

# AI #2 XAI 과제

## 목차

1. 실습 내용
2. 개요
3. 결과 - 데이터 전처리
4. 결과 - 모델 구축 및 평가 실행
5. 결과 - 교차검증 전략 비교 실행
6. 결과 - 하이퍼파라미터 튜닝
7. 결과 - SHAP 기반 해석 (Bar Summary Plot, Beeswarm Plot)
8. 결과 - SHAP 기반 해석 (Waterfall Plot)
9. Final Report - SHAP 결과 해석 및 임상 활용 방안

### ▼ 1. 실습 내용

```
# XAI: Explainable AI

[HPO Method]
1. Random Search
2. Bayesian Search(TPE:Tree-structured Parzen Estimator) using Optuna

[XAI Method]
SHAP

[Data]
UCI 활용 데이터 - Breast Cancer (15)
https://archive.ics.uci.edu/

[Note]
HPO Example로 오류방지를 위한 최소한의 전처리만 수행

[Task]
XAI를 활용한 모델의 해석성 제공하는 코드작성
- Base 예측/분류 모델 선택은 자유
- XAI 결과를 살펴보고 분석내용을 정리해보세요(주석/텍스트셀 통해 코드 파일 내 설명 기입)
- Colab or Jupyter Notebook 파일 제출
```

### ▼ 2. 개요

- 목적
  - UCI Breast Cancer 데이터를 로드하고 전처리 후 XGBoost 모델을 구축 및 평가
  - 교차검증(StratifiedKFold, KFold)과 하이퍼파라미터 탐색 기법(RandomizedSearchCV, Optuna)을 비교하여 최적 성능을 도출
  - SHAP을 활용하여 전역적·집단적·개별적 수준에서 해석력을 확보하고 도메인 지식과 연결
- 구현
  - 데이터 로드: UCI Breast Cancer 데이터셋
  - 데이터 전처리: 타겟(Diagnosis)을 이진화(M=1, B=0), StandardScaler로 범위 스케일링, 상관관계수 0.9 이상인 중복 변수 제거

- 모델 구축: XGBClassifier
- 모델 평가: 정확도, AUC, 분류리포트, 혼동행렬, feature importance
- 교차검증: KFold, StratifiedKFold
- 하이퍼파라미터 최적화: Random Search, Optuna TPE
- 모델 해석 (SHAP)
  - Bar Summary Plot: 전역적 중요도(평균 |SHAP|)를 통해 주요 변수 확인
  - Beeswarm Plot: 변수 값 크기(빨강/파랑)와 방향성(+/-)에 따른 분포 해석
  - Force / Waterfall Plot: 3가지 개별 환자 샘플(예측 확률 극단/불확실, SHAP 영향력 최대, 도메인 특이 케이스)을 선택하여 모델이 어떤 요인 때문에 해당 예측을 내렸는지 설명

### ▼ 3. 결과 - 데이터 전처리

- 결측치 확인
  - 존재하지 않으므로 결측치 전처리 불필요
- 타깃 변수 분포 (불균형 여부)
  - B(양성, Benign) 357건 (62.7%), M(악성, Malignant) 212건 (37.3%)
  - 6:4의 클래스 불균형이 존재하므로 단순 정확도(Accuracy)만 보면 불균형에 의해 성능이 과대평가될 수 있으므로 재현율(Recall), 정밀도(Precision), F1, AUC 등 불균형에 강한 지표로 성능 평가하기.
- 숫자형 변수 분포
  - 변수마다 범위 차이가 매우 크다. 예를들어 area1은 최소 143 ~ 최대 2501 인데 smoothness1은 평균 0.096, 최대 0.16이다. 같은 모델 입력에 넣으면 큰 값이 작은 값을 묵살할수있다.
  - 30개의 숫자형 변수에 StandardScaler를 적용해서 평균 0, 분산 1의 분포로 변환.
- 상관관계 기반 피쳐 제거
  - 피쳐가 30개로 좀 많음편인데 피쳐 수가 줄어들면서 해석력이 향상되고 과적합 위험이 완화된다. 서로 강하게 상관된 피쳐들은 대표성 있는 하나만 남기고 나머지는 제거하는 게 좋으므로 상관계수가 높은 피쳐는 제거하기.
  - 상관계수 0.9 이상인 피쳐 10개 제거: ['perimeter1', 'area1', 'concave\_points1', 'perimeter2', 'area2', 'radius3', 'texture3', 'perimeter3', 'area3', 'concave\_points3']
- 해석 가능성 기준 피쳐 유지
  - 의료 데이터라는 특성상, 의사가 직관적으로 이해할 수 있는 변수를 남기는 게 중요하므로 smoothness, symmetry 같은 형태학적 지표는 모델 해석에 유리하므로 유지 가치가 있다. 제거하지 않기.

### ▼ 4. 결과 - 모델 구축 및 평가 실행

- 정확도 (0.9474)
  - 모델이 꽤 잘 맞히고 있다 (정확도 95%).
  - 정확도는 클래스 불균형 상황에서 높게나오기쉬워서 확인해야하는데 이 데이터셋은 불균형이 심하지 않아(0 클래스 72, 1 클래스 42) 정확도가 어느 정도 신뢰할만하다고 생각된다.
- AUC (0.9864)
  - AUC가 0.98이라는 것은 특정 임계값에서만 잘 작동하는 게 아니라 전반적으로 좋은 분류 성능을 갖고 있음을 의미한다. 특히 불균형 데이터셋에서는 AUC가 더 안정적인 지표인데 이 데이터셋은 약간의 불균형 데이터셋인데 높은 가까운 구분력을 보여준다.
- 분류 리포트
  - 클래스 0: Precision 0.93, Recall 0.99 → 정상 환자를 거의 다 잡아내고, 일부를 암으로 잘못 분류하는 경우가 극히 적음.
  - 클래스 1: Precision 0.97, Recall 0.88 → 암으로 예측했을 때 대부분 맞지만, 실제 암 환자 중 일부(약 12%)를 놓침.

