Hennessy: analysis of fraudulent amazon reviews

Cybercrime and Fraud Detection 23/24

1st Eduardo Mosca 276661
LUISS Guido Carli University
Management and Computer Science
Rome, Italy
eduardo.mosca@studenti.luiss.it

2nd Andrea Stella 270861
LUISS Guido Carli University
Management and Computer Science
Rome, Italy
andrea.stella@studenti.luiss.it

3rd Fabrizio Dari 269651
LUISS Guido Carli University
Management and Computer Science
Rome, Italy
fabrizio.dari@studenti.luiss.it

4th Nadeer Salem 274491
LUISS Guido Carli University
Management and Computer Science
Rome, Italy
nadeer.salem@studenti.luiss.it

5th Rafael Evangelista Monteiro CS05158
LUISS Guido Carli University

Rome, Italy
r.evangelistamontei@studenti.luiss.it

***Abstract*—In this document we will delve into a dataset containing information on fraudulent Amazon.com reviews with the goal of analyzing data to extract valuable information on fraud dynamics. This will be done through data science practices and the implementation and interpretation of machine learning (ML) models in the context of fraud detection.**

I. Introduction

Our dataset contains information on a given amazon review's: rating, purchase verification, product category, the product ASIN, and text fields containing product title, review title and review text. On top of those fields we have a label telling us whether that row is describing a fraudulent review or not.

- Rating
- Purchase Verification
- Product Category
- Product ASIN
- Product Title
- Review Title
- Review text
- Label(label1 or label2)

It is important throughou this analysis to remember that we chose to consider label2 as the fraudulent label from the start of the work.

II. Exploratory Data Analysis - Part 1

The first step to extracting vauable information is getting to know the dataset, and through data visualization and checking feature distribution we do just that. Our main insights from the (first) EDA[Fig.1] were:

- Amazon review ratings tend to be close to 5 more often than not, with 5 being the most frequent rating by far.
- In the data both product category and label classes are all equally distributed, this is not the case in reality and might be a result of sampling stratification.
- "YES" and "NO" in the verified purchase fields are similarly distributed, even though "YES" is the more frequent one.

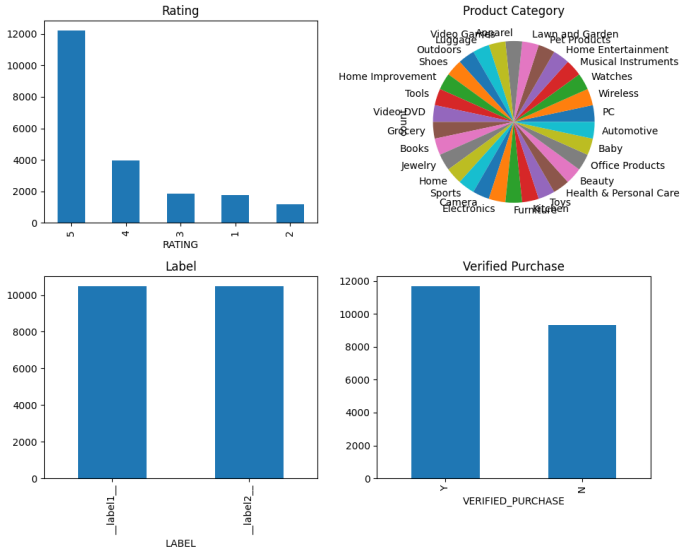

Fig. 1. Distributions for Rating, Product Category, Label and Verified Purchase

III. Feature Engineering

Once we took a first look at the values in some fields, we realized there are many variables involved with an Amazon review, and we began to think about how to extract more data so as to build a more powerful models. This practice of adding new columns or values from ones already present in our dataset is called "Feature Engineering", and so we move forward by adding the following columns:

- "Review Length": built to be the number of characters included in a review's text.

- ”Review Title Length”: analogue to the first, but for a given review’s title.

