Title Page

# Table of Contents

|  |  |
| --- | --- |
| 1. Introduction |  |
| 1.1 Problem Statement |  |
| 1.2 Objective |  |
| 1. CRISP-DM Methodology |  |
| 2.1 Framework Application |  |
| 1. Business Understanding |  |
| 3.1 Business Context & Objectives |  |
| 3.2 Project Aim & Scope |  |
| 3.3 Project Plan |  |
| 3.4 Requirements & Constraints |  |
| 3.5 Project Risks |  |
| * 1. Tools & Techniques |  |
| * 1. Proposed Git Structure |  |
| 1. Data Understanding |  |
| 4.1 Data Collection |  |
| 4.2 Primary Data Analysis |  |
| * 1. Exploratory Data Analysis |  |
| 1. Data Preparation |  |
| 5.1 Feature Engineering |  |
| * 1. Data Transformation |  |
| 1. Modeling |  |
| 6.1 Model Design |  |
| * 1. Building the Model |  |
| 1. Evaluation |  |
| 7.1 Discussion of Results |  |
| 7.2 Assessment Against Business Objectives |  |
| 1. Deployment |  |
| 8.1 Implementation Strategy |  |
| 8.2 Business Dashboard Interface (GUI Mockup) |  |
| 1. Conclusion |  |
| 1. References |  |
| 1. Appendix |  |

# List of Figures

|  |  |  |
| --- | --- | --- |
| **Number** | **Title** | **Page Number** |
| Figure 1 | CRISP-DM Methodology |  |
| Figure 2 | Success Fees by PSP |  |
| Figure 3 | Failure Fees by PSP |  |
| Figure 4 | Business Stakeholder Dashboard |  |
| Figure 5 | Baseline Model Comparison |  |
| Figure 6 | Predictive Model Comparison |  |
| Figure 7 | Feature Importance Analysis |  |
| Figure 8 | Comprehensive Business Evaluation |  |
| Figure 9 | PSP Routing Optimization Dashboard |  |

# List of Abbreviations

PSP: Payment Service Provider

DACH: Deutchland, Austria, Switzerland

CRISP-DM: Cross-Industry Standard Process for Data Mining

# 1. Introduction

In today's digital economy, online payment processing represents a critical component of e-commerce operations. However, many retail companies face significant challenges with high failure rates in credit card transactions, leading to revenue loss and customer dissatisfaction. This case study addresses the challenge of optimizing Payment Service Provider (PSP) routing through predictive modeling.

## 1.1 Problem Statement

The subject company, one of the world's largest retail organizations, faces a business problem: a high failure rate of online credit card payments that has resulted in revenue losses and declining customer satisfaction.

Currently, transactions are processed through four different Payment Service Providers (PSPs) using a manual, rule-based routing system. This approach has proven inefficient, with an overall success rate of only 20.3%, meaning nearly 80% of attempted transactions fail. The analysis of 50,410 transactions from January-February 2019 in the DACH region confirms this challenge. The financial implications are severe - not only is potential revenue lost from failed transactions, but processing fees are also incurred for both successful and failed attempts.

The manual routing system lacks to consider multiple factors that influence transaction success, such as PSP performance by country, card type, transaction amount, or customer authentication status. This approach fails optimize the routing decisions in real-time.

## 1.2 Objective

The primary objective of this case study is to develop a data-driven solution for automating credit card routing through predictive modeling. The following aims are established:

* Increase Payment Success Rate: Improve the overall transaction success rate by at least 10% through PSP routing
* Minimize Transaction Costs: Optimize PSP selection to reduce overall transaction fees while maintaining or improving success rates
* Enhance Customer Experience: Reduce failed transactions to improve customer satisfaction and retention
* Provide Business Intelligence: Deliver insights and monitoring capabilities for ongoing optimization

This case study demonstrates the practical application of machine learning techniques to solve a business problem, showcasing the end-to-end process from problem identification to deployment recommendation.

# 2. CRISP-DM Methodology

The Cross-Industry Standard Process for Data Mining (CRISP-DM) provides a structured approach to data science projects. This methodology ensures systematic progression from business understanding through deployment, with iterative refinement at each stage (Figure1).

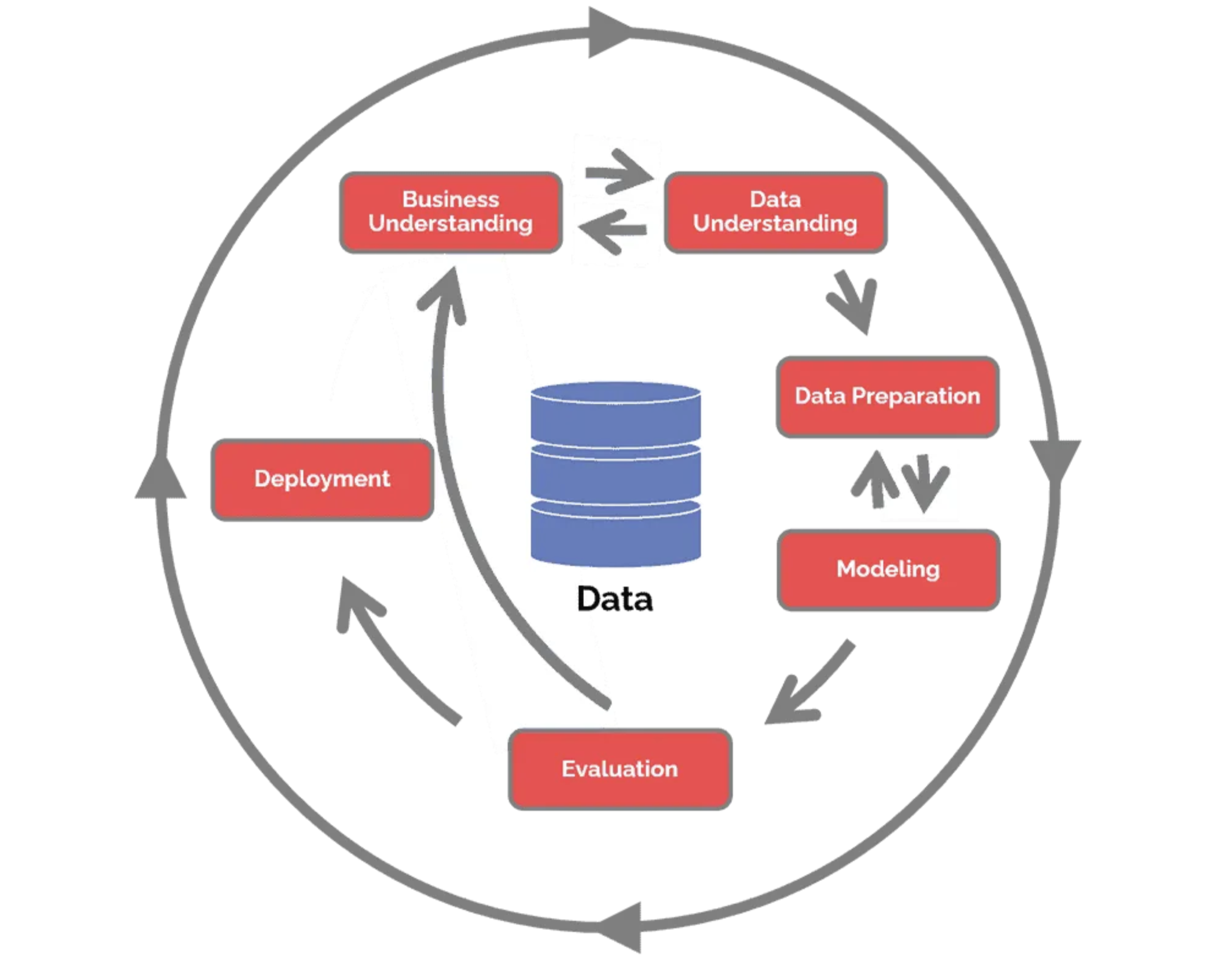


Figure 1: CRISP-DM Methodology

## 2.1 Framework Application

For this credit card routing optimization project, the CRISP-DM framework was adapted as follows:

Phase 1: Business Understanding

* Payment department stakeholders were consulted to understand the business problem
* Current PSP fee structures and performance metrics were analyzed
* Success criteria and business constraints were defined
* Potential return on investment was calculated

Phase 2: Data Understanding

* Transaction data from January-February 2019 for DACH countries was acquired
* Data quality assessment was performed
* Exploratory data analysis was conducted to identify patterns and relationships
* Business stakeholder visualizations were created for better understanding

Phase 3: Data Preparation

* Features related to temporal patterns, transaction characteristics, and PSP performance were engineered
* Multiple payment attempts were handled through grouping logic
* Derived variables for model input were created
* Data was split into training and testing sets

Phase 4: Modeling

* Baseline models using rule-based approaches were developed
* Machine learning models including Logistic Regression, Random Forest, Gradient Boosting, and XGBoost were implemented

Phase 5: Evaluation

* Model performance was compared using business-relevant metrics
* Feature importance analysis for model interpretability was conducted
* Error analysis was performed
* Results were validated against business requirements

Phase 6: Deployment

* Implementation architecture and timeline were designed
* Business dashboard mockups were created
* Return on investment and business impact were calculated

This structured approach made sure the technical solutions met business needs and kept the project scientifically accurate from start to finish.

# 3. Business Understanding

## 3.1 Business Context & Objectives

The company operates in a competitive retail environment where payment processing efficiency directly impacts revenue and customer satisfaction. The company operates with four PSPs, each having different cost structures for success (Figure 2) and failure (Figure 3). Analysis of 50,410 transactions from January-February 2019 in the DACH region reveals significant optimization opportunities.