- 모델은 정상 환자(클래스 0)에 대해 매우 민감하게 반응하지만 암 환자(클래스 1) 중 일부를 놓치고 있음. 의료 도메인에서는 Recall(재현율)이 더 중요할 수 있으므로 threshold 조정이나 클래스 가중치 보정 같은 방법으로 보완이 필요
- 혼동행렬  
[[71 1]  
[ 5 37]]
  - 실제 0(정상) 중 71명을 맞히고 1명을 암으로 잘못 분류함 -> 거짓양성(FP) 거의 없음.
  - 실제 1(암) 중 37명을 맞히고 5명을 정상으로 잘못 분류함 -> 거짓음성(FN)이 존재.
  - 의료도메인은 과잉진료가 차라리 낫다. 암 환자를 정상으로 예측한 경우 (FN)가 더 문제이므로 recall 지표인 이 부분이 보완이 필요하다.
- 특성 중요도
  - concavity1이 40% 비중으로 가장 중요한 변수이며, compactness3, radius1 등이 다음으로 중요한것으로 나온다.
  - 실제 유방암 데이터 문헌에서 일관되게 보고되는 중요한 특징들과 일치할 경우 모델이 중요한 변수를 잘 포착하고 있다고 볼수있는데 도메인 지식이 부족하여 잘 모르겠음.

## ▼ 5. 결과 - 교차검증 전략 비교 실행

- KFold
  - 사용된 데이터셋 클래스 비율이 B: 62.7%, M: 37.3%로 약간 불균형이 있기 때문에 Kfold를 사용하면 어떤 Fold에는 M이 적게 들어가고 어떤 Fold에는 많아질 수 있다.
  - 이러면 Fold 간 성능 편차가 커지거나, 특정 Fold에서 모델이 과소평가/과대평가될 가능성이 있다
- StratifiedKFold
  - StratifiedKFold는 클래스 비율을 유지해서 나누기 때문에 불균형 셋이어도 각 Fold가 전체 분포를 잘 대표한다
  - 실제로 결과 평균 성능 AUC도 아주조금 높고 (0.98 -> 0.99, 0.99), 표준편차가 5 fold에선 절반 수준으로 줄었다 (0.01 -> 0.005, 0.01)
  - 결론1: 현재와같은 불균형 데이터일 때는 StratifiedKFold가 더 공정하고 안정적인 평가를 제공하므로 이번 데이터셋에서도 적합한것으로 생각된다
- 5-fold vs. 20-fold
  - AUC는 비슷하고(0.99, 0.99) 표준편차는 2배 차이난다(0.005, 0.01). CI 폭은 20 fold가 더 넓다(0.01, 0.03)
  - 5 fold는 데이터셋을 크게 나누고 20 fold로 늘어난다면 한 Fold당 검증 데이터가 적어진다. 데이터가 적다 보니 Fold별 AUC 변동이 커진다.
  - 20 fold에서 평균적으로 조금 정확해보이지만(AUC) 결과 불안정성이 크고(표준편차) 실제 성능이 0.96~1.02 범위까지 널뛰기할 수 있다(CI 폭).
  - 결론2: 데이터가 많은 경우엔 20-fold도 의미 있지만 현재 데이터셋(569개 샘플)에서는 5-fold가 더 합리적으로 생각된다.
- 결론
  - 모델이 악성(M)과 양성(B)을 잘 구분하며(AUC=0.99)
  - 모델이 특정 데이터 샘플에 과도하게 의존하지 않아서 일관성 있는 결과를 내지만(표준편차 0.0058: 매우작음)
  - AUC가 높은게 실제 배포 상황에서는 데이터 편향이나 작은 잡음에도 민감할 수 있다 즉 훈련 데이터 분포와 다른 새로운 데이터에서 성능이 유지되는지확인해야한다.

## ▼ 6. 결과 - 하이퍼파라미터 튜닝

- 튜닝 대상인 하이퍼파라미터의 특징을 파악해보기
  - n\_estimators (트리 개수): 크면 학습을 오래 시킴, 성능을 높일 수 있음, overfitting
  - max\_depth (트리 깊이): 깊으면 데이터의 세부 패턴까지 학습해 overfitting
  - learning\_rate (학습률): 크면 빠르게 학습하지만 최적점을 지나쳐 버릴 수 있어 overfitting. 작으면 한 스텝씩 조심스럽게 학습해 일반화 성능은 좋아지지만 많은 트리 n\_estimators가 필요할수있음.