A. *Benford’s Law*

We also check whether these variables follow Benford’s Law to have confirmation of their reliability, so we check the distribution of their first digits[Fig.2 and Fig.3] and confirm that they fulfill the Law’s requirements.

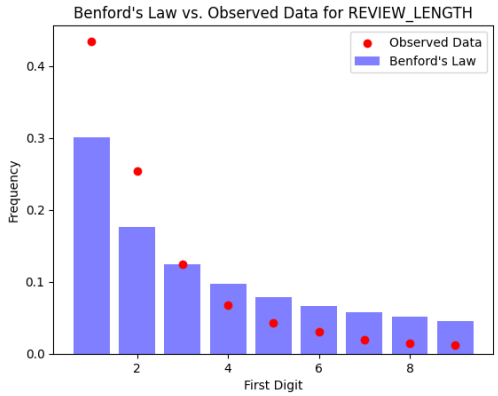

Fig. 2. First-digit frequency for ”Review Length” against Benford Distribution

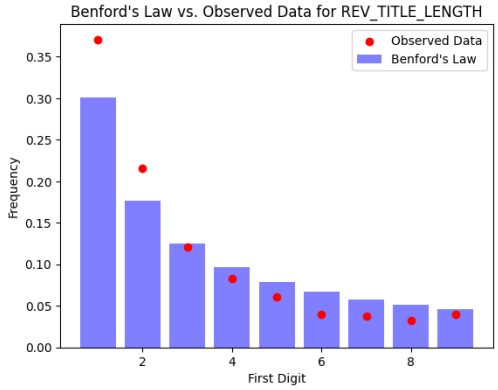

Fig. 3. First-digit Frequency for ”Review Title Length” against Benford Distribution

IV. Natural Language Processing for Sentiment Analysis

Even having extracted length fields for text columns, there is much left on the table. In fact, having review content and its title, we are able, through Natural Language Processing(NLP), to assess a given review’s sentiment; *i.e.* whether it contains positive or negative attitude towards the product. Adding these additional fields to our dataset can empower ML models and grant us more insight on fraud dynamics. The (sequential) approach taken to analyze review sentiment was the following:

- Lower casing
- Punctuation removal
- Removing special characters
- Tokenizing words
- Removing stop words
- Stemming words
- Removing emojis URLs and mentions.

By then joining review tokens back, the review sentiment (or polarity) score was taken to be the sum of each token’s polarity score, normalized to be between 1 and -1. Values above zero described positive sentiment while those below 0 negative.

V. Correlation Analysis

We move on to a correlation coefficient interpretation for our variables, making some adjustments to the categorical columns we encode categorical fields such as ”LABEL” and ”VERIFIED PURCHASE” to a numeric format (0/1) so that we can use them in our correlation analysis. We then developed a matrix and proceeded to use it to create a heat map, resulting in the following[Fig. 4]: Looking at the heatmap, we

