Submitted by Aryan Sindwani(09) & Anurag Joshi(07)
Group-2(Semester III)

DATA MINING ASSIGNMENT - 1

For a given data set, answer the following questions:

Q1. Identify and describe the given dataset.

- ➢ The dataset used for the assignment is BitcoinHeistRansomwareAddressDataset Data Set.
- ➢ It includes the entire Bitcoin transaction graph from 2009 January to 2018 December with a time interval of 24 hours.
- ➢ It contains features on the heterogeneous Bitcoin network to identify ransomware payments.
- ➢ It is a multivariate time-series dataset with 2916697 number of instances.
- ➢ There are 10 attributes associated with each instance of the dataset.

Features or attributes of the dataset along with explanations are as follows:

1. address (nominal) : Bitcoin address
2. year (numeric) : Year of the bitcoin transaction
3. day (numeric) : Day of the year. First day represented by 1 & last day is represented by 365.
4. length (numeric) : Length is designed to quantify mixing rounds on Bitcoin, where transactions receive and distribute similar amounts of coins in multiple rounds with newly created addresses to hide the coin origin
5. weight (numeric) : Weight quantifies the merge behavior i.e., the transaction has more input addresses than output addresses
6. count (numeric) : count feature is designed to quantify the merging pattern i.e., it represents information on the number of transactions
7. looped (numeric) : It is intended to count how many transactions split their coins, move these coins in the network by using different paths and finally, and merge them in a single address.

8. neighbors (numeric)
9. income (numeric) : Satoshi amount (1 bitcoin = 100 million satoshis)
10. label (nominal) : Name of the ransomware family

Q2. How data mining might prove to be beneficial for the given dataset?

The data mining techniques, if successfully applied, could prove very beneficial to the given dataset. After preprocessing of dataset various visualizations techniques can be applied for understanding the dataset better. Also by using other techniques we could efficiently unearth interesting patterns from the dataset such as:

➢ which month or a set of months contains the most ransomware payments every year?, or
➢ which year in the past 10 years proved to be most affected by the ransomware transactions both in terms of total incurred income and the frequency of the transactions?, etc.

For the given dataset predictive analysis can be very useful. It can be used to predict (classify) if a given transaction instance of the dataset belongs to a ransomware or a non-ransomware payment so that the transaction could be flagged in time.

Multiple clustering techniques like K-means, DBSCAN algo, etc can also be applied to make useful clusters of transactions, and can further explore & extract information from various formed clusters.

Q3. What type of data mining task (classification, clustering, etc.) would be relevant for the given dataset? Illustrate with an example.

Data mining techniques such as clustering could be relevant for the given dataset. We could perform clustering of all the transactions in the dataset so that transaction instances showing similar behaviour could be clustered together and further used for extracting various patterns from them. For example, the transaction instances belonging to different ransomware families such as Cryptxxx, cryptolocker etc. might be grouped separately into different clusters and white transactions (that are non-ransomware payments) could be grouped in a separate cluster. Now, using the formed clusters we could find various patterns & answer the questions such as :

I. What is the average 'income' of each of the found clusters? This could help in finding out which ransomware family has the most or the least 'income' associated with it.

II. Which cluster among all has the most number transactions associated with them using the 'count' attribute?

Predictive analysis would also be relevant for the given dataset. Using the 'label' attribute given in the dataset we could train a model that can classify a given transaction instance to be either belonging to the 'ransomware payment' class label or to the 'non-ransomware payment' (white payment) class label.

Q4. For at least five attributes, discuss data quality issues. What might be an appropriate response to the quality issues?

There are 10 features or attributes of a given instance in the dataset. Among the 10 attributes 8 attributes are numeric (have either ratio or interval values), and other 2 attributes are nominal (one of which is the address of the transaction & other is the label given to it).

Data quality issues and the appropriate responses to some of the attributes are as follows:

1. year (numeric) : Year of the bitcoin transaction

   a. Data Quality Issues: 'year' attribute according to the collected dataset must only contain the years from 2009 to 2018, so there could be some cases where due to some human or machine error the year goes out of this range. There could also be a case in which the year attribute is completely missing.

   b. Responses to data quality issues: As a response to the mentioned data quality issues we could find such instances with out-of-range or missing year attribute values and replace it by the median or mode of all the year attribute values or by boundary years (2009 & 2018).

2. day (numeric) : Day of the year. First day represented by 1 & last day is represented by 365.

   a. Data Quality Issues: The day of the year might be missing or might be out of range [1, 365] due to some insufficient data collection or human error respectively.

   b. Responses to data quality issues: As a response to the mentioned data quality issues we could find such instances with out-of-range or missing day attribute values and replace it by the median or mode of all the year attribute values or by boundary years (1 & 365).

3. looped (numeric-Integer) : looped feature is designed to quantify the looping pattern i.e., it represents information on the number of times the transaction is looped.

    a. Data Quality Issues: Majority of values in this for this feature are 0.

    b. Responses to data quality issues: looped can thus be removed from the dataset to reduce the dimensionality of the dataset and minimize the cost of the training model.

4. income (numeric) : Satoshi amount (1 bitcoin = 100 million satoshis)

    a. Data Quality Issues: The attribute ***income*** in the dataset has a very wide range (between 3.000e7 & 4.996e13) and very different scale from other attributes and so this could severely affect the multivariate analysis, if performed.

    b. Responses to data quality issues: we can apply normalization techniques for scaling and normalizing the data.

5. weight (numeric) : Weight quantifies the merge behavior i.e., the transaction has more input addresses than output addresses

    a. Data Quality Issues: The range of weight varies from 3.61e-94 to 1.94e+03 which is a very wide range of values and at a different scale than other attributes.

    b. Responses to data quality issues: The feature needs to be scaled. Normalization and Standardization are the two methods through which the feature can be scaled. Mean normalization is best suited here as it will also reduce the effect of outliers.

Q5. For at least one attribute, discuss an appropriate normalization or data reduction technique. Illustrate with the help of an example on a small subset of the data.

The attribute ***income*** in the dataset ranges between 3.000e7 & 4.996e13 which as can be seen is quite a wide range of values. Hence, we can apply normalization technique to the ***income*** attribute of the dataset to make it fall between a smaller range of values.
For the normalization technique, there are many to choose from; we could either perform min-max normalization (converts the range of values between 0 & 1), standard normalization, or mean normalization to it.