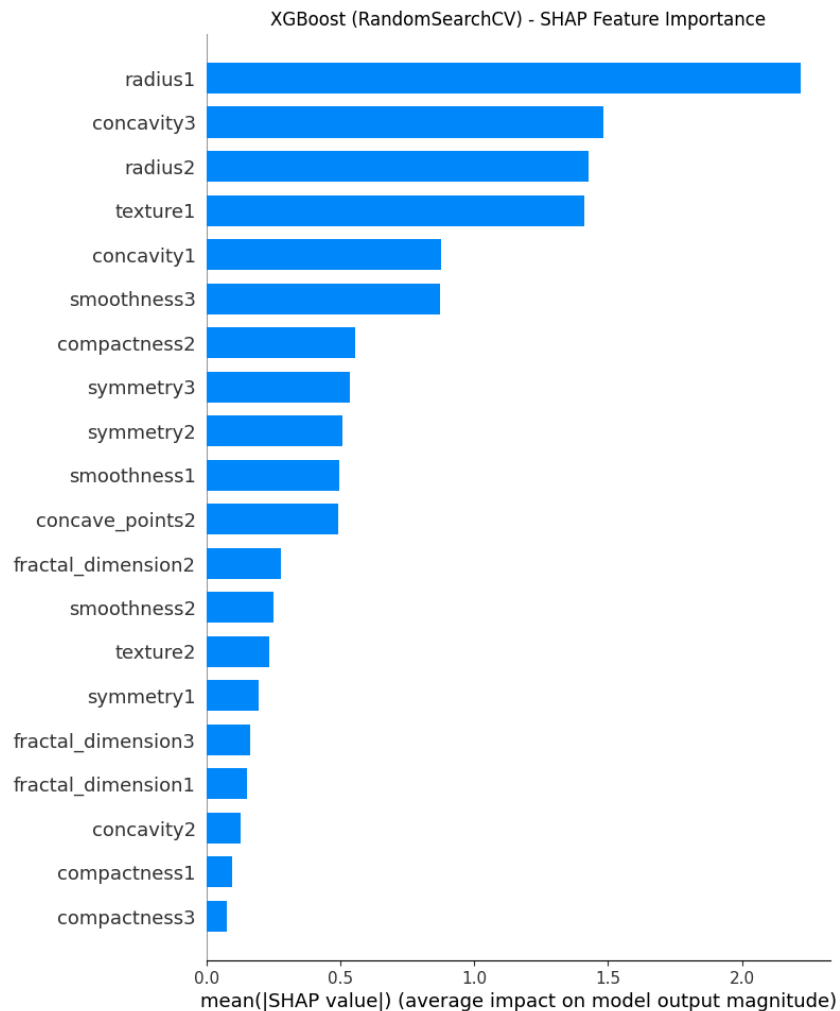
- subsample (샘플 비율): 크면 variance가 크고 overfitting.
- colsample\_bytree (특성 샘플링 비율): 1.0이면 모든 피처를 쓰고 낮추면 랜덤성을 주므로 모델 다양성을 높여 overfitting 완화.
- RandomizedSearchCV 결과 및 해석
  - n\_estimators = 200 ([50, 100, 200, 300] 중 200 → 중간수치 선택)
  - max\_depth = 6 ([3, 4, 6, 8, 10] 중 6 → 중간수치 선택)
  - learning\_rate = 0.2 ([0.01, 0.05, 0.1, 0.2] 중 0.2 → 빠르게 학습)
  - subsample = 0.7 ([0.7, 0.8, 0.9, 1.0] 중 0.7 → 랜덤성 부여해서 과적합 억제)
  - colsample\_bytree = 1.0 ([0.7, 0.8, 0.9, 1.0] 중 1.0 → 모든 특성을 사용)
  - 해석: Random Search에서는 학습률을 크게 두고 빠른 수렴을 유도하면서, 트리 깊이를 6으로 제한해 과적합을 막았다. 또 데이터 샘플은 70%만 써서 안정성을 확보했고 피처 수는 과도하지않아 전부 사용했다.
  - 성능: CV AUC = 0.9936 (높은 성능)
- Optuna TPE 결과 및 해석
  - n\_estimators = 233 (50~300중 233 → random search보다 조금 더 많음 → 작은 learning\_rate 보완)
  - max\_depth = 8 (3~10중 8 → random search보다 깊은 트리 → 복잡한 패턴까지 학습)
  - learning\_rate = 0.0196 (0.01~0.3 중 0.0196 → 아주 작은 값 → 신중하게 학습, 대신 많은 트리가 필요)
  - subsample = 0.823 (0.7~1.0 중 0.823 → 나름 랜덤성 부여로 과적합 방지)
  - colsample\_bytree = 0.943 (0.7~1.0 중 0.943 → 전체 피처의 94%만 사용해서 random search에 비해 약간 랜덤성 추가)
  - 해석: Optuna TPE는 random search에 비해 학습률을 매우 작게 두고(0.2 → 0.02) 트리 깊이를 늘려(6→8) 복잡한 구조를 학습함. 트리 개수는 비슷하지만 200에서 233개로 늘렸고, subsample도 random(0.7)보다는 높지만 (0.823) 낮춰서 조절하였고 colsample을 낮춰서(1.0 → 0.943) 샘플 샘플링과 피쳐 샘플링으로 overfitting을 방지했다. 결론적으로 Optuna TPE는 학습률을 작게 두고 천천히 정밀하게 더 복잡한 모델을 만들었다.
  - 성능: CV AUC = 0.9924 (Random Search보다 살짝 낮지만 충분히 높기도 하고 더 안정적이고 일반화 지향적인 조합을 찾아냈다고 생각됨)
- 최종 결과해석
  - 성능 수치로는 Random Search가 조금 더 높아서(0.9936 vs 0.9924) 최적 파라미터로는 Random search가 선택되었지만
  - AUC는 둘다 높기도 하고 Optuna의 특성상 Optuna 결과가 일반화 성능에 더 유리할 가능성도 있다고 생각됨.
- 실제 현장에서의 활용?
  - Random Search
    - 빠른 학습률(0.2)과 비교적 단순한 구조(max\_depth=6, n\_estimators=200)로도 AUC 0.9936이라는 매우 높은 성능을 보였기 때문에 의료 현장에서 빠른 예측이 필요할 때 유리할 것으로 생각됨
    - 검진센터나 실시간 판독 지원 시 빠르고 계산 자원이 덜 드는 Random Search 모델이 적합할수있다
  - Optuna
    - 매우 작은 학습률(0.02)과 깊은 트리(max\_depth=8, n\_estimators=233)를 사용하여, 더 세밀하게 데이터를 학습했고 이는 재현율(Recall)을 더 확보해 악성 환자를 놓치지 않는 방향으로 일반화 성능을 강화했을 가능성이 있음
    - 의료 진단에서는 한 명의 악성 환자라도 놓치지 않는 것(민감도/Recall 우선) 이 중요하기 때문에 진단 보조, 재검 필요성 판단과 같은 실제 임상 적용 시 Optuna 모델이 더 신뢰할 만하다고 생각된다.

