**Report: Analysis and Improvement of AWS Fraud Detection Machine Learning Model**

**Enhancing the Accuracy of AWS Fraud Detection Machine Learning Model**

**Introduction**

This analysis aims to assess the performance of an AWS fraud detection machine learning model by means of several criteria to forecast fraudulent behavior. Examined are possible upgrades and present performance problems of the model to guarantee more accurate fraud detection.

**Methodology**

The analysis involves evaluating the model’s decision-making process based on the importance values of the parameters it uses. The key parameters considered are IP address, email address, billing date and city, user agent, billing postal code, and phone number. The focus is on understanding how each parameter influences the model's predictions.

**Analysis of Model Performance and Statistics**

The analysis examines the decision-making process of the model depending on the relevance values of the used parameters. Key factors considered include IP address, email address, billing date and city, user agent, billing postal code, phone number. The main emphasis is on realizing how every parameter affects the forecasts of the model.

**Examining Model Performance and Statistics**

Numerous important model statistics are displayed below. Presenting important metrics and visualizations helps one assess the model's performance in differentiating between legitimate and fraudulent transactions.

**Model Type and AUC**

**Model Type:** Online Fraud Insights   
**AUC:** 0.95; 0.94–0.96 uncertainty range

With an Area Under the Curve (AUC) of 0.95 the model shows great capacity for discriminating between real and fraudulent events. With a near 1.0 AUC score, the model shows general good classification performance.

**Distribution of Score**

The score distribution graph shows how different score ranges distribute real and fraudulent events. Applying a rule with a 500: model score threshold will help to:

**The true positive rate (TPR)** is 93.2% of all fraudulent events—that is, fraudulent transactions found.   
13.7% of valid events are wrongly categorized as fraud (legal transactions misclassified as fraud).

The histogram shows that whereas fake events are more dispersed with higher scores, most authentic events are concentrated at lower score ranges. This distribution emphasizes the trade-off between misclassifying real events and spotting fraud, therefore supporting the ability of the model to identify it.

**The Confusion Matrix**

Using 100,000 sample events—including 95,189 valid and 4,811 fraudulent events—the confusion matrix offers a comprehensive analysis of the model's categorization performance.

**True Positives (TP):** 4,454 accurately noted fraudulent events.   
**False Negatives (FN):** 324 false negatives labeled as valid.   
**False Positives (FP):** 13,072 real events misclassified as fraud.  
82,150 authentic events properly identified as True Negatives (TN).

These figures let us develop the following performance measures:   
TP / (TP + FP) = 4,454 / (4,454 + 13,072) = 0.254 (25.4%). Precision, or positive predictive value   
TP / (TP + FN) = 4,454 / (4,454 + 324) = 0.932 (93.2%)   
TN / (TN + FP) = 82,150 / (82,150 + 13,072) = 0.862%, or specificity.   
Accuracy: (TP + TN) / (total samples) = (4,454 + 82,150) / 100,000 = 0.866 (86.6%)

With a high True Positive Rate (93.2%), the model shows itself to be rather good in spotting fraudulent transactions. The False Positive Rate of 13.7% suggests, nevertheless, that a considerable percentage of lawful transactions are wrongly identified as fraudulent. Practical applications are mostly concerned about this high probability of false positives since it can cause consumer discontent and pointless actions. Although the model's high AUC value—0.95—is encouraging—the study of the confusion matrix suggests the necessity of more improvement. Improving the whole dependability and customer satisfaction of the fraud detection system depends on balancing the True Positive Rate with the False Positive Rate.

**Advice**

**Threshold Correction:** Adjust the decision threshold to strike a better mix between recall and accuracy.   
Reevaluation of the important values of several parameters helps to guarantee a more balanced approach by means of which over-reliance on any one parameter may be minimized.

**Features Engineering:** Create further tools to help distinguish real from fraudulent transactions, hence lowering false positives. Aiming to improve the generalizing capacity of the model and lower bias, include more varied and balanced data to retrain it.

A screenshot of a computer

Description automatically generated

Picture: Overall score distribution of the model

**ROC Curve analysis**

The Receiver Operating Characteristic (ROC) curve offers a graphic depiction of how well the model distinguishes between legal and fraudulent transactions. Plotting the True Positive Rate (TPR) versus the False Positive Rate (FPR) at several threshold levels, the curve shows.

A screenshot of a computer

Description automatically generated

Picture: ROC Curve of the model

**Discussion**

The great dependence of the model on the billing city parameter raises issues since it results in unreliable forecasts. For instance, all transactions from billing cities called "AL" are pronounced valid; those from "AZ" are judged fraudulent. This implies that the model is not sufficiently evaluating other critical factors such as IP address, email address, and user agent, which are necessary for correctly spotting fraud.

A screenshot of a computer

Description automatically generated

Picture: Proof of over reliance on billing\_state

**Recommendations**

These steps are advised to raise the dependability and accuracy of the model:

Review the parameter importance to guarantee a more equitable evaluation of every criterion. Improve current features or design new ones combining several criteria to lessen dependence on a single characteristic. Re-training the model using a more balanced dataset helps to avoid any one parameter from controlling the predictions. Frequent audits and model testing will help to guarantee correct predictions and balanced parameter consumption.

References:  
Documentation for AWS Machine Learning   
Best Practices for Detection of Fraud

A screenshot of a computer

Description automatically generated

Picture: Thresholds for detection of fraud

A table of text with numbers

Description automatically generated with medium confidence

As mentioned previously, the model detected most of the information from AZ as fraud.

A table of numbers and letters

Description automatically generated with medium confidence

We can see that the same repetition exists when the model is parsing through the data of city of AL as well, as it is detecting most of the inputs as legitimate.

**Conclusion**

The present performance of an AWS fraud detection machine learning model is investigated in this analysis together with the over-reliance of this model on the billing city parameter for fraud prediction. This imbalance compromises the accuracy of the model even with its potential. Advice is given on re-prioritizing parameters to improve the dependability and precision of the model.