Illustration of applying standard normalization (or standardization) to the ***income*** attribute is as follows:

income attribute original values from subset of the dataset (containing 200 values):

```
Name: income, Length: 200, dtype: float64

0      100050000.0
1      100000000.0
2      200000000.0
3       71200000.0
4      200000000.0
                ...
                ...
195   200000000.0
196     64882935.0
197     74030000.0
197     74030000.0
198   144066297.0
199   200000000.0
```

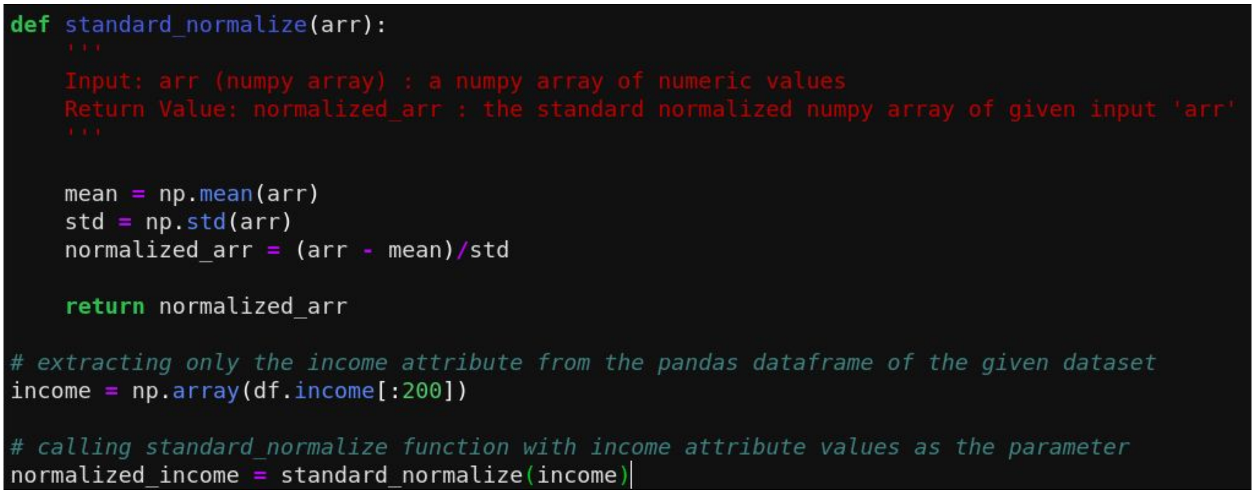
Calculations:

1. Mean of values, **mean** : **156302656.32**

2. Standard Deviation of the values, **std** : **129784827.14**

For each value *X*, the standard normalized value is calculated as: (*X* - **mean**)/**std**

Code Snippet:



```python
def standard_normalize(arr):
    '''
    Input: arr (numpy array) : a numpy array of numeric values
    Return Value: normalized_arr : the standard normalized numpy array of given input 'arr'
    '''

    mean = np.mean(arr)
    std = np.std(arr)
    normalized_arr = (arr - mean)/std

    return normalized_arr

# extracting only the income attribute from the pandas dataframe of the given dataset
income = np.array(df.income[:200])

# calling standard_normalize function with income attribute values as the parameter
normalized_income = standard_normalize(income)
```

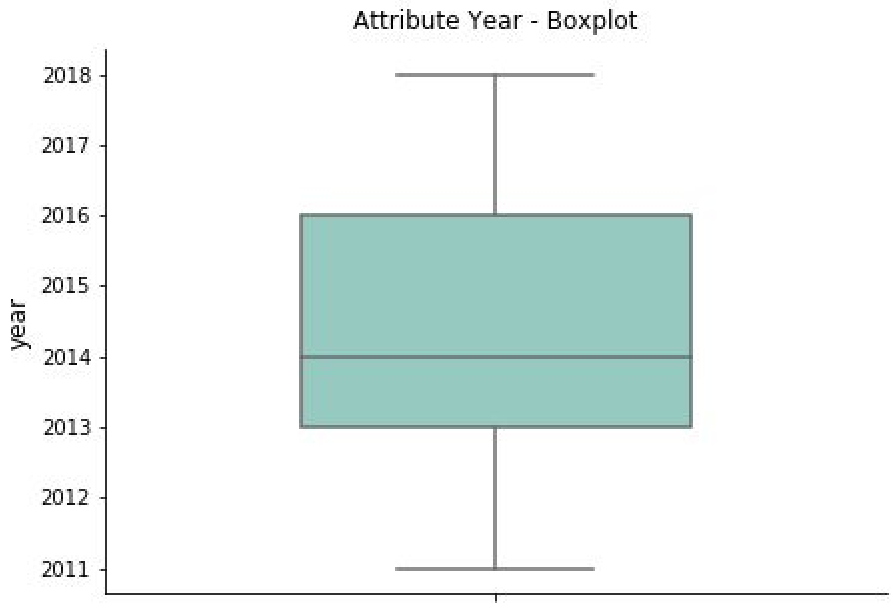
After normalization, ***income*** attribute values are transformed as follows:

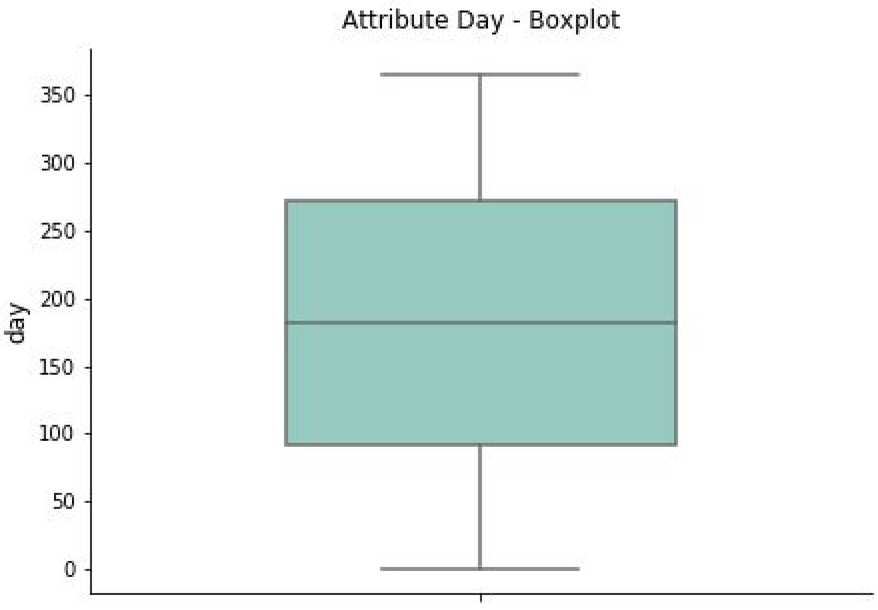
```
Name: income, Length: 200, dtype: float64

0     -0.433430
1     -0.433815
2      0.336691
3     -0.655721
4      0.336691
              …
              ...
195   0.336691
196  -0.704395
197  -0.633916
198  -0.094282
199    0.336691
```