## ▼ 7. 결과 - SHAP 기반 해석 (Bar Summary Plot, Beeswarm Plot)

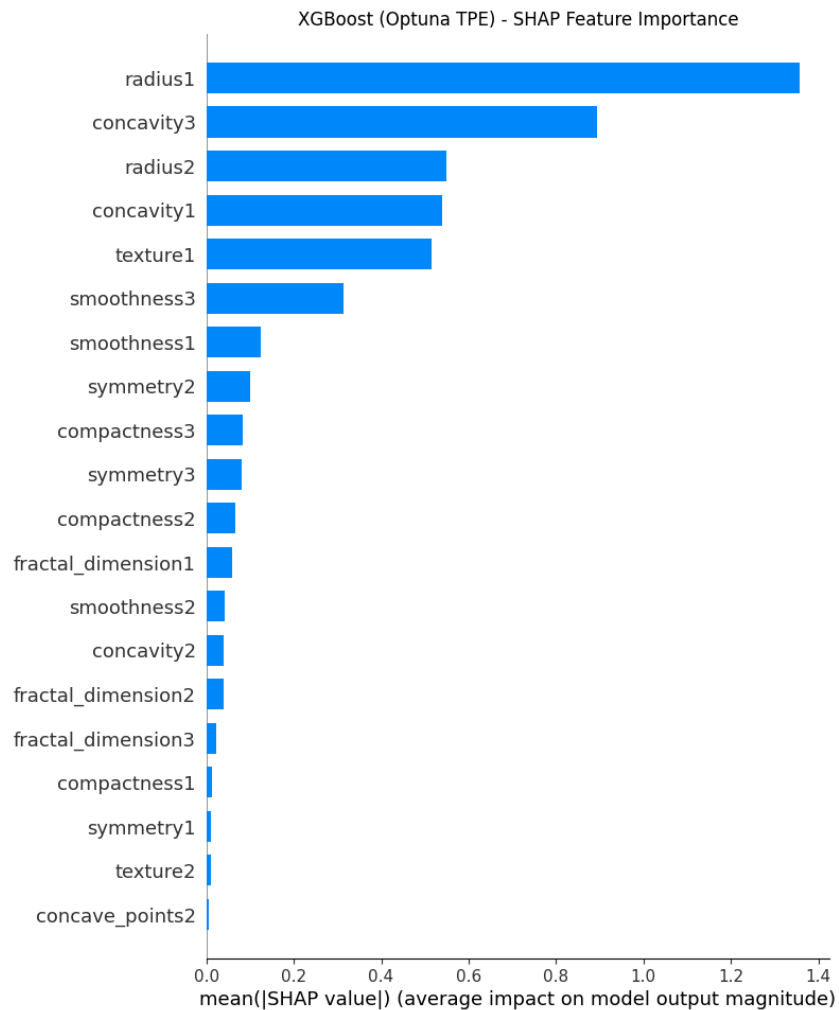
### 1. Bar Summary Plot

- 플롯 해석법

- Bar Summary Plot으로 확인 가능한것: 전역적 중요도 크기
- 봐야하는것: 막대 길이
  - 막대 길이: 길수록 전체 모델 예측에서 해당 변수가 차지하는 중요도가 크다.
- RandomSearchCV 결과
  - ▼ 플롯



- radius1 (2.219)와 concavity3 (1.483), radius2 (1.428), texture1 (1.412)가 가장 길다 → 전역적으로 중요한 변수임을 의미.
- compactness2 (0.557), symmetry3 (0.534), smoothness1 (0.496)은 중간 정도 기여.
- Optuna TPE 결과
  - ▼ 플롯



- radius1 (1.357)와 concavity3 (0.895), radius2 (0.550), concavity1 (0.540), texture1 (0.516)이 상위 그룹.
- 나머지 변수(smoothness3 0.315 이하)는 기여도가 급격히 낮음.

- 해석

- 전역적으로 보면 두 모델 모두 radius1과 concavity 관련 변수들을 가장 중요한 특징으로 본다.
- 차이는 RandomSearchCV는 texture1도 radius와 비슷한 수준으로 중요하게 평가했고, Optuna TPE에서는 concavity1이 더 높게 올라왔다
- 따라서 Optuna는 형태적 지표(concavity)를 더 강조한 반면, RandomSearch는 크기 지표(radius, texture)를 좀 더 고르게 반영했다.