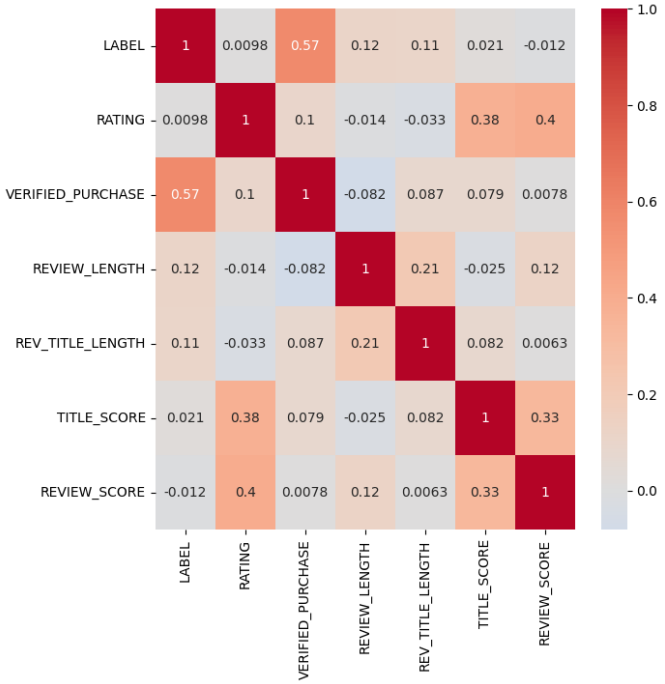

Fig. 4. Correlation coefficient Heatmap for numeric variables

see a lack of very negatively or very positively correlated pairs of variables. There is a positive, strong correlation between the review and title scores and the rating; which suggests that the more positive the sentiment, the higher the star rating. A moderate correlation (0.33) between the scores of the titles and reviews, meaning the sentiment expressed in titles is quite consistent with the one of the reviews. ”LABEL” and ”VERIFIED PURCHASE” have an unexpectedly high positive correlation; that would suggest that a verified purchase more frequently calls for ”label2”, which is supposed to be the fraudulent one. Most of the other correlations were around zero.

VI. Exploratory Data Analysis - Part 2

Wanting to better understand our new variables, we once again plot them and visualize their characteristics[Fig 5]:

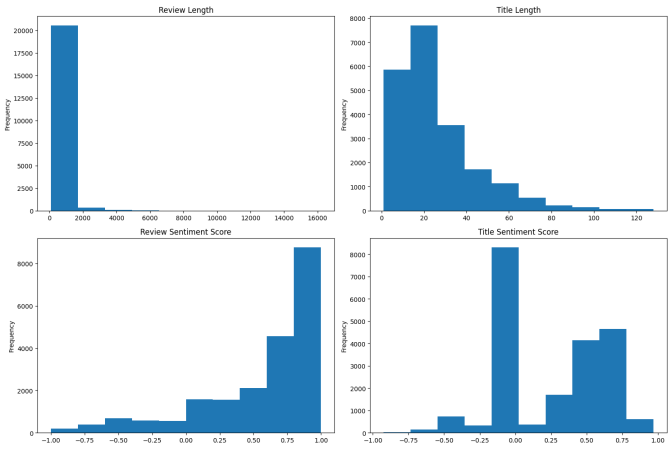

Fig. 5. Histograms for: Review length(top left), Title length(top right), Review score(bottom left), Title score(bottom right)

- We notice that review length is very skewed to the right, therefore before training models we will make sure to remove some outliers beforehand.
- Other variables also skewed, and most title scores show neutral sentiment around 0.

VII. Fraud Subset Analysis

In an effort to extract more information of fraudulent review behavior, we analysed a subset comprised of all the points with ”LABEL” equal to ”label2”, which in our analysis is considered the fraudulent one. After plotting the variables we noticed that their distributions are pretty much the same as in the complete dataset, with the exception of ”VERIFIED PURCHASE” being 1 way more than 0 in the fraud subset[Fig. 6].

