Salma’s Script

Slide 1

Hi, today our group will be presenting on Anomaly Detection to Identify Fraudulent Credit Card Transactions. Our group consists of Razvan, Samanthi, Amar, Farjana and myself Salma.

Slide 2

So when looking for a topic to base our project on we wanted to look for real world problems that a significant amount of the population may be exposed to. And after looking through multiple potential datasets we had come across credit card fraud detection.

In this age of digital transactions, millions of credit card transactions occur every day generating lots of data. And unfortunately, among these millions of transactions there are fraudulent ones that cost both cardholders and credit card companies billions of pounds every year.

The scale and complexity of the data makes it extremely difficult for human to identify meaningful patterns that would indicate a fraudulent transaction be described as finding a needle in a haystack. To counteract this companies have adopted the use of machine learning techniques which has now become a lot more widespread. Machine learning provides us with a fast and effective way to extract relevant information from large datasets and identify patterns that could potentially indicate fraud.

Our project aims to show machine learning techniques to detect fraudulent and non-fraudulent credit card transactions.

Slide 3

So, after identifying our project topic and dataset we then moved on to analysing our dataset which you can see here on the left. The picture shows a glimpse of the dataset containing transactions made by credit cards in September 2013 by European cardholders over a period of 2 days. Out of total of 284,807 transactions, 492 were fraudulent.

So, what do we know about the dataset? We retrieved the dataset from the link seen at the bottom right-hand corner leading to a Github page where the dataset is housed. The dataset is an anonymised, normalised real world dataset.

Slide 4

As the dataset was already normalised, we confirmed the dataset was normalised and started to clean the data by getting rid of duplicate and NaN values. To better understand the dataset we created a histogram for each of the numeric variables in our dataset using the Matplotlib library. This allows us to better see the distribution of the data and identify patterns/anomalies. We concluded that the Principal Component Analysis (PCA) components are distributed in a Gaussian manner and are therefore likely standardized. The majority of the components appear to be centered around zero. Also, we noted that the amount variable is not displayed on the histogram, indicating that its distribution may be skewed.

Slide 5

As we began our exploration of the credit card transaction dataset, we conducted an in-depth analysis to better understand its characteristics. Our exploratory data analysis included the creation of visualizations, one of which is shown on the right side of the screen. This visualization displayed the number of transactions that were mentioned earlier, which enabled us to properly identify that our dataset is severely imbalanced. In fact, there is only one fraud transaction for every 577 non-fraud transactions.

While this may seem concerning, it is actually beneficial for credit cardholders, as they would prefer a lower incidence of fraudulent transactions. If the distribution were more balanced, they would likely consider switching to a different credit card company. It's also worth noting that this kind of imbalance is not unique to fraud detection, but is often observed in other real-life classifications, such as claim prediction and default prediction.

Slide 6

After conducting exploratory data analysis, we moved on to selecting the appropriate tools to analyze the dataset and carry out the fraud detection task. We utilized the machine learning libraries, Scikit Learn, Tensorflow, and Keras, to classify transactions as either fraudulent or legitimate by detecting anomalies.

To ensure that our models were accurate, we split the dataset into training and testing sets and employed supervised, unsupervised, and deep learning models. We then evaluated the performance of each model using metrics such as accuracy, precision, and recall.

The simplified version of our approach is presented on this slide, and our team members will provide further details on the specific techniques employed during the analysis.