coef_df['coef'])
plt.xticks(rotation = 90)
```

Out[106]:
```
([0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18],
 <a list of 19 Text xticklabel objects>)
```



For comparison, we can try Gradient Boosting algorithm also

In [111]:
```
from sklearn.ensemble import GradientBoostingClassifier
```

In [113]:
```
gbc = GradientBoostingClassifier()
```

In [ ]:

In [114]:
```
gbc.fit(X_train, y_train)
```

```
C:\Users\george\AppData\Local\Continuum\anaconda3\lib\site-packages\sklearn\utils\validation.py:761: DataConversionWar
ning: A column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples, ), for ex
ample using ravel().
  y = column_or_1d(y, warn=True)
```

Out[114]:
```
GradientBoostingClassifier(criterion='friedman_mse', init=None,
              learning_rate=0.1, loss='deviance', max_depth=3,
              max_features=None, max_leaf_nodes=None,
              min_impurity_decrease=0.0, min_impurity_split=None,
              min_samples_leaf=1, min_samples_split=2,
              min_weight_fraction_leaf=0.0, n_estimators=100,
              n_iter_no_change=None, presort='auto', random_state=None,
              subsample=1.0, tol=0.0001, validation_fraction=0.1,
              verbose=0, warm_start=False)
```

In [115]:
```
gbc.score(X_test, y_test)
```

Out[115]:
```
0.532871972318339
```

In [116]:

```
df
```

Out[116]:

Individual Key	Household Key	Member Flag	City	State - Grouped	ZIP5	ZIP9	FSV CMSI Flag	FSV Credit Card Flag	FSV Deposit Program Flag	FSV Home Equity Flag	...	SVC Facility Name	SVC Facility Type	Total Cost
10000003.0	10462590.0	Y	NEW HAVEN	CT	6511.0	65111349.0	0	0	0	0	...	NaN	NaN	NaN
52211550.0	4500791.0	Y	WEST WARWICK	RI	2893.0	28933850.0	0	1	0	0	...	ASTRO WRECKER SERVICE	independent repair	32.50
52211550.0	4500791.0	Y	WEST WARWICK	RI	2893.0	28933850.0	0	1	0	0	...	Astro Wrecker Service	independent repair	30.00
52211550.0	4500791.0	Y	WEST WARWICK	RI	2893.0	28933850.0	0	1	0	0	...	ASTRO WRECKER SERVICE	independent repair	32.50
52211550.0	4500791.0	Y	WEST WARWICK	RI	2893.0	28933850.0	0	1	0	0	...	ASTRO WRECKER SERVICE	independent repair	30.00
52211550.0	4500791.0	Y	WEST WARWICK	RI	2893.0	28933850.0	0	1	0	0	...	AAA SNE RI LIGHT SERVICE	mobile battery service	53.00
52211550.0	4500791.0	Y	WEST WARWICK	RI	2893.0	28933850.0	0	1	0	0	...	Astro Wrecker Service	independent repair	30.00
52211550.0	4500791.0	Y	WEST WARWICK	RI	2893.0	28933850.0	0	1	0	0	...	Astro Wrecker Service	independent repair	32.00
52211550.0	4500791.0	Y	WEST WARWICK	RI	2893.0	28933850.0	0	1	0	0	...	Astro Wrecker Service	independent repair	32.00
52211550.0	4500791.0	Y	WEST WARWICK	RI	2893.0	28933850.0	0	1	0	0	...	ASTRO WRECKER SERVICE	independent repair	32.50
1606764.0	4317516.0	Y	TIVERTON	RI	2878.0	28781026.0	0	0	0	0	...	NaN	NaN	NaN
2766867.0	11622991.0	Y	WARWICK	RI	2889.0	28892920.0	0	0	0	0	...	Aaa Sne Ri Light Service	mobile battery service	58.85
2766867.0	11622991.0	Y	WARWICK	RI	2889.0	28892920.0	0	0	0	0	...	AAA SNE RI FLEET FULL SERVICE	other	53.00

