Explainability of AIML Decisions

Arghyajit Debnath
Member of AI/ML Intern Team 74

Overview

This report highlights the implementation of explainability methods to interpret AI model predictions in the context of financial loss prediction from cybersecurity incidents. Techniques used include SHAP and LIME, which help improve transparency and trust in AI decisions.

Anomaly Detection

An Isolation Forest model was used to detect anomalies with 5% contamination.

Anomaly Detection Results

Normal : 2850
Anomaly : 150

Regression Model Comparison

Four models were compared based on MSE and R^2. Linear Regression was chosen as the best model.

Linear Regression: MSE=0.98, R2=-0.004

Random Forest : MSE=1.06, R2=-0.085

Gradient Boosting: MSE=1.00, R2=-0.021

XGBoost : MSE=1.27, R2=-0.301

Best Model : Linear Regression

3. SHAP Explainability

SHAP was used to evaluate global and local feature impact on model output.

4. LIME Explainability

LIME was applied to explain individual predictions using local feature contributions.

Saved Artifacts

- `best_regressor_model.joblib` — trained model
- `isolation_forest_model.joblib` — anomaly detection model
- `label_encoders.pkl` — saved encoders for categorical features
- `scaler.pkl` — saved StandardScaler for reproducibility

6) SHAP & LIME Visualizations

Commands + Code + Text ▷ Run all

```python
    return (name, model, mse, r2)

models = [
    ('Linear Regression', LinearRegression()),
    ('Random Forest', RandomForestRegressor(n_estimators=100, random_state=42)),
    ('Gradient Boosting', GradientBoostingRegressor(n_estimators=100, random_state=42)),
    ('XGBoost', XGBRegressor(n_estimators=100, random_state=42, verbosity=0))
]
results = [evaluate_model(n, m) for n, m in models]
results.sort(key=lambda x: x[2])
best_name, best_model, _, _ = results[0]
print(f"Best Model: {best_name}")
```

```
Linear Regression: MSE=0.98, R2=-0.004
Random Forest: MSE=1.06, R2=-0.085
Gradient Boosting: MSE=1.00, R2=-0.021
XGBoost: MSE=1.27, R2=-0.301
Best Model: Linear Regression
```

