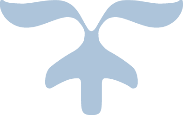


유방암 데이터셋을 활용한 기계학습 모델 분석 보고서

학습을 통한 유방암 예측



2025 APRIL 14

우 성 욱

<목 차>

내용

[요약 2](#_Toc195571286)

[1. 프로젝트 개요 3](#_Toc195571287)

[1.1. 목표 3](#_Toc195571288)

[1.2. 데이터 3](#_Toc195571289)

[2. 데이터 구성 4](#_Toc195571290)

[2.1. 데이터 구성 4](#_Toc195571291)

[2.2. 이상치, 결측 치 및 중복 데이터 유무 4](#_Toc195571292)

[3. 프로젝트 흐름 5](#_Toc195571293)

[4. 모델링 6](#_Toc195571294)

[5.1. 사용 모델 6](#_Toc195571295)

[5.2. 모델 사용한 이유 6](#_Toc195571296)

[5. F1-score와 재현율이 중요한 이유 8](#_Toc195571297)

[6. 실험 결과 9](#_Toc195571298)

# 요약

유방암 프로젝트는 scikit-learn의 유방암 데이터셋을 활용하여 다양한 기계학습 모델의 성능을 비교 분석하였습니다. 로지스틱 회귀, KNN, 랜덤 포레스트, 그래디언트 부스팅 등의 분류 알고리즘을 적용하고, 하이퍼파라미터 최적화 및 PCA를 통한 차원 축소 기법을 활용하여 최적의 모델을 찾았습니다. 그 결과, 그래디언트 부스팅 분류기가 97.66%의 정확도로 가장 높은 성능을 보였으며, 랜덤 포레스트에 대한 하이퍼파라미터 최적화와 XGBoost 모델 최적화를 통해 성능을 더욱 향상시켰습니다. 또한 PCA를 통한 차원 축소 후에도 높은 분류 성능을 유지할 수 있었습니다.

# 프로젝트 개요

## 목표

* 유방암 데이터셋을 활용하여 암 진단을 위한 최적의 기계학습 모델 개발
* 다양한 분류 알고리즘의 성능 비교 및 분석
* 모델 최적화를 통한 예측 정확도 향상

## 데이터

* scikit-learn의 유방암 데이터셋(Breast Cancer Wisconsin Dataset)
* 총 569개의 샘플과 30개의 특성 포함
* 이진 분류: 양성(benign, 1)과 악성(malignant, 0) 종양 구분

1.3 주요과제

* 최적의 분류 모델 선정
* 하이퍼파라미터 최적화
* 성능 평가 지표 선정 및 적용

# 데이터 구성

## 데이터 구성

샘플 수: 569개

특성 수: 30개

타겟 변수: 종양의 양성(1) 또는 악성(0) 여부

## 이상치, 결측 치 및 중복 데이터 유무

결측치: 없음

이상치: 데이터 탐색 과정에서 특별한 이상치는 발견되지 않음

중복 데이터: 없음

데이터의 전반적인 품질은 양호한 것으로 확인됨

# 프로젝트 흐름

기본 모델 삭습 및 평가

* 로지스틱 회귀
* K-최근접 이웃
* 랜덤 포레스트
* 그래디언트 부스팅
* 의사결정 트리
* AdaBoost
* 보팅 분류기

하이퍼파라미터 최적화

* GridSearchCV를 통한 랜덤 포레스트 최적화
* RandomizedSearchCV를 통한 랜덤 포레스트 최적화
* Hyperopt를 통한 XGBoost 최적화

# 모델링

## 사용 모델

* 로지스틱 회귀(Logistic Regression)
* K-최근접 이웃(KNN)
* 랜덤 포레스트(Random Forest)
* 그래디언트 부스팅(Gradient Boosting)
* 의사결정 트리(Decision Tree)
* AdaBoost
* 보팅 분류기(Voting Classifier)
* XGBoost

## 모델 사용한 이유

**로지스틱 회귀**:

* 해석력이 좋고 구현이 간단한 기본 모델
* 이진 분류 문제에 적합

**K-최근접 이웃(KNN)**:

* 비모수적 방법으로 복잡한 결정 경계를 학습 가능
* 구현이 간단하고 직관적인 모델

**랜덤 포레스트**:

* 의사결정 트리의 앙상블로 과적합 방지 효과
* 특성 중요도 분석이 가능

**그래디언트 부스팅**:

* 순차적인 학습을 통해 예측 오차를 줄여나가는 강력한 앙상블 기법
* 일반적으로 높은 예측 성능을 보임

**의사결정 트리**:

* 모델의 해석이 용이
* 다른 앙상블 모델과의 성능 비교를 위한 기준 모델

**AdaBoost**:

* 약한 학습기를 조합하여 강한 학습기를 만드는 앙상블 기법
* 오분류된 샘플에 가중치를 부여하는 방식으로 학습

**보팅 분류기**:

* 여러 모델의 예측을 통합하여 최종 예측을 결정
* 개별 모델의 장점을 결합하여 성능 향상 가능

**XGBoost**:

* 그래디언트 부스팅의 최적화된 구현체
* 효율적인 학습과 높은 예측 성능

# F1-score와 재현율, AUC가 중요한 이유

* 유방암 데이터셋에서 \*\*재현율(Recall)\*\*이 중요한 이유:

심각한 결과 예방: 유방암을 가진 환자를 정상으로 잘못 분류하면(위음성), 치료 지연으로 생명이 위험할 수 있습니다.

