

# Agentic Fraud Detection System

A Machine Learning Approach to Financial Security

İTÜ

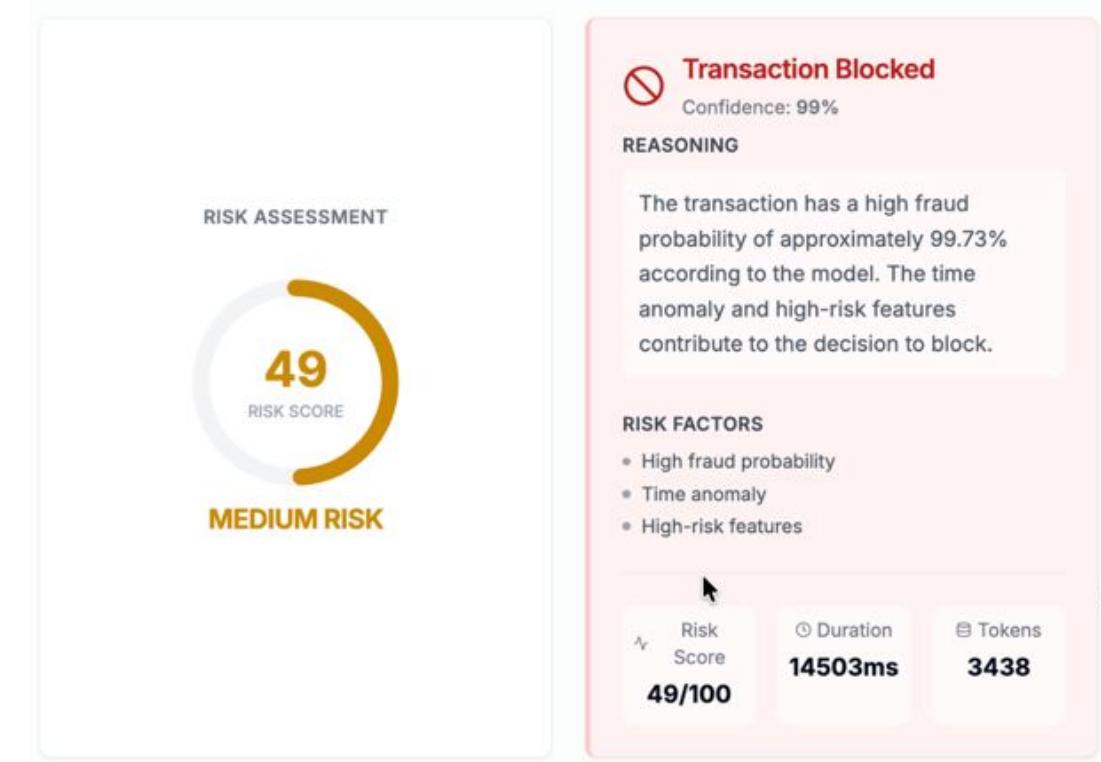


Term Project

Business Analytics for Managers

Students

Oğuzhan Kır & Fırat Ölçüm



# Presentation Outline

---

- 1 Introduction & Motivation
- 2 Exploratory Data Analysis
- 3 Feature Engineering
- 4 Model Development
- 5 Results & Performance
- 6 System Architecture
- 7 Business Impact
- 8 Deployment & Technical Stack
- 9 Conclusion & Future Work

# Introduction & Motivation

---

## The Challenge

- Financial fraud costs billions annually.
- Traditional systems lack adaptability and have high false positives.
- ML models need explainability for stakeholder trust.

