# ML 과제 보고서: 신용카드 사기 탐지

---

## 1. 데이터 로드 및 기본 탐색

데이터의 기초 구조와 클래스 불균형 상태를 확인합니다.

- **데이터 구성**: creditcard.csv (V1~V28, Time, Amount, Class)
- **클래스 비율 (전체)**:
  - 정상 (0): 99.83% (극심한 불균형)
  - 사기 (1): 0.17%

---

## 2. 샘플링 및 전처리

데이터의 크기를 조정하고 수치형 변수의 스케일을 맞춥니다.

### 무작위 샘플링 (Downsampling)

- **방법**: 사기 거래(Class=1) 전체 유지 + 정상 거래(Class=0) 10,000건 무작위 추출
- **목적**: 분석 효율성 증대 및 데이터 불균형의 1차적 완화

### 데이터 전처리

- **Standard Scaling**: Amount 변수의 값 편차가 크므로 평균 0, 표준편차 1로 표준화 수행.
- **변수 제거**: 스케일링 전 원본 Amount 변수 제거 및 $X, y$ 분리.

---

## 3. 학습 및 테스트 데이터 분할

- Train(8) : Test(2)
- 전략: stratify=y 설정을 통해 학습셋과 테스트셋에서도 클래스 비율이 원본과 동일하게 유지되도록 분할함.

```
Class
0    7999
1     394
Name: count, dtype: int64
Class
0    2001
1      98
Name: count, dtype: int64
Class
0    0.953056
1    0.046944
Name: proportion, dtype: float64
Class
0    0.953311
1    0.046689
Name: proportion, dtype: float64
```

---

## 4. SMOTE 적용 (Oversampling)

> **SMOTE를 적용해야하는 이유**
>
> SMOTE(Synthetic Minority Over-sampling Technique)는 소수 클래스의 데이터를 단순히 복제하는 것이 아니라, 인접한 데이터들 사이의 가상 데이터를 생성하여 오버샘플링함.
> 이를 통해 모델이 사기 거래의 특성을 더 풍부하게 학습하여 Recall을 높일 수 있음.

- 적용 결과

```
Before SMOTE:
Class
0    7999
1     394
Name: count, dtype: int64
After SMOTE:
Class
0    7999
1    7999
Name: count, dtype: int64
```

---

## 5. 모델 학습 및 평가

- **선정 모델**: `RandomForestClassifier`
- **평가 지표**: Precision, Recall, F1-score, PR-AUC

```
              precision    recall  f1-score   support

           0       0.99      1.00      1.00      2001
           1       0.95      0.89      0.92        98

    accuracy                           0.99      2099
   macro avg       0.97      0.94      0.96      2099
weighted avg       0.99      0.99      0.99      2099

Class 0 - Recall: 0.9975, F1-score: 0.9960
Class 1 - Recall: 0.8878, F1-score: 0.9158
PR-AUC: 0.9538
```

---

## 6. 최종 성능 평가 및 제언

- 목표: Recall ≥ 0.80, F1 ≥ 0.88, PR-AUC ≥ 0.90
  - 달성