Nathaniel Tan
Fraud-Scam Detection Report

1. Background. Financial institutions confront a growing threat from New Account Fraud (NAF), which involves criminals using synthetic identities to open and subsequently exploit financial service accounts. The swift, anonymous processes enabled by digital banking and account creation have significantly amplified this risk. Traditional fraud detection is often reactive, focusing on transactions after an account is opened. Consequently, the industry is shifting toward proactive, data-driven strategies designed to detect and prevent NAF at the point of account creation, thus mitigating financial and reputational damage.

2. Problem Statement. Financial exploitation continues to rise, costing institutions and consumers billions annually. Traditional fraud detection methods often identify fraudulent accounts only after significant damage has occurred. How can financial institutions design a system to identify NAF before it escalates.

3. Objectives. This project aims to develop a data-driven approach to detect patterns indicative of potential NAF during their onboarding processes. The project will (1) identify attribute-based clusters of applicants using unsupervised learning; (2) build a predictive model to assess whether an application is potentially suspicious based on the attributes provided during onboarding and; (3) combine both clustering insights and predictive modelling to create a hybrid framework that supports proactive fraud detection.

4. Datasets. There are mainly two datasets used for this project from two different sources.

    a. ScamWatch. The first data set used for this project is obtained from ScamWatch[1], an initiative managed by the Australian Competition and Consumer Commission. The dataset contains real-world reports of scams and fraudulent activities submitted by consumers and

[1] The link for the first dataset (ScamWatch) is https://www.scamwatch.gov.au/research-and-resources/scam-statistics

organisations, dated from 2022 to 2025. Refer to **Annex A** for the data dictionary associated with this dataset.

(1) Caveat. While the ScamWatch dataset offers a realistic view of fraud occurrences, several limitations should be considered when interpreting the results.

(a) Data Completeness. The dataset is compiled from incidents reported by members of the public. Consequently, some incidents may be under-reported or missing details.

(b) Geographical Scope. This dataset represents scams that were reported in Australia and may not fully capture the global fraud patterns. Therefore, generalising the findings as representative of worldwide behaviour may not be accurate in real-world contexts.

b. Kaggle. The second data set used for this project is the Bank Account Fraud Dataset[2]. The dataset consists of approximately one million records and 30 features, providing a large and diverse sample of data related to bank account activities. The data set had a mix of numerical, categorical and binary variables representing customer demographics and transaction details. Refer to **Annex B** for the data dictionary for this dataset.

(1) Caveat. While the dataset provides a valuable foundation for experimentation, there are several limitations that should be considered when interpreting the results.

(a) Synthetic Data. The dataset is artificially generated and may not fully capture the complexity or unpredictability of a real-world fraud pattern. Model performance on this data set might not directly translate to real banking environments.

(b) Features Simplification. Some features in the dataset are anonymised representations of the actual financial

[2] The link for the second dataset (Kaggle) is https://www.kaggle.com/datasets/sgpjesus/bank-account-fraud-dataset-neurips-2022

indicators, which may limit the depth of real-world applicability.

(c) Ethical Consideration. The dataset does not include personal information, ensuring compliance with the Personal Data Protection Act (PDPA), making it safe for research and exploration.

5. Data Preprocessing.

a. ScamWatch. EDA and Data Wrangling were done on the ScamWatch dataset.

(1) Structure. The data set consists of 127,864 rows and nine columns. Of the nine columns, six were categorical and three were numerical.

(2) Duplication. There was no duplication found in the dataset.

(3) Nulls. There was a total of 65,208 nulls (51%) of the data in one of the features 'Amount_lost'.

(a) Handling Nulls. As this was going to be used for a clustering model, the null values were all dropped because they would introduce noise to the model with those nulls.

(4) Outliers. Under the feature of 'Amount_lost', there were a few outliers which had amounts more than one million. Decided to remove them as the amount was at the extreme.

(5) Features Removal. One feature was removed prior to the model training.

(a) Start of Month. This feature was excluded from the model training as the model aims to profile the victims to scam and has no relevance to the date. This might introduce noise to the model if they were kept.

(b) Address State. This feature was excluded from the model training because the project aims to build a victim

profile that can be used as a general clustering instead of focusing only on Australia.

(c) Category_Level_2. This feature was excluded from the model training as there was a feature on category level 3, which was more specific rather than general.

(d) Cleaning of Data. Some of the data were classified as unspecified and others were vague. These data were dropped to have a clean and specific victim profiling.

(6) Features Engineering. A new feature for amount loss grouping was created as the amount loss has too big a range from $0 to $6,000,000. The new feature was categorised as shown in **Table 1**.

Table 1: Amount Loss Group

Original (Amount Loss)	New (Amount Loss Group)
< 100	Very Low Loss
< 1,000	Low Loss
< 10,000	Medium Loss
< 100,000	High Loss
< 500,000	Very High Loss
> 500,000	Extreme Loss

a. Kaggle Dataset. Exploratory Data Analysis (EDA) and Data Wrangling were done on the Kaggle dataset.

(1) Structure. The data set consists of 1,000,000 rows and 32 columns. Of the 32 columns, eight were binary, five were categorical and 19 were numerical.

(2) Duplication. There was no duplication found in the dataset.

(3) Nulls. There was no null found in the dataset.

(4) Negative Values. Some of the columns contain negative numbers, which are supposed to represent null in the dataset. This might be due to the field not being a compulsory field when customers are creating their new accounts. After replacing all negative numbers with nulls in the dataset, there were a huge

number of nulls. Refer to **Table 2** for features with more than 1% of missing values.

Table 2: Missing Values in Percentage (%)

Fraud Bool	Not Fraud (0)	Fraud (1)
intended_balcon_amount	74.09	88.38
prev_address_months_count	71.06	91.89
bank_months_count	25.23	37.54

(a) Handling the Nulls. After exploring the dataset, it was observed that many of the null values likely resulted from non-mandatory fields during the bank account creation process. These missing entries represent genuine absences of information rather than data collection errors. Therefore, no rows or columns were dropped and no imputation was performed. The negative values were retained to accurately reflect real-world application scenarios where incomplete information is a meaningful indicator that the system may need to capture and analyse as part of the fraud detection process.

(5) Outliers. It was found that 14 of the features had outliers. Refer to **Annex C** for the corresponding boxplots illustrating these distributions

(a) Handling the Outliers. Although outliers were present, none were extreme enough to significantly distort the overall data distribution. Given the objective for fraud detection, these outliers were retained as they might represent unusual but meaningful indicators of potentially fraudulent activity.

(6) Correlation. A correlation analysis was performed to examine the strength of association between each feature and the fraud label. This analysis helped identify which variables showed stronger linear relationship with the fraudulent outcomes and provide an initial indication of which attributes might contribute meaningfully to the model performance. Refer to **Annex D** for the correlation visualisations.

(7) Features Removal. One feature was removed prior to the model training.

(a) Device Fraud Count. This feature was excluded from the model training because it contained only a single constant value (0) across all records, providing no variability or informational value for the model. Retaining such a feature would not contribute to the learning and would introduce unnecessary noise in the analysis.

6. Methodology and Model Design. The project methodology utilises the ScamWatch dataset for victim profiling and the Kaggle dataset for NAF detection. Both datasets were analysed using a data-driven approach focusing on unsupervised and supervised learning methods to uncover the underlying patterns.

a. Unsupervised Learning (Clustering). The purpose of the clustering stage is to segment ScamWatch victims into meaningful behavioural groups based on their categorical attributes. These segments help reveal underlying patterns that may be associated with increased vulnerability to specific scam types, forming the basis for more targeted prevention and fraud-risk profiling efforts.

(1) Data Preparation. After the data preprocessing, which includes removing irrelevant features and dropping non-substantial entries, one-hot encoding is applied to the categorical variables. It was observed that many victims shared identical attribute combinations. To avoid overweighting duplicated patterns, a deduplication with a frequency-weighting approach was applied.

(2) Initial Clustering Model. Several clustering algorithms were initially tested directly on the high-dimensional one-hot encoded data. The models that were used and rated with a silhouette score[3] were:

(a) K-Means. K-Means was evaluated across multiple values of k and the best silhouette score obtained was

[3] Silhouette Score of 1 indicates a clearly defined, dense and well-separated cluster, while a score close to 0 suggests an overlapping cluster with little separation.

0.152 with K = 24, where K refers to the number of clusters identified.

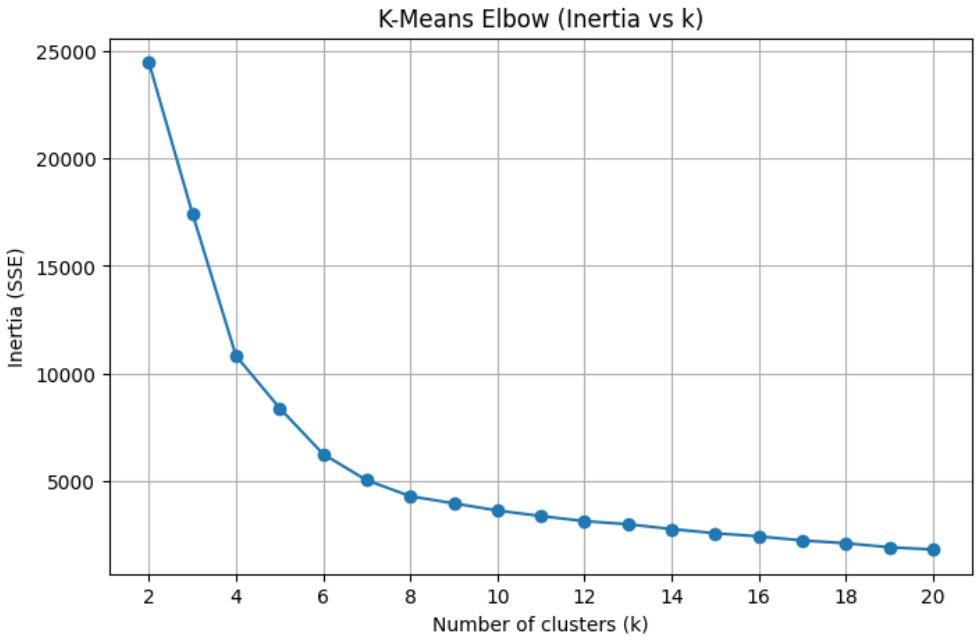
(b) HDBSCAN. HDBSCAN was evaluated based on multiple parameters. The model consistently produced a silhouette score of less than 0.1 with only two clusters. These clusters were too broad and did not offer meaningful segmentation for profiling victims.

(c) Hierarchy. Using average linkage and Hamming/ Manhattan distance, the best silhouette score obtained was 0.188 with K = 2. Like HDBSCAN, the model produced only two generalised clusters and lacked interpretability.

(3) Dimensional Reduction using Uniform Manifold Approximation and Projection (UMAP). To address this, the UMAP algorithm was applied to reduce the high-dimensional binary features matrix into a compact numerical representation.

(4) K-Mean on UMAP Embeddings. After dimensional reduction with UMAP, the K-Means model was applied. The silhouette score obtained was 0.542 with K = 7. The elbow method was used to identify a suitable value of K (See **Figure 1**).

Figure 1: K-Means Elbow graph



(5) Cluster Profiling. A total of seven victim clusters were identified from the ScamWatch dataset. See **Annex E** for the profiling details.

(6) Insights. During the exploratory stage, some general victim profile patterns were observed. See **Annex F** for the corresponding visualisations.

(a) Age. The age group with the highest proportion of reported scam victims was 35 to 44 years old.

(b) Gender. Male victims were reported more frequently than female victims.

(c) Scam Method. Online platforms were the most common channel through which scams were executed.

(d) Amount Lost. The most frequently reported loss amount was below $1,000.

(e) Scam Category. Investment scams were the most prevalent category among other reported cases.

b. Supervised Learning (Classification). The purpose of classification is to predict the binary outcome based on the historical data in the dataset. This will help to make a prediction based on the attributes keyed in when the customer is trying to create an account.

(1) Data Preparation. After the data preprocessing, which includes removing irrelevant features and applying one-hot encoding to all categorical variables. The dataset was then split into a training and a testing set. The dataset was found to be highly imbalanced, with fraudulent cases representing a small portion of records. To avoid data leakage, the split was performed based on the month feature.

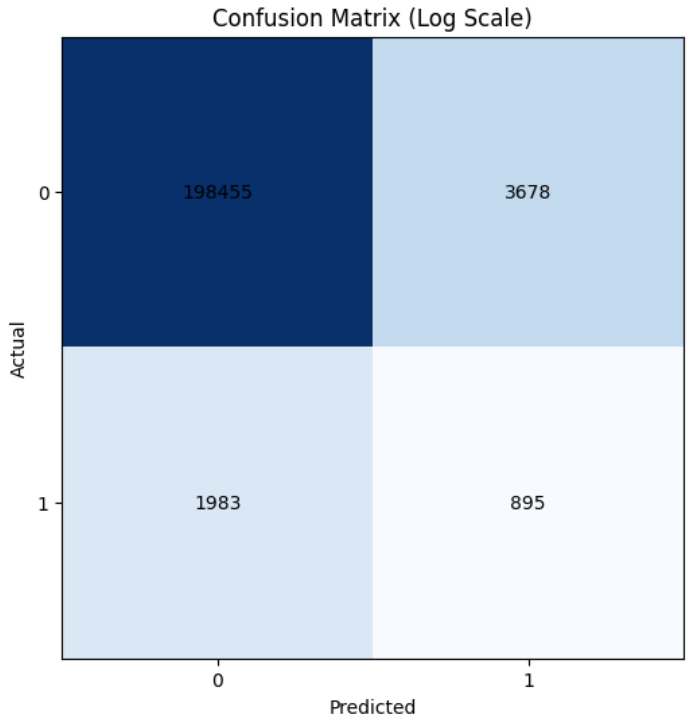
(a) Training Set. The dataset was split into the months of 0 to 5.

(b) Testing Set. The dataset was split into the months of 6 and 7.

(2) Baseline Model. Logistic Regression model selected as the baseline classifier. Because of severe class imbalance, a random under-sampler was applied to reduce the number of non-fraud records in the training set. Under-sampling was chosen over oversampling to prevent overfitting of the model.

(a) Logistic Regression. The Logistic Regression model performed unexpectedly well and achieved the best overall evaluation metrics among all models tested, making it the final chosen classifier. A confusion matrix of the model is shown in **Figure 2**.

Figure 2: Confusion Matrix for Logistic Regression



(b) Precision. The model achieved a precision of 0.196, meaning that many flagged cases were false positives. This is expected in fraud detection, where it is typically more important to catch fraud than to minimise false alarms.

(c) Recall. The model achieved a recall of 0.311, which means that the model successfully identified about 1 out of every 3 real fraud cases.

(d) F1 Score. The model achieved an F1-score of 0.24. Given the severe dataset imbalance, such a value is reasonable and expected.

(e) Accuracy. The model achieved an accuracy of 0.972. However, due to the dataset imbalance, accuracy is not a meaningful metric for evaluating fraud-detection performance.

(3) XGBoost. XGBoost was selected as the second supervised model due to its strong performance in handling large datasets, ability to capture non-linear relationships and effectiveness in handling imbalanced classes.

(a) Scale Pos Weight (SPW). The first XGBoost model was trained on the original imbalanced dataset using the SPW parameter to compensate for the low proportion of fraud cases. This approach increases the penalty for misclassifying fraudulent records and improves the model's sensitivity. However, despite tuning of the model, it did not exceed the performance of the Logistic Regression baseline.

(b) Under-Sampling. A second modelling pipeline was trained by applying random under-sampling to balance the training data before the modelling. After trying various tuning of the hyperparameters, this model did improve the model's recall and F1-score relative to the SPW model, but did not outperform the Logistic Regression baseline.

(4) Random Forest. A Random Forest classifier was also evaluated using SMOTE to address class imbalance. While recall improved relative to an untuned baseline, precision dropped and overall F1-score remained below that of Logistic Regression. Given the higher complexity and lower net benefit, Random Forest was not selected for deployment.

(5) Comparing Results. All models were evaluated on the same train-test split. The results of all models are summarised and shown in **Table 3**. Although Logistic Regression produced the lowest recall value, it achieved the highest precision and consequently the highest F1-score, indicating the best overall balance between detecting fraud and reducing false alarms.