## Our Solution

- **XGBoost Model:** 99.76% AUC on temporal test data.
- **Agentic Framework:** ReAct-based reasoning for transparency
- **Real-time:** <50ms end-to-end inference latency

# Exploratory Data Analysis

---

İTÜ



## Dataset Overview

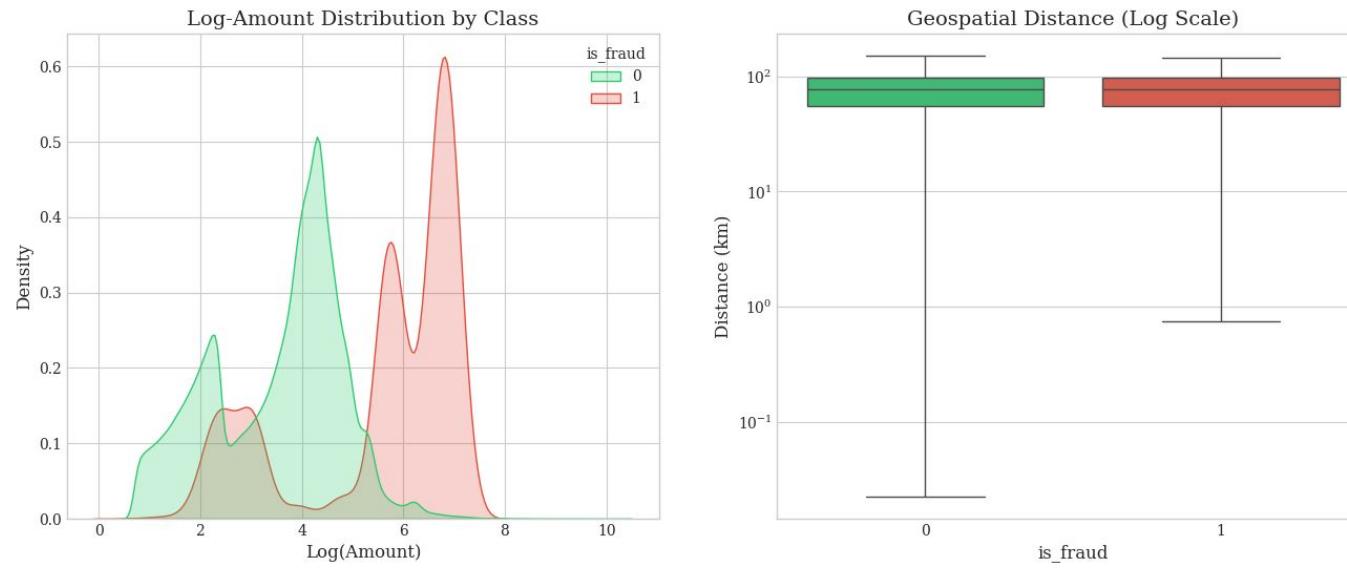
- 1,296,675 synthetic credit card transactions
- 7,506 fraudulent cases (0.58% prevalence)
- Temporal coverage: January 2019 – June 2020 (18 months)
- Features: Temporal, geospatial, transactional, demographic
- Designed to mirror real-world transaction distributions

# Exploratory Data Analysis

## Distributional Analysis

- Fraud shows bimodal distribution (low & high amounts).
- Log transformation improves class separation.
- Similar median distances but fraud shows higher variance.