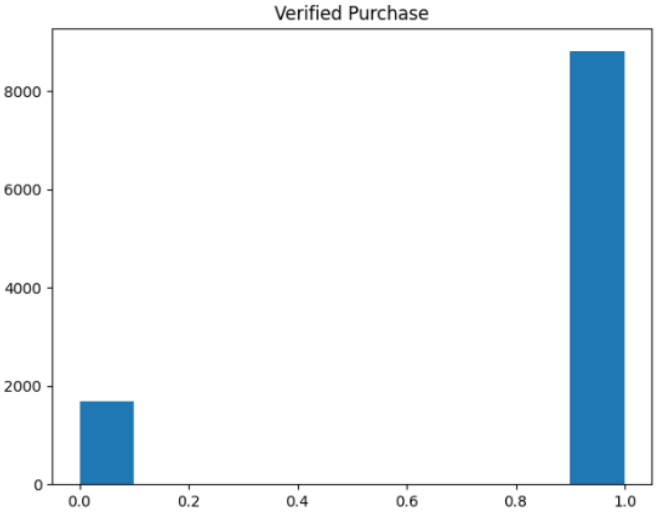

Fig. 6. ”VERIFIED PURCHASE” distribution in the ”label2” subset

VIII. Anomaly Detection

Beyond the labels contained in the dataset, we can employ an ML model for anomaly detection in order to find the most anomalous rows, that is, points which deviate from the norm in a significant statistical manner. Using the Isolation Forest algorithm which is based on the fact that in the hyperspace anomalous rows are easier to separate from the rest, we set up the analysis knowing that frauds represent 50% of rows in our dataset, this is similar to the general belief that fake reviews comprise around 30% or 40% of all online reviews, so we set the ”contamination” parameter of our model to 50%, this will determine what proportion of the dataset we see as anomalous and at the same time determine a threshold for anomaly scores so that rows with a score exceeding said threshold are seen as anonymous. We do not feed the ”LABEL” variable to our IF model, but we keep all the others which hold some value(all except ID variables and non-processed text columns). When evaluating model behavior versus the original labels, metric scores such as AUC, Precision and Recall end up being around 0.5, but more importantly, they all end up **equal**, which is something that can be explained by the contamination parameter choice causing the number of false positives and false-negatives to be the same all around, or anyway by the fact that we always flag 50 percent of the data as anomalous. At the same time it tells us that only 50 percent of statistical anomalies are actually frauds.

A. Anomaly Subset Analysis

Extracting a subset only composed of anomalous rows, we were able to see that for our model’s result, anomalies were very balanced between label1 and label2, as well as for ”VERIFIED PURCHASE”, for ”PRODUCT CATEGORY” it was the usual equally distributed, while ”RATING” was a bit more equally distributed than in the normal dataset, but still with 5 being the most frequent value. As for length and score columns, there was no major difference from those of the complete data. Comparing to a dataset of only non-anomaly rows, ”VERIFIED PURCHASE” is distributed in a fifty-fifty manner for anomalies(that is, 50 percent of anomalous purchases are verified), while in the non-anomalous subset its more towards a 60-40, where 60 percent of non-anomalies are verified.

B. AD Conclusions

To close this section, it is important to remember that AD is a statistical measure with the ability of finding patterns which distinguish a few observations from the rest of a dataset, while our analysis may lead to thinking that there is no clear-cut difference between labels, the insights gained are statistical first, and don’t allow us to conclude that anomalies are frauds; with that said it is true that an outlier analysis has given us a different view of possible malicious activity recorded in the dataset.

IX. Principal Components Analysis

We want to further comprehend the nature of our data points having gained the insights from previous analyses, therefore we reduce the dimensionality of the data through Principal Components Analysis(PCA) so as to be able to plot the points in a 3D space. It is worth noting that the three components built as a result from the entire dataset have a portion of retained variance equal to 0.99999. From our first plot[Fig.7], where we color points differently simply based on whether their LABEL is 0(label1) or 1(label2), we find that frauds and non-frauds lie very close for the most part, and even both deviate from the majority of the points(although label2 points deviate the most).