Table 3: Comparing Results of Models

Model	Imbalance Strategy	Precision	Recall	F1-Score
Logistic Regression	Undersampling	0.20	0.31	0.24
XGBoost	SPW	0.07	0.77	0.12
XGBoost	Undersampling	0.15	0.37	0.21
Random Forest	SMOTE	0.10	0.39	0.16

7. Deployment Recommendation. To fully operationalise the findings of this project, a two-tier hybrid fraud detection pipeline is recommended.

a. Stage One. The first stage involves deploying the Logistic Regression model as the primary front-line classifier during the new bank account opening process. Although the model reflects a trade-off between recall and precision, it offers a strong overall balance and is computationally lightweight, making it highly suitable for real-time, high-volume transactional systems. When an application is flagged as potentially fraudulent, financial institutions may add additional automated identification verification steps or even escalate for manual review before allowing the account to proceed.

b. Stage Two. The second stage incorporates behavioural intelligence derived from the unsupervised clustering analysis of the ScamWatch dataset. These clusters can function as an external risk multiplier. Applications whose attributes closely resemble one of the higher-risk victim profiles will be flagged out and financial institutions can then prioritise the high-risk cases for enhanced verification, ensuring additional scrutiny is applied.

8. Conclusion. This project developed a hybrid fraud detection framework combining behavioural clustering and supervised predictive modelling to identify NAF during the onboarding process. Through the ScamWatch

dataset, seven behavioural victim clusters were identified, offering an initial foundation for understanding groups that may be prone to scams.

While the clustering results provide meaningful insights, they may be somewhat ambiguous due to the limitations in the data, such as incomplete victim information, self-reported records and geographic bias. Nonetheless, they serve as a useful starting point for behavioural risk profiling. These clusters should not be treated as static but should be continuously refined and retrained as more comprehensive data becomes available, ensuring that the behavioural profiles stay up to date with the evolving fraud patterns.

For supervised learning, several models were tested under different imbalance-handling strategies. Logistic Regression ultimately demonstrated the best balance of recall and precision, achieving the highest F1-score. Its simplicity and interpretability also support real-time operations.

Overall, this hybrid approach provides a scalable and proactive foundation for early fraud detection. With continuous data updates and model retraining, both clustering and predictive models will strengthen over time, enabling financial institutions to stay ahead of emerging fraud tactics.

Annexes:

A. Data Dictionary for Scam Watch
B. Data Dictionary for Kaggle Bank Account Fraud Dataset
C. Boxplot for Outliers
D. Correlation of Features to Fraud
E. Victim Cluster Profiling
F. General Victim Profiling

Annex A: Data Dictionary for ScamWatch

1. The data Dictionary for ScamWatch are show in **Table A-1** below.

Table A-1: Data Dictionary for ScamWatch

Features	Remarks
StartOfMonth	Indicates the month and year of the reported incident. All records are standardised to the first day of the month, regardless of the actual reported date
Address_State	The Australian state where the complainant resides
Scam___Contact_Mode	The primary method through which the scammer contacted the victims
Complainant_Age	Age group of the individual who reported the scam
Complainant_Gender	Gender of the individual who reported the scam
Category_Level_2	High-level classification of the scam type
Category_Level_3	More specific subtype of the scam under the Level 2 category providing additional details
Amount_lost	The total financial loss reported by the victim for the incident
Number_of_reports	The number of individual reports associated with the same scam incident. Higher values may indicate repeated attempts or multiple victims reporting the same event

Annex B: Data Dictionary for Kaggle Bank Account Fraud Dataset

1. The data Dictionary for Kaggle Bank Account Fraud are show in **Table B-1** below.

Table B-1: Data Dictionary for Kaggle Bank Account Fraud

Features	Remarks
income	Annual income of the applicant (in decile form). Ranges between [0.1, 0.9]
name_email_similarity	Metric of similarity between email and applicant's name. Higher values represent higher similarity. Ranges between [0, 1]
prev_address_months_count	Number of months in previous registered address of the applicant, i.e. the applicant's previous residence, if applicable. Ranges between [-1, 380] months (-1 is a missing value)
current_address_months_count	Months in currently registered address of the applicant. Ranges between [-1, 429] months (-1 is a missing value)
customer_age	Applicant's age in years, rounded to the decade. Ranges between [10, 90] years
days_since_request	Number of days passed since application was done. Ranges between [0, 79] days
intended_balcon_amount	Initial transferred amount for application. Ranges between [-16, 114] (negatives are missing values)
payment_type	Credit payment plan type. 5 possible (annonymised) values
zip_count_4w	Number of applications within same zip code in last 4 weeks. Ranges between [1, 6830]
veloctity_6h	Velocity of total applications made in last 6 hours *i.e.*, average number of applications per hour in the last 6 hours. Ranges between [-175, 16818]
velocity_24h	Velocity of total applications made in last 24 hours *i.e.*, average number of applications per hour in the last 6 hours. Ranges between [1297, 9586]
velocity_4w	Velocity of total applications made in last 4 weeks *i.e.*, average number of applications per hour in the last 6 hours. Ranges between [2825, 7020]
bank_branch_count_8w	Number of total applications in the selected bank branch in last 8 weeks. Ranges between [0, 39]

Features	Remarks
date_of_birth_distinct_emails_4w	Number of emails for applicants with same date of birth in the last 4 weeks. Ranges between [0, 39]
employment_status	Employment status of the applicant. 7 possible (annonymised) valuees
credit_risk_score	Internal score of application risk. Ranges between [-191, 389]
email_is_free	Domain of application email (either free or paid)
housing_status	Current residential status for applicant. 7 possible (annoymised) values.
phone_home_valid	Validity of provided home phone
phone_mobile_valid	Validity of provided mobile phone
bank_months_count	How old is previous account (if held) in months. Ranges between [-1, 32] months (-1 is a missing value)
has_other_cards	If applicant has other cards from the same banking company
proposed_credit_limit	Applicant's proposed credit limit. Ranges between [200, 2000]
foreign_request	If origin country of request is different from bank's country
Source	Online source of application. Either browser (INTERNET) or app (TELEAPP)
session_length_in_minutes	Length of user session in banking website in minutes. Ranges between [-1, 107] minutes (-1 is a missing value)
device_os	Operative system of device that made request. Possible values are: Windows, macOS, Linux, X11, or other
keep_alive_session	User option on session logout
device_distinct_emails	Number of distinct emails in banking website from the used device in last 8 weeks. Ranges between [-1, 2] emails (-1 is a missing value)
device_fraud_count	Number of fraudulent applications with used device. Ranges between [0, 1]
month	Month where the application was made. Ranges between [0, 7]
fraud_bool	If the application is fraudulent or not