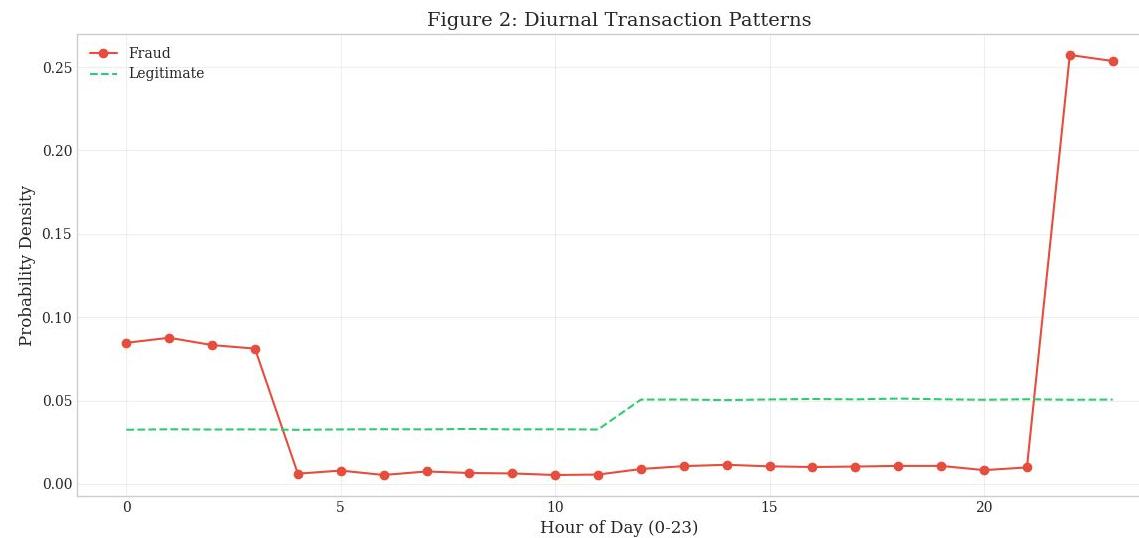
Figure 1: Distributional differences in Amount and Location



# Exploratory Data Analysis

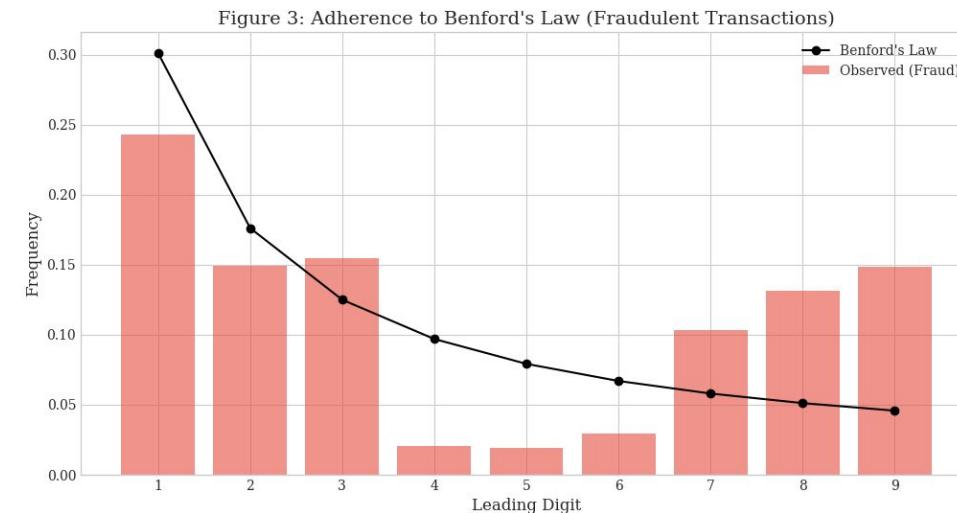
## Temporal Pattern Recognition

- 25%+ of fraud occurs in 21:00-23:00 window.
- Legitimate transactions uniform (3-5%) across day.
- Near-zero fraud during business hours (05:00-20:00).
- Created hour\_risk\_score feature with elevated weights.



## Benford's Law Analysis

- Benford's Law is that in natural datasets, leading digits follow logarithmic distribution. Deviation indicates fraud.
- Fraud shows strong deviation from expected.
- Over-representation of digits '1' (24%) and '9' (15%).
- Engineered benford\_log\_prob feature.



## Feature Engineering Strategy

- **Temporal Features:**

- Cyclical Encoding: hour\_sin, hour\_cos, day\_sin, day\_cos.

- **Geospatial Analytics:**

- Haversine formula for geodesic distance.

- **Financial Profiling:**

- log\_amt, relative amounts, Benford's Law probability.

- **Behavioral Aggregates:**

- Rolling windows (24h, 7d, 30d): transaction counts, amounts, frequencies.