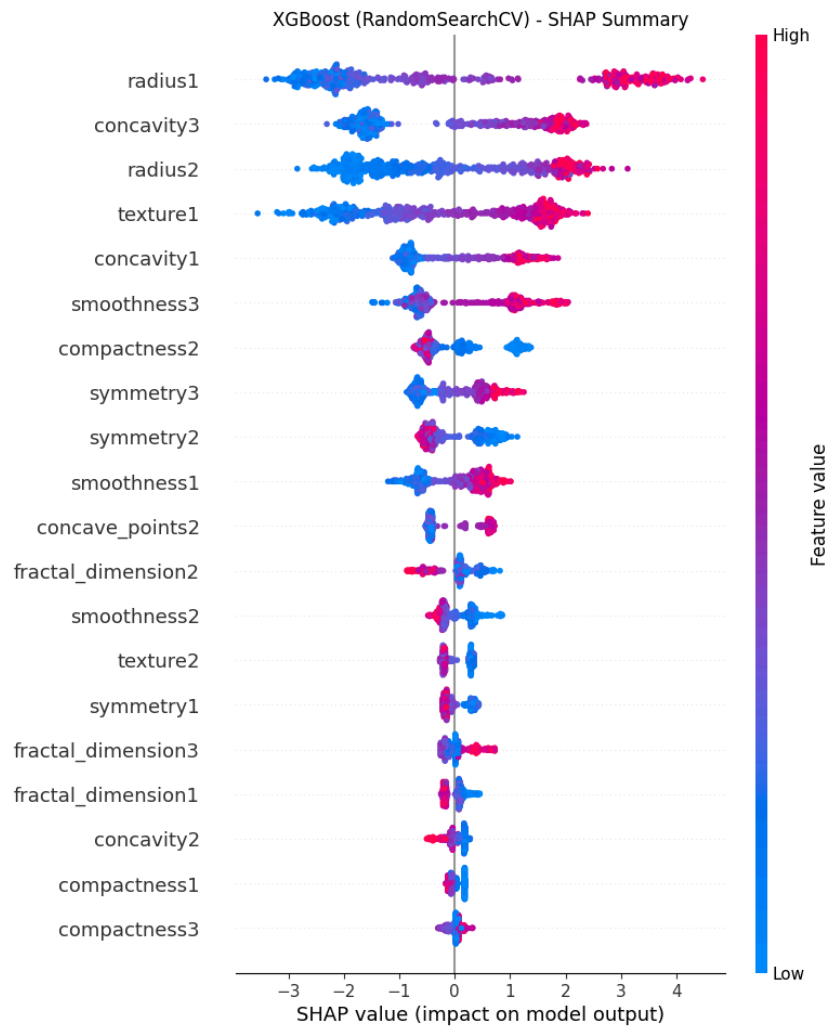
## 2. Beeswarm Plot

- 플롯 해석법

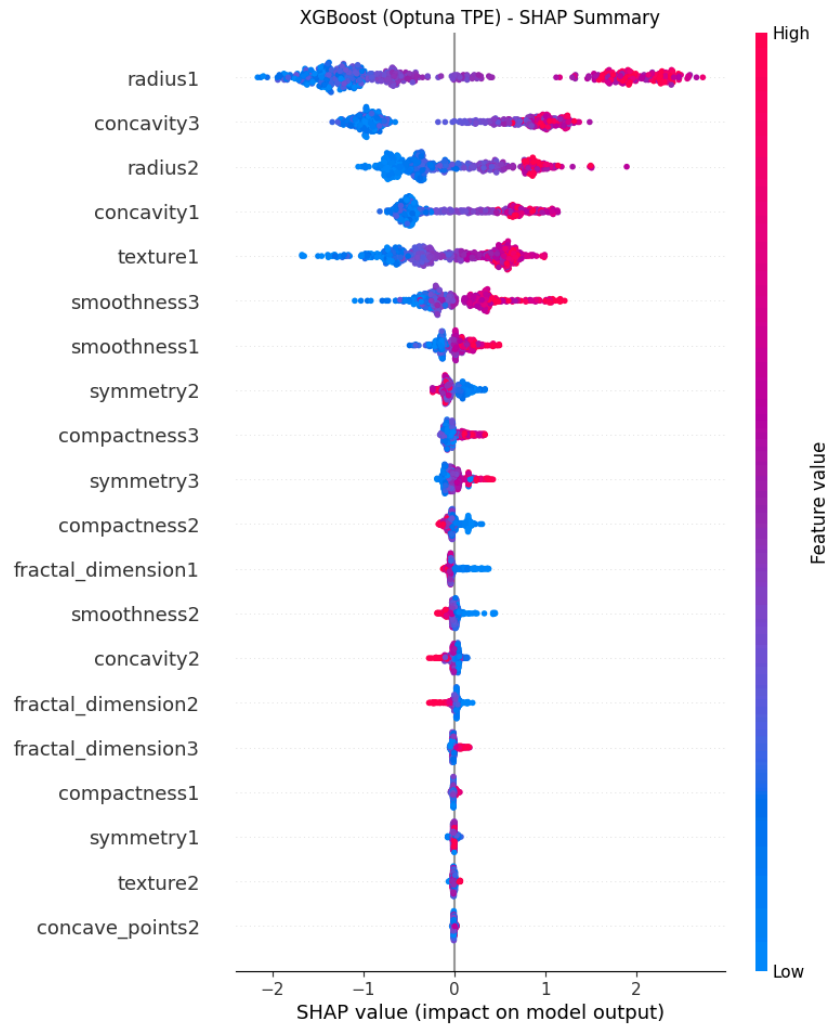
- Beeswarm Plot으로 확인 가능한것: 전역적 중요도 + 값의 방향성/분포
- 봐야하는것: 막대 길이, 방향성, 색깔
  - 막대 길이: 분포가 넓을수록 샘플별로 변수 효과가 다양하다.
  - 방향성: +일수록 예측값을 올린다, -일수록 예측값을 낮춘다.
  - 색깔: 빨강 = 변수 값이 크다, 파랑 = 변수 값이 작다.

- RandomSearchCV 결과

- ▼ 플롯



- radius1: 빨강(값 큼)일 때 오른쪽(+), 파랑(값 작음)일 때 왼쪽(-) → radius가 크면 악성 확률↑, 작으면↓.
- concavity3: 값이 클수록(+ 방향) 악성 확률↑.
- texture1: 값이 큰 샘플은 오른쪽(↑), 작은 샘플은 왼쪽(↓).
- 전반적으로 주요 변수들은 선형적으로 값이 클수록 악성 확률을 올리는 패턴을 보이는것으로 생각됨
- Optuna TPE 결과
  - ▼ 플롯



- radius1, concavity3에서 동일한 패턴 확인됨.
- 다만 Optuna에서는 concavity1 기여도가 texture1보다 크며, 샘플별 변동 폭도 상대적으로 큼.

#### ▼ 기여도?

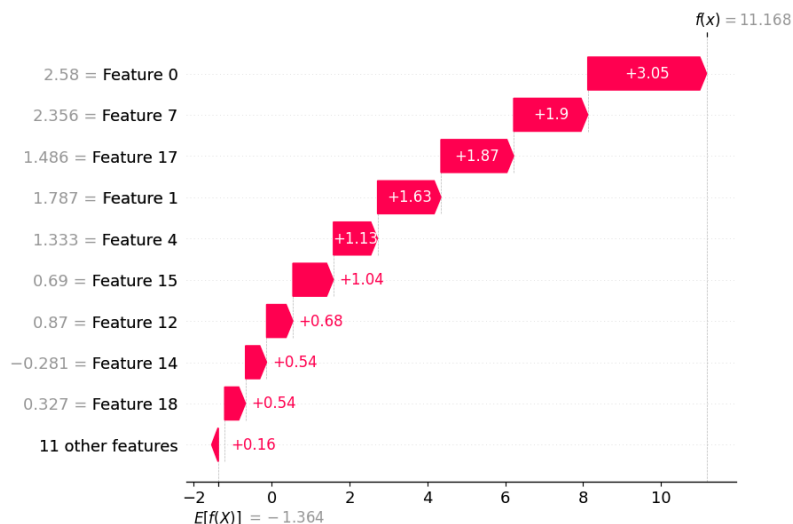
- texture1이 concavity1(-1.5 ~ +1)보다 더 넓은 범위로 퍼져 있음(-2~+2) 즉 가로 길이 더 길긴 하지만, concavity1이 가운데(0 근처)에 밀집된 부분이 더 두껍고 일부 샘플에서 영향이 크게 나타난다(대부분의 점들은 0 근처에 몰려 있지만, 몇몇 점들이 오른쪽(+1 이상)이나 왼쪽(-1 이하)까지 튀어나와 있음)
- concavity1이 밀집도가 높고 특정 방향으로 영향력이 크게 쏠려 있어서 concavity1 기여도가 texture1보다 크다. texture1은 전체적으로 넓게 퍼져 있어서 평균적으로 다양한 샘플에서 고르게 영향을 미침.
- 즉 Optuna는 concavity 기반 분포 차이를 더 민감하게 반영.
- 해석
  - RandomSearchCV는 radius(크기)와 concavity(형태)가 비슷하게 작동하는 균형적 구조.
  - Optuna는 concavity 변수의 분포 차이에 따른 영향을 더 집중적으로 반영.
  - 두 모델 모두 공통적으로
    - 빨강+양수 → 변수 값이 크면 악성 확률이 높다고 말함.
    - 파랑+음수 → 값이 작으면 악성 확률이 낮아진다고 말함.

## ▼ 8. 결과 - SHAP 기반 해석 (Waterfall Plot)

- 플롯 해석법
  - Waterfall Plot으로 확인 가능한 것: 개별 사례 해석



- 봐야하는것: 막대 길이, 방향성
  - 막대 길이: 길수록 해당 샘플의 예측에 기여한 정도가 크다.
  - 방향성: +일수록 예측값을 올리는 요인, -일수록 낮추는 요인.
- 개별 사례 해석할 샘플?
  - 3가지 기준으로 선택해보기.
- 1. 예측 확률 기준
  - 척도: 모델이 출력한 predict\_proba 값(악성일 확률)
  - 선택된 샘플
    - 모델이 가장 확신 있게 악성이라고 본 샘플과, 모델이 가장 불확실하게 분류한 샘플
    - index 236: 예측 확률 1.0000 → 모델이 거의 100% 악성으로 판단. → 전형적인 악성 특징을 강하게 가진 환자 케이스일 가능성.
    - index 297: 예측 확률 0.8534 → 0.5보다는 높지만 애매한 수준. → 경계선적 특성을 가진 케이스로, 모델이 불확실해 함.
- 2. SHAP 기여 강도 기준
  - 척도: 각 샘플별 SHAP 값 절댓값의 합( $\sum |SHAP|$ )(특정 샘플에서 모델이 얼마나 강하게 반응했는지)
  - 선택된 샘플
    - SHAP 영향력이 가장 큰 샘플
    - index 0: SHAP 총합 15.8481 → 여러 피쳐들이 예측에 강하게 작용한 대표적 사례. → 모델 의사결정이 가장 뚜렷하게 드러나는 샘플로 해석 가치가 높음.
- 3. 도메인 특이 케이스 기준
  - 척도: radius1 값(종양 크기)을 기준으로 가장 작은 반경을 가진 사례 선택
  - 선택된 샘플
    - 반경이 가장 작은 샘플
    - index 101: radius1 = -2.0296 (스케일링 값, 실제로는 매우 작은 반경) → 작은 종양임에도 악성/양성을 어떻게 판단했는지 확인할 수 있는 케이스. → 임상적으로도 "작지만 의미 있는 종양"을 설명하는 데 활용 가능.
- 분석 결과
  - 샘플 index=236 & 297 (극단적 악성 vs 불확실 케이스)
    - RandomSearchCV
      - ▼ 플롯 & 결과