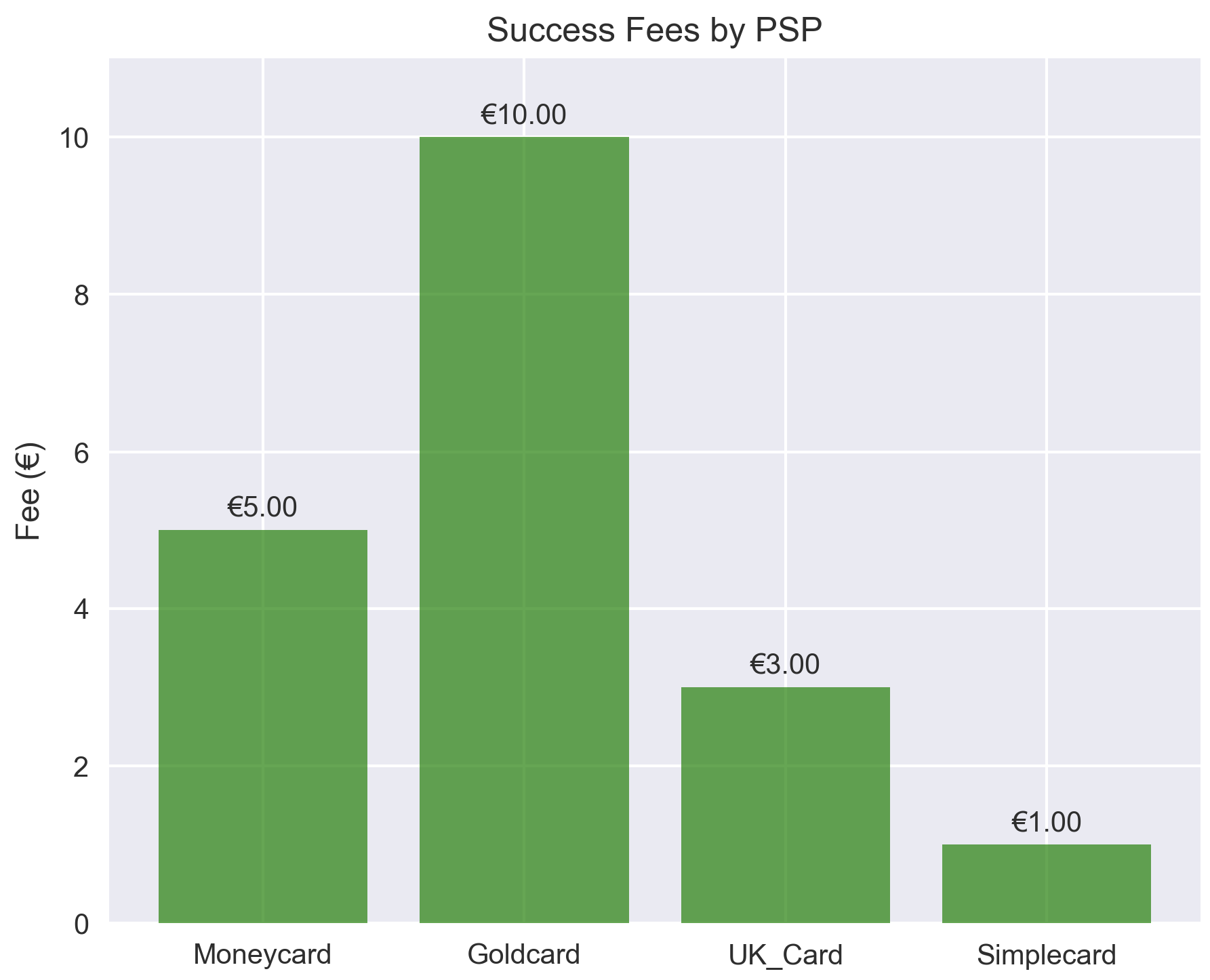


Figure 2: Success Fees by PSP

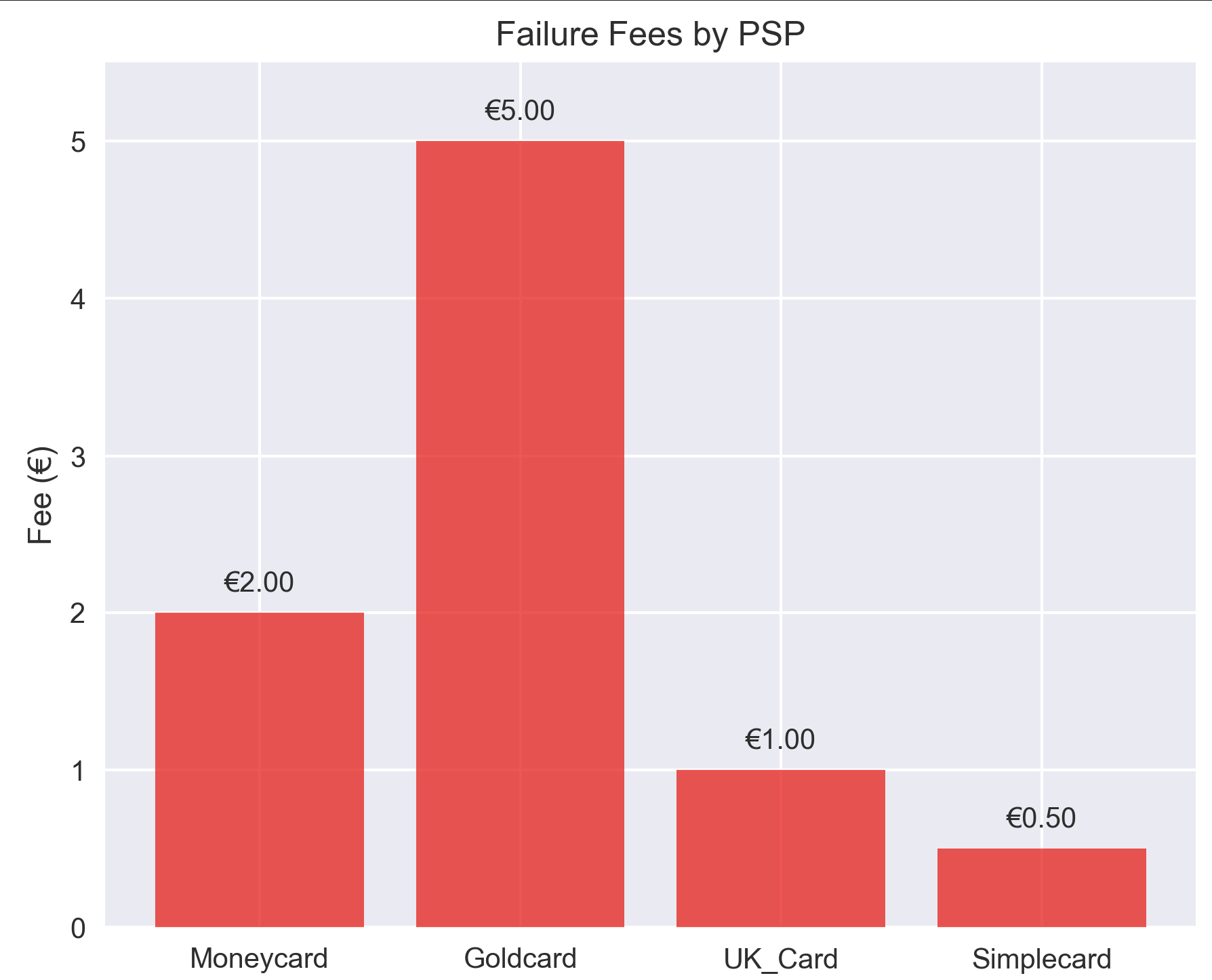


Figure 3: Failure Fees by PSP

Current State:

* 20.3% overall success rate with manual PSP routing
* Significant performance variation across PSPs: Goldcard (40.6% success rate), Simplecard (22.1%), UK\_Card (19.4%), Moneycard (16.8%)
* Volume-performance mismatch: UK\_Card processes 52.5% of transactions but achieves only 19.4% success rate

PSP Cost Analysis: Transaction fees represent a significant cost component that varies by PSP and outcome:

* Current Annual Fees: €132,894 based on existing routing patterns
* Optimized Annual Fees: €82,750 with intelligent routing (€50,144 potential savings)
* Cost Structure: Success fees range from €1.00 (Simplecard) to €10.00 (Goldcard); failure fees range from €0.50 to €5.00

Business Objectives:

* Increase Payment Success Rate: Improve overall transaction success rate by at least 10%
* Minimize Transaction Costs: Optimize PSP selection to reduce fees while maintaining performance
* Automate Routing Decisions: Replace manual rule-based system with data-driven approach

## 3.2 Project Aim & Scope

The aim of this project is to create a predictive model that automates the routing of credit card transactions to the optimal PSP, thereby increasing the success rate of online payments and reducing associated costs.

Success Metrics:

* Primary: Payment success rate improvement ≥10%
* Secondary: Transaction cost reduction ≥15%
* Tertiary: Model accuracy ≥80%
* Business Impact: Annual savings target

## 3.3 Project Plan

Timeline: 20 weeks total

Phase 1: Discovery & Analysis (Weeks 1-6)

* Business requirements gathering
* Data acquisition and quality assessment
* Exploratory data analysis
* Stakeholder presentation of findings

Phase 2: Model Development (Weeks 7-14)

* Feature engineering and data preparation
* Baseline model development
* Advanced model training and optimization
* Model validation and comparison

Phase 3: Implementation Planning (Weeks 15-18)

* Deployment architecture design
* Integration planning with existing systems
* Performance testing and optimization
* User acceptance testing

Phase 4: Deployment & Monitoring (Weeks 19-20)

* Production deployment
* Monitoring system implementation
* User training and documentation
* Go-live support

## 3.4 Requirements & Constraints

Functional Requirements:

* Real-time routing decisions (< 100ms response time)
* Process 100% of credit card transactions
* Maintain audit trail for regulatory compliance
* Provide business dashboard for monitoring
* Support A/B testing for continuous improvement

Non-Functional Requirements:

* Scalable to handle peak traffic loads
* Secure handling of sensitive payment data
* Integration with existing payment gateway

Business Constraints:

* No modification to existing PSP contracts
* Fallback to current system if model fails
* Cost-effective implementation approach
* Regulatory compliance in all DACH countries

## 3.5 Project Risks

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Risk Category** | **Risk Description** | **Impact Level** | **Probability** | **Mitigation Strategy** |
| Technical | Data Quality Issues - Incomplete or inconsistent transaction data | High | Medium | Comprehensive data validation and cleaning procedures |
| Integration Complexity - Difficulty integrating with existing systems | Medium | Medium | API-based architecture and phased deployment approach |
| Business | Regulatory Compliance - Payment data privacy and security requirements | High | Low | Adherence to GDPR and PCI-DSS standards |
| PSP Relationship - Changes in PSP contracts or fee structures | Medium | Low | Flexible model architecture to accommodate changes |
| Operational | Model Drift - Performance degradation over time | Medium | High | Continuous monitoring and automated retraining pipeline |
| System Downtime - Prediction service unavailability | High | Low | Fallback to rule-based routing and redundant infrastructure |
| Scalability Issues - Increased transaction volume beyond capacity | Medium | Medium | Scalable cloud-based deployment architecture |

## Tools & Techniques

1. Programming Languages & Libraries:

* Python: Primary development language
* Pandas: Data manipulation and analysis
* NumPy: Numerical computing
* Scikit-learn: Machine learning algorithms
* XGBoost: Gradient boosting implementation
* Matplotlib/Seaborn: Data visualization
* Plotly: Interactive visualizations

1. Machine Learning Techniques:

* Random Forest: Ensemble learning for robust predictions
* XGBoost: Gradient boosting for high performance
* Logistic Regression: Linear model for interpretability

1. Development Environment:

* Jupyter Notebook: Interactive development and analysis
* Git: Version control and collaboration
* Anaconda: Package management and environment setup
* Docker: Containerization for deployment consistency