# Model Development

---

## Model Training Strategy

- **Temporal Split (Forward Validation):**

**Training Set (80%):** Jan 2019 – Mar 2020 (1,037,340 lines of data).

**Test Set (20%):** Apr 2020 – Jun 2020 (259,335 lines of data).

**Prevents temporal leakage:** Model trained on historical data only.

- **Nested Cross-Validation:**

**Outer Loop (5-fold):** Performance estimation.

**Inner Loop (3-fold):** Hyperparameter optimization with Optuna.

100 trials per model using TPE sampler.

- **Candidate Models**

XGBoost, LightGBM, Random Forest.

# Results & Performance

---

## Test Performance Results

- **XGBoost:** Precision 93.7%, Recall 82.5%
- **Random Forest:** Precision 94.4%, Recall 77.4%
- **LightGBM:** Precision 26.8%, 51.7% Recall

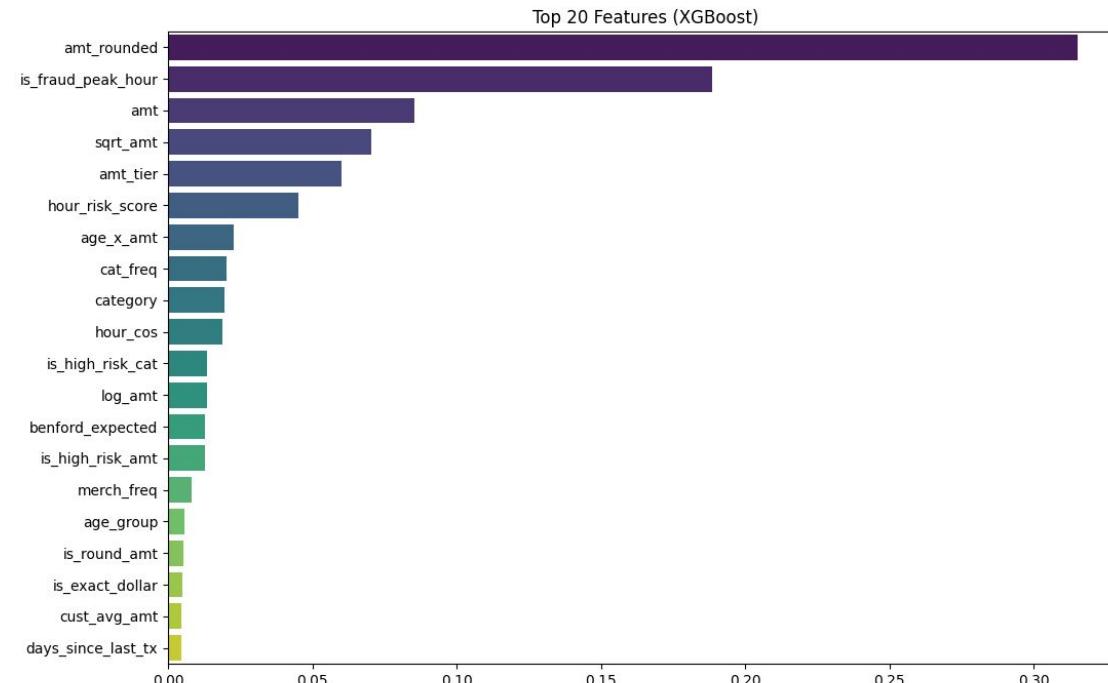
**Winner:** XGBoost with near-perfect class separation

Model	Recall	Precision	Real Cost
<b>XGBoost</b>	0.825098	0.937223	135350
<b>LightGBM</b>	0.517555	0.268647	392670
<b>RandomForest</b>	0.773732	0.943695	174710

# Model Development

## Top Features

- **amt\_rounded (0.30)** : Round amounts
- **is\_fraud\_peak\_hour (0.20)** : 21:00-23:00 flag
- **amt, sqrt\_amt, amt\_tier** : Amount variations
- **hour\_risk\_score (0.05)** : Temporal risk
- Amount features dominate top 6 positions