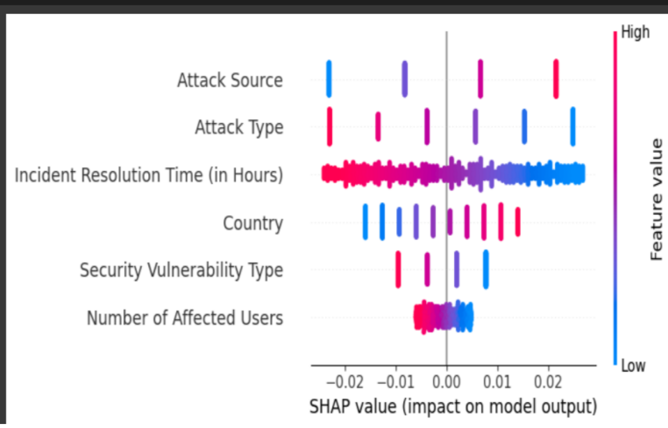
6 Explainability with SHAP

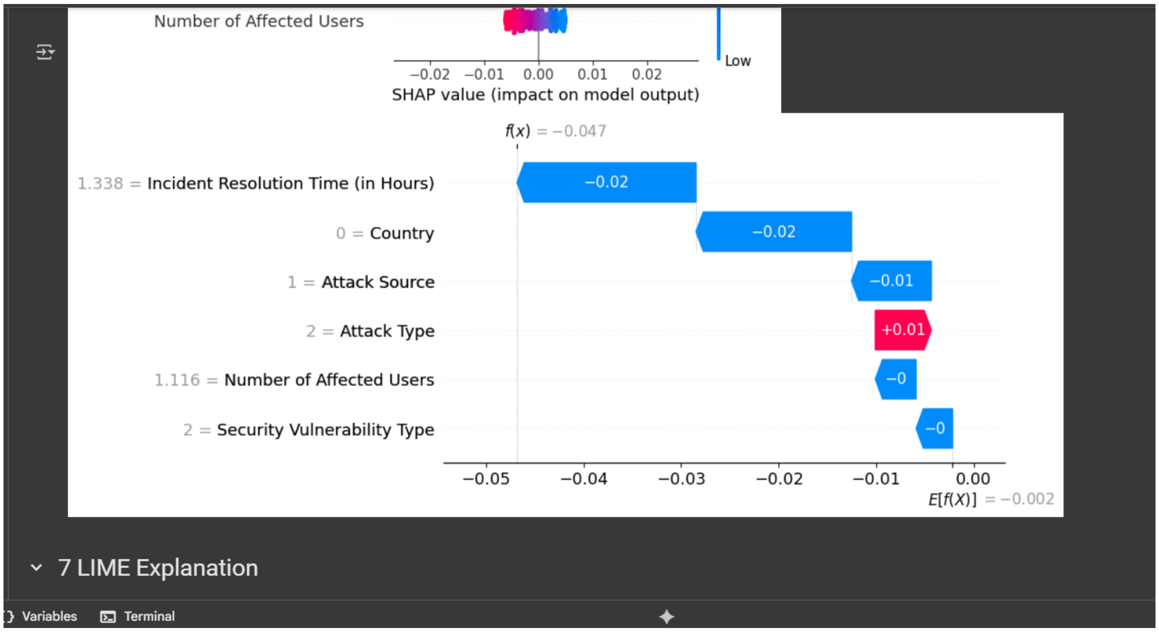
```python
[6] explainer = shap.Explainer(best_model, X_train)
    shap_values = explainer(X_test)
    shap.summary_plot(shap_values, X_test)
    shap.plots.waterfall(shap_values[0])
```

High

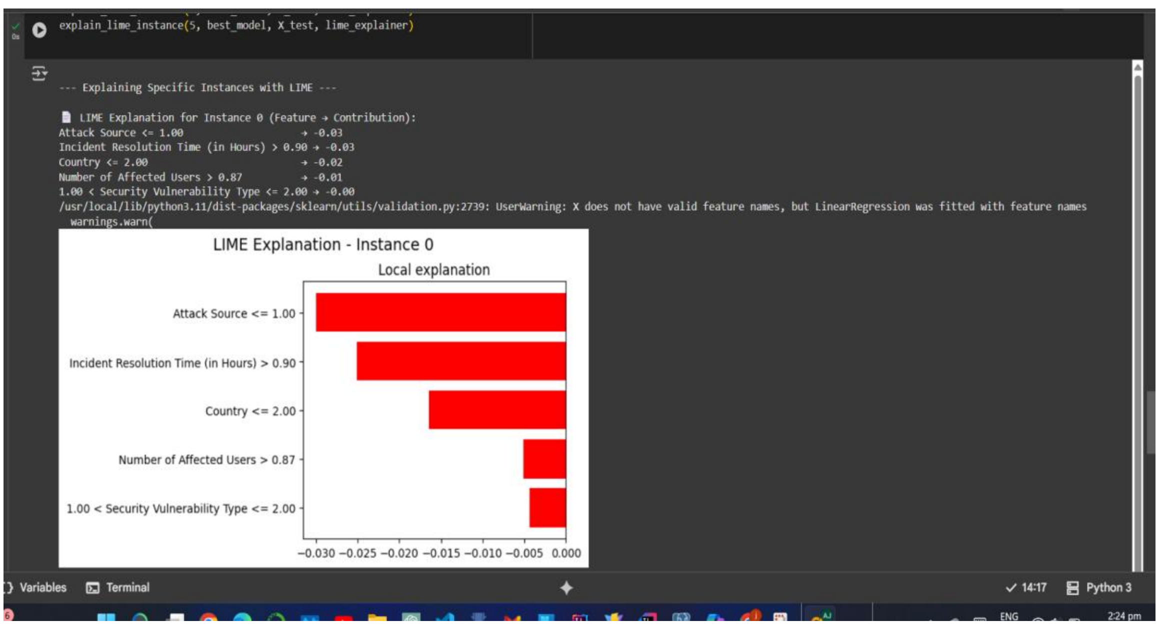
6 Explainability with SHAP

```python
explainer = shap.Explainer(best_model, X_train)
shap_values = explainer(X_test)
shap.summary_plot(shap_values, X_test)
shap.plots.waterfall(shap_values[0])
```