초기 발견의 중요성: 유방암은 조기 발견 시 생존율이 크게 높아지므로, 가능한 많은 실제 환자를 식별하는 것이 중요합니다.

의료 윤리적 측면: "해를 끼치지 말라"는 의료 윤리에 따라, 질병을 놓치는 것보다 추가 검사를 권장하는 것이 바람직합니다.

* 유방암 데이터셋에서 F1-score가 중요한 이유:

불필요한 생검 방지: 재현율만 높이려 하면 위양성이 늘어날 수 있으며, 이는 불필요한 생검과 환자의 심리적 부담을 초래합니다.

의료 자원 효율성: F1-score는 정밀도와 재현율의 균형을 고려해, 의료 자원을 보다 효율적으로 사용하도록 도와줍니다.

환자 경험 최적화: 위양성이 많으면 환자의 불안과 의료 시스템에 대한 신뢰도 하락으로 이어질 수 있습니다.

* 유방암 데이터셋에서 \*\*AUC (Area Under the Curve)\*\*가 중요한 이유:

모델의 전반적인 판별 능력 평가: AUC는 다양한 임계값(threshold)에서 모델이 양성과 음성을 얼마나 잘 구별하는지를 종합적으로 보여줍니다. 높은 AUC는 모델이 환자(양성)와 정상인(음성)을 잘 구분하고 있음을 의미합니다.

클래스 불균형 문제 대응: 유방암 데이터는 양성 사례(암 환자)가 적은 편인데, AUC는 클래스 불균형 상황에서도 비교적 신뢰할 수 있는 평가 지표입니다.

모델 비교 및 선택에 유용: 여러 모델 중 가장 성능이 좋은 모델을 고를 때 AUC는 전반적인 성능을 정량적으로 비교하는 데 유리합니다.

확률 예측의 품질 평가: AUC는 모델이 예측한 확률 값의 품질을 반영하기 때문에, 단순 이진 분류 결과보다 더 정교한 성능 평가가 가능합니다.

# 실험 결과

6.1 기본 분류 모델 성능 비교

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 분류기 | 정확도 | 정밀도 | 재현율 | F1 | AUC |
| Logistic Regression | 0.9298 | 0.9358 | 0.9533 | 0.9444 | 0.9914 |
| K Neighbors Classifier | 0.9240 | 0.9196 | 0.9626 | 0.9406 | 0.9581 |
| Random Forest Classifier | 0.9532 | 0.9541 | 0.9720 | 0.9630 | 0.9781 |
| Gradient Boosting Classifier | 0.9474 | 0.9623 | 0.9533 | 0.9577 | 0.9866 |
| Decision Tree Classifier | 0.8947 | 0.9320 | 0.8972 | 0.9143 | 0.8939 |
| AdaBoost Classifier | 0.9415 | 0.9533 | 0.9533 | 0.9533 | 0.9889 |
| Voting Classifier | 0.9415 | 0.9533 | 0.9533 | 0.9533 | 0.9833 |

**Random Forest Classifier가 재현율(0.9720)과 F1(0.9630) 모두 가장 높아서,  
유방암 데이터셋에서는 가장 좋은 성능이 좋은 것을 알 수 있습니다.**

그러므로 Random Forest를 좀 더 최적화를 하기 위해서 GridSearchCV를 한 후 범위를 좀더 좁혀서 RandomizedSearchCV를 하였습니다. GridSearchCV를 하여 n\_estimators는 150, max\_depth는 4, min\_samples\_split는 8, min\_samples\_leaf는 4가 나와 RandomizedSearchCV의 n\_estimators를 50~200, max\_depth를 2~6, min\_samples\_split를 1~10, min\_samples\_leaf를 1~6을 주었습니다. 그래서 최적 하이퍼 파라미터는 {'bootstrap': False, 'max\_depth': 5, 'min\_samples\_leaf': 3, 'min\_samples\_split': 5, 'n\_estimators': 137}식으로 나왔습니다. 이때의 정확도는 0.9474, 정밀도는 0.9455, 재현율은 0.9720, f1은 0.9585, AUC는 0.9793으로 F1은 떨어졌지만 AUC가 올랐습니다.

6.2 XGBoost 하이퍼파라미터 최적화 결과

{'colsample\_bytree': 0.7254819122205993, 'learning\_rate': 0.198562507758628, 'max\_depth': 7.0, 'min\_child\_weight': 2.0}으로 나와 XGBoost로 해보았을 때, 정확도는 0.9561, 정밀도는 0.9737, 재현율은 0.9610, F1은 0.9673, AUC은 0.9944로 나왔습니다