Individual Key	Household Key	Member Flag	City	State - Grouped	ZIP5	ZIP9	FSV CMSI Flag	FSV Credit Card Flag	FSV Deposit Program Flag	FSV Home Equity Flag	...	SVC Facility Name	SVC Facility Type	Total Cost
2766867.0	11622991.0	Y	WARWICK	RI	2889.0	28892920.0	0	0	0	0	...	Aaa Sne Ri Light Service	mobile battery service	53.00
2766869.0	11622991.0	Y	WARWICK	RI	2889.0	28892920.0	0	0	0	0	...	NaN	NaN	NaN
2766868.0	11622991.0	Y	WARWICK	RI	2889.0	28892920.0	0	0	0	0	...	NaN	NaN	NaN
2766868.0	11622991.0	Y	WARWICK	RI	2889.0	28892920.0	0	0	0	0	...	AAA SNE RI FLEET FULL SERVICE	other	53.00
2766868.0	11622991.0	Y	WARWICK	RI	2889.0	28892920.0	0	0	0	0	...	AAA SNE RI LIGHT SERVICE	other	53.00
2766868.0	11622991.0	Y	WARWICK	RI	2889.0	28892920.0	0	0	0	0	...	AAA SNE RI LIGHT SERVICE	other	53.00
2766868.0	11622991.0	Y	WARWICK	RI	2889.0	28892920.0	0	0	0	0	...	Assured Collision Inc	body shop	29.00
13746947.0	579810.0	Y	CENTRAL FALLS	RI	2863.0	28631322.0	0	0	0	0	...	KING'S SERVICE CENTER	service station	28.00
1788453.0	7187017.0	Y	WARWICK	RI	2888.0	28882811.0	0	0	0	0	...	AAA SNE RI LIGHT SERVICE	mobile battery service	53.00
1788452.0	7187017.0	Y	WARWICK	RI	2888.0	28882811.0	0	0	0	0	...	AAA SNE RI LIGHT SERVICE	other	53.00
1788452.0	7187017.0	Y	WARWICK	RI	2888.0	28882811.0	0	0	0	0	...	AAA SNE RI LIGHT SERVICE	other	53.00
1788452.0	7187017.0	Y	WARWICK	RI	2888.0	28882811.0	0	0	0	0	...	AAA SNE RI LIGHT SERVICE	mobile battery service	53.00
1788452.0	7187017.0	Y	WARWICK	RI	2888.0	28882811.0	0	0	0	0	...	AAA SNE RI LIGHT SERVICE	mobile battery service	53.00

	Household Key	Member Flag	City	State - Grouped	ZIP5	ZIP9	FSV CMSI Flag	FSV Credit Card Flag	FSV Deposit Program Flag	FSV Home Equity Flag	...	SVC Facility Name	SVC Facility Type	Total Cost
Individual Key														
1788455.0	7187017.0	Y	WARWICK	RI	2888.0	28882811.0	0	0	0	0	...	Aaa Sne Ri Fleet Full Service	mobile battery service	58.85
14243585.0	7728088.0	Y	BARRINGTON	RI	2806.0	28065003.0	0	0	0	0	...	AAA SNE RI LIGHT SERVICE	other	53.00
14243587.0	7728088.0	Y	BARRINGTON	RI	2806.0	28065003.0	0	0	0	0	...	AAA SNE RI FLEET FULL SERVICE	mobile battery service	53.00
14243587.0	7728088.0	Y	BARRINGTON	RI	2806.0	28065003.0	0	0	0	0	...	Aaa Sne Ri Light Service	mobile battery service	53.00
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
4458026.0	1588987.0	Y	WARWICK	RI	2886.0	28861711.0	0	0	0	0	...	NaN	NaN	NaN
12849942.0	16604128.0	Y	PROVIDENCE	RI	2906.0	29063709.0	0	0	0	0	...	East Side Service Center Inc	service station	47.00
12849942.0	16604128.0	Y	PROVIDENCE	RI	2906.0	29063709.0	0	0	0	0	...	Aaa Sne Ri Light Service	mobile battery service	58.85
12849942.0	16604128.0	Y	PROVIDENCE	RI	2906.0	29063709.0	0	0	0	0	...	Aaa Sne Ri Light Service	mobile battery service	58.85
12849942.0	16604128.0	Y	PROVIDENCE	RI	2906.0	29063709.0	0	0	0	0	...	Aaa Sne Ri Light Service	mobile battery service	53.00