Annex C: Boxplot for Outliers

1. The Boxplot for the all features is shown in **Figure C-1 to C-17**.

Figure C-1: income boxplot

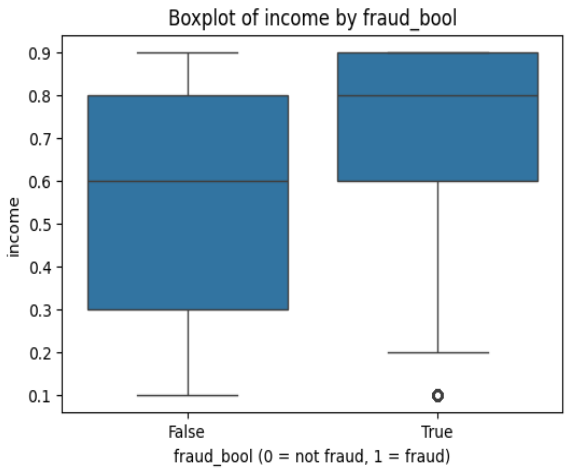


Figure C-2: name email similarity boxplot

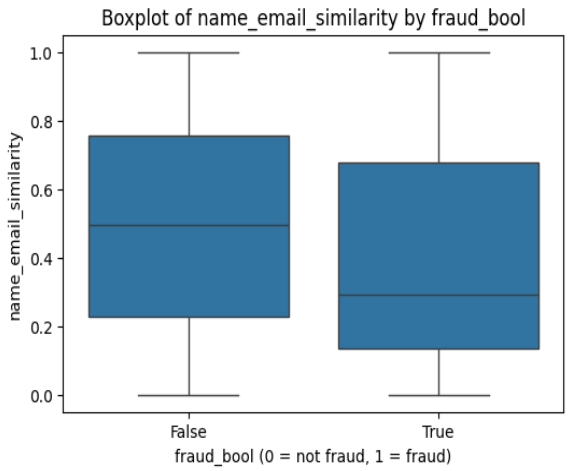


Figure C-3: prev add month count boxplot

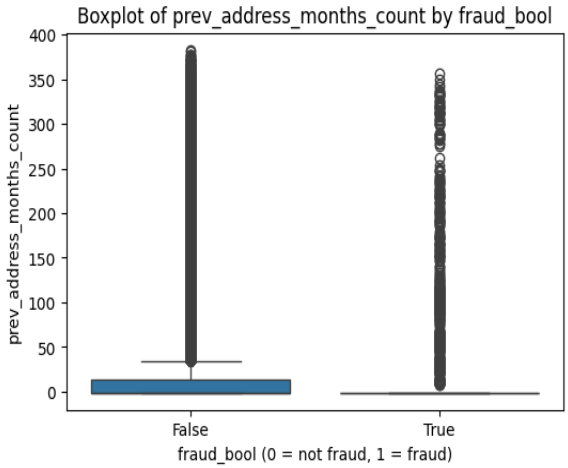


Figure C-4: current add month count boxplot

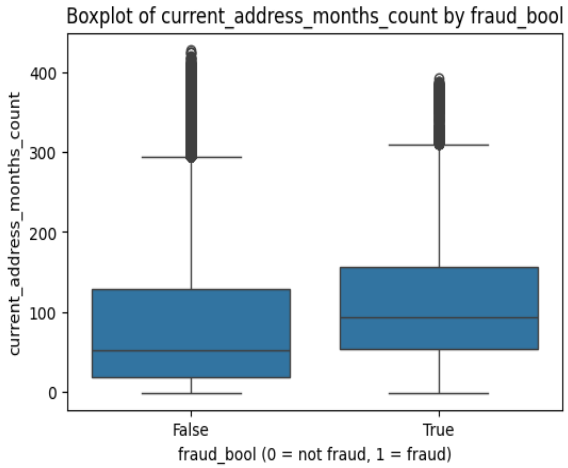


Figure C-5: customer age boxplot

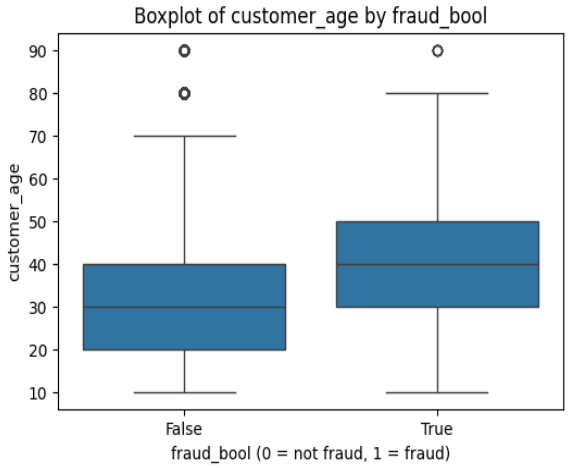


Figure C-6: name email similarity boxplot

Figure C-7: intended balcon amount boxplot

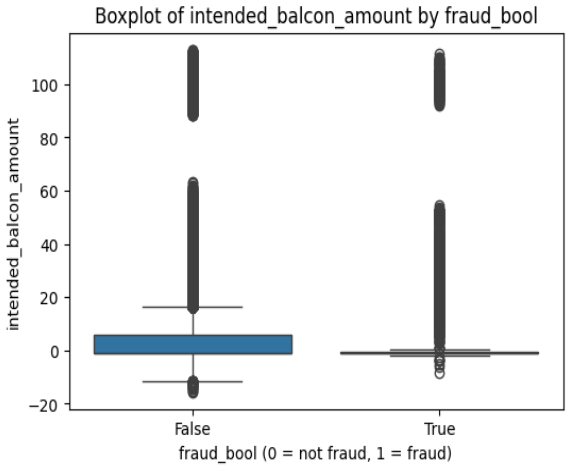


Figure C-8: zip count 4w boxplot

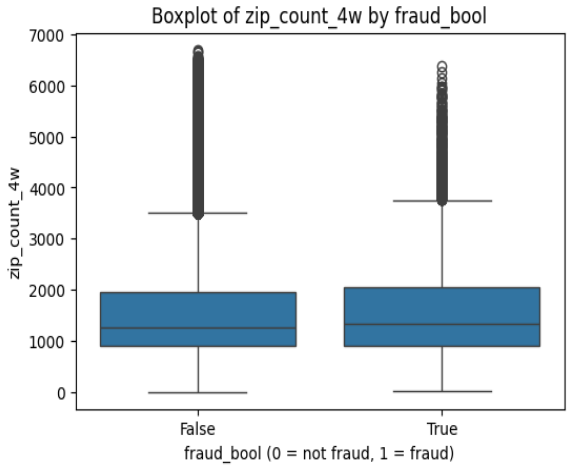


Figure C-9: velocity 6h boxplot

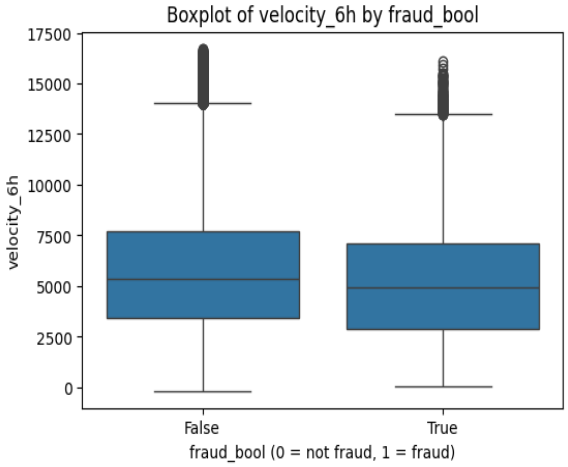


Figure C-10: velocity 24h boxplot

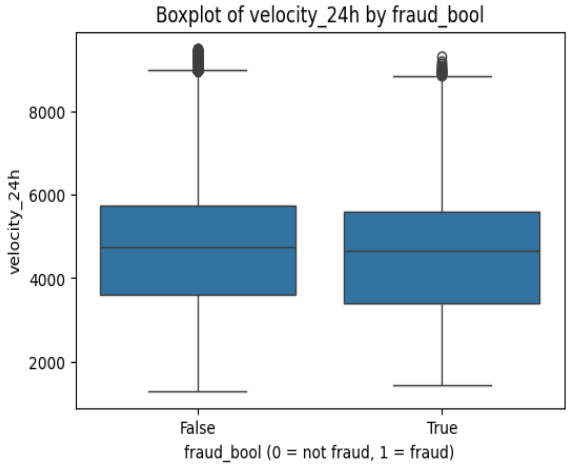


Figure C-11: velocity 4w boxplot

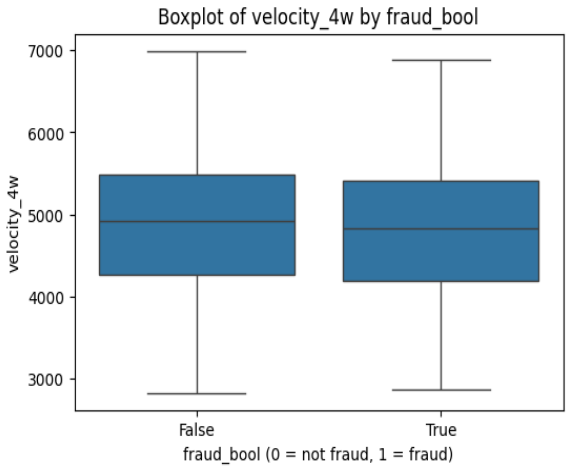


Figure C-12: bank branch count 8w boxplot

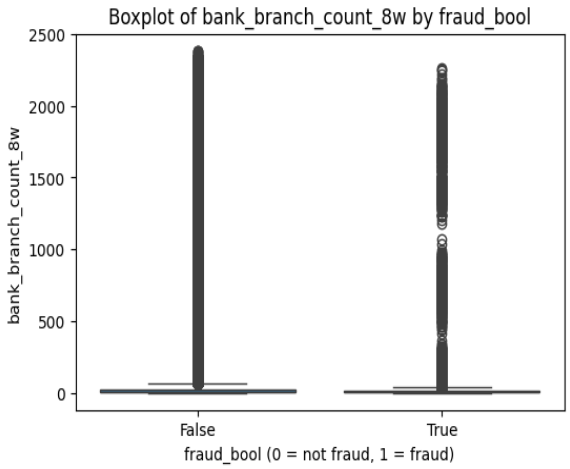


Figure C-13: D.O.B distinct emails 4w boxplot

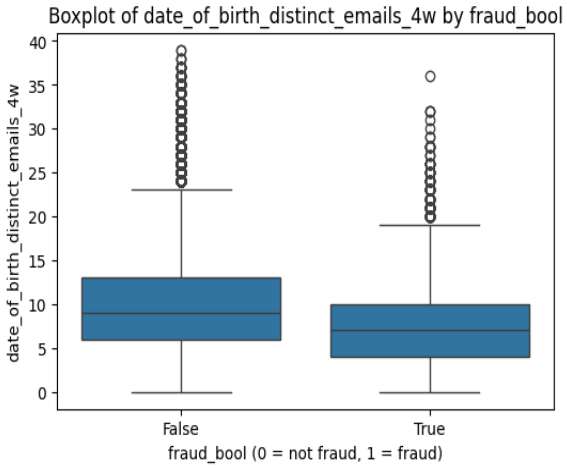


Figure C-14: credit risk score boxplot

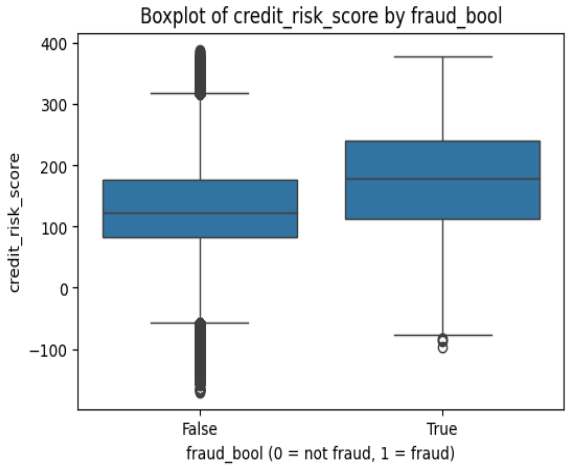


Figure C-15: bank months count boxplot

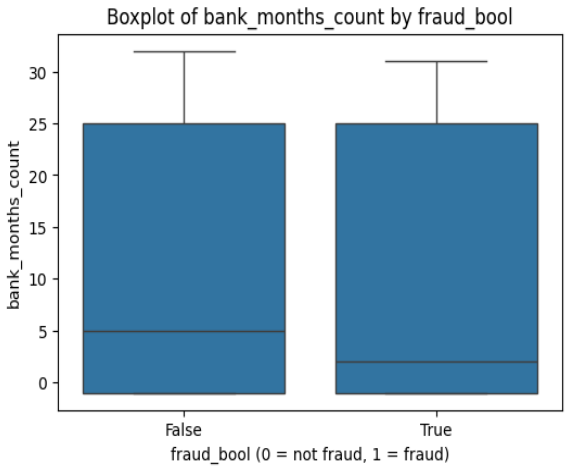


Figure C-16: proposed credit limit boxplot

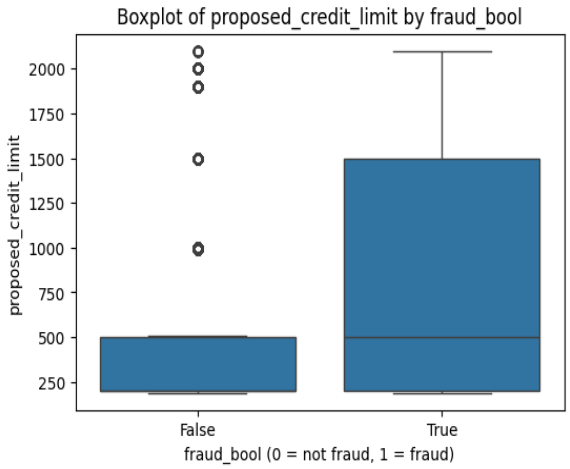
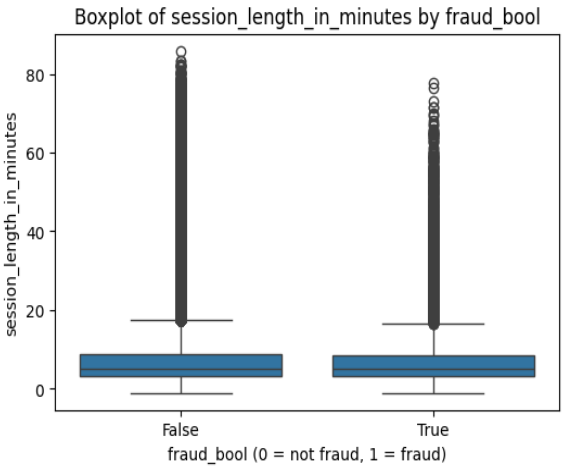