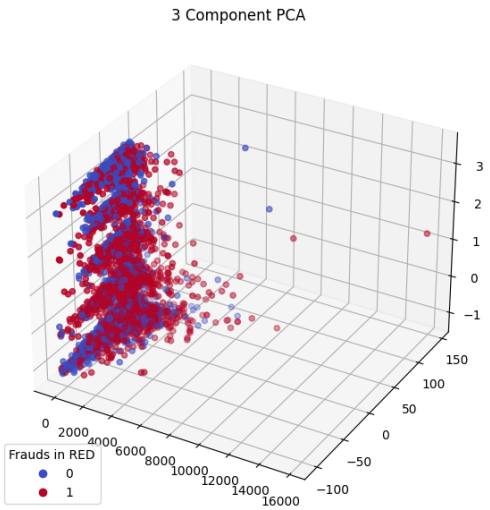

Fig. 7. 3-Component PCA

A. PCA With Anomaly Detection

Visualizing anomalies in the 3D space was more troublesome[Fig.8], only after a while did we find out that non-anomalies were hiding on the opposite side, surrounded by the anomalies in the red points. Since we set the contamination parameter to 0.5 for our model earlier, knowing that we have 50 percent anomalies and 50 percent non-anomalies, we assume that blue points are covered up by red ones, making us think non-anomalies are the ones most closely following these 5 levels of height the data follows in the 3D space.

B. PCA with Clustering

Given the apparent similarity of different label points in a 3D space, we move to substantiate this observation through a clustering step. Implementing the K-Means algorithm with 2 clusters[Fig.9], we see that a great deal of points, both frauds and non-frauds are grouped together, with the second cluster representing the rest of the points. By exploring cluster properties we get the following insight[Table 1]:

By "Fraud Percentage" we intend the portion of frauds making up a cluster, while "Overall Fraud Percentage" is intended to be the portion of frauds in the entire dataset

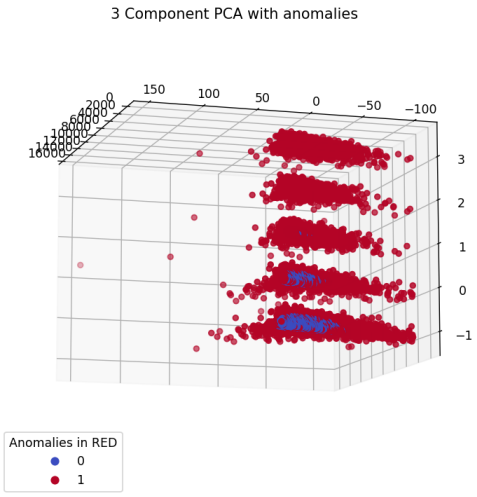

Fig. 8. 3D PCA anomaly detection visualization

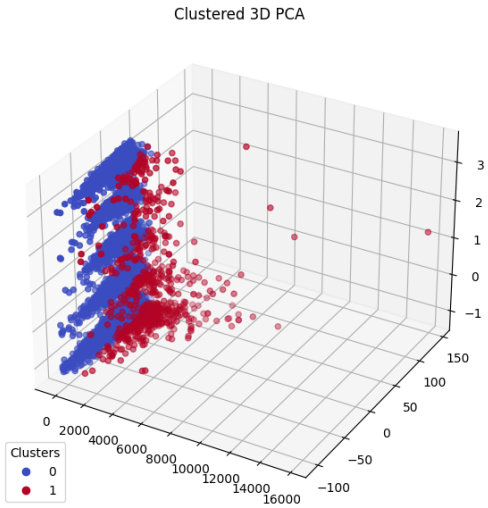

Fig. 9. Clustered 3D PCA

Cluster	Points in Cluster	Fraud %	Overall fraud%
0	20206	48%	94%
1	794	78%	6%

TABLE I
Cluster dynamics

contained in that single cluster; know that these numbers might change on different code runs but should stay similar.

X. Shallow Decision Tree Analysis

A modern problem in machine learning applications is the lack of interpretability of models, the choice of a model brings performance but also a degree of interpretability. A simple model that has an acceptable performance but also can be interpreted very easily is the decision tree, that is why we fit a shallow one (that is, a tree with low max depth, in this case with a maximum depth of 3) on our data to visualize fraud

characteristics. The fitted model was not perfect but achieved acceptable scores[Table 2]:

Metric	Score
AUC	0.81
Precision	0.75
Recall	0.88
Accuracy	0.80

TABLE II
TEST SET METRICS FOR SHALLOW DECISION TREE MODEL

Since we've seen the model to be somewhat reliable, let's visualize its splits in order to extract the aforementioned interpretability[Fig. 10]: Unfortunately its not so visible in the