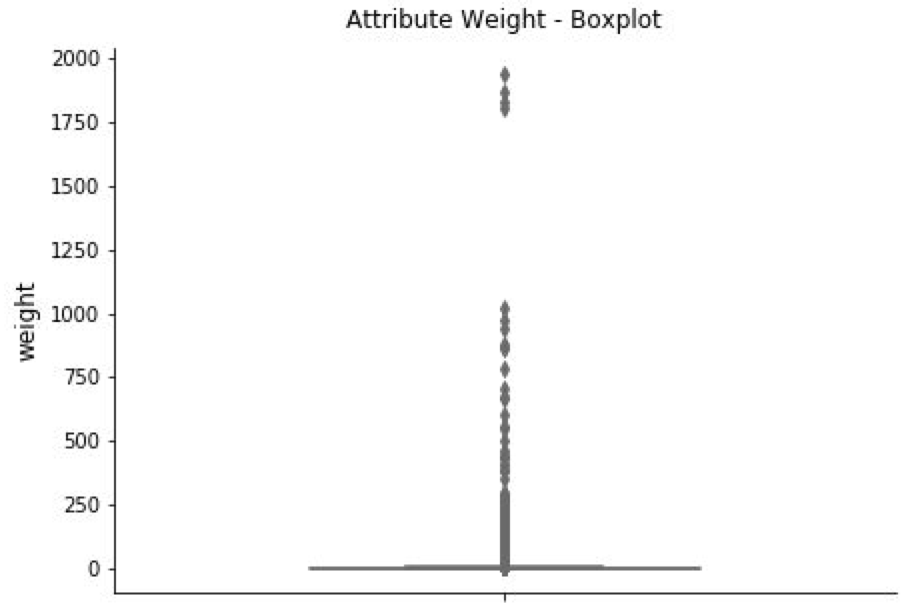
New range of the income attribute after standard normalization is between -0.97316 & 5.196273.

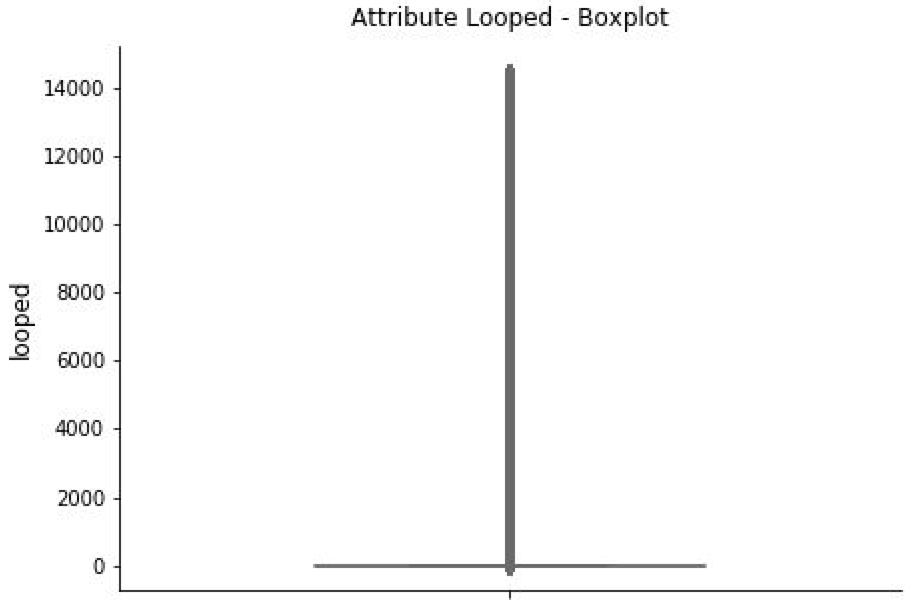
Q6. For any five attributes from the dataset, apply different visualization techniques (histogram 1-D and 2-D, scatter plot 2-D and 3-D, box-plot) using MATLAB, R or Python.

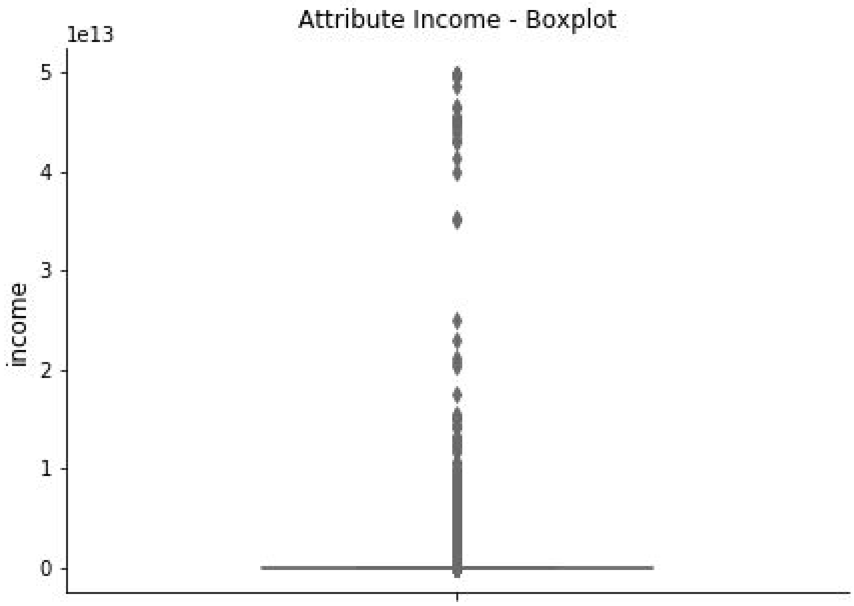
BOXPLOTS:


Attribute Year - Boxplot
year
2018
2017
2016
2015
2014
2013
2012
2011

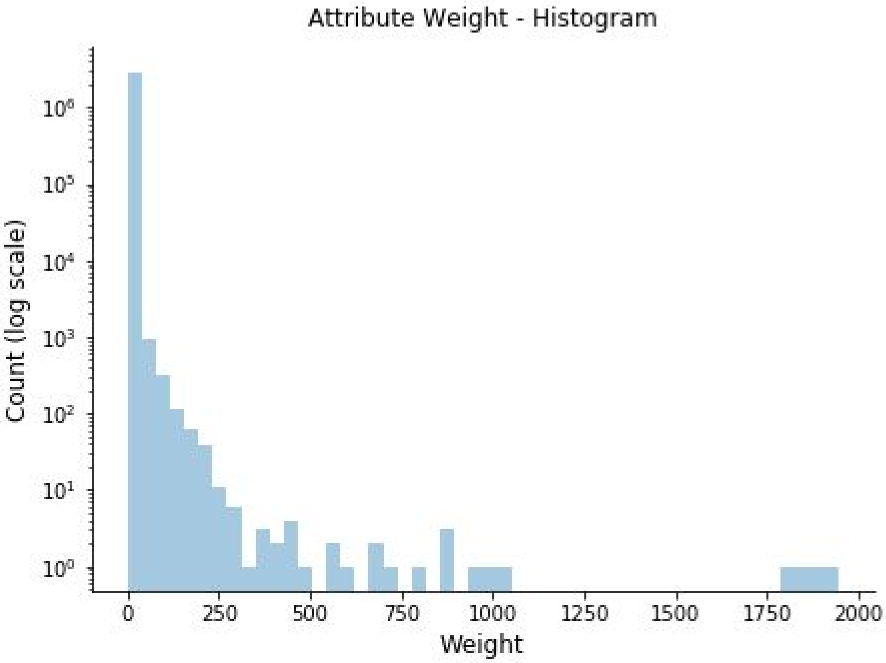

Attribute Day - Boxplot
day
350
300
250
200
150
100
50
0

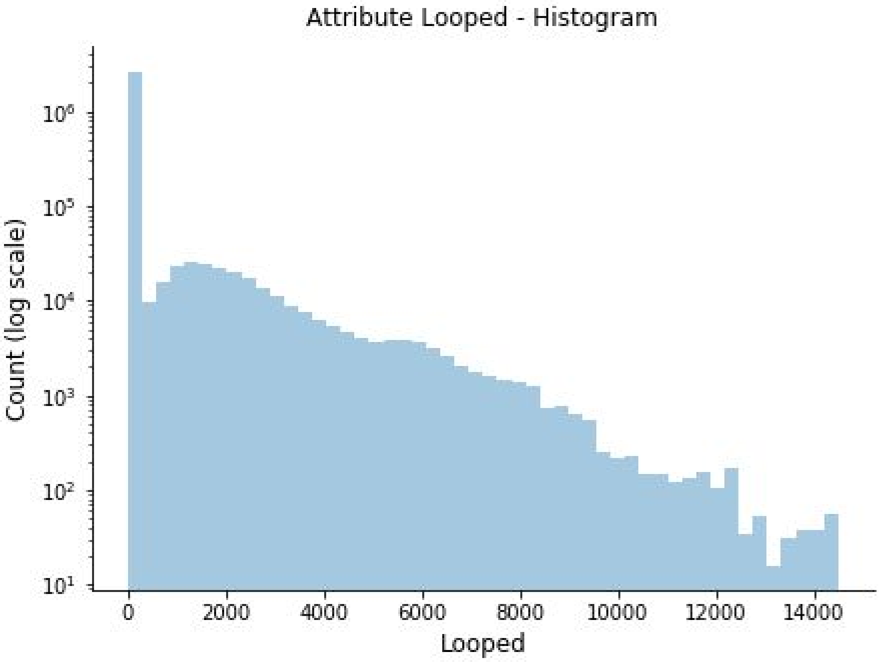

Attribute Weight - Boxplot
weight
2000
1750
1500
1250
1000
750
500
250
0

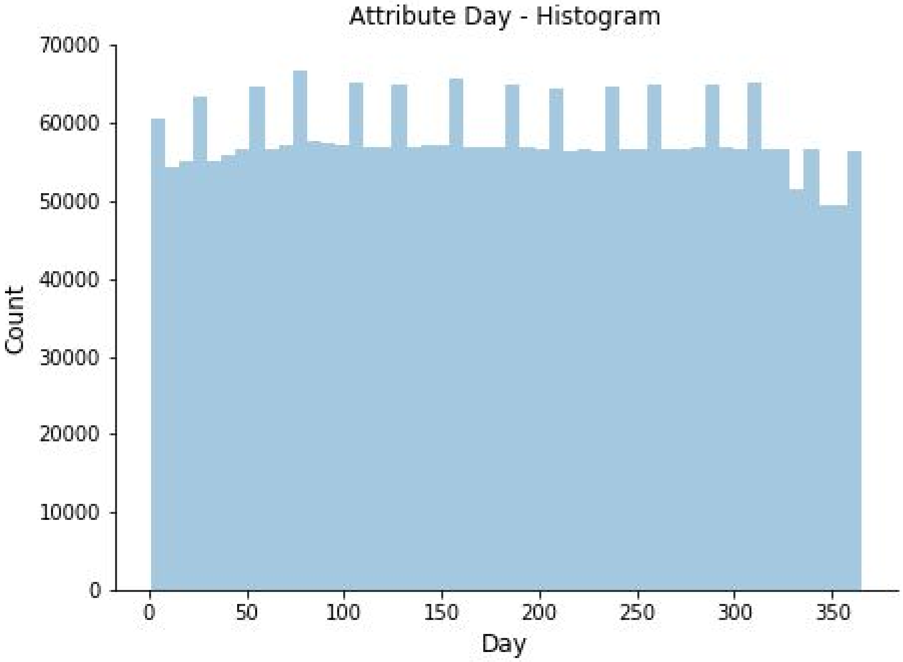

Attribute Looped - Boxplot
looped
14000
12000
10000
8000
6000
4000
2000
0

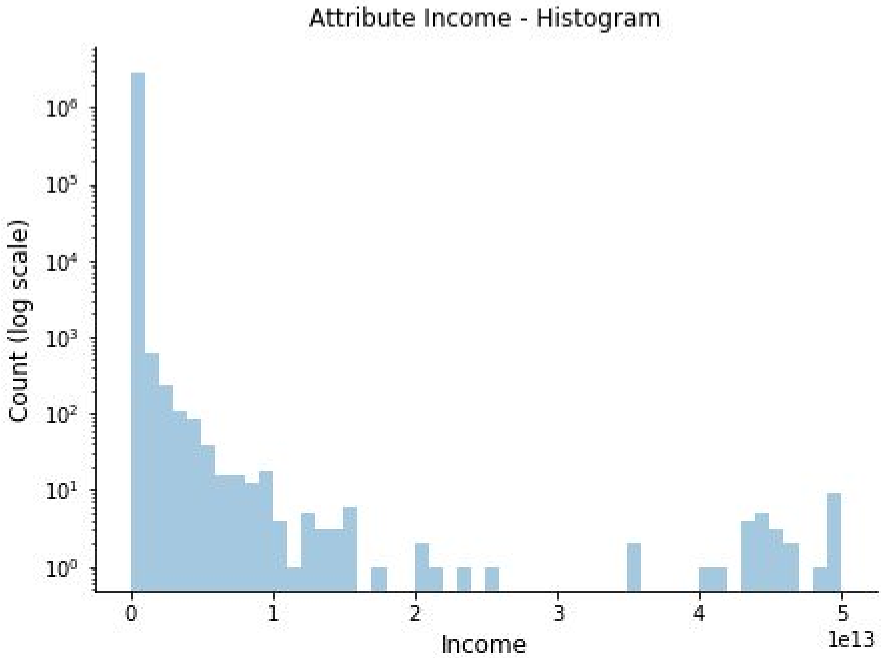

Attribute Income - Boxplot
1e13
5
4
3
2
1
0
income

HISTOGRAMS:

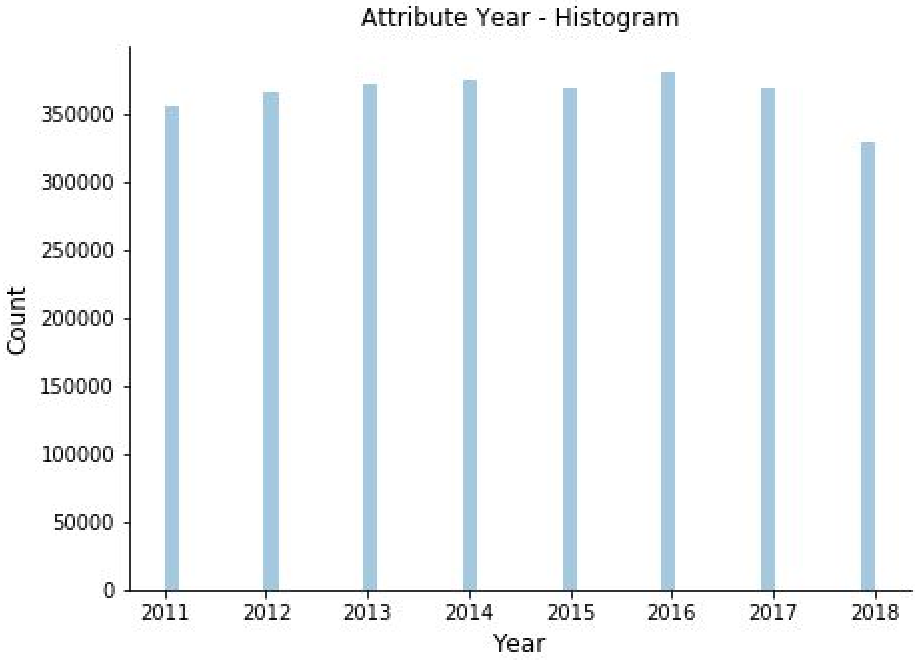

Attribute Weight - Histogram
10^6
10^5
10^4
10^3
10^2
10^1
10^0
Count (log scale)
0
250
500
750
1000
1250
1500
1750
2000
Weight


Attribute Looped - Histogram
Count (log scale)
10^6
10^5
10^4
10^3
10^2
10^1
0
2000
4000
6000
8000
10000
12000
14000
Looped


Attribute Day - Histogram
70000
60000
50000
40000
30000
20000
10000
0
Count
0
50
100
150
200
250
300
350
Day