Figure C-17: session length in mins boxplos



Annex D: Correlation of Features to Fraud

1. The correlation of features to fraud for the all features is shown in **Figure D-1 to D-23**.

Figure D-1: Income Correlation

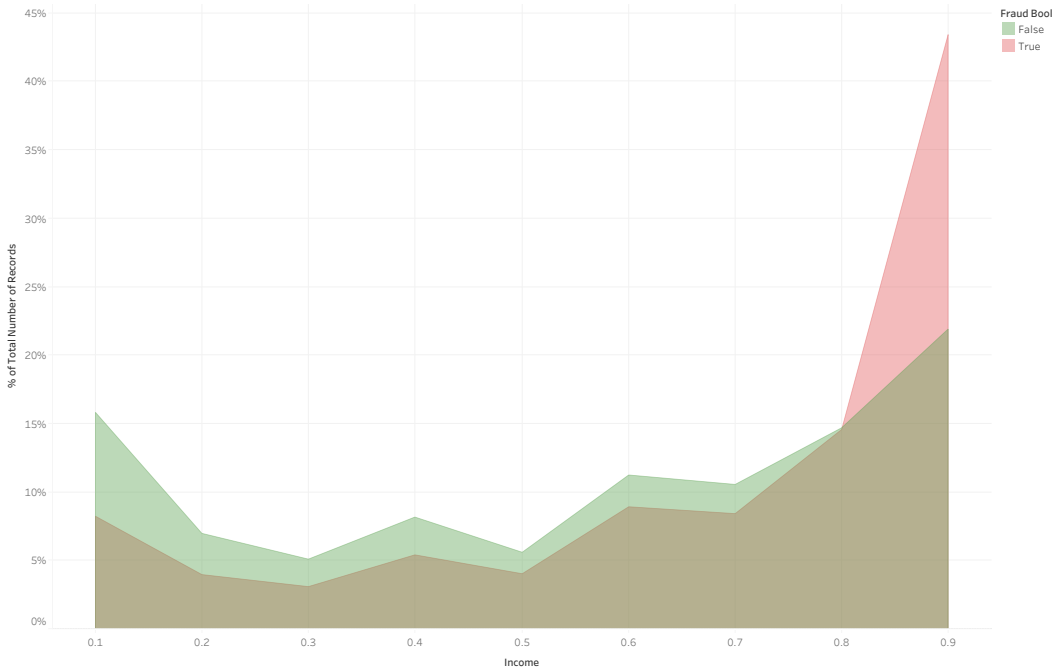


Figure D-2: Email Similarity Correlation

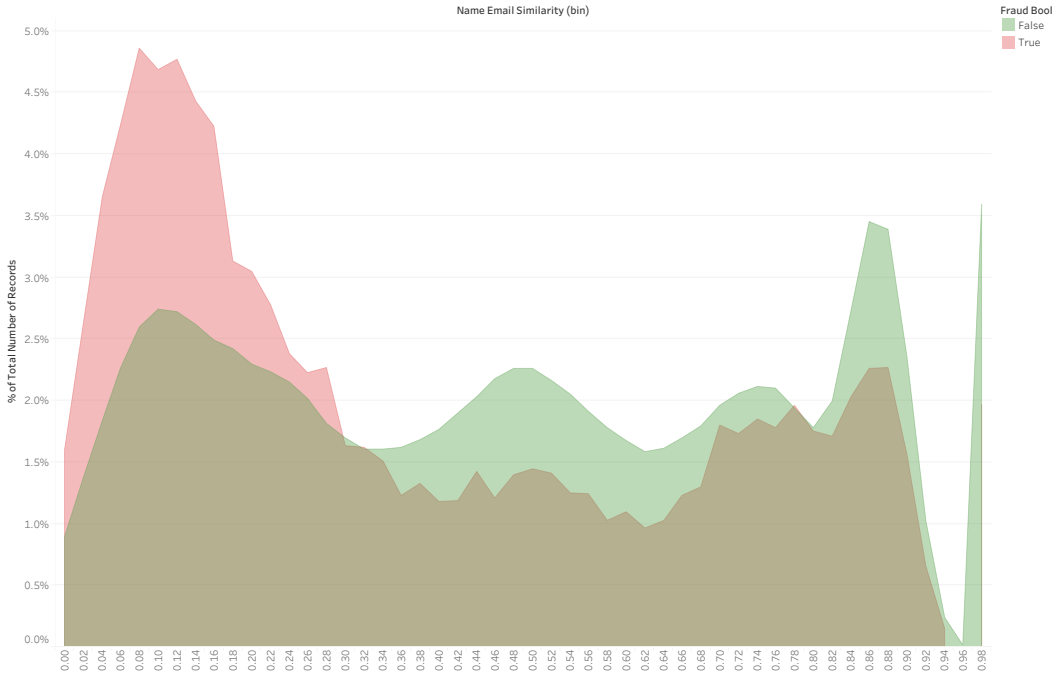


Figure D-3: Prev Address Mount Count Correlation

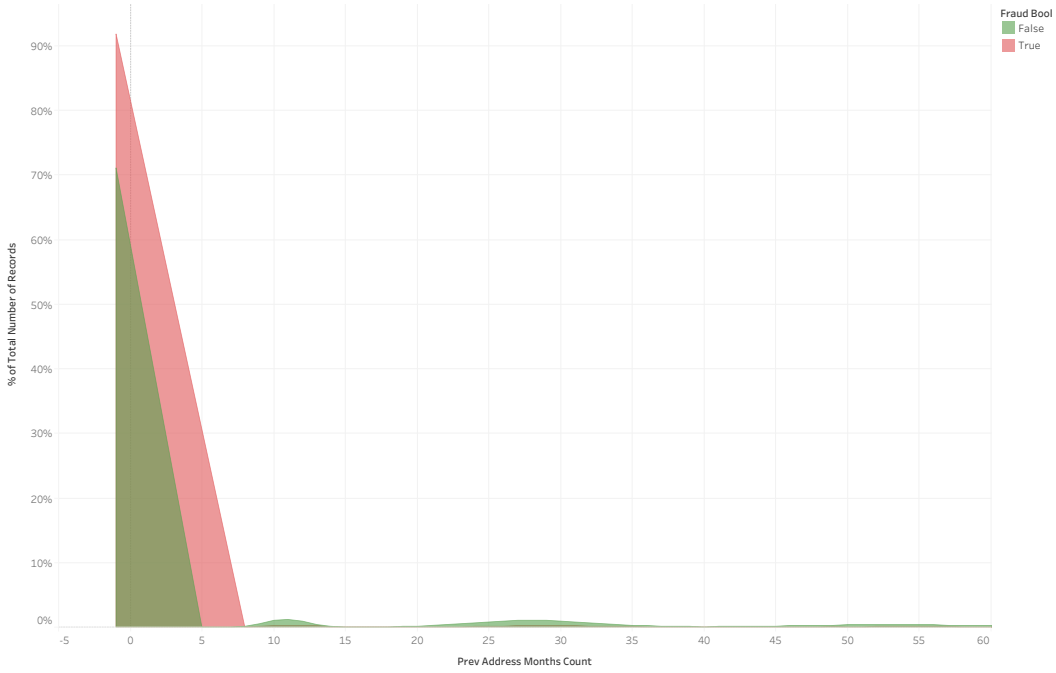


Figure D-4: Current Address Month Count Correlation

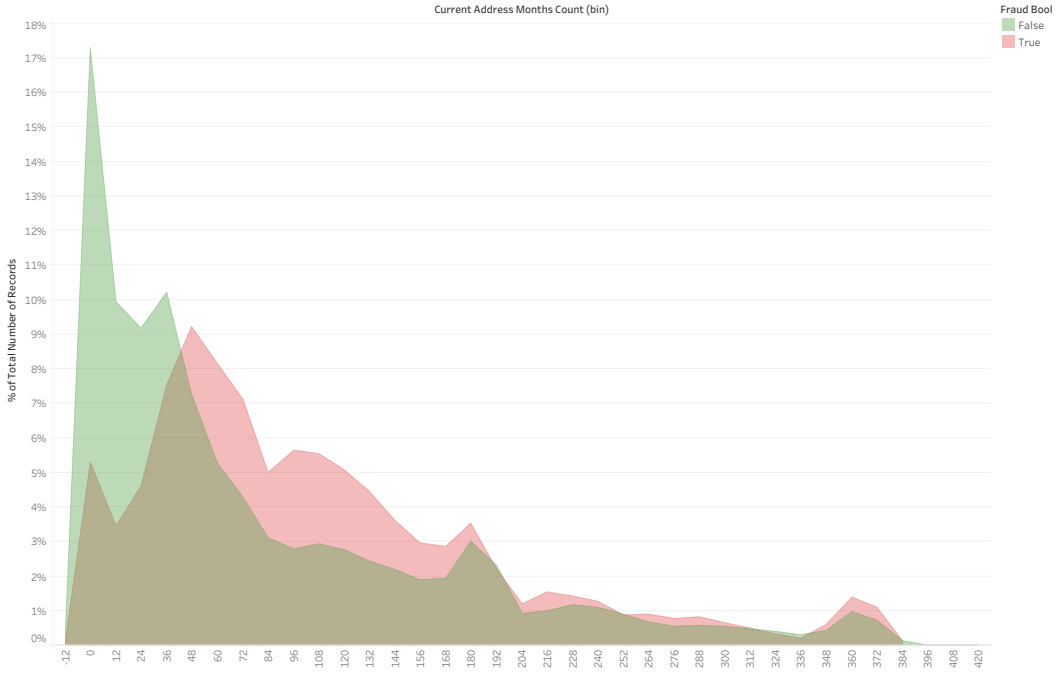


Figure D-5: Customer Age Correlation

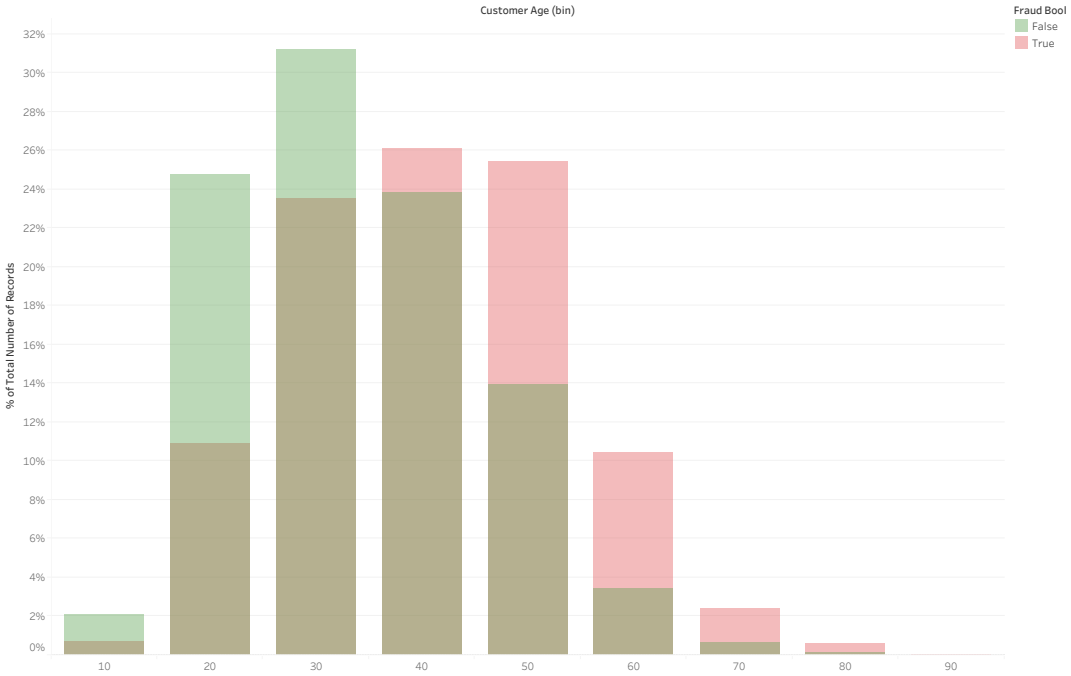


Figure D-6: Days Since Request Correlation