12849942.0	16604128.0	Y	PROVIDENCE	RI	2906.0	29063709.0	0	0	0	0	...	Aaa Sne Ri Light Service	mobile battery service	53.00
12849942.0	16604128.0	Y	PROVIDENCE	RI	2906.0	29063709.0	0	0	0	0	...	Aaa Sne Ri Fleet Full Service	mobile battery service	58.85
12849941.0	16604128.0	Y	PROVIDENCE	RI	2906.0	29063709.0	0	0	0	0	...	NaN	NaN	NaN
22426406.0	45466286.0	Y	BRISTOL	RI	2809.0	28092304.0	0	0	0	0	...	NaN	NaN	NaN

Individual Key	Household Key	Member Flag	City	State - Grouped	ZIP5	ZIP9	FSV CMSI Flag	FSV Credit Card Flag	FSV Deposit Program Flag	FSV Home Equity Flag	...	SVC Facility Name	SVC Facility Type	Total Cost
22426405.0	45466286.0	Y	BRISTOL	RI	2809.0	28092304.0	0	0	0	0	...	Safe-Way Auto Sales Inc	body shop	34.00
19764804.0	15397653.0	Y	PORTSMOUTH	RI	2871.0	28712143.0	0	0	0	0	...	NaN	NaN	NaN
19764802.0	15397653.0	Y	PORTSMOUTH	RI	2871.0	28712143.0	0	0	0	0	...	NaN	NaN	NaN
19764801.0	15397653.0	Y	PORTSMOUTH	RI	2871.0	28712143.0	0	0	0	0	...	NaN	NaN	NaN
19764793.0	15397653.0	Y	PORTSMOUTH	RI	2871.0	28712143.0	0	0	0	0	...	NaN	NaN	NaN
16521338.0	13735475.0	Y	BRISTOL	RI	2809.0	28091350.0	0	0	0	0	...	Twigg's Automotive Inc	independent repair	29.00
16521338.0	13735475.0	Y	BRISTOL	RI	2809.0	28091350.0	0	0	0	0	...	Safe-Way Auto Sales Inc	body shop	44.00
16521338.0	13735475.0	Y	BRISTOL	RI	2809.0	28091350.0	0	0	0	0	...	Safe-Way Auto Sales Inc	body shop	29.00
16521338.0	13735475.0	Y	BRISTOL	RI	2809.0	28091350.0	0	0	0	0	...	Safe-Way Auto Sales Inc	body shop	47.00
16521338.0	13735475.0	Y	BRISTOL	RI	2809.0	28091350.0	0	0	0	0	...	SAFE-WAY AUTO SALES INC	body shop	44.00
16521336.0	13735475.0	Y	BRISTOL	RI	2809.0	28091350.0	0	1	0	0	...	EAST SIDE SERVICE CENTER INC	service station	82.00
1619870.0	5462399.0	Y	WAKEFIELD	RI	2879.0	28791421.0	0	0	0	0	...	Aaa Sne Ri Fleet Full Service	mobile battery service	29.43
1619868.0	5462399.0	Y	WAKEFIELD	RI	2879.0	28791421.0	0	0	0	0	...	NaN	NaN	NaN
1619869.0	5462399.0	Y	WAKEFIELD	RI	2879.0	28791421.0	0	0	0	0	...	NaN	NaN	NaN
54745437.0	5462399.0	Y	COVENTRY	RI	2816.0	28167132.0	0	0	0	0	...	NaN	NaN	NaN

Individual Key	Household Key	Member Flag	City	State - Grouped	ZIP5	ZIP9	FSV CMSI Flag	FSV Credit Card Flag	FSV Deposit Program Flag	FSV Home Equity Flag	...	SVC Facility Name	SVC Facility Type	Total Cost
25797262.0	20330346.0	Y	WARWICK	RI	2886.0	28867552.0	0	0	0	0	...	AAA SNE RI LIGHT SERVICE	other	53.00
25797262.0	20330346.0	Y	WARWICK	RI	2886.0	28867552.0	0	0	0	0	...	HERB'S SUNOCO	service station	36.00
25797262.0	20330346.0	Y	WARWICK	RI	2886.0	28867552.0	0	0	0	0	...	AAA SNE RI LIGHT SERVICE	mobile battery service	53.00
28273400.0	8325571.0	Y	WARWICK	RI	2886.0	28868235.0	0	0	0	0	...	Aaa Sne Ri Light Service	mobile battery service	58.85
28273400.0	8325571.0	Y	WARWICK	RI	2886.0	28868235.0	0	0	0	0	...	Aaa Sne Ri Light Service	mobile battery service	58.85
28273400.0	8325571.0	Y	WARWICK	RI	2886.0	28868235.0	0	0	0	0	...	NaN	NaN	NaN

21344 rows × 113 columns

We can see a little improvement in score in Gradient Boosting algorithm

In [ ]: