BIA 652A - Multivariate Data Analytics

Final Project Report

Fall - 2021

Fraud Detection in Electricity and Gas Consumption

Team A5

Malarvizhi Shanmuga Pandian

Monish Bangera

Table of Contents:

1. Introduction:

Tunisia, an African country with 11.7 million population and with 43.5% energy independence. The Tunisian Company of Electricity and Gas (STEG) is a public and non-administrative company, it is responsible for delivering electricity and gas across Tunisia. STEG has established kilometers of pipes and lines to provide electricity and gas to millions of people in Tunisia. But one of the main problems faced by them is the imbalance in energy provided and energy consumed. Most of the common reasons for this energy-loss are the accidental malfunction of the meter, tampering with the meter, bypassing the meter by splicing the pipes. Fraudulent consumption of energy due to these reasons can be identified by technicians inspecting personally. But doing that will cost the company some money for every visit. It is said that STEG suffered tremendous losses in the order of 200 million Tunisian Dinars due to fraudulent manipulations of meters by consumers.

1. Project goal:

Using the client’s billing history, the aim of the project is to detect and recognize clients involved in fraudulent activities. The solution will enhance the company’s revenues and reduce the losses caused by such fraudulent activities. A supervised approach will be followed to detect fraud and irregularities in the energy utilizing company. Fraud detection is treated as a classification problem where the supervised techniques are applied to the set of historic cases of fraud.

1. Data Description:

This dataset is publicly available in one of Tunisian predictive analytics challenge[1] and consists of two sets of data. One dataset with client details and the other with meter readings. A common variable between both dataset is the *Client\_ID* on which both of the datasets are merged into one. The dependent variable is a boolean variable denoting if the particular customer is a fraud or not. The following numbers denote the combined dataset after merging Client and Invoice datasets.

Total no. of rows: 1465

Independent variable count: 17 (Int: 14, Object:4, Float: 1)

Dependent variable: Target (Type: Int) [1 - Fraud, 0 - Not Fraud]

Among all the independent variables there are 12 categorical variables, 5 continuous variables. The final merged dataset consists of 53.2 and 46.8 percentage of fraud and non-fraud customers.

|  |  |
| --- | --- |
| **Variable Name** | **Description** |
| Client\_id | Unique id for the client |
| Invoice\_date | Date of the invoice |
| Tarif\_type | Type of tax |
| Counter\_number | Counter number |
| Counter\_statue | Takes up to 5 values such as working fine, not working, on hold status, etc. |
| Counter\_code | Counter code |
| Reading\_remarque | Remarks given by STEG agent during client's visit |
| Counter\_coefficient | An additional coefficient to be added when standard consumption is exceeded |
| Consommation\_level\_1 | Consumption level 1 |
| Consommation\_level\_2 | Consumption level 2 |
| Consommation\_level\_3 | Consumption level 3 |
| Consommation\_level\_4 | Consumption level 4 |
| Old\_index | Old index |
| New\_index | New index |
| Months\_number | Month number |
| Counter\_type | Type of counter |
| District | District where the client is |
| Client\_catg | Category client belongs to |
| Region | Area where the client is |
| Creation\_date | Date client joined |
| Target | fraud:1 , not fraud: 0 |

Exploratory Data Analysis:

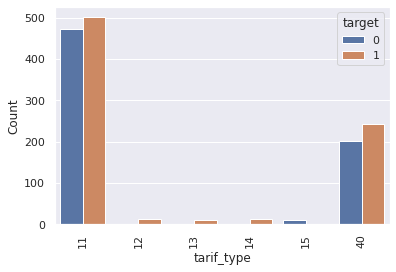
The dataset consists of two sets of data like client data and invoice data. The main idea of data exploration is to find if there will be any pattern with the rate of consumption per month by every customer, by which the client can be categorized as fraud or not. In Fig1 the black line is the average energy consumption by all the customers in a particular month. The red line and green line depict the energy consumption of a fraudulent and non-fraudulent customer respectively every month. A few points were observed from the graph (Fig1) on the consumption rate for every month by customer.

