Case Study Assignment: Intelligent Analysis of IT Security Alerts

Overview

In today's cybersecurity environment, organizations receive thousands of IT security alerts daily. Not all alerts pose the same level of risk, making it crucial to quickly identify and prioritize those that require immediate attention. Your task is to develop an end-to-end data science solution that not only predicts the criticality of security alerts but also provides interpretable insights that can support high-stakes decision-making in a Security Operations Center (SOC).

In this case study, you will work with a IT Security Alerts dataset that includes a rich mix of categorical and numerical features (e.g., alert types, threat intelligence indicators, system metrics) and a target variable, **incident_priority**.

Dataset Description

The IT Security Alerts dataset is designed to mimic real-world operational data. It consists of approximately 8,000 alerts. Each alert record is constructed with a variety of features that span several domains:

1. Metadata:

- **alert_id**: A unique identifier for each alert (formatted like "ALERT_0000001").
- **timestamp**: The date and time when the alert.

2. Alert Characteristics:

- **alert_type**: The general category of the alert, such as *Phishing*, *Malware*, *Brute Force*, *DDoS*, or *Data Exfiltration*.

3. Threat Intelligence:

- **source_ip**: The originating IP address for the alert.
- **ip_reputation_score**: A numerical score (ranging roughly from 0 to 100) reflecting the trustworthiness of the source IP. Higher scores indicate better reputation.

4. Technical Metrics:

- **payload_size**: The size of the data payload involved in the alert.
- **cve_score**: A score derived from the Common Vulnerabilities and Exposures (CVE) framework, indicating the severity of a vulnerability.
- **login_attempts**: This feature represents the number of login attempts detected.

5. System & User Context:

- **user_role**: The role of the user associated with the alert, such as *Admin*, *Developer*, *Finance*, *External Contractor*, or *Support*. This variable can influence the risk level, as alerts involving privileged users may require more scrutiny.

- **system_context**: Describes the type of system or endpoint involved in the alert. For example, for *Brute Force* alerts, the context might be *Critical Server*, *Database*, or *API Gateway*; for others, it could be *End-User Laptop*, *IoT Device*, *Mobile Device*, or *Cloud Instance*.
- **geolocation**: The country associated with the source of the alert.

6. Operational Metrics:

- **cpu_usage_percent**: An estimate of CPU usage at the time of the alert.
- **memory_usage_percent**: Represents the memory usage at the time of the alert.

7. Target Variable:

- **incident_priority**: The final label assigned to each alert, representing its criticality. The labels are *Critical*, *High*, *Medium*, and *Low*.

Case Study Objectives

You are required to develop an end-to-end analytical solution that encompasses the following steps:

1. Exploratory Data Analysis (EDA) & Visual Storytelling

- **Objective:** Understand the dataset's structure, quality, and inherent patterns.
- **Tasks:**
    - Analyze missing values and data quality issues.
    - Visualize the distribution of incident priorities.
    - Explore relationships between alert types, system metrics, and priorities.
    - Create **6 to 8 key visualizations** with narrative explanations.

2. Feature Selection & Engineering

- **Objective:** Identify and select the most relevant features for predictive modeling.
- **Tasks:**
    - Evaluate features for missingness, outliers, and high-cardinality issues.
    - Apply transformations (e.g., one-hot encoding, standard scaling).
    - Optionally, use feature importance techniques to justify your selection.
    - Document your rationale for retaining or discarding features.

3. Clustering Analysis

- **Objective:** Discover natural groupings within the alert data.
- **Tasks:**
    - Use appropriate clustering techniques (e.g., K-Means, K-Prototypes, or DBSCAN) to examine if distinct clusters exist.
    - Describe the characteristics of the clusters (e.g., average payload size, typical alert types).

4. Supervised Machine Learning

- **Objective:** Predict incident priority using supervised models.
- **Tasks:**
    - Build and evaluate at least four classification models.
    - Address class imbalance with suitable techniques, if required.
    - Evaluate models using metrics such as precision, recall, F1-score, and confusion matrices.
    - Discuss model performance and any challenges encountered.

5. Explainable AI (XAI) Integration

- **Objective:** Make your model predictions transparent and interpretable.
- **Tasks:**
    - Integrate an XAI technique (e.g., SHAP or LIME) to provide global and local explanations of your model's decisions.
    - Generate visual outputs (e.g., SHAP summary and force plots).
    - Explain how these interpretations can assist SOC analysts in understanding and trusting the model outputs.

Deliverables

Your final submission should include:

1. **A Well-Commented Notebook or Report:**
    - A clean, organized Jupyter Notebook (or equivalent) that walks through your analysis step by step.
    - Clear markdown cells explaining each section of your work, including EDA insights, feature engineering, clustering results, model evaluations, and XAI interpretations.
2. **Key Visualizations & Narrative:**
    - At least 6 to 8 insightful graphs (as outlined above) with annotations and narratives explaining the operational implications.
    - A storytelling component that ties the visuals into actionable cybersecurity insights.
3. **Feature Selection & Clustering Insights:**
    - A discussion on your feature selection process and any feature engineering techniques used.
    - Clustering analysis and a description of cluster characteristics, if clusters are detected.
4. **Supervised ML Models:**
    - Implementation and evaluation of at least four models (e.g., Random Forest and XGBoost).
    - Detailed evaluation metrics and a discussion of class imbalance or other modeling challenges.

5. **Explainable AI (XAI) Analysis:**
    - Integration of an XAI tool (SHAP or LIME) with visual outputs that explain your model predictions both globally and locally.
    - A narrative on how these insights can be used in a high-stakes SOC environment.
6. **Final Report/Executive Summary:**
    - Summarize your findings, including insights from EDA, key features, clustering results, model performance, and interpretability.
    - Provide actionable recommendations for how security teams can leverage these insights to improve alert triaging and response strategies.

Evaluation Criteria

Your submission will be evaluated based on:

- **Analytical Depth:** How thoroughly you explore and understand the dataset.
- **Visual Storytelling:** Quality and clarity of the 6–8 key visualizations and the narrative connecting them.
- **Technical Rigor:** Effectiveness of feature selection, clustering analysis, and model building.
- **Interpretability:** How well you integrate and explain XAI outputs.
- **Communication:** Clarity of your narrative and practical relevance of your recommendations for a SOC.