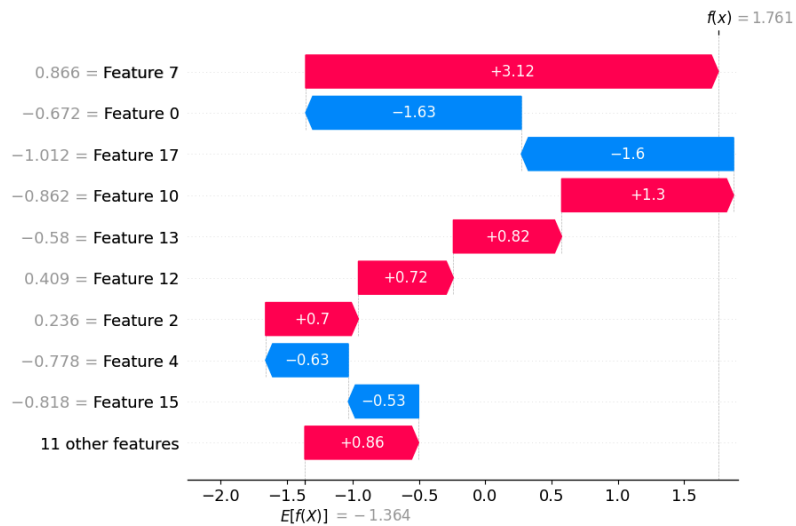
=== [Local Explanation: Print Only] XGBoost (RandomSearchCV) | 샘플 index=236 ===

Base value (bias): -1.3638296620842245

모델 최종 예측값: 11.168308656683315

상위 기여 피처:

	feature	feature_value	shap_value
0	radius1	2.579618	3.050362
7	radius2	2.356193	1.902044
17	concavity3	1.486328	1.867596
1	texture1	1.787269	1.627593
4	concavity1	1.333347	1.132707
15	smoothness3	0.689600	1.040196
12	concave_points2	0.870070	0.675197
14	fractal_dimension2	-0.280626	0.540320
18	symmetry3	0.327187	0.536057
10	compactness2	0.176156	-0.492998



=== [Local Explanation: Print Only] XGBoost (RandomSearchCV) | 샘플 index=297 ===

Base value (bias): -1.3638296620842245

모델 최종 예측값: 1.7612848288574072

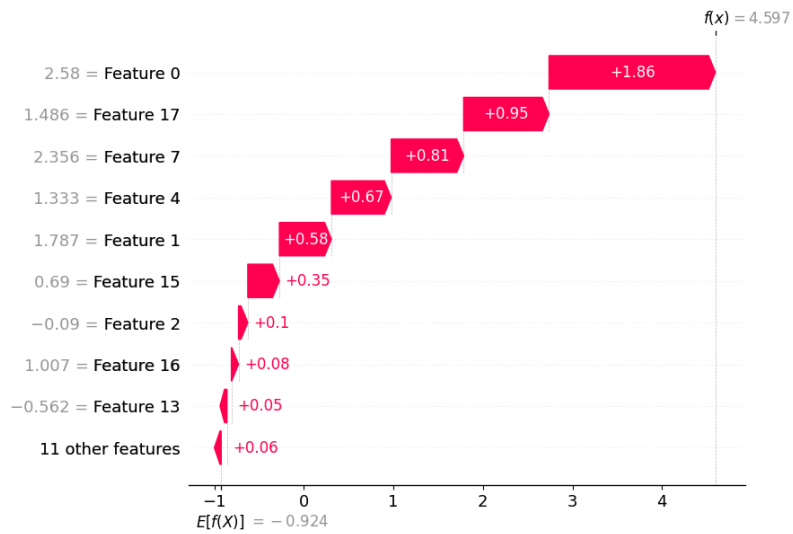
상위 기여 피처:

	feature	feature_value	shap_value
7	radius2	0.865589	3.118733
0	radius1	-0.672344	-1.629278
17	concavity3	-1.012222	-1.603995
10	compactness2	-0.862272	1.299105
13	symmetry2	-0.580244	0.818502
12	concave_points2	0.409395	0.717138
2	smoothness1	0.236249	0.701507
4	concavity1	-0.777772	-0.625091
15	smoothness3	-0.818354	-0.531707
14	fractal_dimension2	-0.392210	0.457600

- 236: radius·concavity 관련 피처가 강하게 양수 → 압도적 악성 예측 (logit 11.17)
- 297: radius2·compactness2는 악성 방향, radius1·concavity3는 양성 방향 → 양성·악성 요인이 충돌 → 불확실한 예측

■ Optuna TPE

▼ 플롯 & 결과



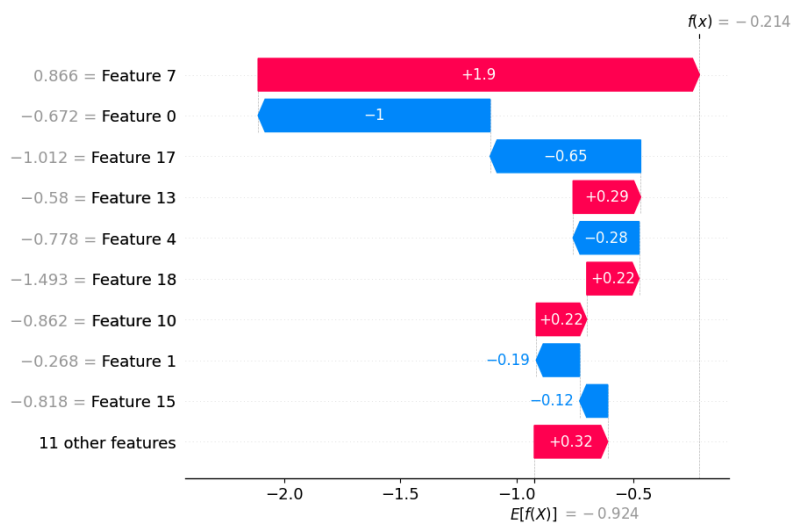
=== [Local Explanation: Print Only] XGBoost (Optuna TPE) | 샘플 index=236 ===