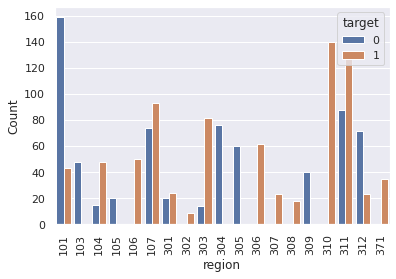
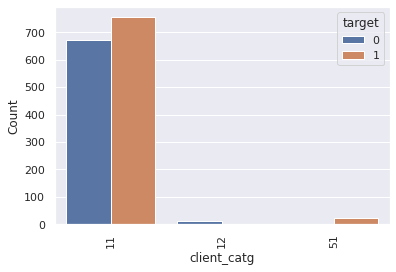
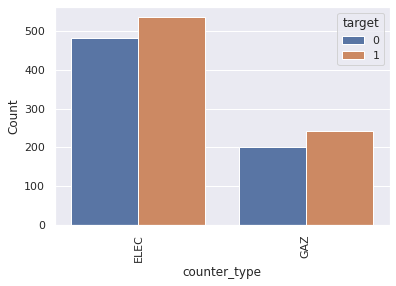
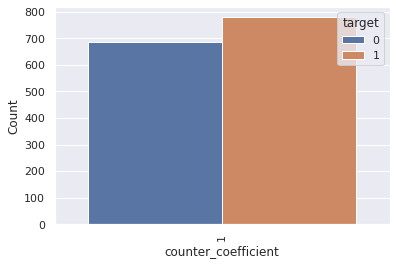
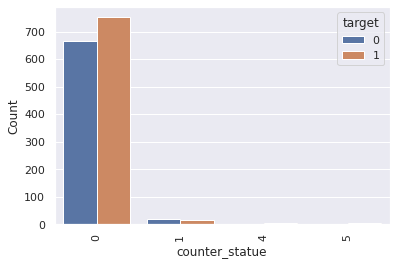
1. Consumption of one non-fraudulent customer is somewhat aligned with the average consumption of all the customers. This means there is no outlier in this case and it is the same in most of the non-fraudulent customers.
2. Whereas, the consumption pattern of a fraudulent customer is unpredictable and fluctuating. Most of the time the consumption is low and it suddenly peaked may also indicate the actual consumption might have not been recorded.

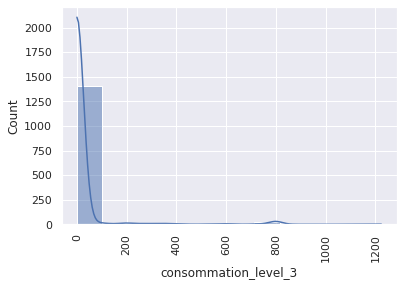
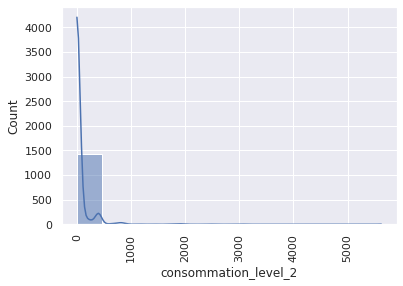
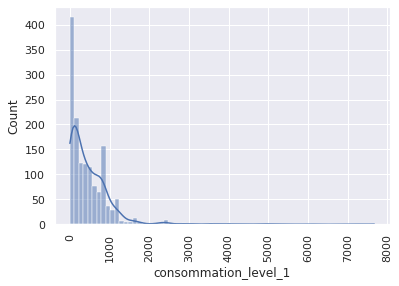
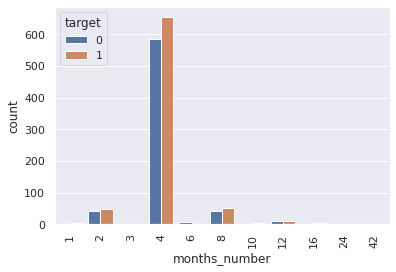
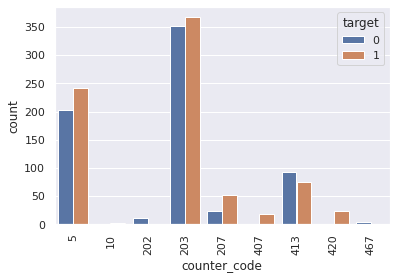
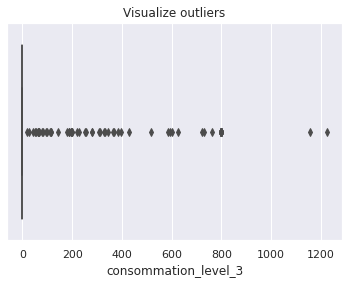
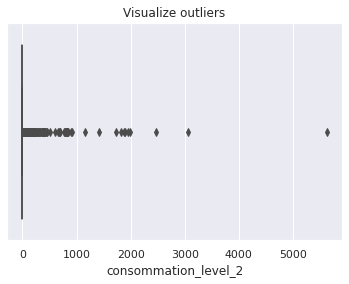
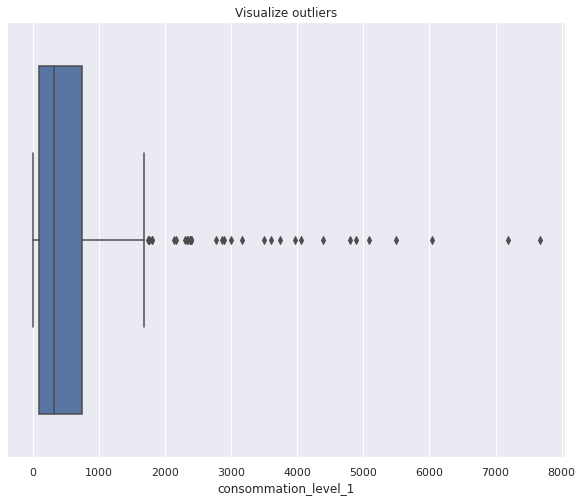


