**Applied Data Science Lab 3 report: Fraud Detection**

**Introduction**

For this project, I attempted to create a method for finding fraudulent transactions in a large data set. To do this, I used a training set that had the fraudulent transactions marked to test various methods and algorithms. Once I had decided on an algorithm, I implemented it on a second, unlabelled, data set and found predictions for which transactions were fraudulent.

Since the first section makes up a substantial proportion of the project, I have split it up into 2 sections. The first section describes my initial analysis of the data set including the production of statistics. The second explains my thought process in trying clustering and other methods for filtering the data to look for fraudulent transactions.

**Task 1 – Part 1: Analysis of the training data**

For each transaction, I was given 5 values: the client id which separated different buyers, transaction value which gave the amount paid in British pence, merchant id with separated different sellers, date and time. Each transaction was also separated by a transaction id.

Since it seemed the most obvious way of looking for fraud, I first looked at transaction value on its own, this is shown in Fig. 1. From that figure, the transactions fit into three distinct price bands which I have separated with red vertical lines. The low-price band includes all transactions cheaper than £57 and the high-price band includes all transactions more expensive than £200. Each price band presumably represents a different kind of shopping. I then calculated some statistics on the transaction value for these three bands. These can be found in Table 1.

While studying the merchants, I discovered that 96.2% of them always make transactions in a single price band. Influenced by this, the merchants were mostly neglected during the production of filtering methods. Similarly, 97.6% of clients always make transactions in the same price band, this is discussed further in Task1 – Part 2.

**A screenshot of a cell phone

Description generated with very high confidence**Fig. : A plot showing the distribution of transaction values for the training data. The red lines mark separations between the three price bands.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Price band | Mean value (p) | Maximum value (p) | Minimum value (p) | Std. Dev. (p) |
| All transactions | - | 49903 | 651 | - |
| High | 36039 | 49903 | 20986 | 4107.4 |
| Middle | 11996 | 19289 | 5897 | 1639.4 |
| Low | 3129.4 | 5368 | 651 | 676.22 |

Finally, I looked at the time and date columns. By plotting them against transaction value I looked for any visible patterns or trends. Despite plotting date as it came and using it to plot day of the week and week against transaction value, I saw no trends for all data. The only visible trend (found when I converted the time into hours of the day) was that the number of transactions tends to decline between 2 am and 7 am (approximately). This, however, didn’t seem likely to be very informative.

When the time and date were plotted against transaction value for a single client there seemed to be more correlation. Although the correlation was not strong, both features were included during clustering since they were determined to sufficiently useful.

**Task 1 – Part 2: Clustering**

Principle component analysis (PCA) was used for two reasons during this portion of the project. First, it was used to visualise data such that general trends could be seen. This was essentially used as a test for the effectiveness of the clustering algorithm. Secondly, it was used at the beginning to negate the “curse of dimensionality” that came about from using one-hot encoding on the categorical merchant data. Although this method worked, the time taken to perform this encoding was greater than the time taken to perform the clustering on the larger number of features. Further, the point was rendered moot by the neglecting of the merchant id as a useful variable in the final filtering method.

The method used came in two parts, price filtering and cluster filtering. The first was computationally fast and so was run first to reduce the number of clients tested in the second part. As mentioned in Task 1 – Part 1, 97.6% of the clients in the first data set were found to have made all of their transactions in a single price band. Since this corresponded to a set of just over 10 clients, containing the set of all clients with fraudulent transactions, it was used as an initial filter - the price filter.

For the cluster filter, the algorithm looped through the remaining clients. For each client, the algorithm used K-means clustering 18 times on all transactions made by that client, using between 2 and 19 (inclusive) clusters. Each time the clustering was used it would look for a cluster that contained only one transaction. This single-element cluster was assumed to contain the outlier and stored. Once all 18 runs were complete, a majority vote would determine which transaction was most likely to be the outlier for that client. If a single-element cluster was never found, it was assumed that the client being tested did not have a fraudulent transaction.

Using this method, the transactions considered fraudulent were those given in Table 2. Of them, the only transaction falsely labelled was transaction 40667. Since this transaction was made by client 175, who did have a fraudulent transaction, this method was considered sufficiently accurate.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Transaction id | 48476 | 67228 | 51558 | 42283 | 40298 | 39108 | 53168 | 34740 | 34543 | 6213 |
| Client id | 47 | 52 | 120 | 171 | 175 | 191 | 248 | 263 | 357 | 496 |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Transaction id | 9071 | 4307 | 73969 | 21605 | 26235 | 75009 | 93974 | 10603 | 21165 | 38820 |
| Client id | 11 | 28 | 93 | 131 | 247 | 266 | 325 | 385 | 412 | 414 |

**Task 2**

With the method finalised in Task 1 – Part 2, the only change was noticing that the boundary between the low- and mid-price bands was closer to £60 for the test data set.

Although many variations on the above method were tested, this was found to be the best predictor. The final version used the values for transaction value, date and time for the cluster filtering to predict that the transactions in Table 3 are fraudulent.