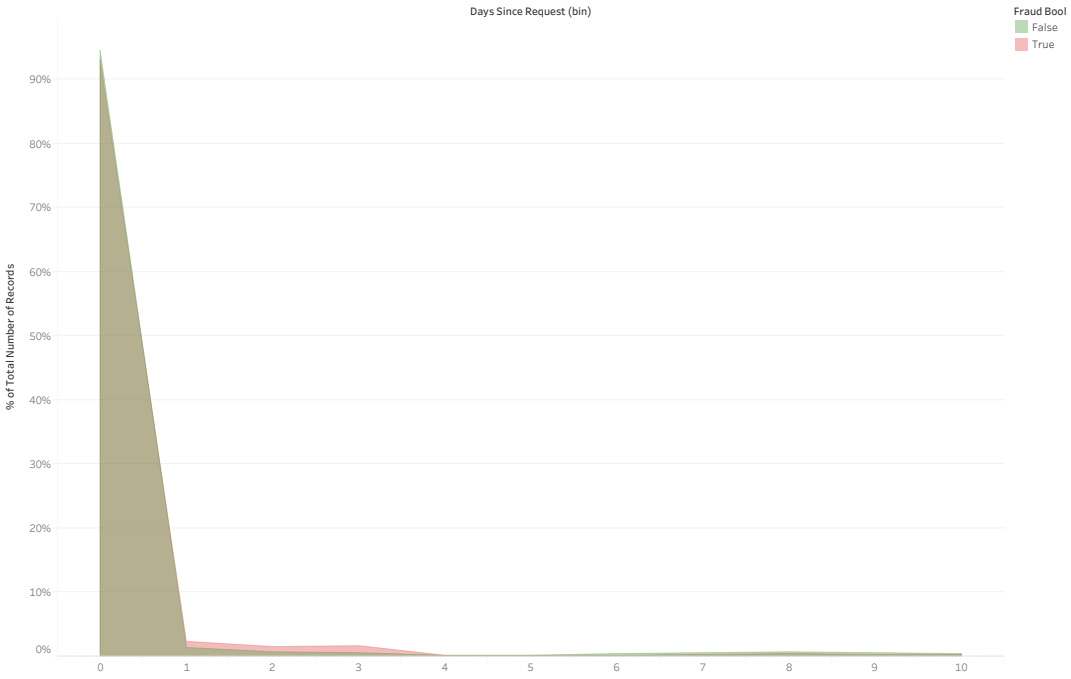


Figure D-7: Payment Type Correlation

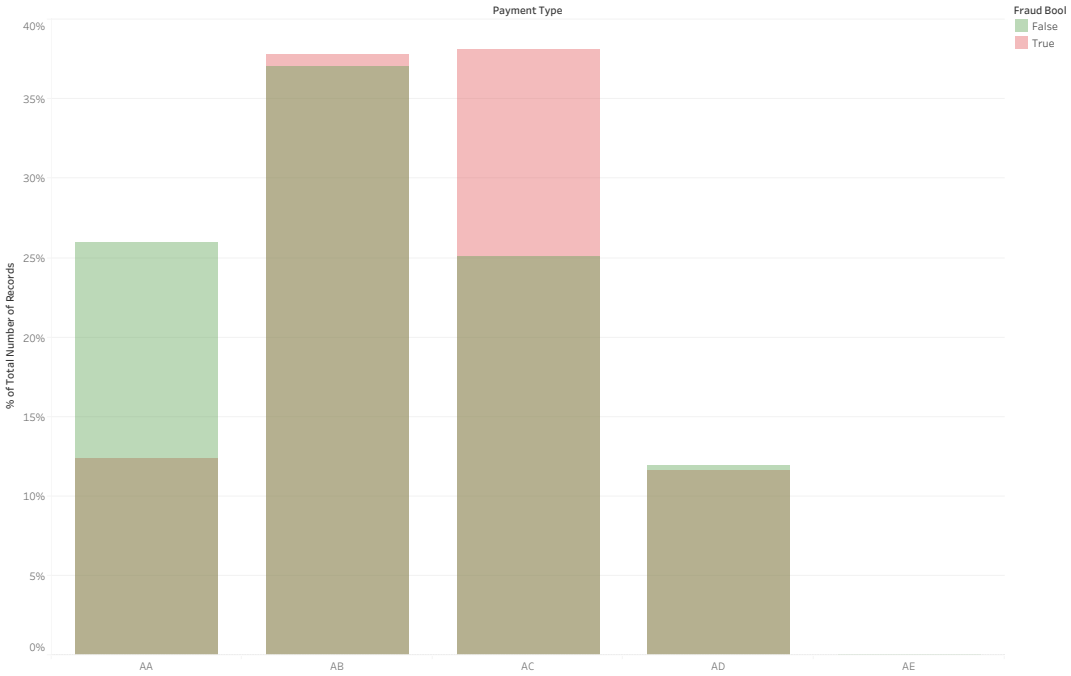


Figure D-8: Same Zip in 4 Weeks Correlation

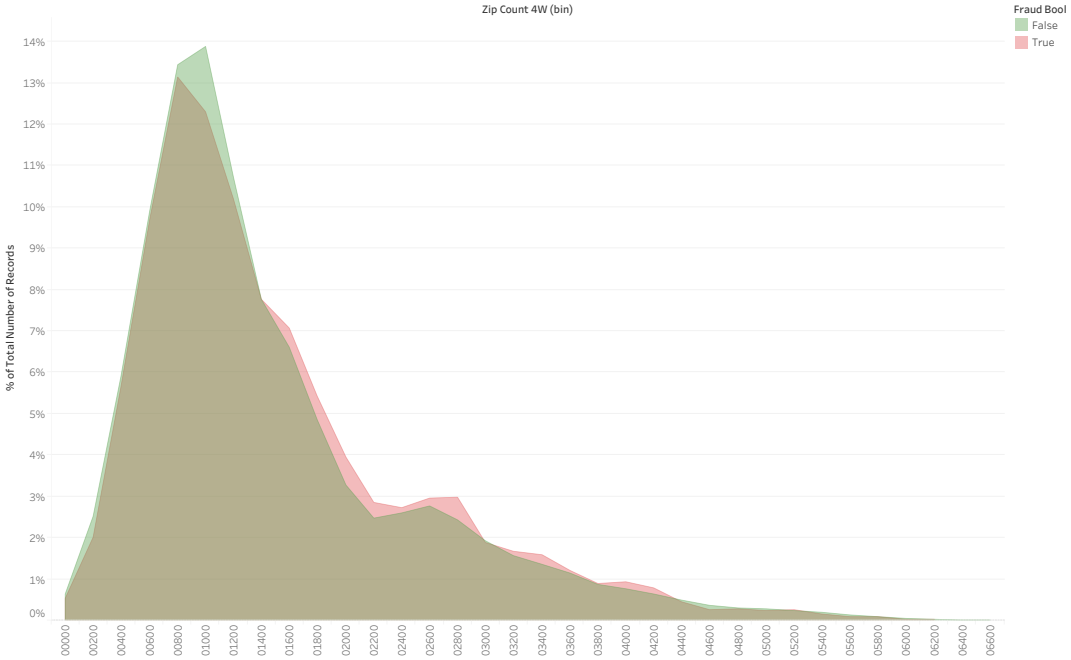


Figure D-9: Bank Months Count Correlation

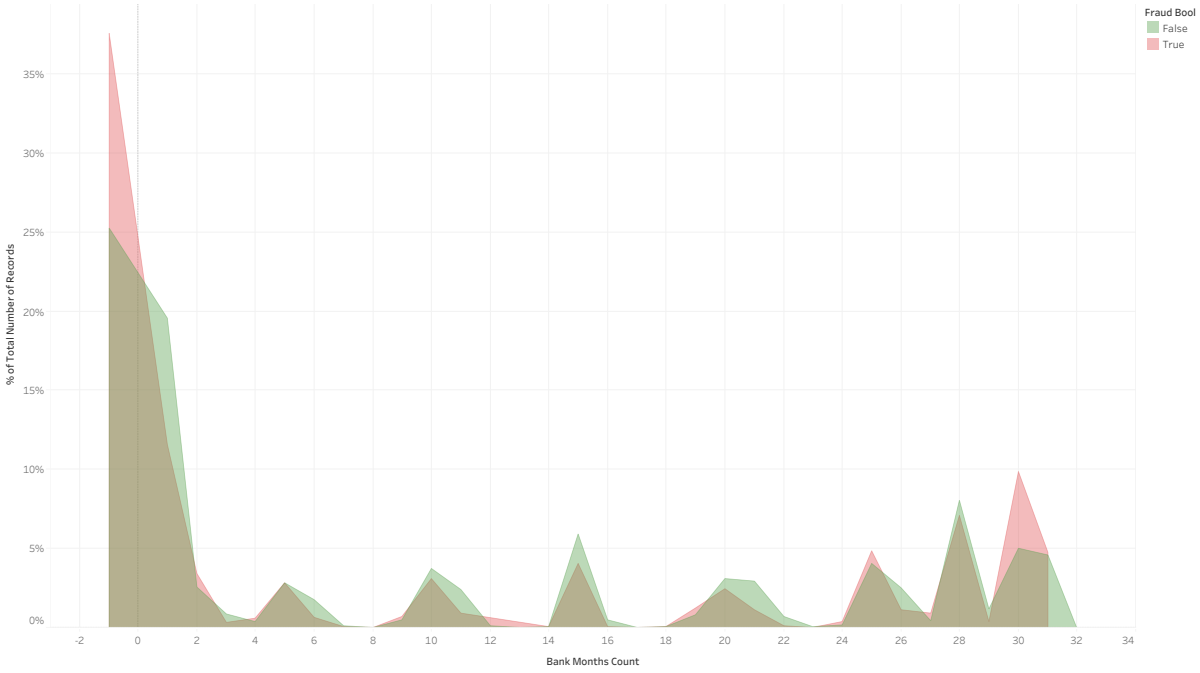


Figure D-10: D.O.B Distinct Emails 4W Correlation

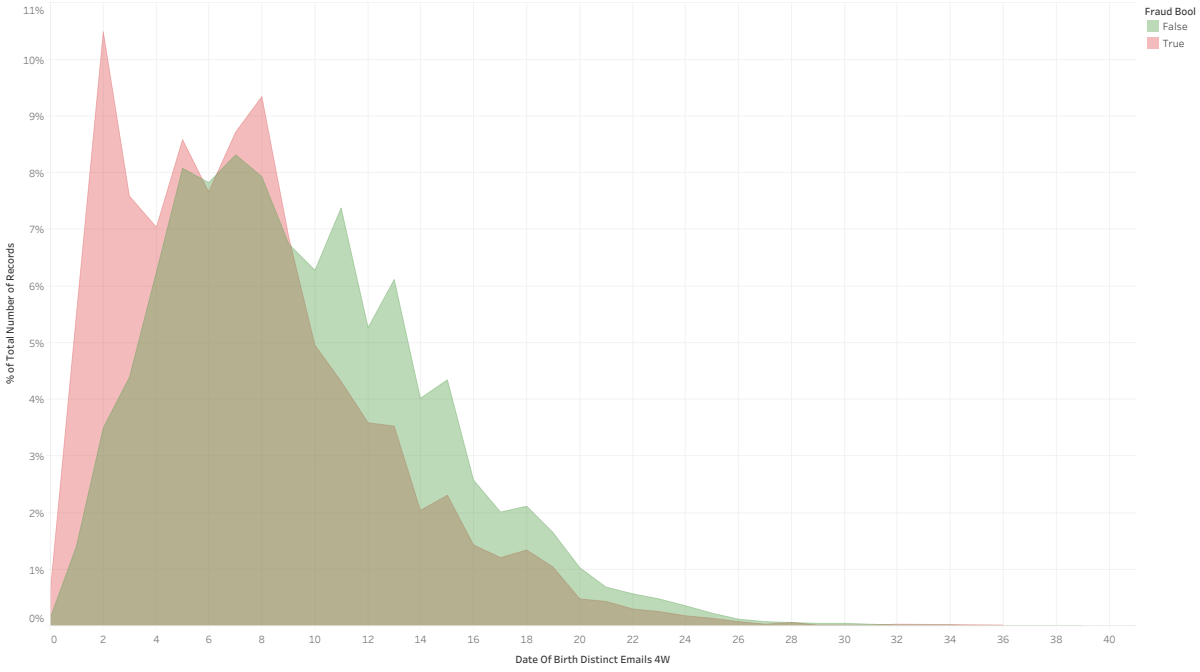


Figure D-11: Employment Status Correlation

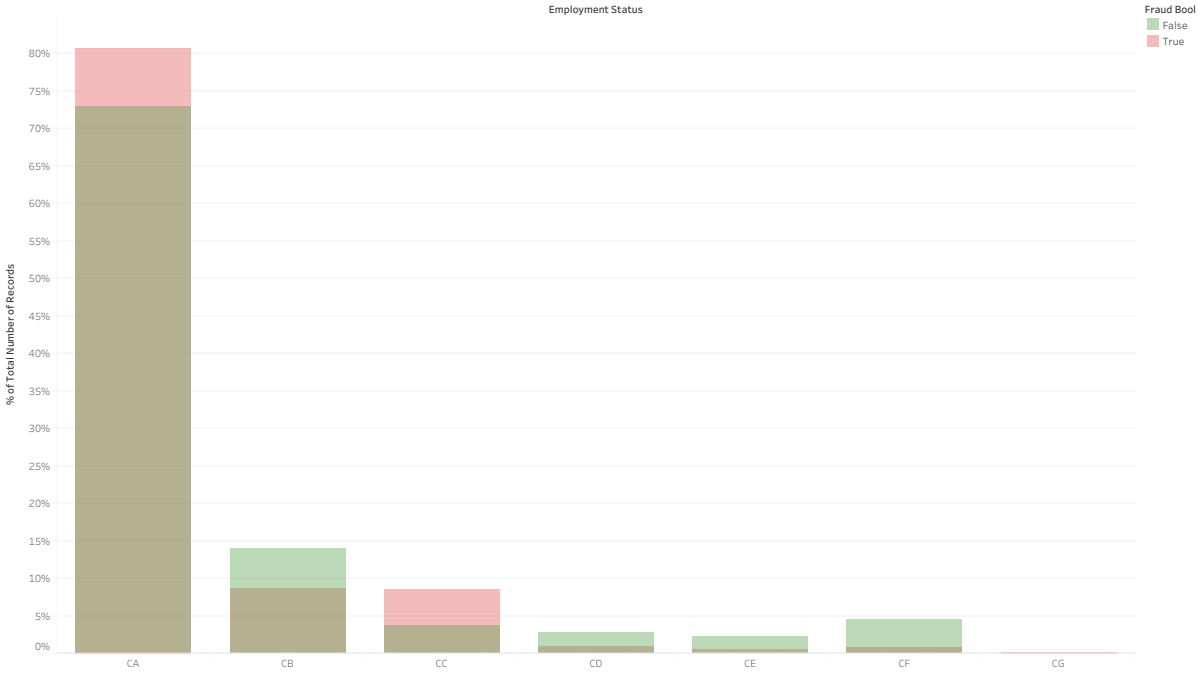


Figure D-12: Credit Risk Score Correlation

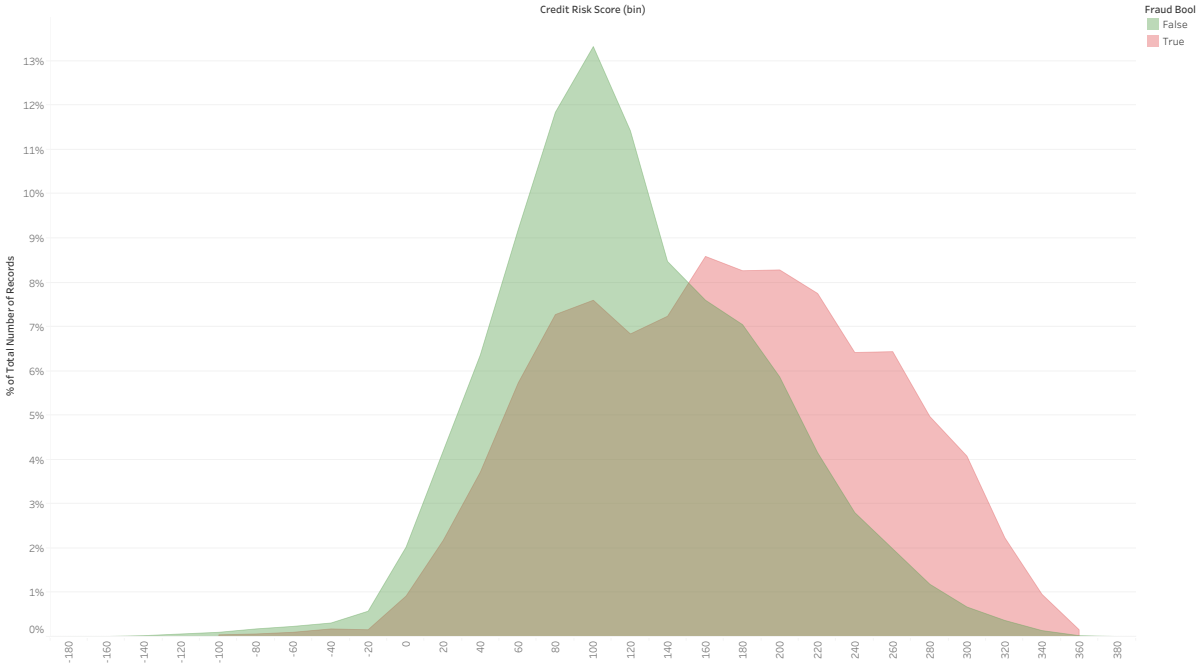


Figure D-13: Housing Status Correlation

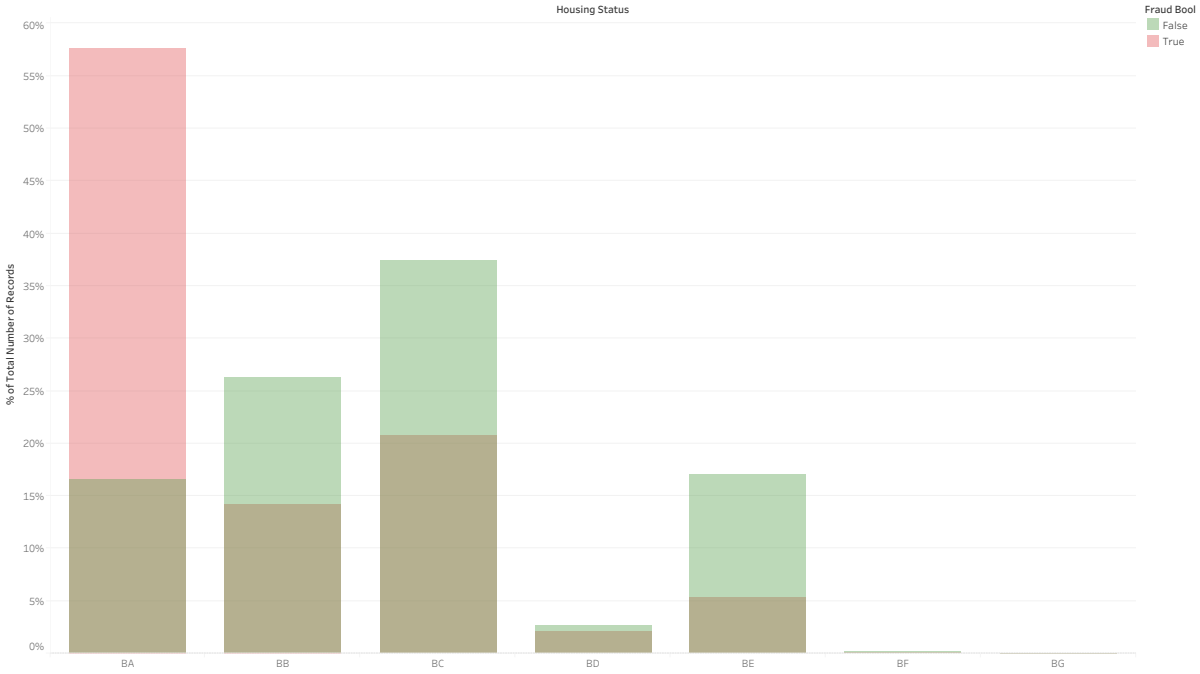


Figure D-14: Credit Limit Correlation