Fig 1. Consumption rate pattern for fraud vs non-fraud vs average

Apart from the above data pattern between variables, there are also a few more graphs drawn to explore the range of data values and count of them in each variable. Those graphs are as follows.







Data Preprocessing:

Below are steps performed as part of data cleaning in the order it is performed.

1. **Removed unimportant independent variables** - features like client\_id, old\_index, new\_index, invoice\_date are removed
2. **Remove null and zero values:** There were no many rows with null or zero values
3. **Create dummy variables**: Categorical variables are encoded using label encoder and as a result, a total of 38 columns is present in the dataset.
4. **Correlation Matrix**: All variables are checked for the high correlation between them and it found that counter\_type\_GAZ and tarif\_type\_40 are highly correlated with correlation coefficient equal to 0.98. Having two highly correlated variables will result in multicollinearity problem, thus the column tarif\_type\_40 is dropped from the dataset for further processing.
5. **Feature scaling** is used to normalize continuous variables using StandardScaler. Doing this helps will convert all the continuous variables in different range of values into a unique value between –3 and +3, helping the model to give equal weightage to all the values while training.
6. **Balancing dataset:** As there was not much difference in the count of fraud and non-fraud customers in the dataset(53.2 and 46.8 percentage respectively), it is not attempted to balance the dataset.
7. **Train and test data split:** Cleaned and processed dataset is now split into train and test data which will be 75 and 25 percentage of the entire dataset.

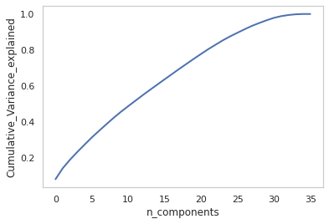
Model Selection

In order to do the fraud clients prediction, a few of the best performing binary classification algorithms are going to be used like Logistic regression, Decision Tree classifier, and Random forest classifier. Principal Component Analysis (PCA) a dimension reduction technique is also used in combination with each algorithm. The data will be fitted into these models and the performance of each algorithm is compared in cases with and without doing PCA before training. For modeling a hyperparameter tuning technique GridSearchCV is used to tune and determine the best estimators in order to improve the performance of the algorithms. The order of algorithms used in the tuning pipeline and the parameter set passed for each to them is described in the following section.