## Agent Architecture & ReAct Workflow

- **Multi-Agent Hierarchy:**

**Coordinator Agent** : Decomposes tasks, delegates to sub-agents, final decision.

**Data Agent** : Uses detect\_anomalies tool (Z-scores, Benford's Law, temporal risk).

**Model Agent** : Uses model\_predictor and calculate\_risk\_score tools.

- **ReAct Loop Implementation:**

**THOUGHT** : Agent analyzes current state and plans next action.

**ACTION** : Agent calls specific tool (consult\_data\_agent, consult\_model\_agent).

**OBSERVATION** : Agent receives tool results and updates reasoning.

- **Tools & Capabilities:**

**detect\_anomalies** : Statistical analysis, Benford's Law, temporal patterns.

**model\_predictor** : XGBoost inference via ONNX Runtime (5ms).

**calculate\_risk\_score** : Combines probability + anomalies → 0-100 score.

**get\_customer\_profile**: Historical spending patterns and baselines.

# System Architecture

Tought → Action → Observation

FraudGuard AI  
Agentic Analysis Dashboard

System Online

**Transaction Details**  
Enter details or generate random data

**CUSTOMER INFO**

First Name	Last Name
Travis	Daniel

Gender Date of Birth

M	1959-03-03
---	------------

Job City Population

Database administrator	14462
------------------------	-------

**PAYMENT DETAILS**

Amount	Merchant
868,09	fraud_Kuhn LLC

Category Date/Time

shopping_pos	2020-12-22 22:18:07
--------------	---------------------

Credit Card Number

2242176657877538
------------------

**LOCATION DATA**

Street	City	State
1327 Rose Causewa	Unknown	Unknown

Zip Code	Latitude	Longitude
00000	34,6323	-89,8855

Merchant Lat	Merchant Long	Transaction Num
34,091227	-90,390612	da7f67d7375f10a05

**RISK ASSESSMENT**

**49**  
MEDIUM RISK

**Transaction Blocked**  
Confidence: 99%

**REASONING**

The transaction has a high fraud probability of approximately 99.73% according to the model. The time anomaly and high-risk features contribute to the decision to block.

**RISK FACTORS**

- High fraud probability
- Time anomaly
- High-risk features

Risk Score: 49/100, Duration: 14503ms, Tokens: 3438

**Agent Reasoning Timeline**

Live Stream

1:34:37 PM

**THOUGHT** coordinator  
Tool: consult\_data\_agent

1:34:32 PM

**ACTION** coordinator  
Calling tool: consult\_data\_agent  
Tool: consult\_data\_agent

1:34:32 PM

**OBSERVATION** coordinator

1:34:37 PM

**OBSERVATION** coordinator  
Engineered Features:  
Transaction Date & Time: 2020-12-22 22:18:07  
Amount: \$868.09  
Merchant: fraud\_Kuhn LLC  
Category: shopping\_pos  
Latitude: 34.6323  
Longitude: -89.8855  
Merchant Latitude: 34.091227  
Merchant Longitude: -90.390612  
Date of Birth: 1959-03-03  
City Population: 14462  
Zip Code: 00000  
State: Unknown  
Job: Database administrator  
Gender: Male  
Anomaly Report:  
Amount Anomaly: No anomaly detected (0.00 standard deviations from customer baseline).  
Time Anomaly: Transaction during high-risk hours (Late night/Early morning) - Medium Severity  
Location Anomaly: Location consistent with home address  
Overall Risk Level: MEDIUM  
The transaction is flagged for a time anomaly as it occurred during high-risk hours. The amount and location do not show anomalies. It is important to monitor transactions during late night/early morning hours for potential fraud risk.