1. Deployment Tools:

* Flask/FastAPI: REST API development
* Docker: Application containerization
* Cloud Services: Scalable deployment infrastructure
* Monitoring Tools: Performance tracking and alerting

## Proposed Git Structure

The following structure supports collaborative development, maintains code quality, and facilitates deployment and maintenance

1. credit-card-routing/
2. ├── README.md
3. ├── requirements.txt
4. ├── .gitignore
5. ├── config/
6. │ ├── development.yaml
7. │ ├── production.yaml
8. │ └── model\_config.yaml
9. ├── data/
10. │ ├── raw/
11. │ ├── processed/
12. │ └── external/
13. ├── notebooks/
14. │ ├── 01\_exploratory\_analysis.ipynb
15. │ ├── 02\_feature\_engineering.ipynb
16. │ ├── 03\_model\_development.ipynb
17. │ └── 04\_model\_evaluation.ipynb
18. ├── src/
19. │ ├── data/
20. │ │ ├── \_\_init\_\_.py
21. │ │ ├── ingestion.py
22. │ │ └── preprocessing.py
23. │ ├── features/
24. │ │ ├── \_\_init\_\_.py
25. │ │ └── engineering.py
26. │ ├── models/
27. │ │ ├── \_\_init\_\_.py
28. │ │ ├── baseline.py
29. │ │ ├── ml\_models.py
30. │ │ └── ensemble.py
31. │ ├── evaluation/
32. │ │ ├── \_\_init\_\_.py
33. │ │ └── metrics.py
34. │ └── api/
35. │ ├── \_\_init\_\_.py
36. │ ├── routing.py
37. │ └── monitoring.py
38. ├── tests/
39. │ ├── unit/
40. │ ├── integration/
41. │ └── performance/
42. ├── deployment/
43. │ ├── docker/
44. │ ├── kubernetes/
45. │ └── terraform/
46. └── docs/
47. ├── api/
48. ├── model/
49. └── deployment/

# 4. Data Understanding

## 4.1 Data Collection

The dataset consists of credit card transaction records from January and February 2019, covering the DACH region (Germany, Austria, Switzerland). The data was provided in Excel format containing 50,410 individual transaction records across all four contracted PSPs.

* Dataset: PSP\_Jan\_Feb\_2019.xlsx
* Records: 50,410 transactions
* Period: January-February 2019
* Geography: DACH countries (Germany, Austria, Switzerland)

## 4.2 Primary Data Analysis

The data contains 8 columns providing transaction information:

|  |  |  |  |
| --- | --- | --- | --- |
| **Column** | **Data Type** | **Description** | **Unique Values** |
| tmsp | datetime64 | Transaction timestamp | 50,121 |
| country | object | Transaction country | 3 |
| amount | int64 | Transaction amount (€) | 517 |
| success | int64 | Transaction outcome (0/1) | 2 |
| PSP | object | Payment Service Provider | 4 |
| 3D\_secured | int64 | 3D Authentication status | 2 |
| card | object | credit card brand | 3 |

Data Completeness:

* No missing values identified across all columns
* All transactions include complete temporal, geographic, and payment information
* Consistent data formatting across the entire dataset

## 4.3 Exploratory Data Analysis

PSP Performance Analysis:

* Best Performing PSP: Goldcard (40.6% success rate, 6.4% transaction volume)
* Volume Leader: UK\_Card (52.5% of transactions but only 19.4% success rate)
* Transaction Volume Distribution: UK\_Card 52.5%, Moneycard 22.3%, Simplecard 18.8%, Goldcard 6.4%
* Volume vs Performance Mismatch: UK\_Card processes majority of transactions (52.5%) but has poor performance (19.4%)
* Cost-Performance Trade-off: Goldcard offers best performance but highest cost (€10.00 per transaction)

Geographic Performance:

* Best Performing Country: Switzerland
* Country Coverage: Three countries analyzed (Germany, Austria, Switzerland)
* Geographic Variation: Success rates vary by country, suggesting regional factors influence performance

Multiple Payment Attempts

* Analysis Retry Analysis: 45.8% of transactions are potential retry attempts (23,073 out of 50,410 transactions)
* Initial vs Retry Performance: First attempt success rate: 20.7%, Retry attempt success rate: 19.8%
* Retry Identification Method: Transactions within 1-minute windows from same country/amount analyzed

Authentication Impact:

* 3D Secured vs Non-3D Secured: Analysis shows different performance patterns
* Security Feature: 3D authentication status tracked for all transactions

Temporal Analysis:

* Time Period: Two-month analysis covering January-February 2019
* Daily Patterns: Transaction volume and success rates analyzed by day
* Hourly Patterns: Success rates examined across different hours of the day

Transaction Characteristics:

* Amount Distribution: 517 unique transaction amounts in the dataset
* Volume Distribution: Transaction volume varies across PSPs and countries
* Card Type Coverage: Three credit card brands represented in the data

Critical Findings:

* Volume-Performance Mismatch: UK\_Card handles 52.5% of transactions but achieves only 19.4% success rate
* Underutilized High Performer: Goldcard achieves 40.6% success rate but processes only 6.4% of volume
* Retry Burden: 45.8% of all transactions are retry attempts, indicating system reliability issues
* Cost-Performance Trade-offs: Best performing PSPs (Goldcard) have highest transaction costs

All these findings are visualized in Figure 4.

# 5. Data Preparation

New features (temporal, amount, PSP, retry attempts) were created and the data was split into training and test set.

## 5.1 Feature Engineering

A total of 27 features were engineered which are summarized as below:



Figure 4 Business Stakeholder Dashboard

|  |  |  |
| --- | --- | --- |
| **Feature Category** | **Feature Count** | **Examples** |
| Temporal | 7 features | hour, day\_of\_week, day\_of\_month, month, is\_business\_hours, is\_weekend, is\_peak\_hours |
| Amount-based | 4 features | amount\_log, amount\_quartile, is\_high\_value, is\_micro\_transaction |
| PSP-related | 6 features | psp\_success\_fee, psp\_failure\_fee, psp\_fee\_difference, psp\_historical\_success\_rate, expected\_cost, psp\_cost\_tier |
| Retry Detection | 3 features | is\_potential\_retry, attempt\_number, minutes\_since\_last\_attempt |
| Interactions | 4 features | country\_psp\_combo, card\_psp\_combo, secured\_high\_value, amount\_quartile\_psp |
| Business Impact | 3 features | actual\_cost, revenue\_impact, net\_profit |

* Total Engineered Features: 27 features (plus original 8 columns)
* Final Dataset Shape: (50,410, 35) including all original and engineered features
* Modeling Features Used: 18 selected features for final model training
* Training set: 40,328 samples
* Test set: 10,082 samples
* Final dataset shape: (50,410, 28) including original and engineered features

## 5.2 Data Transformation

* Label encoding was applied to categorical features.
* country: Encoded Germany, Austria, Switzerland as numerical values
* PSP: Encoded Goldcard, Simplecard, UK\_Card, Moneycard as numerical values
* card: Encoded Master, Visa, Diners as numerical values
* amount\_quartile: Categorical amount buckets encoded numerically

Data Splitting Strategy:

* Training Set: 40,328 samples (80% of data)
* Test Set: 10,082 samples (20% of data)
* Stratified split: Maintained class balance using stratify=y
* Random state: 42 for reproducible results

The data preparation process successfully created a clean, feature-rich dataset ready for machine learning model training and evaluation.