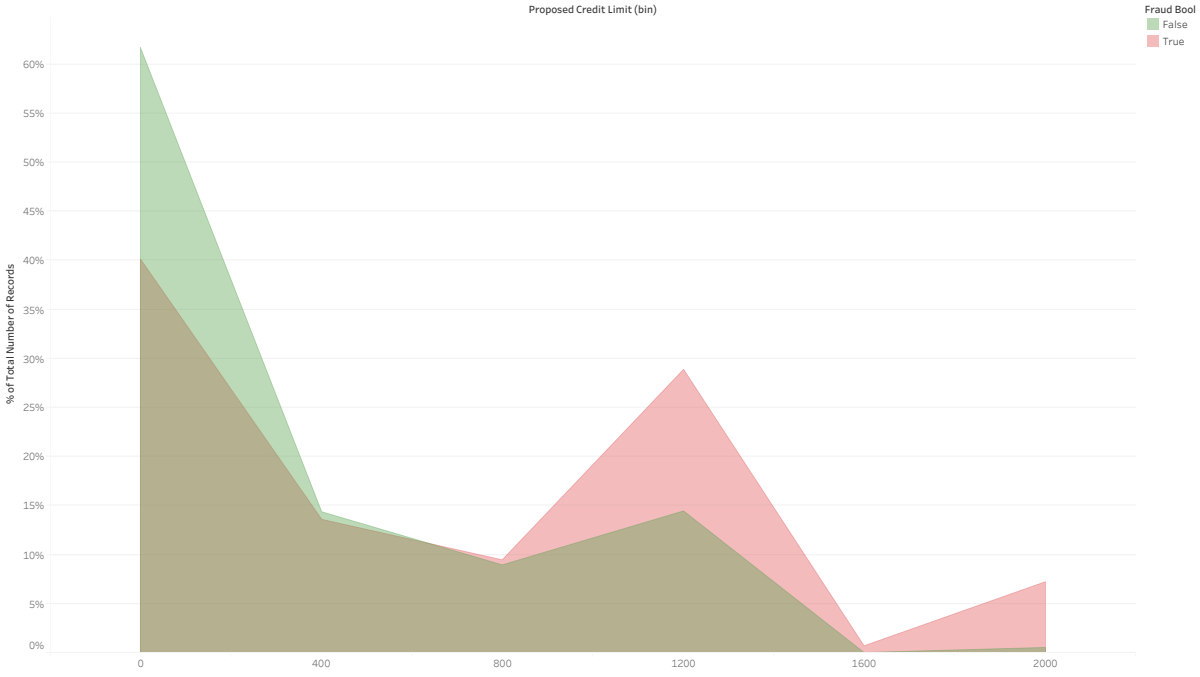


Figure D-15: OS Correlation

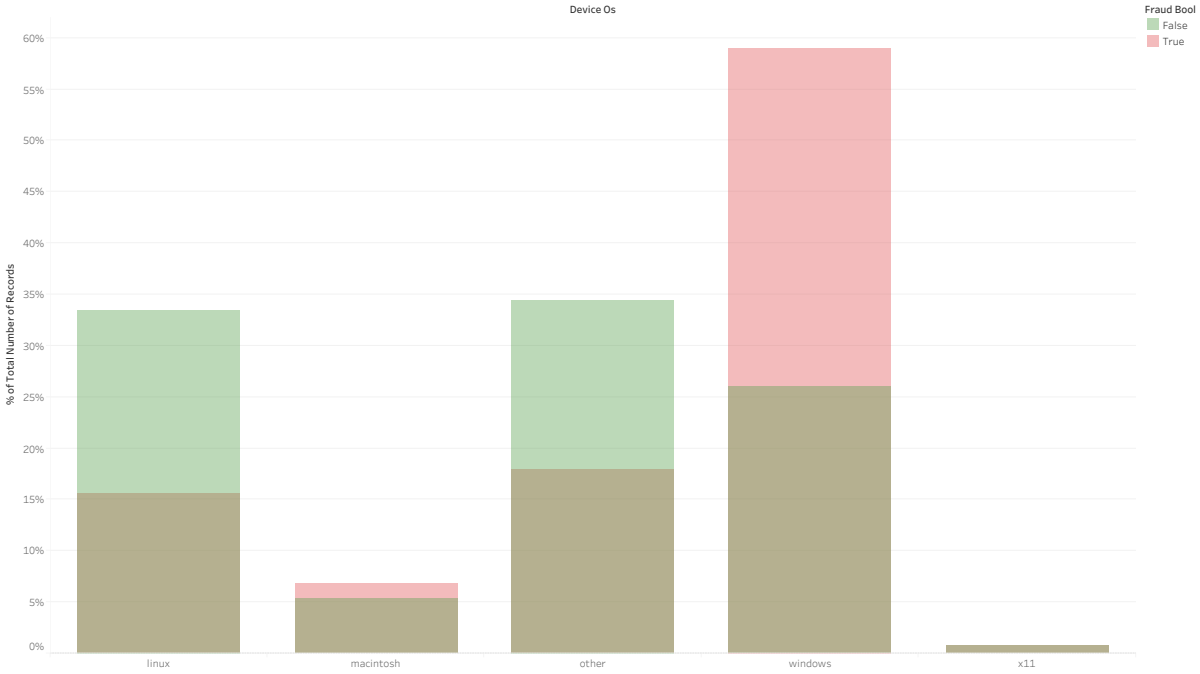


Figure D-16: Distinct Email Correlation

Figure D-17: Free Email Correlation

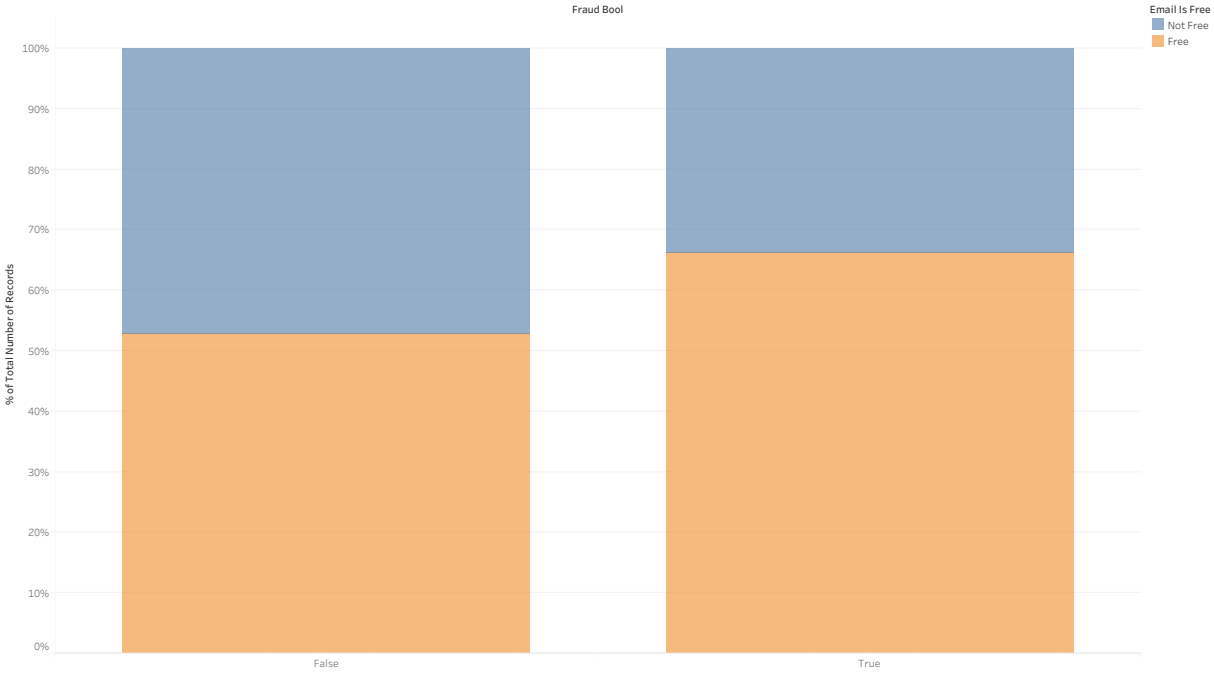


Figure D-18: Phone Home Valid Correlation

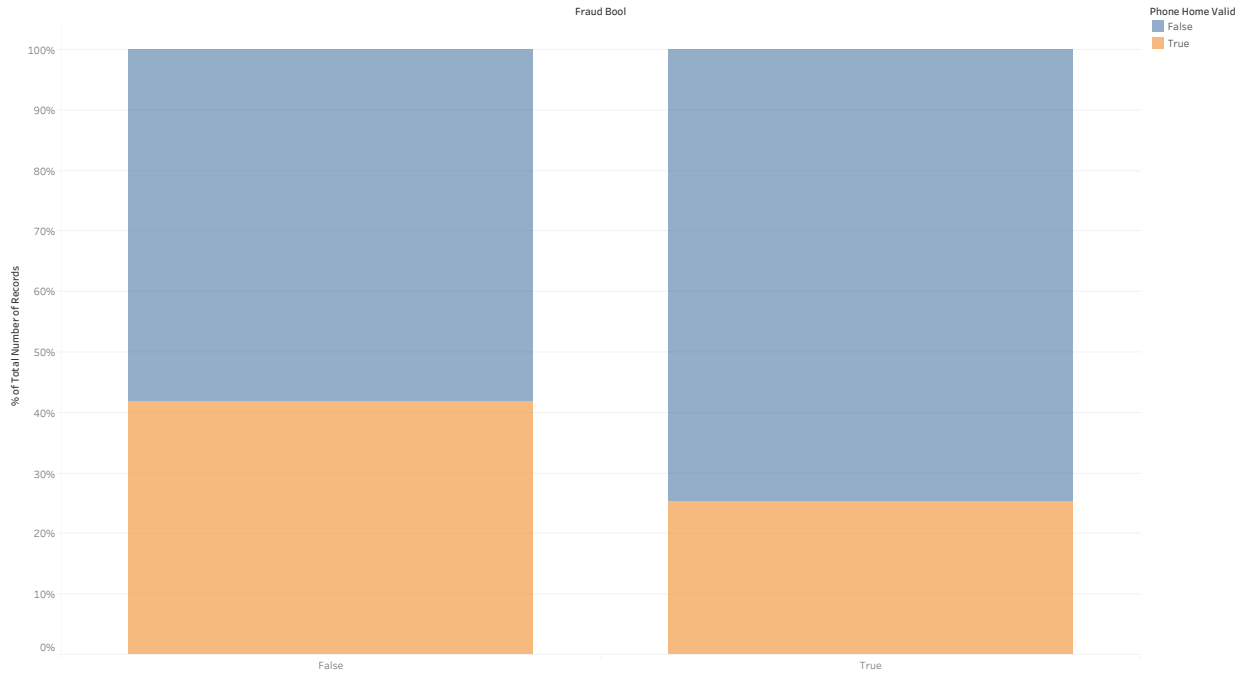


Figure D-19: Phone Mobile Valid Correlation

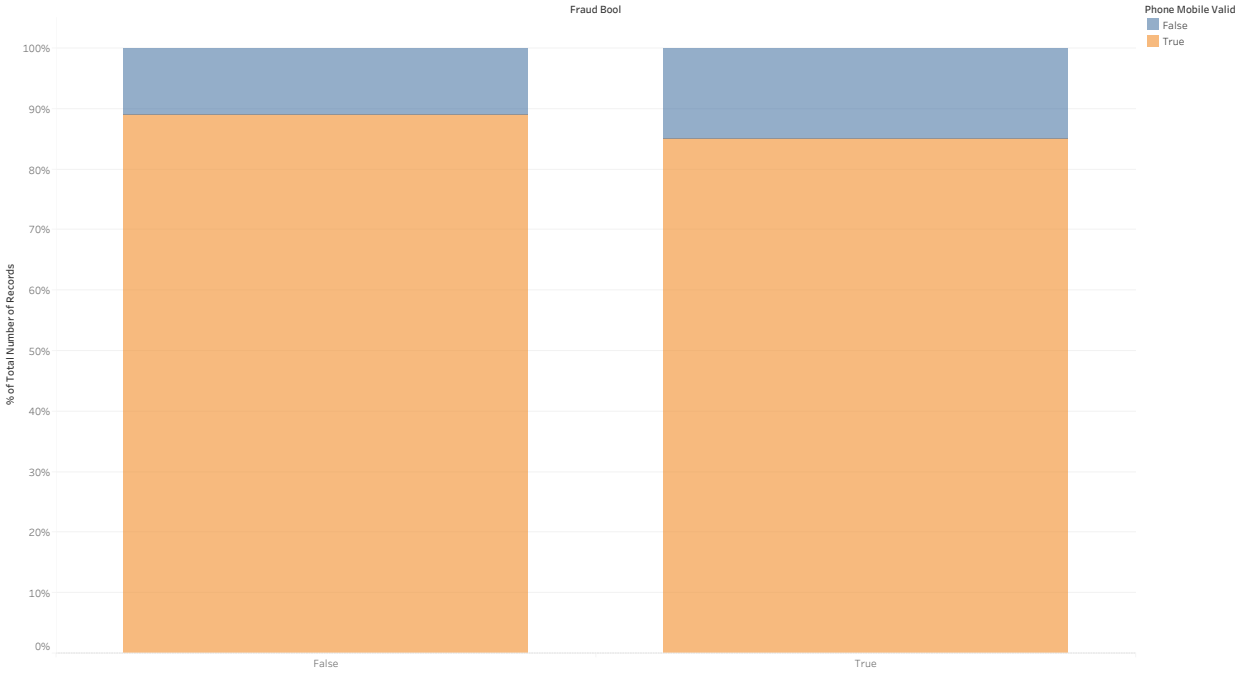


Figure D-20: Has Other Cards Correlation

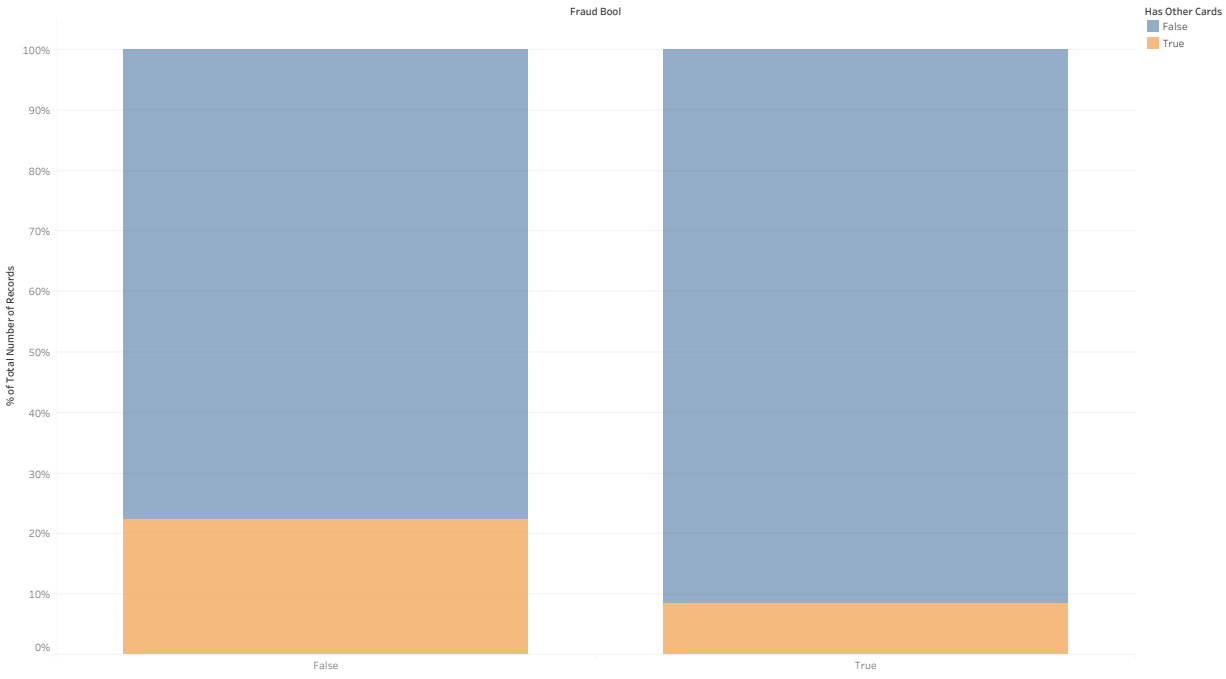


Figure D-21: Foreign Request Correlation

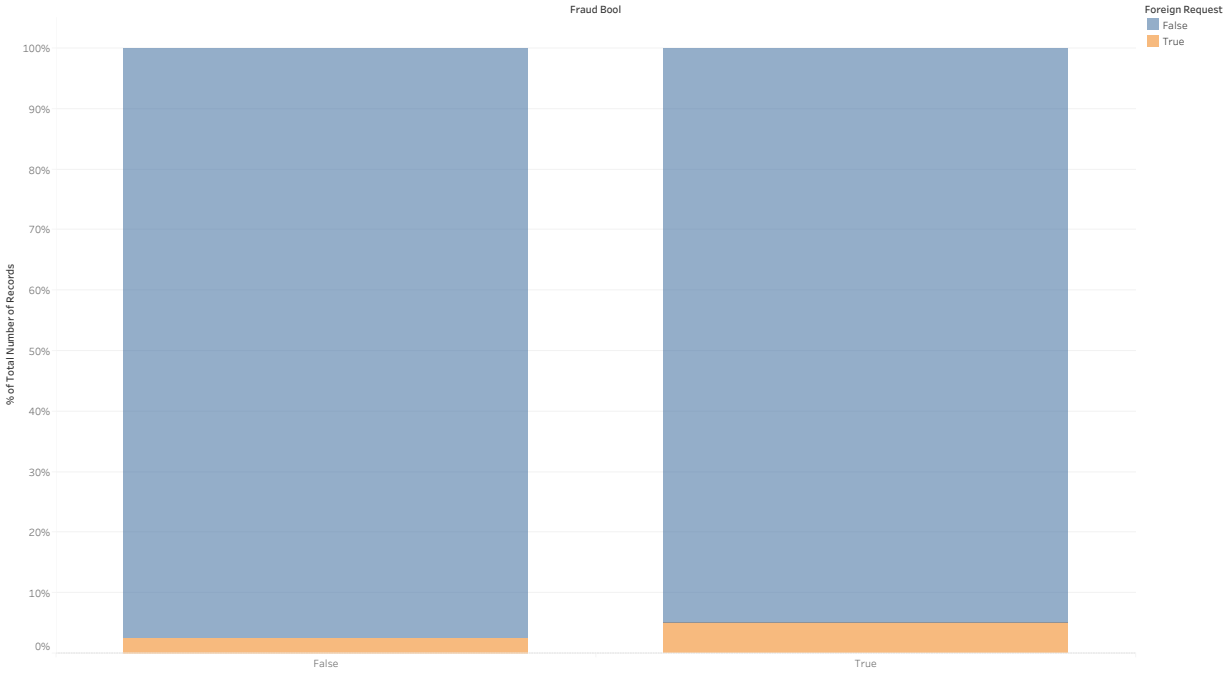


Figure D-22: Source Correlation

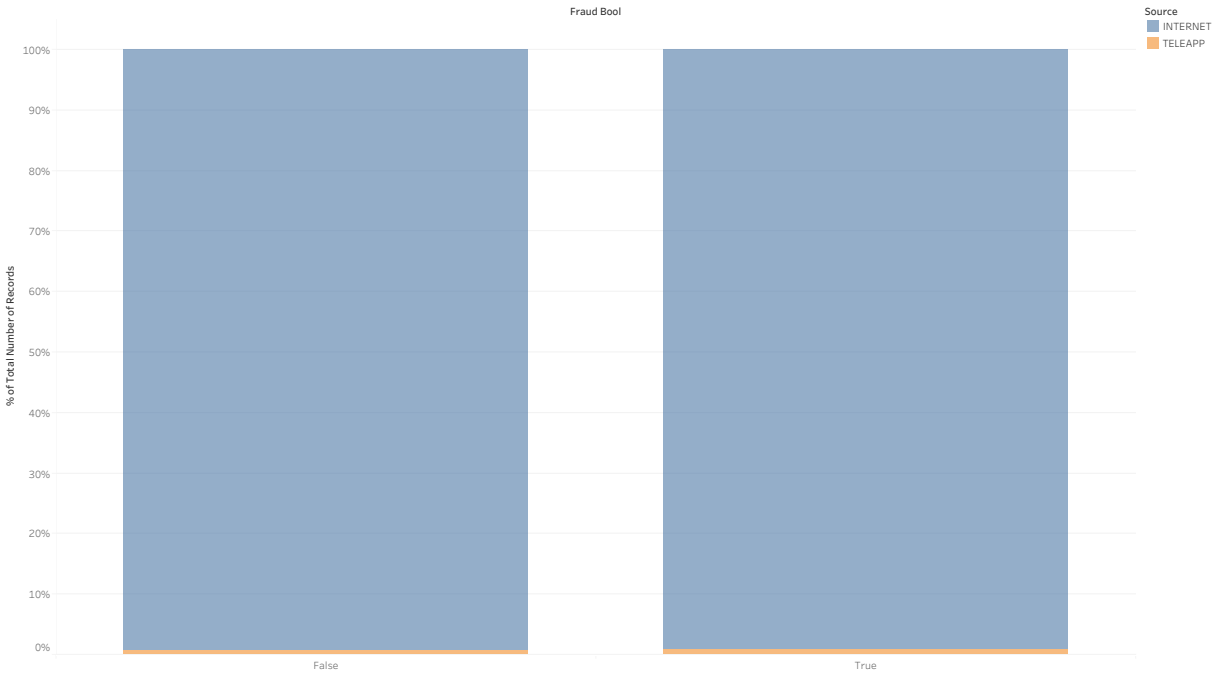
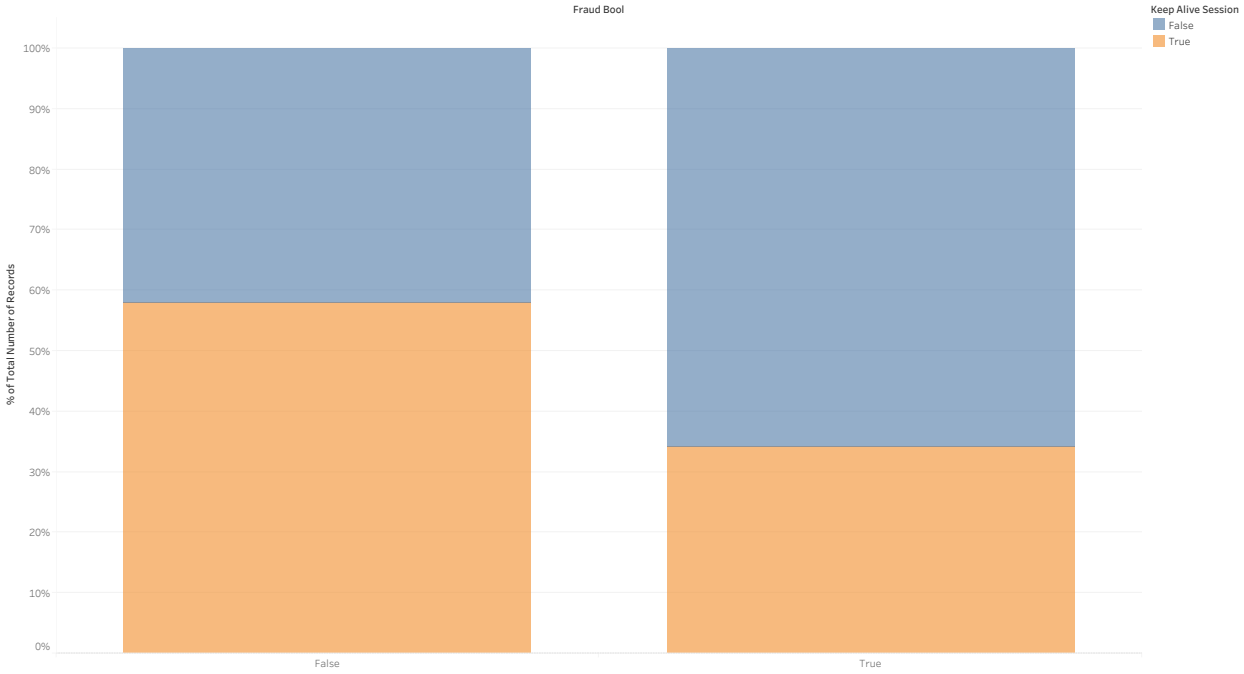


