<FDS 모델 선택 및 학습 과정 정리>

| 순서 | 작업 내용 |
| --- | --- |
| 1 | 모델 학습 |
| 모델 선택 |
| 모델 초기화 |
| 모델 학습 |
| 2 | 모델 검증 |
| 예측 수행 |
| 성능 평가 |
| 하이퍼파라미터 튜닝 |
| 3 | 모델 테스트 |
| 최종 예측 수행 |
| 최종 성능 평가 |
| 4 | 결과 보고 및 배포 |
| 결과 보고 |
| 모델 배포 |
| 모니터링 및 유지보수 |

## 모델 학습

X\_train과 y\_train 데이터를 사용하여 모델을 학습 시킨다.

#### 1.1. 모델 선택

* 선택 기준: 데이터의 특성, 프로젝트 요구사항, 알고리즘의 적합성 등을 고려하여 모델을 선택한다.
* 모델 예시: 랜덤 포레스트, XGBoost, 로지스틱 회귀, 서포트 벡터 머신(SVM)

#### 1.2. 모델 초기화

* 선택한 모델을 초기화하고, 필요에 따라 하이퍼파라미터를 설정한다.
* 예) RandomForestClassifier(n\_estimators=100, max\_depth=10)

#### 1.3. 모델 학습

* X\_train과 y\_train 데이터를 사용하여 모델을 학습시킨다.

## 모델 검증

X\_val과 y\_val 데이터를 사용하여 학습된 모델의 성능을 확인한다. (과적합을 방지하고 모델의 일반화 성능을 평가하는데 중요한 과정이다.)

#### 2.1. 예측 수행

* 학습된 모델을 사용해 X\_val 데이터를 예측한다.

#### 2.2. 성능 평가

* 예측 결과 y\_val\_pred와 실제 라벨 y\_val을 비교하여 모델 성능을 평가한다.
* 평가 지표: 정확도, 정밀도, 재현율, F1-score, ROC-AUC 등

#### 2.3. 하이퍼파라미터 튜닝

* 검증 결과를 바탕으로 모델의 하이퍼파라미터를 조정하여 최적의 성능을 찾는다. 작업은 교차 검증이나 그리드 서치 등의 방법으로 수행될 수 있다.

## 모델 테스트

X\_test와 y\_test 데이터를 사용하여 최종 모델의 성능을 평가한다. (실제 환경에서 모델이 얼마나 잘 작동할지를 예측하는 중요한 과정이다.

#### 3.1. 예측 수행

* 검증을 통해 최적화된 모델로 X\_test 데이터를 예측한다.

#### 3.2. 최종 성능 평가

* y\_test와 y\_test\_pred를 비교하여 모델의 최종 성능을 평가한다.
* 평가 지표: 검증 단계와 동일한 지표를 사용하여 평가한다.

## 결과 보고 및 배포

모델 테스트 후, 결과를 보고하고, 모델을 실제 환경에 배포할 준비를 한다.

#### 4.1. 결과 보고

* 내용: 모델의 성능, 사용된 데이터, 평가 지표, 중요 변수, 모델의 장단점 등을 포함한 보고서를 작성한다.
* 포맷: PDF 보고서, 슬라이드 자료, 또는 기술 문서로 작성한다.

#### 4.2. 모델 배포

* 모델을 실시간 예측 시스템에 통합하거나 API로 배포하여 이상 거래 탐지 시스템으로 운영한다.
* 기술 스택: Flask, FastAPI, Docker, Django 등

#### 4.3. 모니터링 및 유지보수

* 운영 중인 시스템의 성능을 모니터링하고, 필요시 모델을 재학습하거나 조정한다.
* 모니터링 툴: ELK Stack, Prometheus, Grafana