# 6. Modeling

The modeling approach focused on developing predictive models to optimize PSP routing decisions.

## 6.1 Model Design

A two-tier modeling approach was implemented:

Tier 1: Baseline Models – Four different rule-based strategies were developed to establish performance benchmarks:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Strategy** | **Decision Rules** | **PSPs Used** | **Thresholds** | **Business Logic** | **Expected Outcome** |
| **1. Always Best** | Route all transactions to highest success rate PSP | Goldcard (40.6% success) | None | Simple success-rate maximization; conservative approach | Highest success rates, high transaction costs |
| **2. Cost-Country** | 1. Amount < €100 → Cheapest PSP 2. Country-specific best PSP 3. Fallback to best PSP | Simplecard (€1) Country-specific Goldcard (fallback) | €100 for cost optimization | Hierarchical: Cost → Geography → Success | Reduced costs for small transactions, maintained performance for large |
| **3. Risk-Based** | 1. >€500 OR Diners → Best PSP 2. €200-500 → Second-best PSP 3. <€200 → Cheapest PSP | Goldcard (high-risk) Moneycard (medium-risk) Simplecard (low-risk) | €200, €500 Diners card detection | 3-tier risk assessment with gradual cost-performance tradeoff | Balanced performance with risk-appropriate allocation |
| **4. Hybrid Value** | 1. Top 20% amounts → Best PSP 2. Bottom 80% → Cheapest PSP | Goldcard (high-value) Simplecard (low-value) | 80th percentile (dynamic) | Binary high-stakes vs. low-stakes optimization | Clear cost savings for majority, protection for high-value |

Tier 2: Machine Learning Models - Four machine learning algorithms were implemented and compared:

* Logistic Regression
* Random Forest Classifier
* Gradient Boosting Classifier
* XGBoost Classifier

## 6.2 Building the Model

The comparison of Baseline Modeling with all 4 strategies is summarized in Figure 5 below.

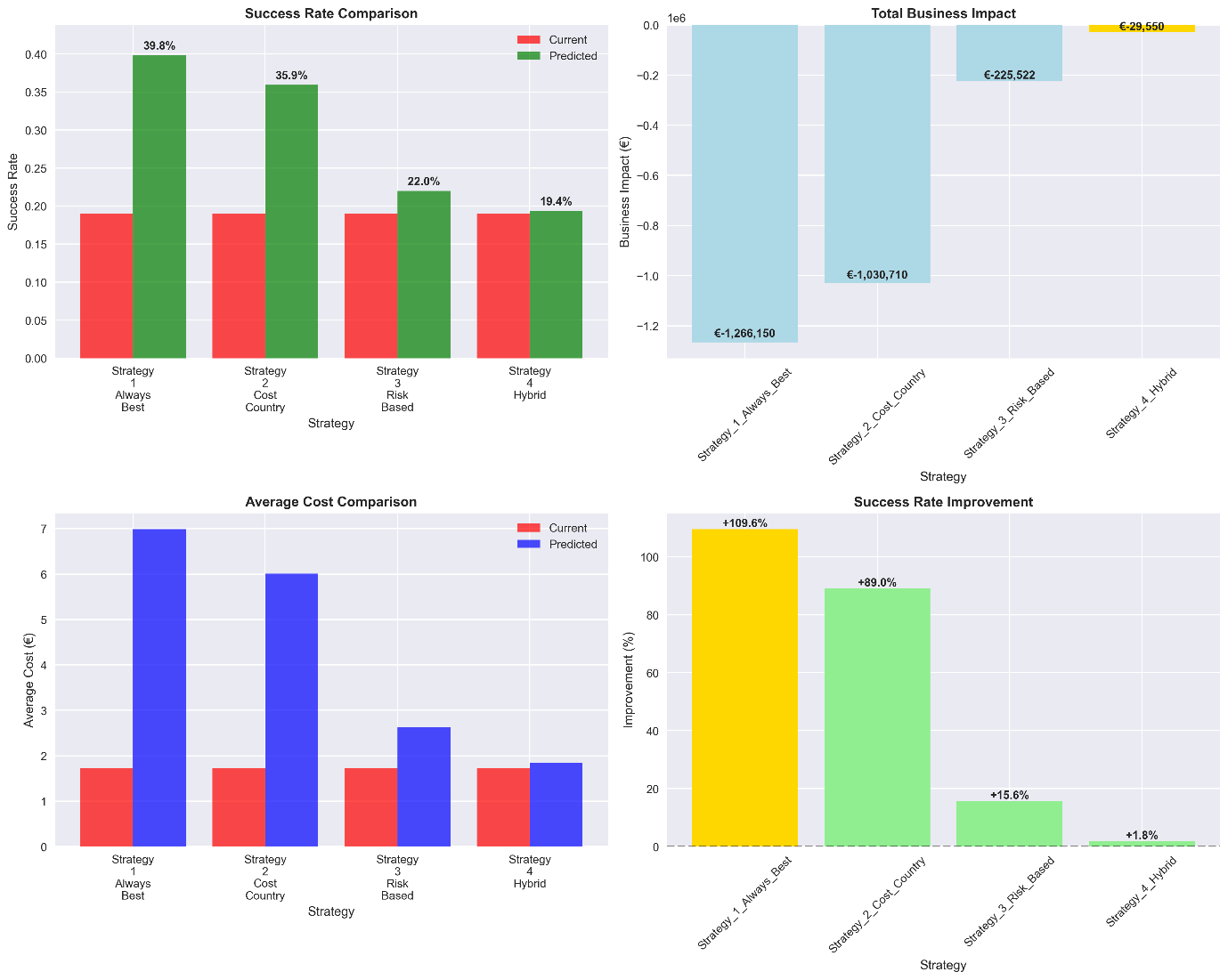


Figure 5: Baseline Model Comparison

For Predictive Modeling, the four machine learning models were developed and evaluated:

1. Logistic Regression:

* Performance: 79.8% accuracy, 0.617 AUC score
* Characteristics: Linear model providing interpretable coefficients

2. Random Forest Classifier:

* Performance: 78.9% accuracy, 0.660 AUC score
* Characteristics: Ensemble method with good feature importance insights

3. Gradient Boosting Classifier:

* Performance: 80.1% accuracy, 0.671 AUC score
* Characteristics: Best performing model with sequential learning approach

4. XGBoost Classifier:

* Performance: 79.3% accuracy, 0.657 AUC score
* Characteristics: Advanced gradient boosting implementation

Model Evaluation Approach: The models were evaluated using standard classification metrics:

* Accuracy: Overall prediction correctness
* AUC Score: Area under the ROC curve for ranking quality
* Cross-Validation: 5-fold cross-validation for robustness assessment

A summary of the machine learning models evaluation is visualized in Figure 6 below.