Figure D-23: Keep Alive Session Correlation



Annex E: Victim Clustering Profiling

1. The seven victims clustering profile that was identify by the K-Means Model is shown in **Figure E-1 to E-7**.

Figure E-1: Cluster 1

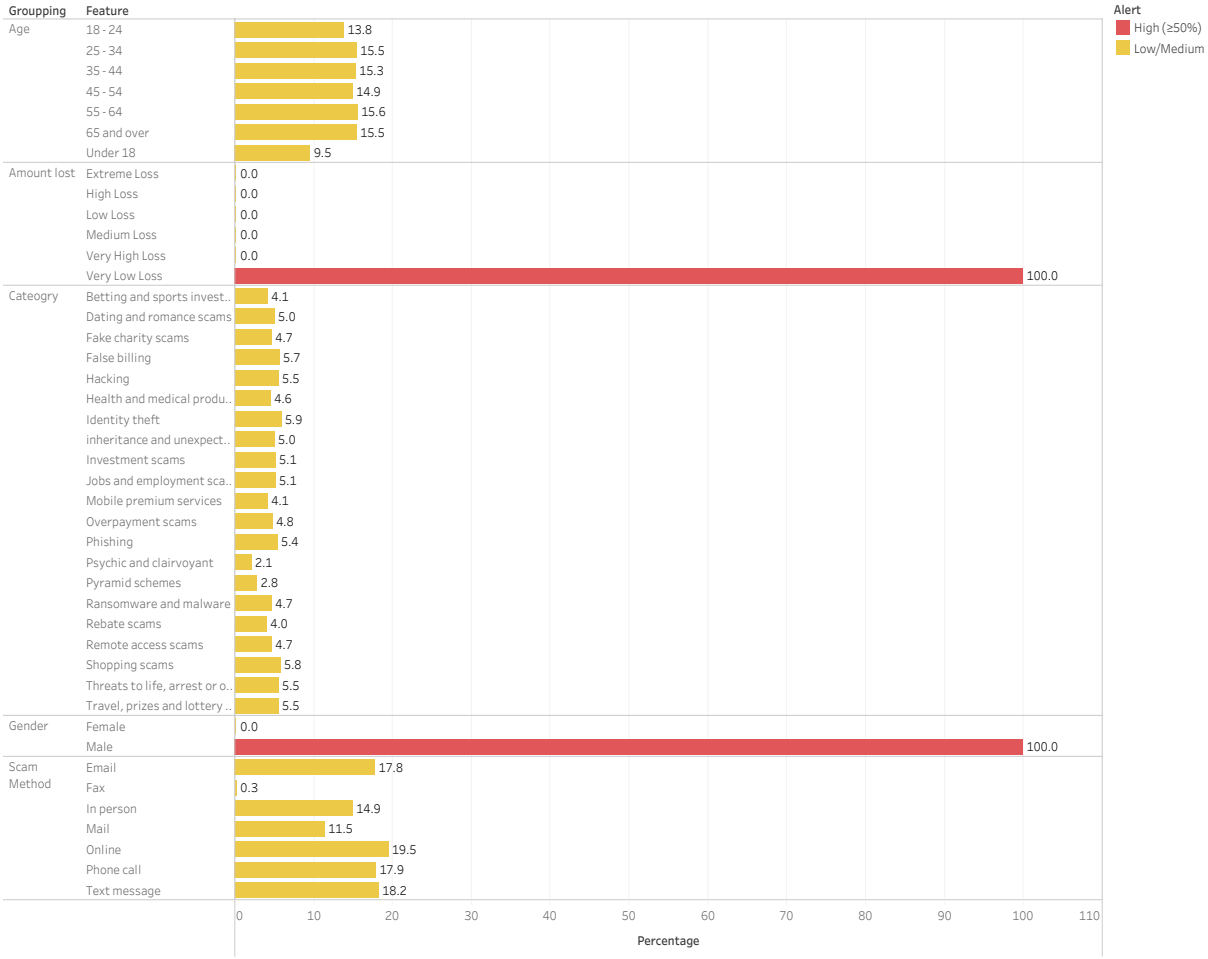


Figure E-2: Cluster 2

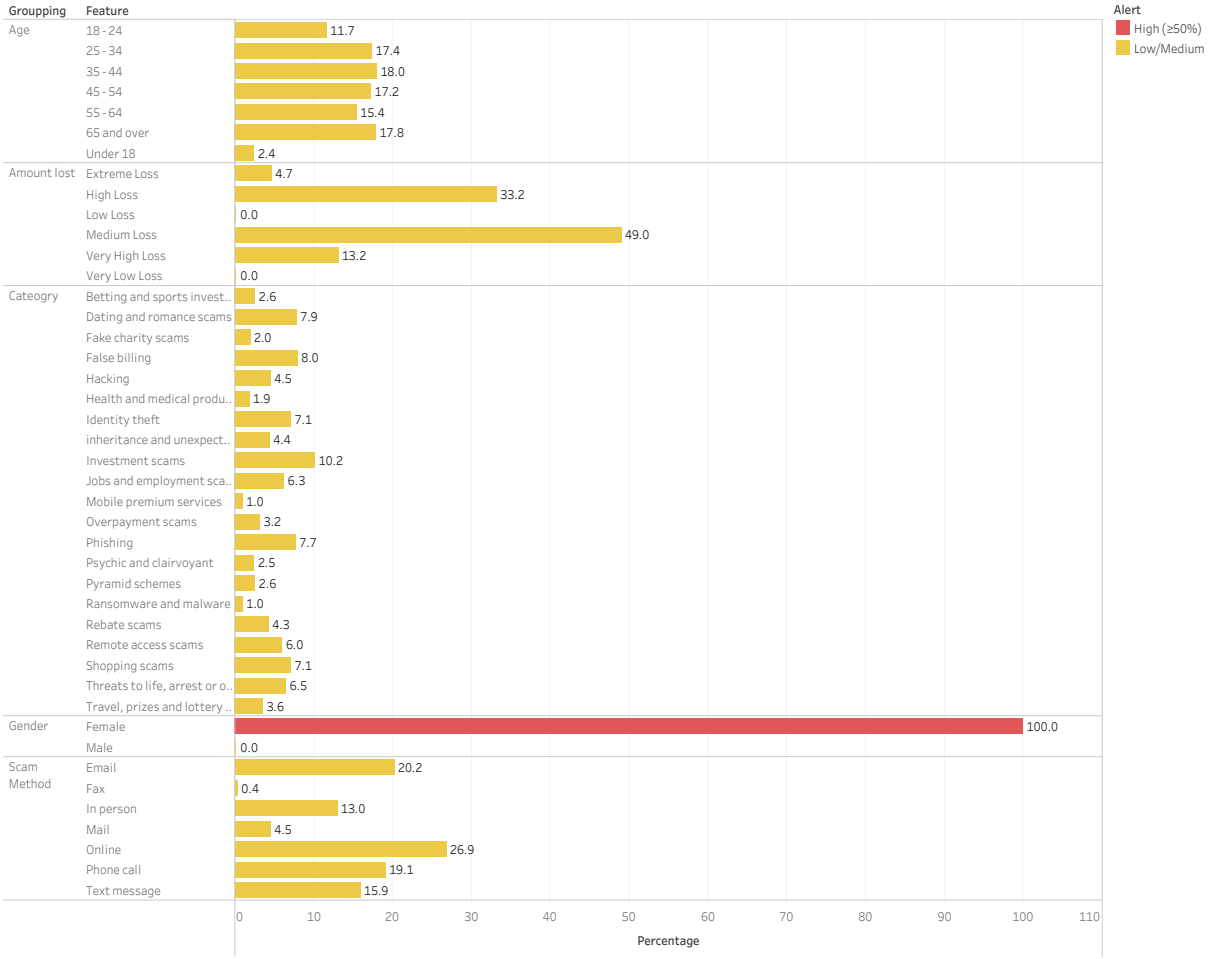


Figure E-3: Cluster 3

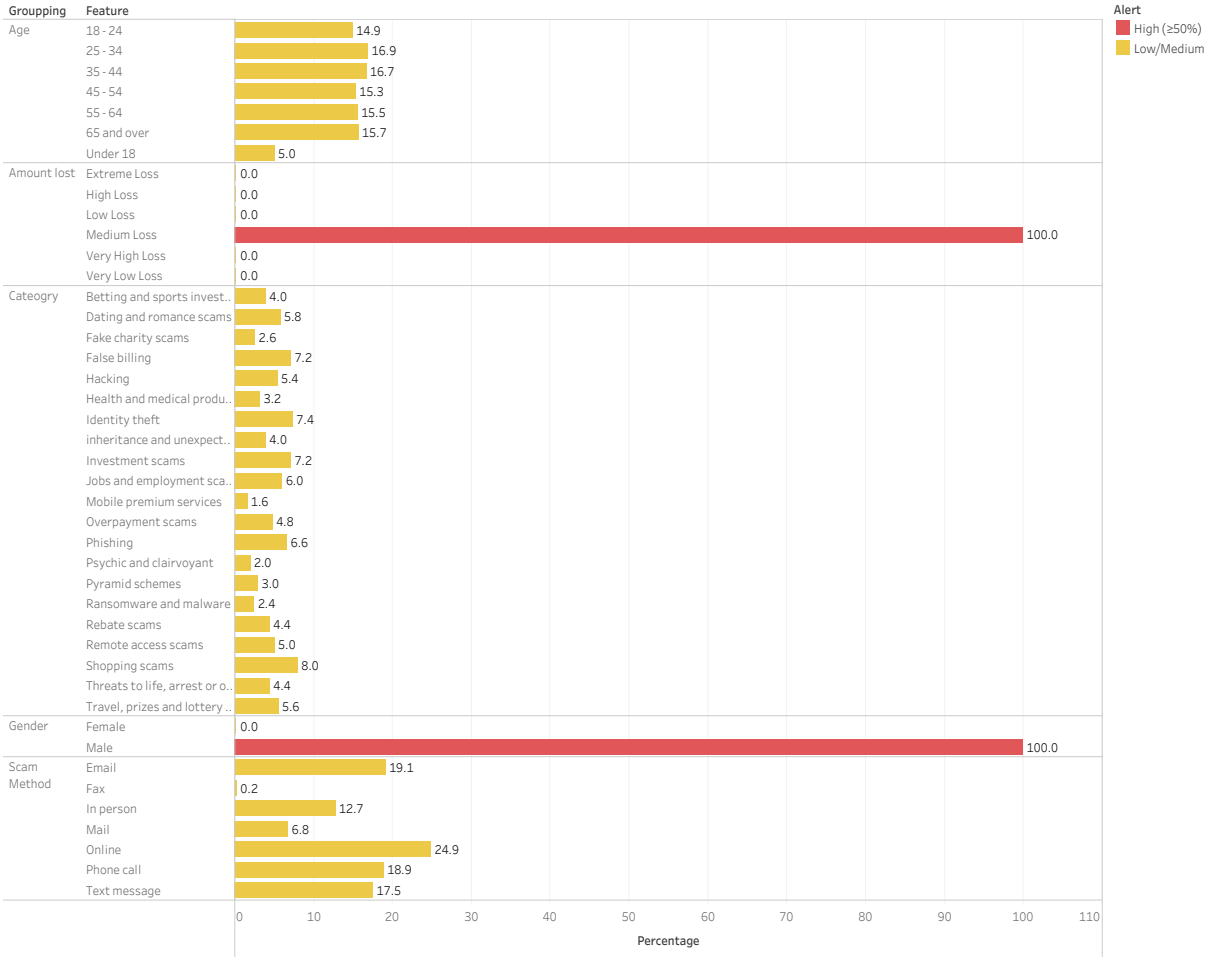


Figure E-4: Cluster 4

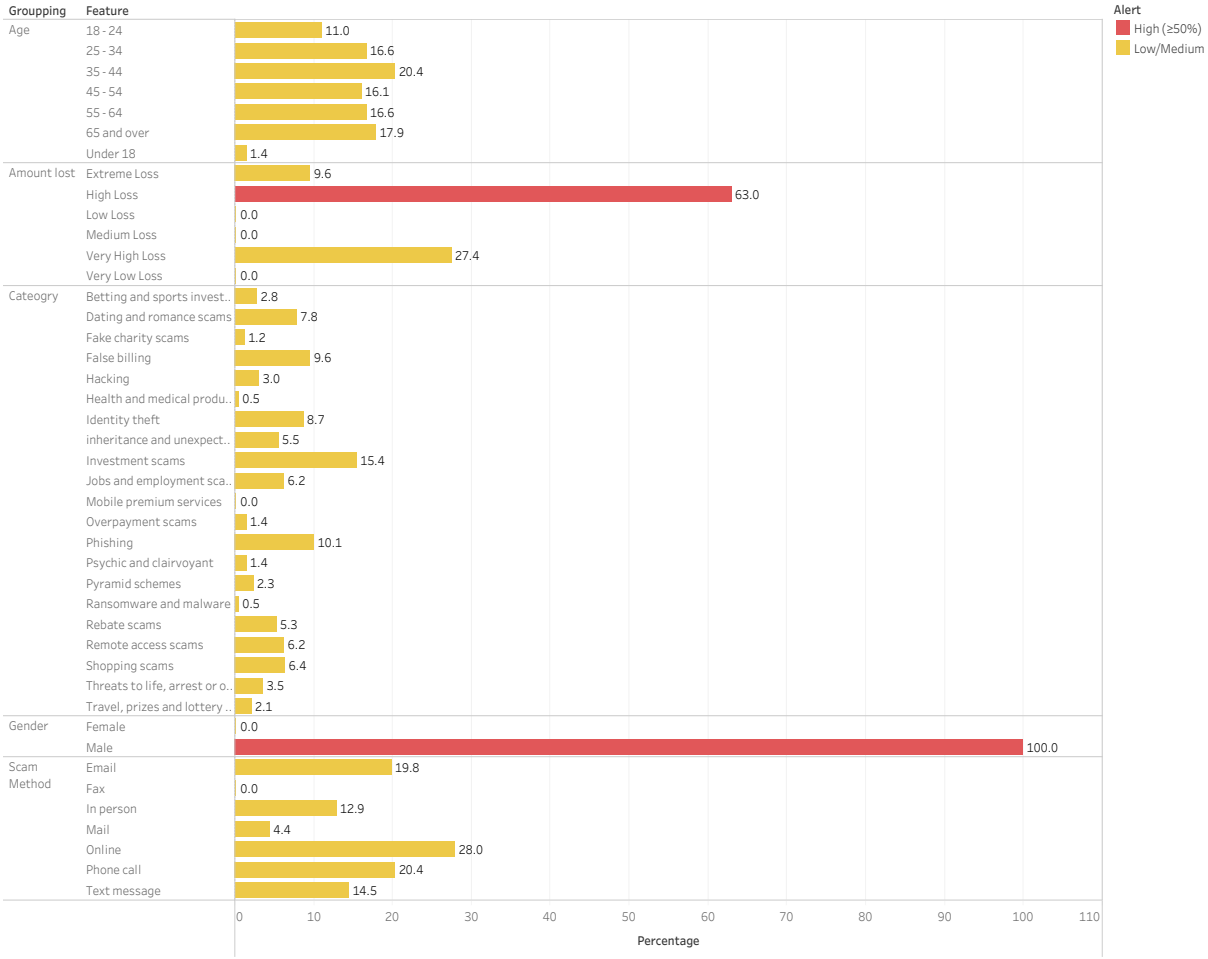


Figure E-5: Cluster 5

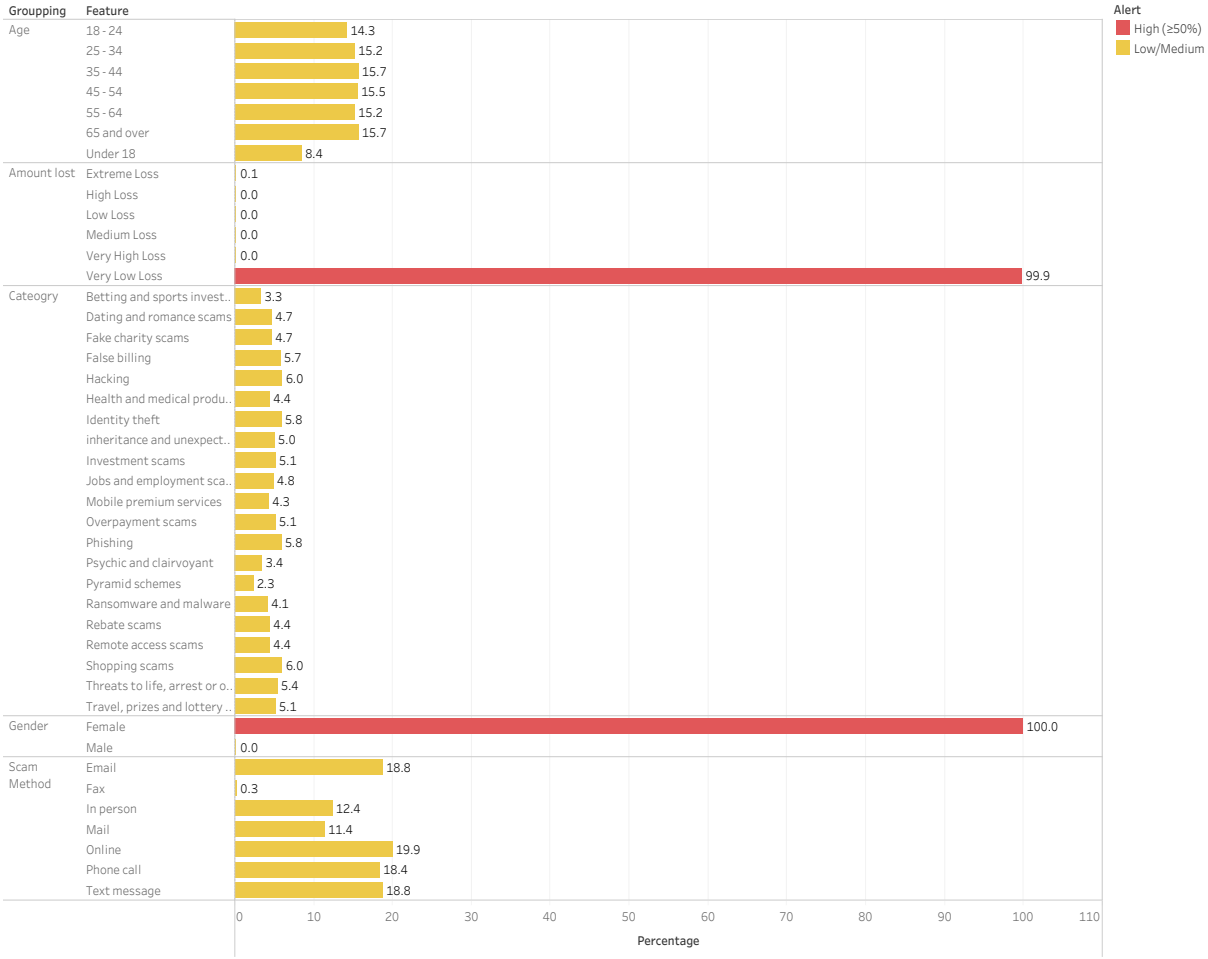


Figure E-6: Cluster 6