Number of Affected Users
Low
−0.02 −0.01 0.00 0.01 0.02
SHAP value (impact on model output)
f(x) = −0.047
1.338 = Incident Resolution Time (in Hours)
−0.02
0 = Country
−0.02
1 = Attack Source
−0.01
2 = Attack Type
+0.01
1.116 = Number of Affected Users
−0
2 = Security Vulnerability Type
−0
−0.05 −0.04 −0.03 −0.02 −0.01 0.00
E[f(X)] = −0.002
7 LIME Explanation
Variables
Terminal

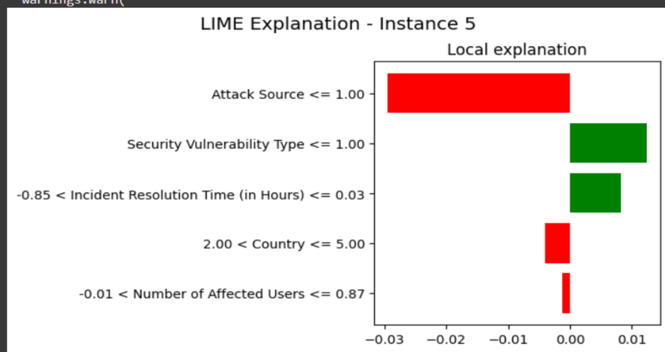


explain_lime_instance(5, best_model, X_test, lime_explainer)
--- Explaining Specific Instances with LIME ---
LIME Explanation for Instance 0 (Feature → Contribution):
Attack Source <= 1.00 → -0.03
Incident Resolution Time (in Hours) > 0.90 → -0.03
Country <= 2.00 → -0.02
Number of Affected Users > 0.87 → -0.01
1.00 < Security Vulnerability Type <= 2.00 → -0.00
/usr/local/lib/python3.11/dist-packages/sklearn/utils/validation.py:2739: UserWarning: X does not have valid feature names, but LinearRegression was fitted with feature names
warnings.warn(
LIME Explanation - Instance 0
Local explanation
Attack Source <= 1.00
Incident Resolution Time (in Hours) > 0.90
Country <= 2.00
Number of Affected Users > 0.87
1.00 < Security Vulnerability Type <= 2.00
−0.030 −0.025 −0.020 −0.015 −0.010 −0.005 0.000
Variables
Terminal
14:17
Python 3

−0.030 −0.025 −0.020 −0.015 −0.010 −0.005 0.000

```
🔷 Model Predicted Financial Loss: $-0.05 Million

📄 LIME Explanation for Instance 5 (Feature → Contribution):
Attack Source <= 1.00                        → -0.03
Security Vulnerability Type <= 1.00          → +0.01
-0.85 < Incident Resolution Time (in Hours) <= 0.03 → +0.01
2.00 < Country <= 5.00                       → -0.00
-0.01 < Number of Affected Users <= 0.87 → -0.00
/usr/local/lib/python3.11/dist-packages/sklearn/utils/validation.py:2739: UserWarning: X does not have valid feature names, but LinearRegression was fitted with feature names
  warnings.warn(
```




```
🔷 Model Predicted Financial Loss: $0.01 Million
```

} Variables Terminal ✓ 14:17 Pyth