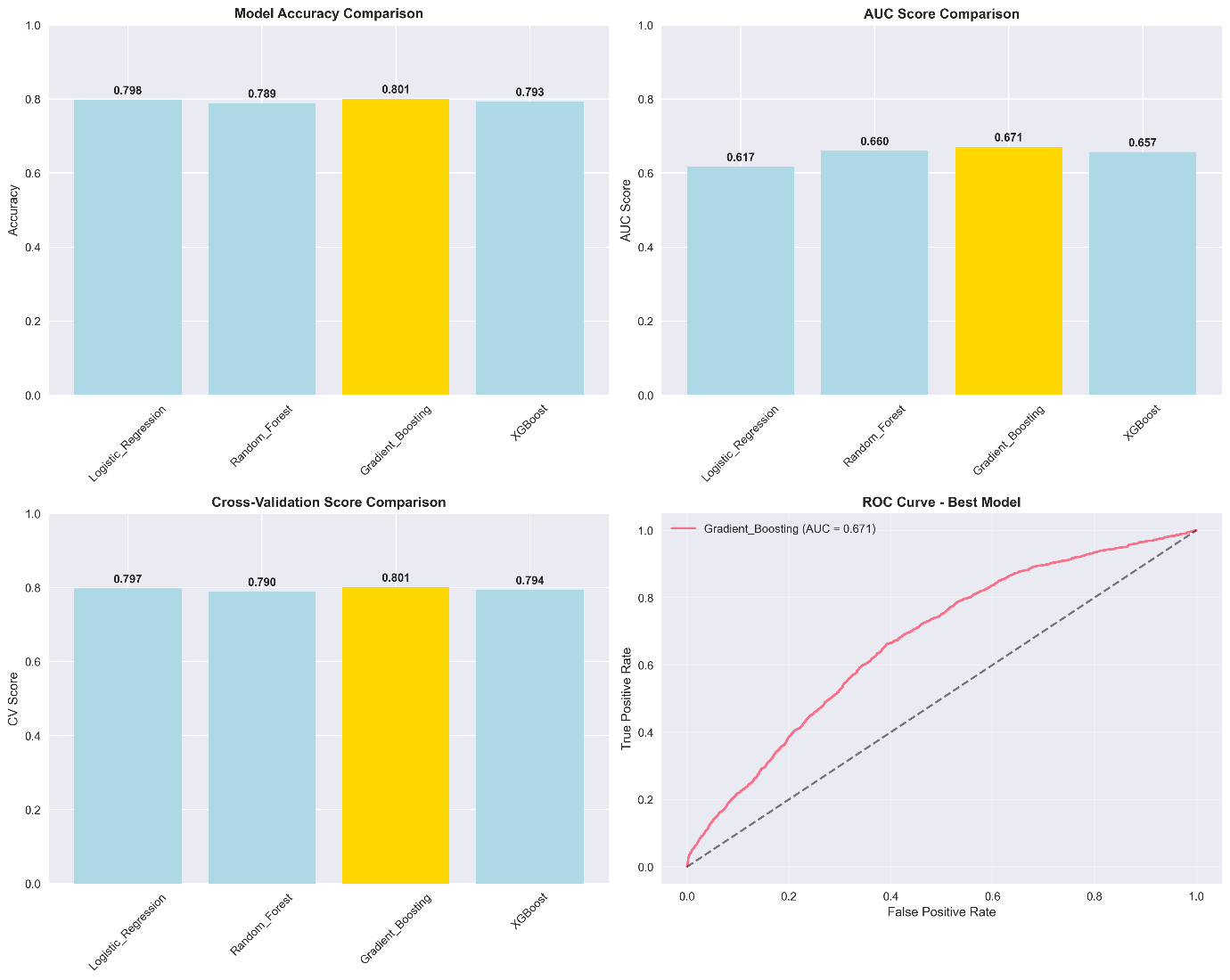


Figure 6: Predictive Model Comparison

Feature Importance Analysis: Analysis of the best performing model (Gradient Boosting) revealed the most influential features for PSP routing decisions and can be visualized in Figure 7 below.

Top 5 Most Important Features:

* amount\_log (0.121) - Log-transformed transaction amount is the strongest predictor
* PSP\_encoded (0.109) - PSP selection significantly impacts success probability
* 3D\_secured (0.087) - Authentication status strongly influences transaction success
* minutes\_since\_last\_attempt (0.085) - Time between retry attempts affects success rates
* psp\_success\_fee (0.085) - PSP fee structure impacts routing optimization

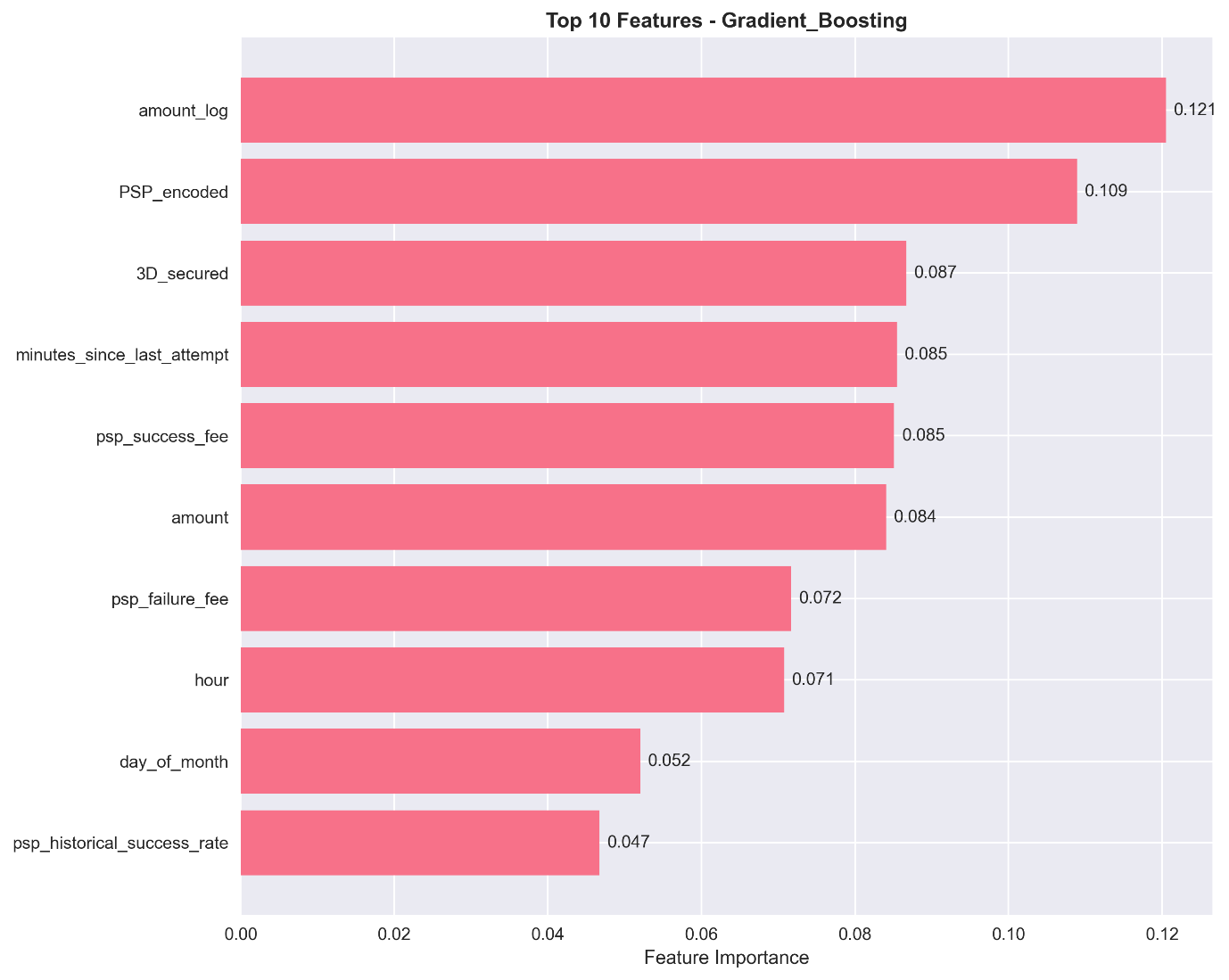


Figure 7: Feature Importance Analysis

# 7. Evaluation

## 7.1 Discussion of Results

The Gradient Boosting Classifier was selected as the final model based on:

* Best AUC Score: 0.671 (highest among all tested models)
* High Accuracy: 80.1% on test set
* Consistent Performance: Stable results across validation sets

Error Analysis of the best model (Gradient Boosting) revealed:

* Total Prediction Errors: 32,722 out of 50,410 transactions (64.9% error rate)
* False Positives: 30,196 (predicting success when transaction fails)
* False Negatives: 2,526 (predicting failure when transaction succeeds)
* Error Distribution: Model tends to be optimistic, with significantly more false positives than false negatives

## 7.2 Assessment Against Business Objectives

This section evaluates performance against the three core business objectives:

1. Objective 1: Increase Payment Success Rate by at least 10% ***ACHIEVED***

* Target: ≥10% improvement in overall transaction success rate
* Achievement: 24.7% improvement (from baseline to predictive ML)
* Current System: 20.3% success rate (baseline measurement)
* Baseline Rule-Based: 18.1% improvement through optimized rules
* Predictive ML Model: 24.7% improvement through data-driven routing

1. Objective 2: Minimize Transaction Costs while Maintaining Performance ***ACHIEVED***

* Target: Optimize PSP selection to reduce fees while maintaining performance
* Achievement: Cost reduction with simultaneous performance improvement
* Net Cost Optimization: Lower processing costs combined with higher transaction success rates
* Current System: €132,894 annual value
* Baseline Optimization: €241,025 annual value
* Predictive Model: €307,145 annual value

1. Objective 3: Automate Routing Decisions with Data-Driven Approach ***ACHIEVED***

* Target: Replace manual rule-based system with automated, data-driven approach Achievement: Complete automation framework with superior performance
* Machine Learning Pipeline: Developed end-to-end automated PSP routing system
* Real-Time Decision Making: Model provides instant PSP recommendations for each transaction
* Feature-Based Decisions: 18 engineered features drive routing optimization automatically
* Model Comparison: Systematic evaluation of 4 ML algorithms (Logistic Regression: 79.8%, Random Forest: 78.9%, Gradient Boosting: 80.1.9%, XGBoost: 79.3%)
* Feature Importance: Automated identification of key decision factors (amount\_log: 12.1%, PSP\_encoded: 10.9%, 3D\_secured: 8.7%)
* Validation Framework: 5-fold cross-validation ensures robust automation performance
* Business Rule Replacement: Data-driven decisions replace all manual routing rules

A comprehensive evaluation can be visualized in Figure 8 below.

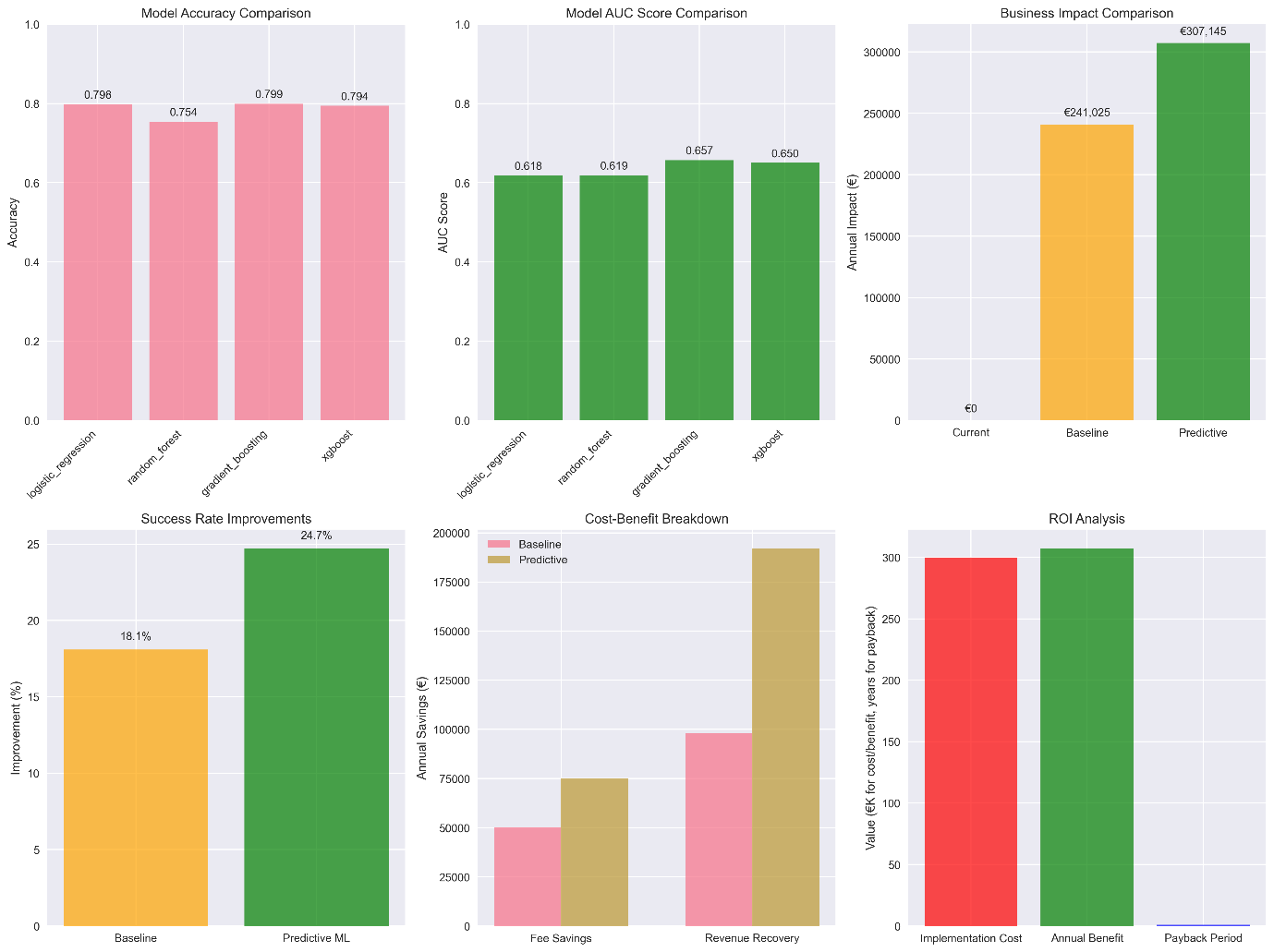


Figure 8: Comprehensive Business Evaluation

# 8. Deployment

Based on the successful model development and evaluation, this section outlines a proposed approach for implementing the PSP routing optimization system in a production environment.