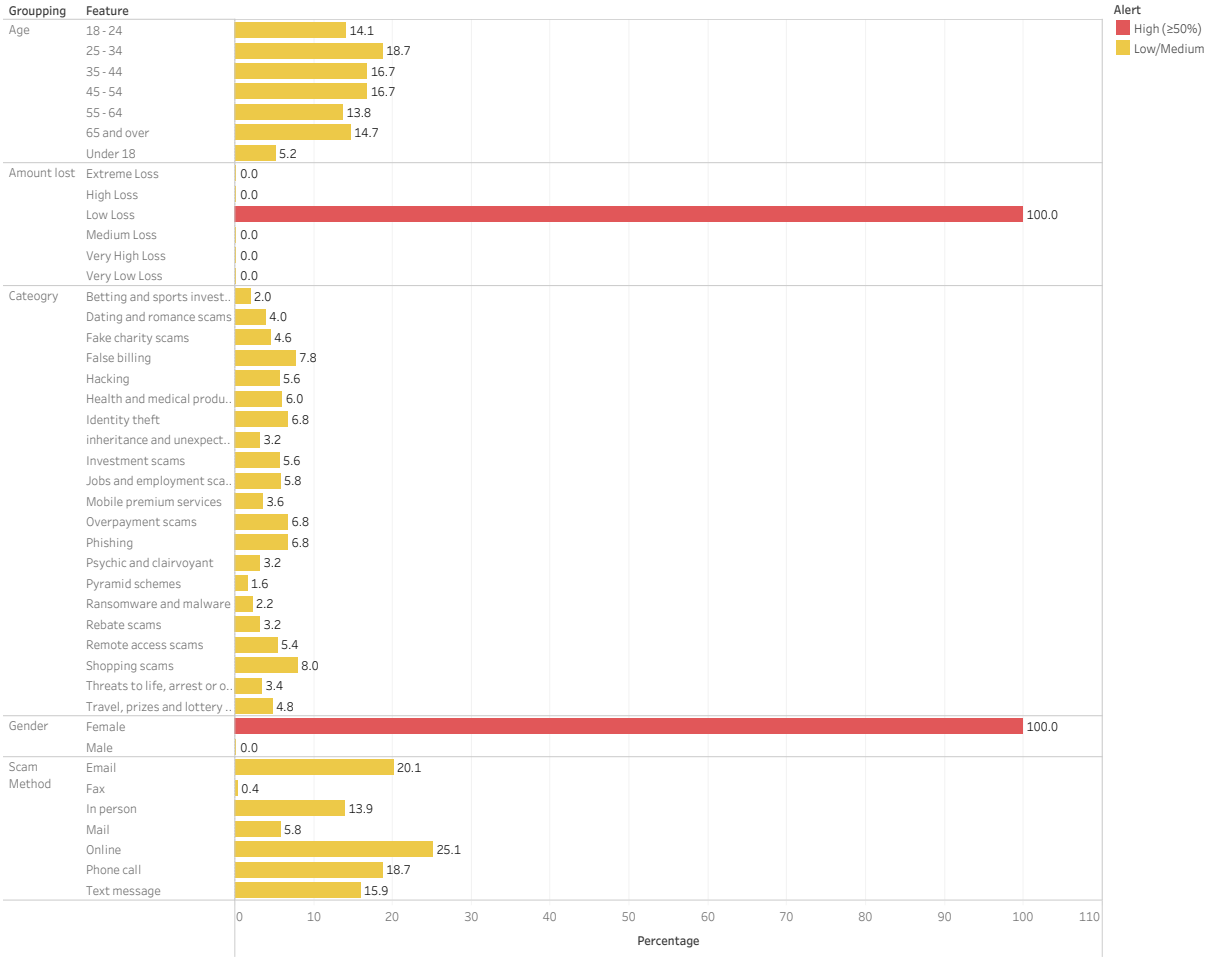
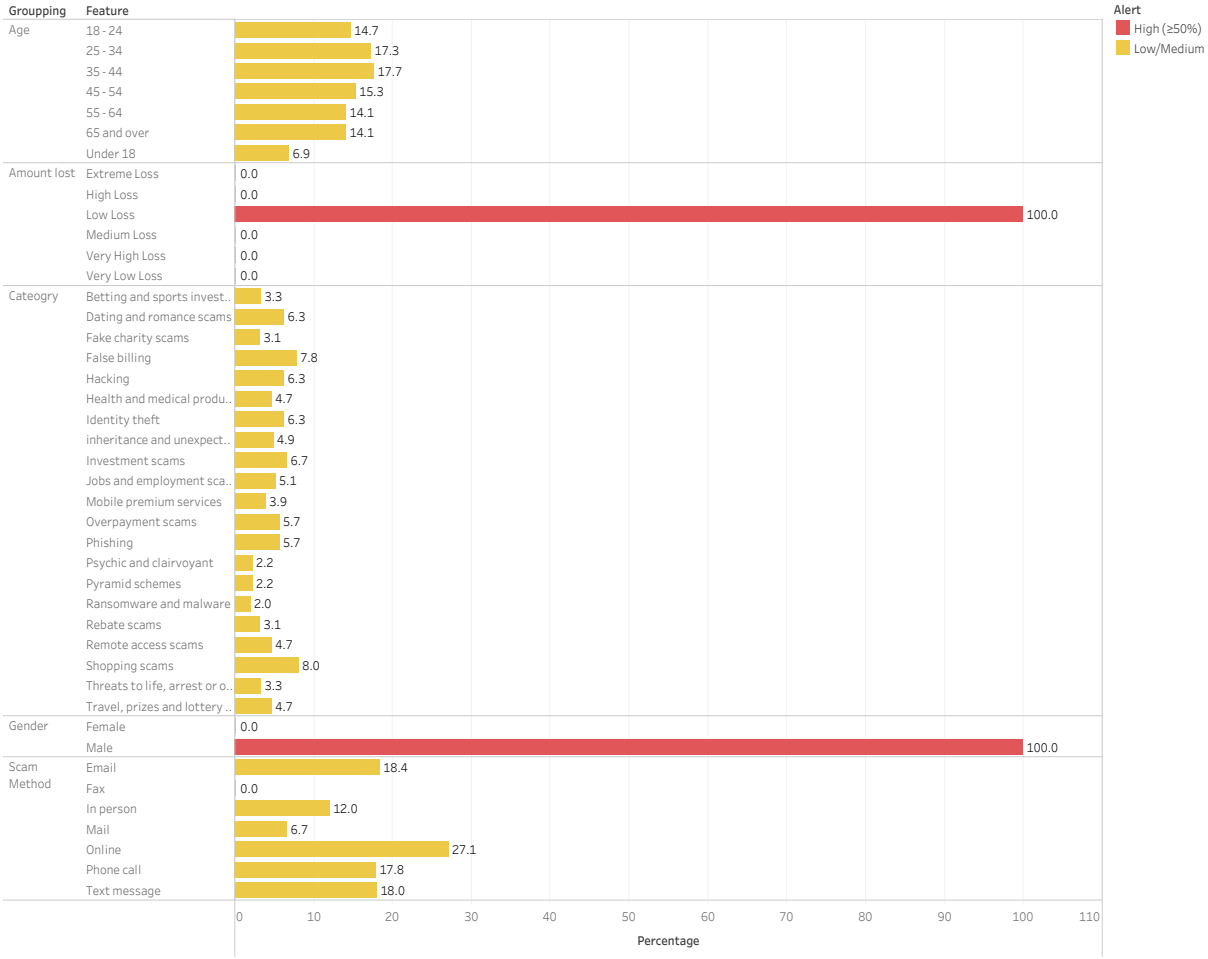


Figure E-7: Cluster 7



Annex F: General Victim Profiling

1. The general victim profiling from ScamWatch is shown in **Figure F-1 to F-5**.
2.

Figure F-1: Overall Victim Age

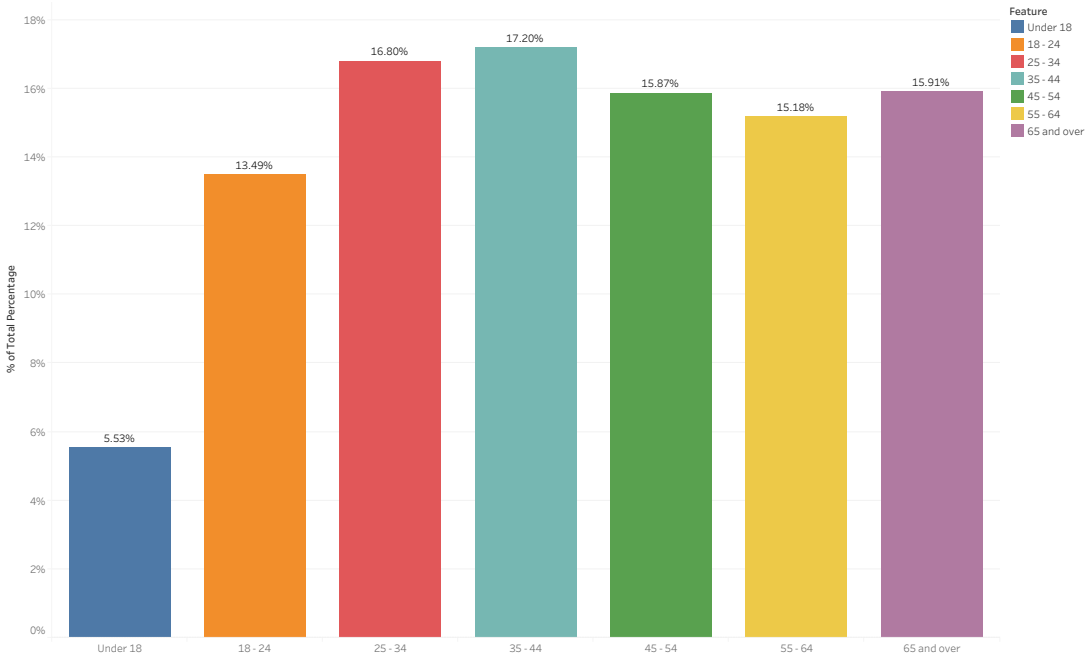


Figure F-2: Overall Victim Gender

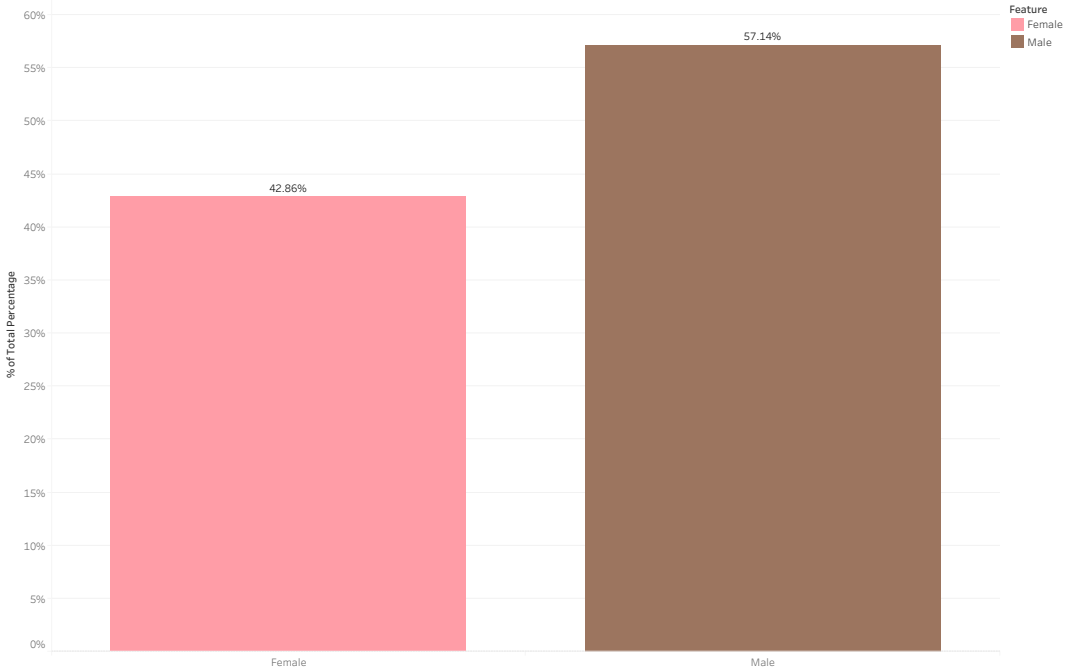


Figure F-3: Overall Victim Scam Method

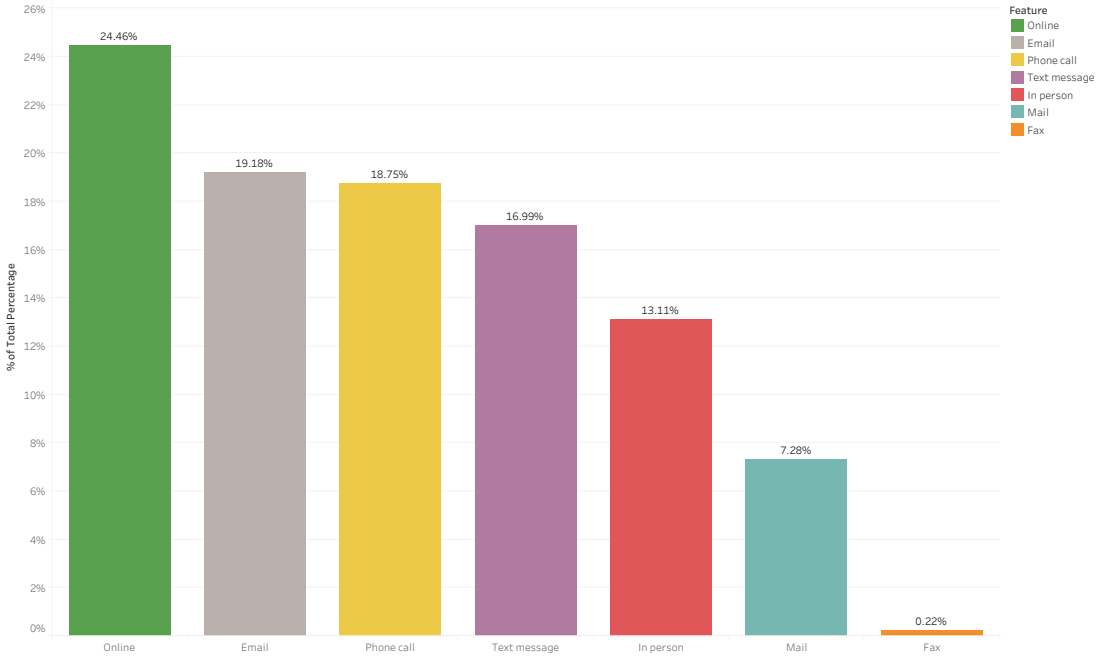


Figure F-4: Overall Victim Scam Amount

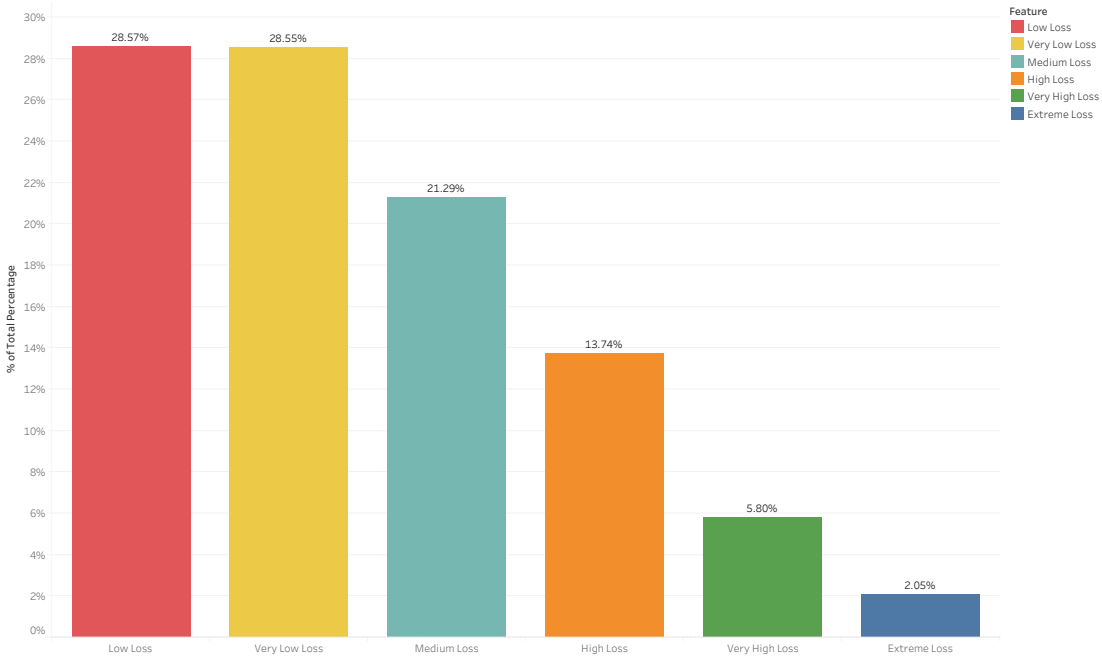


Figure F-5: Overall Victim Scam Category