Base value (bias): -0.9241485576226455

모델 최종 예측값: 4.5968071827564225

상위 기여 피처:

	feature	feature_value	shap_value
0	radius1	2.579618	1.858232
17	concavity3	1.486328	0.954163
7	radius2	2.356193	0.809562
4	concavity1	1.333347	0.667691
1	texture1	1.787269	0.581037
15	smoothness3	0.689600	0.352877
2	smoothness1	-0.090400	0.103665
16	compactness3	1.007232	0.079603
13	symmetry2	-0.562083	0.050454
18	symmetry3	0.327187	0.040983



=== [Local Explanation: Print Only] XGBoost (Optuna TPE) | 샘플 index=297 ===

Base value (bias): -0.9241485576226455

모델 최종 예측값: -0.2141680640541357

상위 기여 피쳐:

	feature	feature_value	shap_value
7	radius2	0.865589	1.896786
0	radius1	-0.672344	-0.996925
17	concavity3	-1.012222	-0.647314
13	symmetry2	-0.580244	0.290723
4	concavity1	-0.777772	-0.283796
18	symmetry3	-1.492816	0.224899
10	compactness2	-0.862272	0.217224
1	texture1	-0.267531	-0.186536
15	smoothness3	-0.818354	-0.120681
5	symmetry1	-0.703240	0.075771

- 236: radius-concavity가 여전히 양수 기여, 다만 크기가 줄어듦 → 악성은 맞지만 더 보수적 예측 (logit 4.60)
- 297: radius2는 악성↑, 하지만 radius1-concavity3 음수 기여를 더 강하게 반영 → 최종적으로 양성 예측 (logit -0.21)

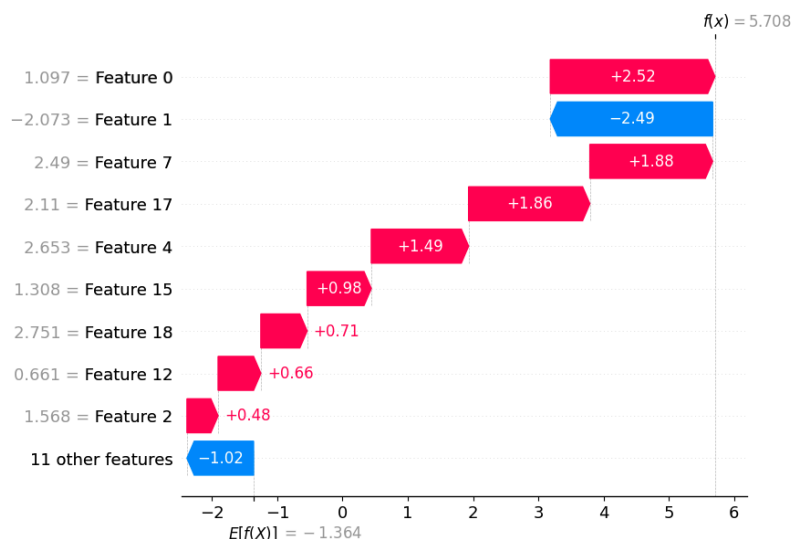
#### ■ 결과 해석

- 236은 가장 악성일 확률이 높은 케이스로써 모델이 얼마나 확신 있게 악성을 구분하는지 확인용이었고 297은 불확실 확률(0.85)을 보인 케이스로써 모델이 어떤 요인 때문에 혼동하는지 확인용이었다.
- 모델이 극단적 악성 샘플(236)은 두 방식 모두 확신 있게 잡아냄
- 불확실 케이스(297)는 RandomSearchCV는 애매하게 악성 쪽으로 기울고, Optuna는 양성 쪽으로 정리 → 튜닝 방법 차이가 불확실성 처리 방식에 반영됨.

#### ○ 샘플 index=0 (SHAP 영향력이 가장 큰 케이스)

##### ■ RandomSearchCV

##### ▼ 플롯 & 결과



=== [Local Explanation: Print Only] XGBoost (RandomSearchCV) | 샘플 index=0 ===

Base value (bias): -1.3638296620842245

모델 최종 예측값: 5.7077748983163685

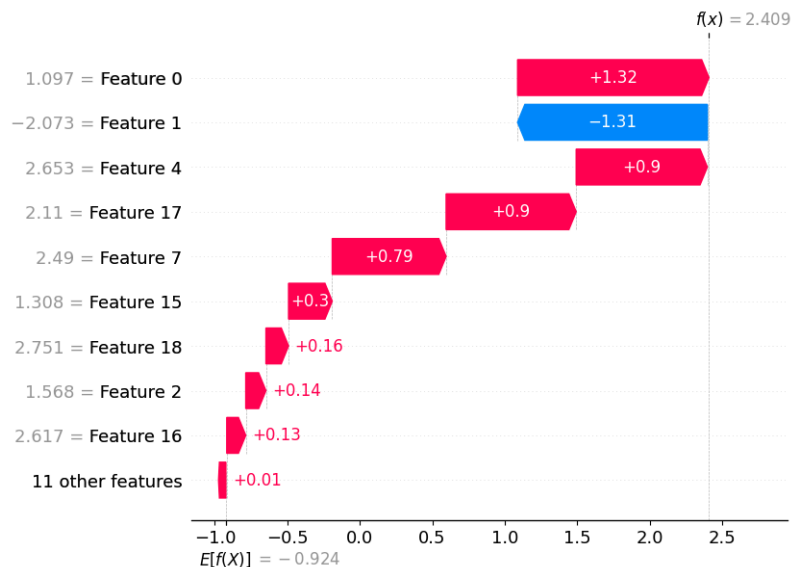
상위 기여 피쳐:

	feature	feature_value	shap_value
0	radius1	1.097064	2.523605
1	texture1	-2.073335	-2.487796
7	radius2	2.489734	1.881625
17	concavity3	2.109526	1.858648
4	concavity1	2.652874	1.491277
15	smoothness3	1.307686	0.983417
18	symmetry3	2.750622	0.709270
12	concave_points2	0.660820	0.655305
2	smoothness1	1.568466	0.477047
14	fractal_dimension2	0.907083	-0.456227