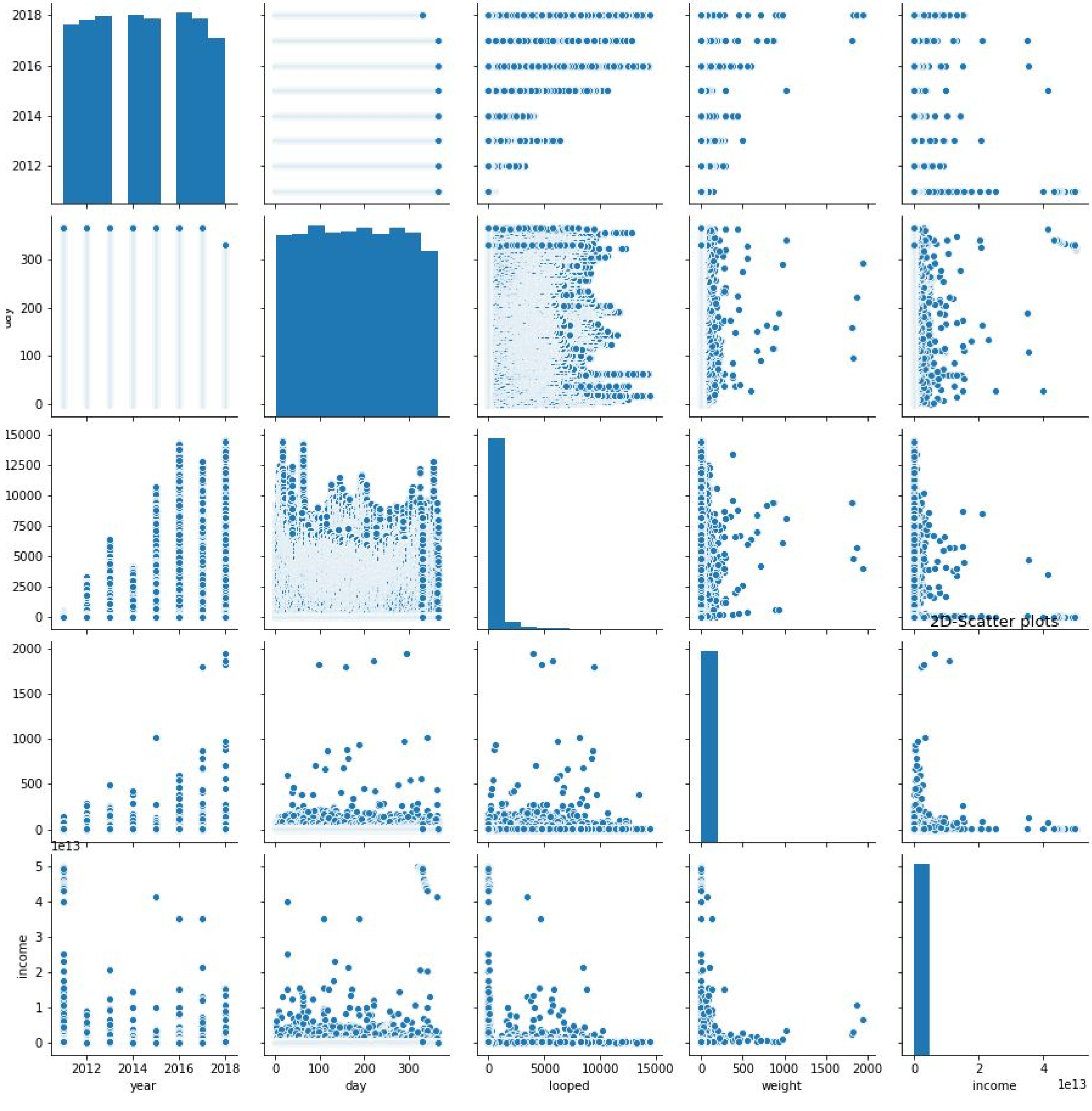


Attribute Income - Histogram

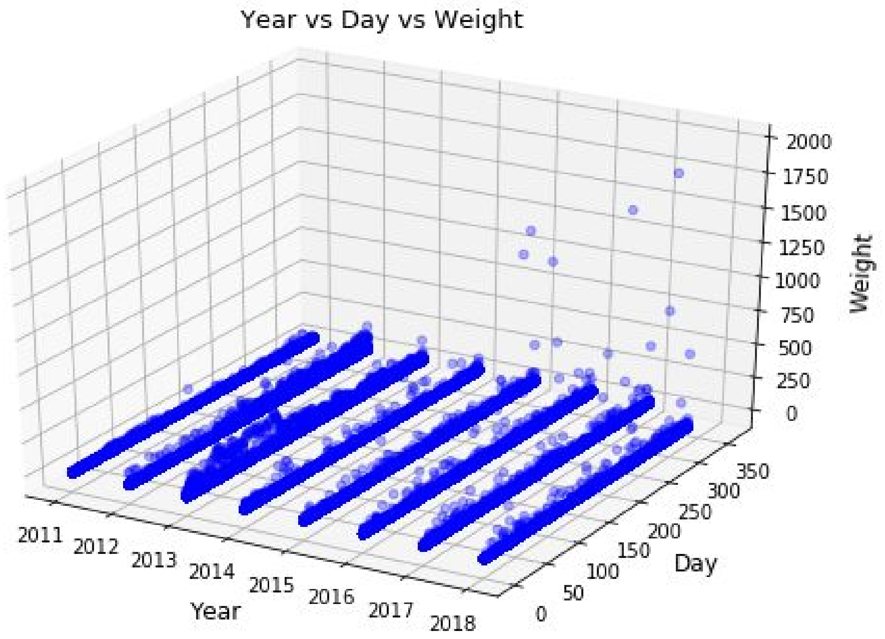


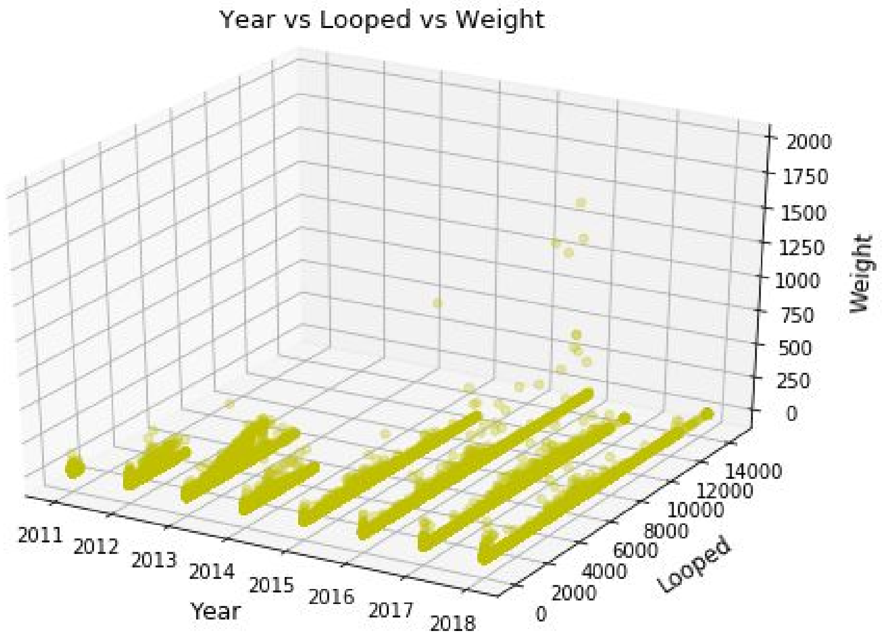
Attribute Year - Histogram

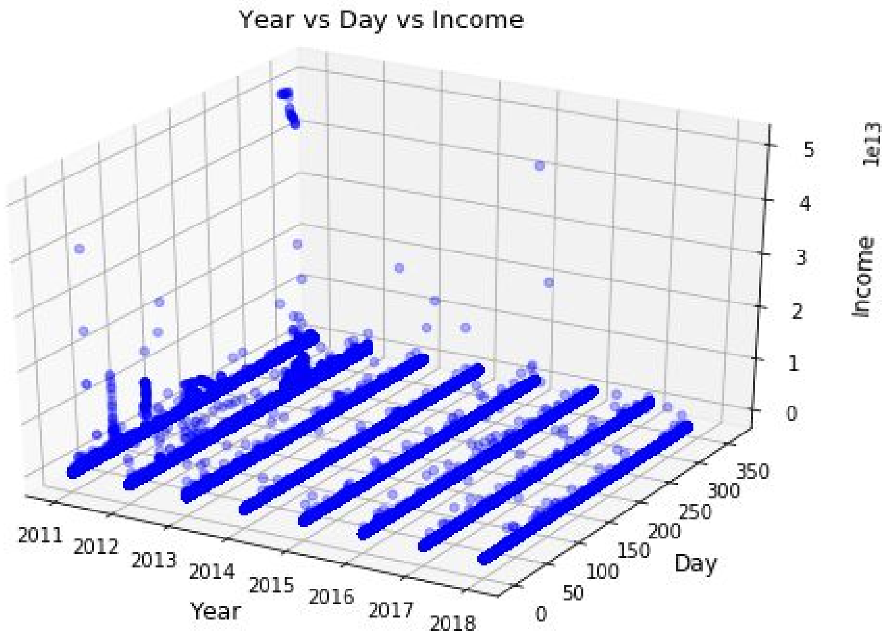
2D SCATTER PLOTS

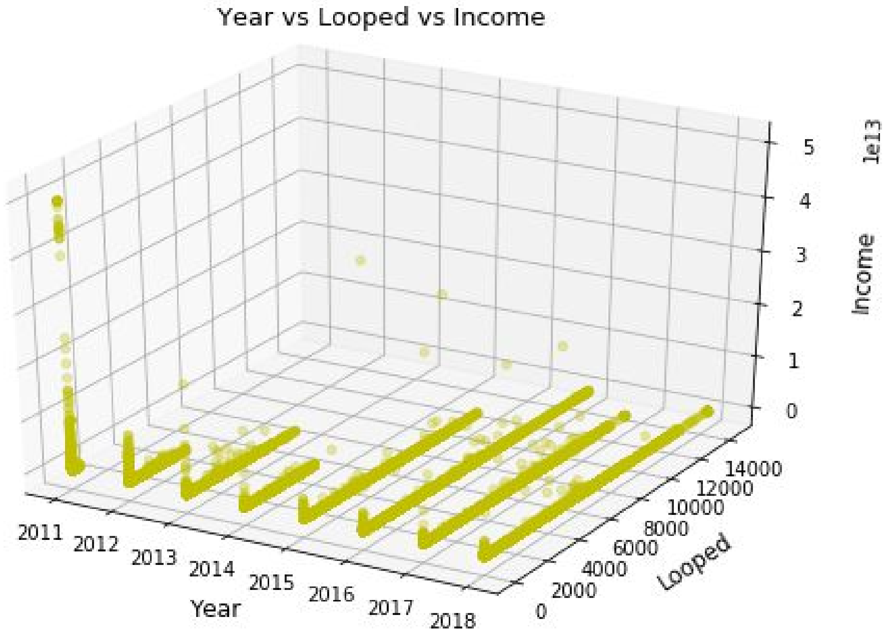


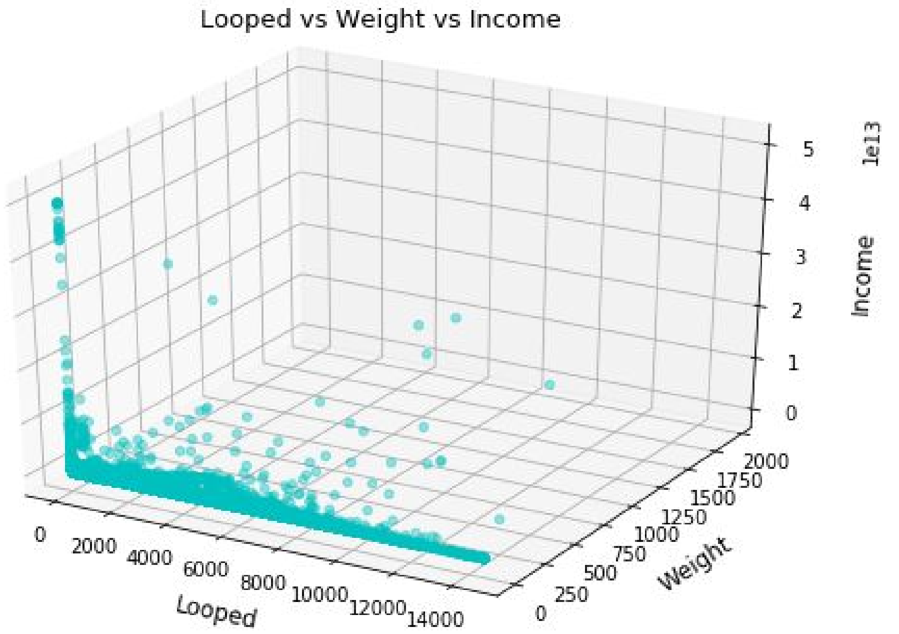
3D SCATTER PLOTS

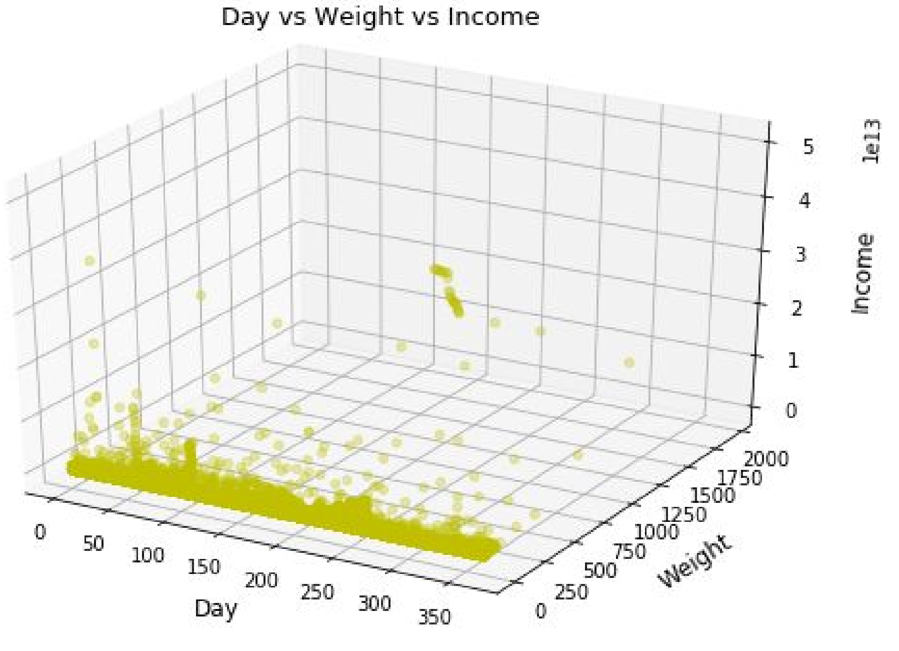


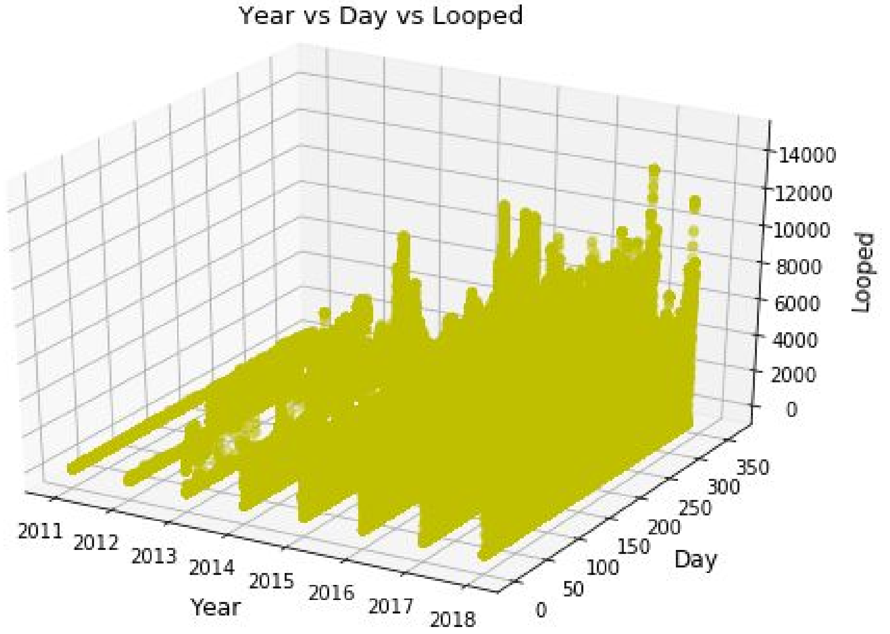


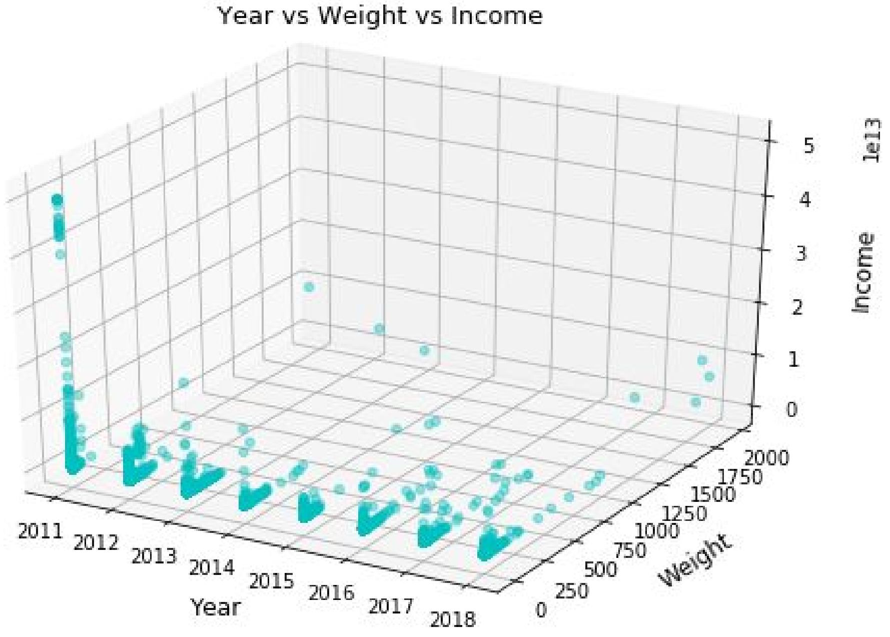

Year vs Day vs Income
1e13
5
4
3
2
1
0
Income
350
300
250
200
150
100
50
0
Day
2011