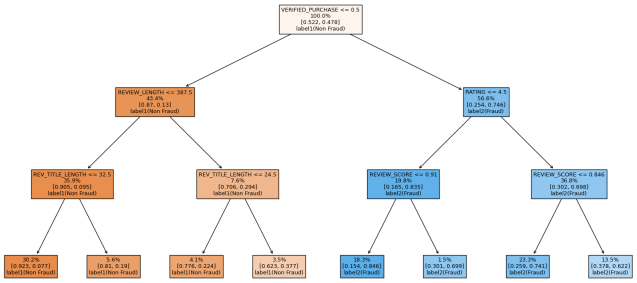

Fig. 10. Shallow decision tree splits on data columns

document, but the tree has one main dynamic, the first split is done on "VERIFIED PURCHASE", if its 0, we go left where all the tree nodes are flagged as "label1(non-fraud)", meanwhile if the purchase is verified we go right where all nodes are flagged as the opposite, that is "label2(fraud)". The results seem to contradict what we might think at first glance, or maybe the choice of fraudulent labels should be reviewed.

XI. XGBOOST CLASSIFICATION

As mentioned before, models bring in some amount of performance and interpretability with them, now we employ a very powerful model who is able to extract abstract and complex patterns from the data, to see how efficient we could be in a live setting at detecting or flagging possible frauds. Therefore we set up an XGB Classifier, which is a complex model based on slow-learning trees: we first split the data into a training and test set, we go on to perform 5-fold cross validation to find the best model hyperparameters from looking at the training data, then we feed the test set to our best model and evaluate its performance. Our best model achieved the following metrics[Table 3]: It is interesting to see that a

Metric	Score
AUC	0.80
Precision	0.74
Recall	0.88
Accuracy	0.8

TABLE III
TEST SET METRICS FOR XGB CLF MODEL

complex model such as XGB has achieved similar metrics to a very simple one like the shallow decision tree, and in some cases (Precision and AUC) it was slightly below the latter. Even so, the little interpretability we get from the XGB Clf model[Fig.11] tells us that its reasoning is completely different from the Shallow decision tree, seeing as "VERIFIED PURCHASE" is regarded as the least important feature by the last model, with length fields being the most important ones. The F-Score in this case refers to the number of times

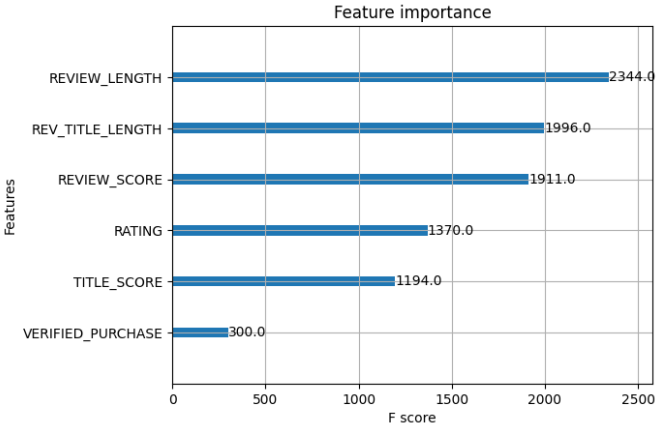

Fig. 11. F-Score variable importance for XGB Clf model

a variable is split on by the model's trees.

XII. CONCLUSIONS

Concluding our analysis, a recap of our insights follows:

- Statistically, frauds and non-frauds are not so distinguishable.
- Both classes may follow anomalous patterns.
- Simple and complex models both achieve good results at classifying labels in this dataset, but give more importance to different types of variables.
- Verified purchase is a main indicator of review class.
- Frauds and non-frauds show pretty similar sentiments(both around 0.5)