**1.1 Principal Component Analysis(PCA)**

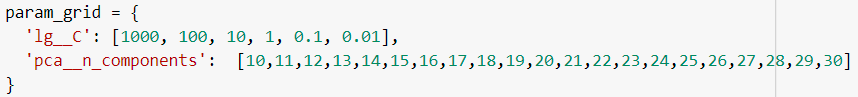
PCA is commonly used for dimension reduction with minimal loss of information to improve model performance. Principal components are the new variables created by calculating eigenvector and eigenvalues from covariance matrix of the original dataset. For our dataset, in total of 36 (total number of independent variable in cleaned dataset) eigenvectors and eigenvalues will be created. Principal components represents the angle of line that captures most of the information i.e. maximal amount of variance. From this, a Feature Vector is eigenvector with dimension passed as parameter. Thus, multiplying the feature vector and standardized original dataset will give a dimension reduced dataset.

Final Dataset = FeatureVectorT  \* StandardizedOrginalDatasetT

Comparison of number of principal components and the cumulative variance is represented in the graph below. 

**2.1 Logistic Regression**

Logistic regression is a statistical model that in its basic form uses a logistic function to model a binary dependent variable, as its output value strictly ranges between 0 and 1. This algorithm predicts a hypothesis which is the estimated probability of how confident can the predicted value be the actual value when given an input. Logistic function is given by the formula 1/(1+e-x) where x is the input variable. Once the probability is calculated, logistic regression uses ‘maximum likelihood estimation’ loss function which is a conditional probability i.e. probability above 0.5 is considered as class 1 and below 0.5 is considered as class 0. The parameter grid used for GridSearchCV is given in fig



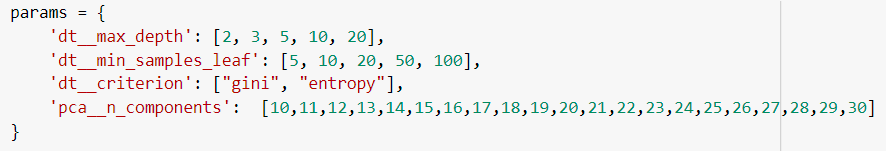
**2.2 Decision Tree classifier**

Decision Tree Classifier uses the dataset features to create yes/no questions and continually split the dataset until it isolates all data points belonging to each class. The node is called the root node and each further nodes are split into two nodes. Leaf nodes are the last nodes created when the process is stopped and expectation is that it consists of all or majority of data items from one class. At each node split, the feature that should be used to split is selected based on purity of the resulting nodes. Two main loss functions that compares the class distribution before and after the split are Gini Impurity and Entropy.

* Gini Impurity is the measure of variance across different classes. It is calculated using the formula G(node) = Σk=1 pk(1-pk) where pk and (1- pk) are the probability of picking a data point from class k respectively. And the value pk is derived from following formula

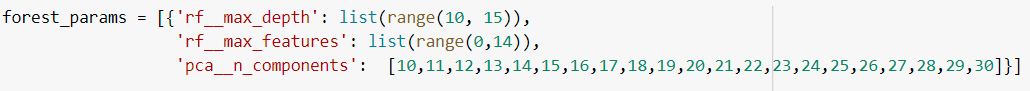
pk = number of observation with class k/ all observations in node

* Entropy determines the measure of observations from different classes in one node. The split is only performed if Entropyresulting nodes > Entropyparent node and the value of entropy is given by the formula



**2.3 Random Forest Classifier**

Collection of decision tree classifier is called forest. The dataset is randomly split that are used to generate decision tree. Each individual decision tree make use of gini or entropy function to determine the splits at each node. After every decision tree completes predicting class, the most popular class determined by voting is declared as the predicted class by random forest classifier.



**Evaluation**

Performance of each algorithm is measured using various metrics like accuracy, precision score, recall score, f1-score, area under curve, etc. Here for performance evaluation Area Under Curve (AUC) of each algorithm with and without PCA is compared.

References:

[1] <https://zindi.africa/competitions/ai-hack-tunisia-4-predictive-analytics-challenge-1/data>

[2] <https://towardsdatascience.com/decision-tree-classifier-explained-in-real-life-picking-a-vacation-destination-6226b2b60575>

[3] <https://www.datacamp.com/community/tutorials/random-forests-classifier-python>