

과제 주제: 특성 선택 및 윈도우 길이 조정, 모델 조정을 통한 정밀도 향상 모델 구현

## 과제 목표:

학생들은 주어진 데이터셋을 사용하여 파생 변수 특성 선택, 윈도우 길이 조정 및 모델 조정을 통해 모델의 성능을 개선하는 방법을 학습하고, 이를 실습을 통해 적용해본다. 최종적으로 개선된 모델의 정밀도를 높이는 것을 목표로 한다.

## 세부 과제 내용:

### 1. 데이터 준비:

- 주어진 데이터셋을 로드하고, 필요한 전처리를 수행하세요.
- 이진 분류 문제를 다루고 있으므로, 클래스 레이블과 주요 특성들을 확인하세요.

### 2. 특성 선택(Feature Selection):

- 주어진 특성들 중 중요한 특성을 선택하여 사용하세요.
- 선택한 특성들의 이유와 선택 과정에서의 판단을 설명하세요.

### 3. 윈도우(Window) 길이 조정:

- 데이터가 시계열일 경우, 윈도우 크기를 조정하여 모델 성능을 향상시키세요.
- 서로 다른 윈도우 크기를 사용해보고, 최적의 윈도우 길이를 찾아 설명하세요.

### 4. 모델 조정(Model Tuning):

- 머신러닝 모델을 하나 선택하세요 (e.g., 로지스틱 회귀, 랜덤 포레스트, XGBoost, LSTM 등).
- 모델의 주요 하이퍼파라미터를 조정하여 성능을 개선하세요.
  - 예시: 랜덤 포레스트의 경우 트리의 개수, 최대 깊이 등의 하이퍼파라미터를 조정.
  - 하이퍼파라미터 튜닝을 위해 Grid Search 또는 Random Search를 사용할 수 있습니다.

- 조정된 하이퍼파라미터와 그 이유를 설명하세요.

**5. 모델 평가 및 비교:**

- 모델의 성능을 정밀도(Precision) 평가 지표로 비교하세요.
- 특성 선택, 윈도우 길이 조정, 하이퍼파라미터 조정 전과 후의 모델 성능을 비교하세요.
- 각 과정이 모델 성능에 어떤 영향을 미쳤는지 분석하세요.

**6. 보고서 작성:**

- 실험 과정과 결과를 보고서로 작성하세요.
- 보고서에는 다음이 포함되어야 합니다.

1. **특성 선택 과정 및 결과**
2. **윈도우 길이 조정 과정 및 최적 값 선택 이유**
3. **모델 선택 및 하이퍼파라미터 조정**
4. **성능 비교** (조정 전후의 성능 차이 설명)
5. **최종 결론:** 가장 좋은 성능을 보인 모델과 그 이유에 대한 설명

**제출물:**

1. Python 코드 파일 (Jupyter Notebook 형식 권장)