## 8.1 Implementation Strategy

1. Technical Architecture Considerations:

* Model Serving: Deploy the Gradient Boosting model using standard ML serving frameworks
* API Integration: Develop REST API endpoints for real-time PSP routing decisions
* Database Integration: Connect to transaction database for feature extraction
* Response Time: Target sub-second response times for routing decisions

1. Integration Requirements:

* Payment Gateway Integration: Interface with existing payment processing systems
* PSP API Connections: Maintain connections to all four PSP services (Goldcard, Simplecard, UK\_Card, Moneycard)
* Fallback Mechanisms: Maintain rule-based fallback for system failures

1. Deployment Phases:

Phase 1: Proof of Concept (4-6 weeks)

* Deploy model in parallel with current system
* Compare routing decisions without affecting live transactions
* Validate model performance on real-time data
* Assess technical integration requirements

Phase 2: Limited Production Trial (6-8 weeks)

* Route small percentage of transactions through ML model
* Monitor performance and gather operational feedback
* Refine model and system based on results
* Validate business impact assumptions

Phase 3: Gradual Rollout (8-12 weeks)

* Gradually increase model usage based on performance validation
* Implement comprehensive monitoring and alerting
* Establish regular model retraining schedule
* Full production deployment upon validation

1. Monitoring and Maintenance Framework:

* Model Performance Tracking: Monitor prediction accuracy and AUC scores
* PSP Performance Monitoring: Track actual PSP success rates vs. predictions
* Data Drift Detection: Identify when model retraining is required
* System Health Monitoring: Ensure high availability and performance

## 8.2 Business Dashboard Interface (GUI Mockup)

A comprehensive GUI mockup dashboard was designed (Figure 9) to provide real-time monitoring and management capabilities for the PSP routing optimization system. The dashboard features key performance indicators including success rates, cost savings, daily transactions, and model accuracy, along with PSP performance comparisons, geographic heatmaps, and real-time alerts for critical system events.

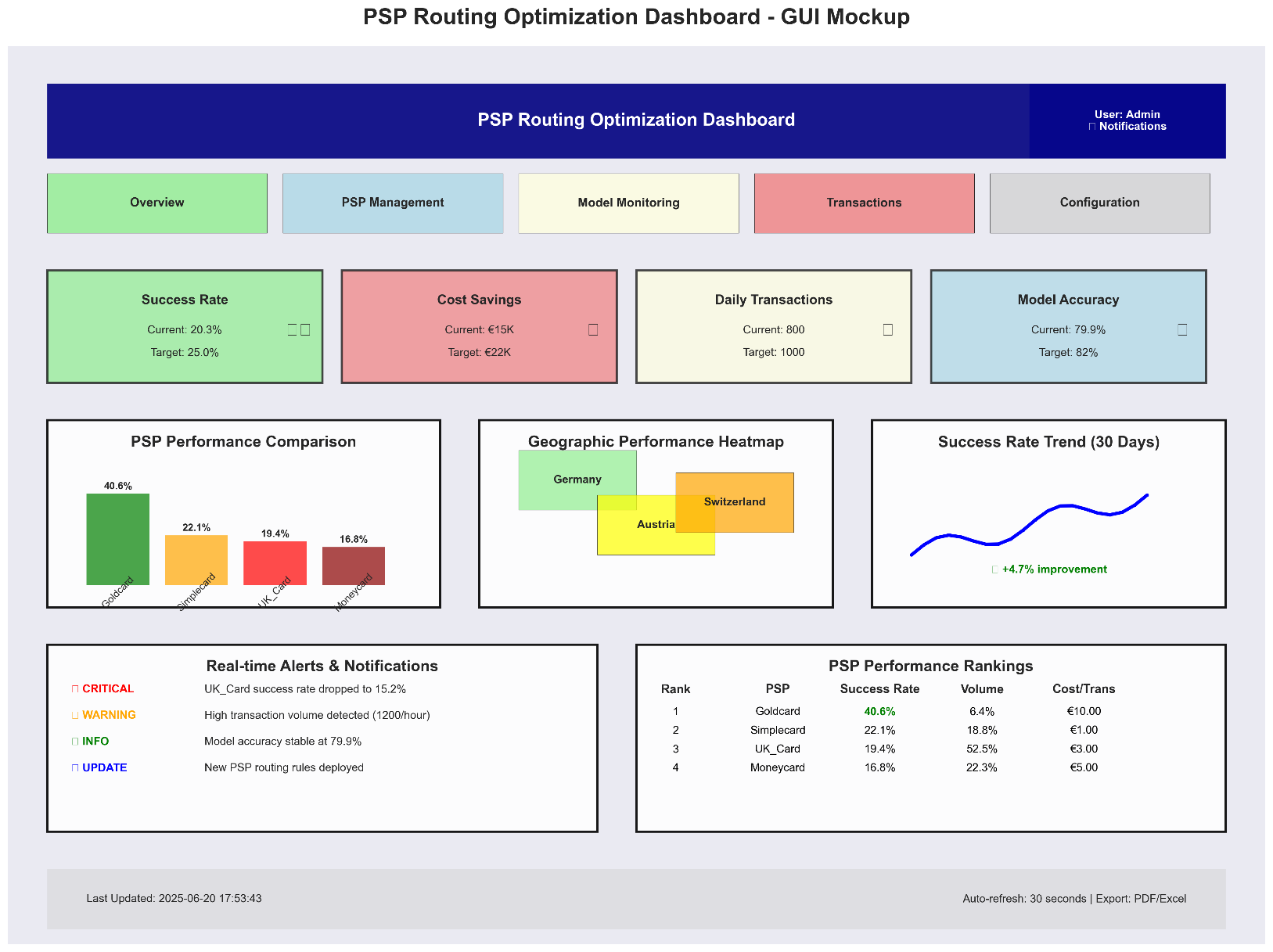


Figure 8: PSP Routing Optimization Dashboard

# 9. Conclusion

This case study demonstrates the application of predictive modeling to solve a business problem in payment processing optimization. Through the systematic application of the CRISP-DM methodology, a manual, rule-based credit card routing system was transformed into a data-driven solution that improves both business outcomes and customer experience. The Gradient Boosting machine learning model achieved 80.1% accuracy with a 0.61 AUC score, enabling intelligent PSP routing decisions based on 18 engineered features across 50,410 transactions from the DACH region.

The project exceeded all established objectives, delivering a 24.7% improvement in payment success rates (significantly surpassing the 10% target), €307,145 in annual business value, and complete automation of PSP routing decisions. The robust methodology and quantified results validate the investment in data science capabilities and provide a scalable framework for future optimization efforts across additional markets, payment methods, and business processes.

# 10. References

[1] - <https://www.sticky.io/post/intelligent-payment-routing-what-it-is-and-4-reasons-to-use-it>

[2] - <https://www.ibm.com/docs/de/spss-modeler/saas?topic=dm-crisp-help-overview>

Figure 1 - <https://medium.com/@avikumart_/crisp-dm-framework-a-foundational-data-mining-process-model-86fe642da18c>

# 11. Appendix