2012
2013
2014
2015
2016
2017
2018
Year

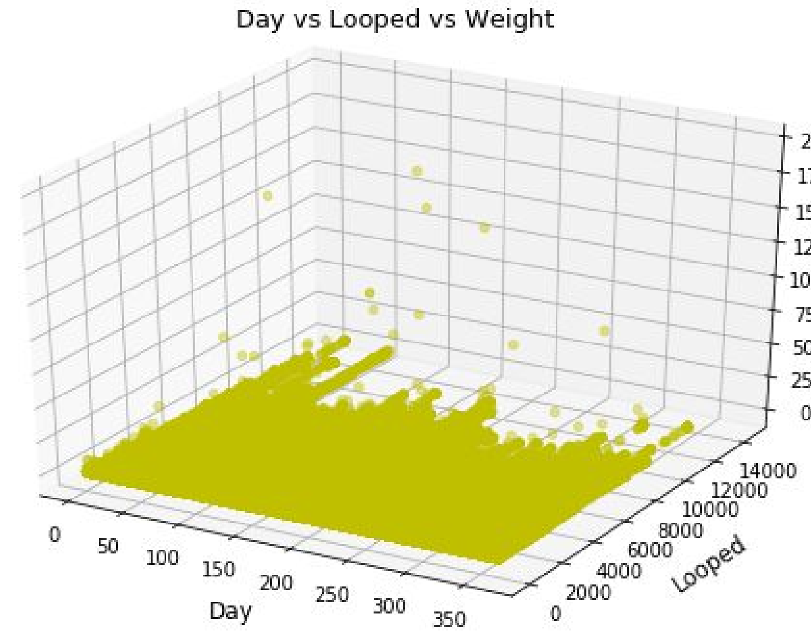

Year vs Looped vs Income
1e13
5
4
3
2
1
0
Income
14000
12000
10000
8000
6000
4000
2000
0
Looped
2011
2012
2013
2014
2015
2016
2017
2018
Year

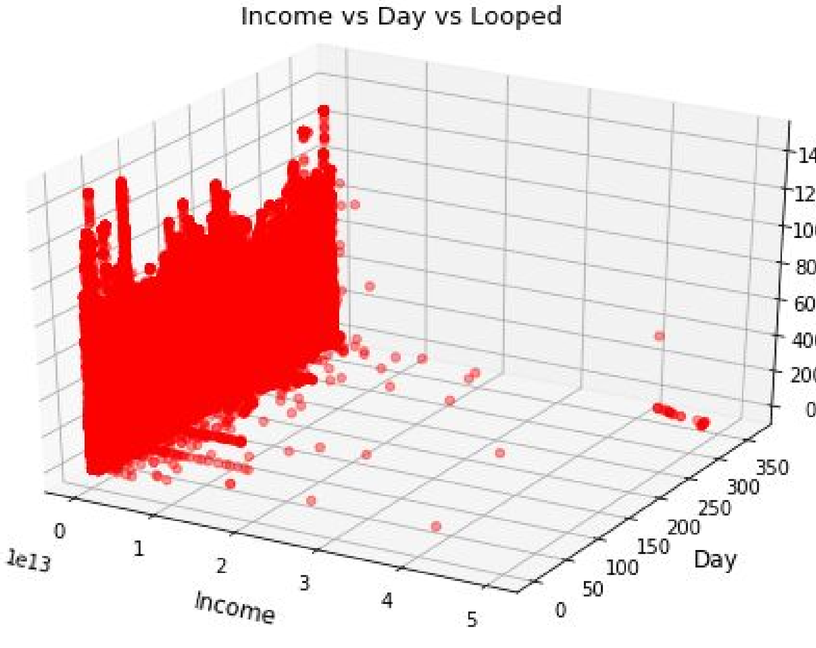

Looped vs Weight vs Income
1e13
Income
0
1
2
3
4
5
Looped
0
2000
4000
6000
8000
10000
12000
14000
Weight
0
250
500
750
1000
1250
1500
1750
2000


Day vs Weight vs Income
1e13
Income
0
1
2
3
4
5
Day
0
50
100
150
200
250
300
350
Weight
0
250
500
750
1000
1250
1500
1750
2000


Year vs Day vs Looped
Looped
14000
12000
10000
8000
6000
4000
2000
0
Day
350
300
250
200
150
100
50
0
Year
2011
2012
2013
2014
2015
2016
2017
2018


Year vs Weight vs Income
1e13
Income
5
4
3
2
1
0
Weight
2000
1750
1500
1250
1000
750
500
250
0
Year
2011
2012
2013
2014
2015
2016
2017
2018


Day vs Looped vs Weight
Weight
2000
1750
1500
1250
1000
750
500
250
0
Day
0
50
100
150
200
250
300
350
Looped
0
2000
4000
6000
8000
10000
12000
14000


Income vs Day vs Looped
Looped
14000
12000
10000
8000
6000
4000
2000
0
Income
1e13
0
1
2
3
4
5
Day
0
50
100
150
200
250
300
350