1:34:40 PM

**THOUGHT** coordinator  
Tool: consult\_model\_agent

1:34:40 PM

**ACTION** coordinator  
Calling tool: consult\_model\_agent  
Tool: consult\_model\_agent

1:34:43 PM

**OBSERVATION** coordinator  
The model predicts a high fraud probability of approximately 99.73% for the transaction at hand, categorizing it as a MEDIUM risk. The risk score assigned is 49, with the model contributing the entire score.  
Key contributing features to the high fraud probability include the transaction amount, merchant name, transaction category, location details (latitude and longitude), city population, job title, date of birth, and gender. These features collectively indicate a high likelihood of fraud for this

FraudGuard AI Dashboard: Transaction details, risk assessment, and decision

## Cost-Benefit Analysis

- Baseline (no model): \$1,501,200 in fraud losses.
- XGBoost Model: \$908,220 net benefit.
- ROI: 60.5% reduction in fraud-related costs.

## Operational Metrics

- 6,194 fraud cases detected and prevented.
- Only 416 false positives (6.3% false alarm rate).
- 1,312 fraud cases missed (17.5%).

## Additional Benefits

- Real-time prevention at point-of-sale.
- Explainable decisions for regulatory compliance.
- Continuous improvement through analyst feedback.

# Deployment & Technical Stack

---

## Backend Stack

### FastAPI

Asynchronous Python web framework

### LangChain

Agent orchestration & LLM integration

### ONNX Runtime

Optimized model inference (5ms latency)

### WebSocket

Real-time streaming to frontend

## Frontend Stack

### Next.js

React 18 + TypeScript framework

### CSS

Tailwind CSS + shadcn/ui components

### RealTime

Real-time React timeline visualization

## Docker Deployment

### Compose

Multi-container setup with Docker Compose

### Backend

Python 3.11-slim (512MB memory limit)

### Frontend

Node 20-alpine (256MB memory limit)

### Monitoring

Production ready with health monitoring

# Conclusion & Future Work

---

## Key Achievements

- State-of-the-art performance: ROC-AUC 0.9976.
- Production ready: <50ms latency, containerized deployment.
- Explainable AI: Multi-agent architecture bridges accuracy and trust.
- Business impact: \$908K annual savings (60.5% ROI).

## Future Enhancements

- Federated Learning: Train on decentralized data while preserving privacy.
- Graph Neural Networks: Detect collusion rings and mule accounts.
- Reinforcement Learning: Optimize decision thresholds based on outcomes.
- Online Learning: Continuous adaptation to emerging fraud tactics.

## Recommendations

- Deploy XGBoost with agentic explainability as primary system.
- Implement monthly retraining with sliding temporal window.

# References

---

- [1] Dataset:** Cartella, F. (2018). Simulated Credit Card Transactions Dataset. Kaggle.  
<https://www.kaggle.com/datasets/kartik2112/fraud-detection>
- [2] XGBoost:** Chen, T., & Guestrin, C. (2016). XGBoost: A Scalable Tree Boosting System. Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.
- [3] Benford's Law:** Durtschi, C., Hillison, W., & Pacini, C. (2004). The Effective Use of Benford's Law to Assist in Detecting Fraud in Accounting Data. *Journal of Forensic Accounting*, 5(1), 17-34.
- [4] Imbalanced Learning:** He, H., & Garcia, E. A. (2009). Learning from Imbalanced Data. *IEEE Transactions on Knowledge and Data Engineering*, 21(9), 1263-1284.
- [5] ReAct Framework:** Yao, S., et al. (2023). ReAct: Synergizing Reasoning and Acting in Language Models. International Conference on Learning Representations.
- [6] Temporal Validation:** Cerqueira, V., Torgo, L., & Mozetič, I. (2020). Evaluating Time Series Forecasting Models: An Empirical Study on Performance Estimation Methods. *Machine Learning*, 109(11), 1997-2028.
- [7] Explainable AI:** Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). "Why Should I Trust You?": Explaining the Predictions of Any Classifier. Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.
- [8] Feature Engineering:** Zheng, A., & Casari, A. (2018). Feature Engineering for Machine Learning: Principles and Techniques for Data Scientists. O'Reilly Media.