- radius1(+2.52), concavity3(+1.86), concavity1(+1.49) → 악성↑
- texture1(-2.49) → 양성↓
- 결과: 서로 강한 요인이 상충, 결국 악성으로 분류 (logit 5.71)

#### ■ Optuna TPE

##### ▼ 플롯 & 결과



=== [Local Explanation: Print Only] XGBoost (Optuna TPE) | 샘플 index=0 ===

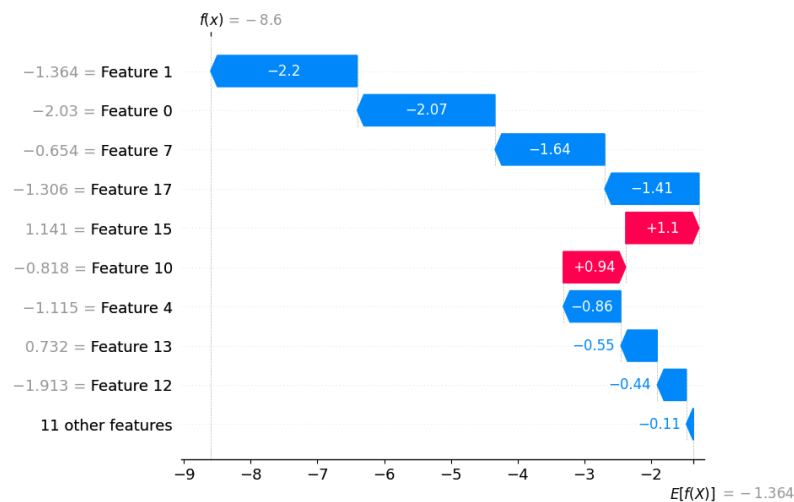
Base value (bias): -0.9241485576226455

모델 최종 예측값: 2.4085309240193253

상위 기여 피쳐:

	feature	feature_value	shap_value
0	radius1	1.097064	1.320182
1	texture1	-2.073335	-1.309145
4	concavity1	2.652874	0.904851
17	concavity3	2.109526	0.897306
7	radius2	2.489734	0.785772
15	smoothness3	1.307686	0.301214
18	symmetry3	2.750622	0.156071
2	smoothness1	1.568466	0.139268
16	compactness3	2.616665	0.131074
10	compactness2	1.316862	-0.086311

- 기여 패턴은 동일하나 기여 크기 축소 (예: radius1 +1.32, texture1 -1.31)
- 결과: 여전히 악성 예측이지만 확신도 낮음 (logit 2.41)
- 결과 해석
  - 0은 SHAP 총합이 가장 큰 샘플로써 모델이 "가장 복잡하게 해석하는" 사례를 분석하기 위함이었다.
  - 이 케이스는 모델 내부에서 가장 많은 변수들이 강하게 작용하는 복잡 패턴.
  - 두 방식 모두 같은 해석(크기·불규칙성↑ = 악성, texture↓ = 양성), 하지만 Optuna는 절대값이 낮게 나왔는데 과도한 확신을 안하는듯하다.
- 샘플 index=101 (radius1 가장 작은 케이스)
  - RandomSearchCV
    - ▼ 플롯 & 결과



=== [Local Explanation: Print Only] XGBoost (RandomSearchCV) | 샘플 index=101 ===

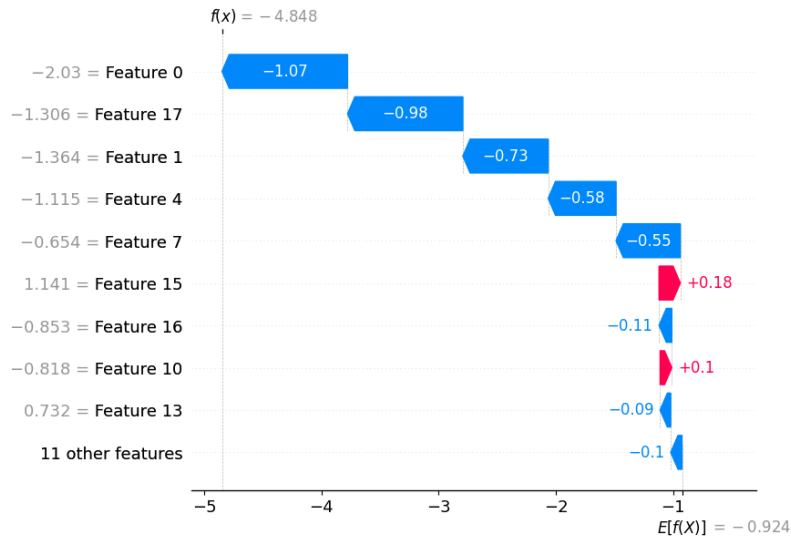
Base value (bias): -1.3638296620842245

모델 최종 예측값: -8.599803687339062

상위 기여 피쳐:

	feature	feature_value	shap_value
1	texture1	-1.363580	-2.195881
0	radius1	-2.029648	-2.065146
7	radius2	-0.653527	-1.644821
17	concavity3	-1.305831	-1.413159
15	smoothness3	1.141110	1.099035
10	compactness2	-0.818119	0.937021
4	concavity1	-1.114873	-0.863753
13	symmetry2	0.732247	-0.545934
12	concave_points2	-1.913447	-0.435124
2	smoothness1	1.468835	0.367233

- radius1(-2.07), texture1(-2.20), concavity3(-1.41) → 모두 양성 기여
- logit -8.60 → 강한 양성 판정
- Optuna TPE
  - ▼ 플롯 & 결과



=== [Local Explanation: Print Only] XGBoost (Optuna TPE) | 샘플 index=101 ===

Base value (bias): -0.9241485576226455

모델 최종 예측값: -4.848266282488437

상위 기여 피쳐:

	feature	feature_value	shap_value
0	radius1	-2.029648	-1.071077
17	concavity3	-1.305831	-0.983422
1	texture1	-1.363580	-0.727923
4	concavity1	-1.114873	-0.578903
7	radius2	-0.653527	-0.548200
15	smoothness3	1.141110	0.181355
16	compactness3	-0.852841	-0.109004
10	compactness2	-0.818119	0.100028
13	symmetry2	0.732247	-0.090747
9	smoothness2	1.049716	-0.082535

- 같은 변수들이 양성 기여(radius1 -1.07, concavity3 -0.98)
- 다만 기여 절대값이 줄어 logit -4.85 → 양성은 맞지만 완화된 확신

#### ■ 결과 해석

- 101은 도메인 특이 케이스로써 "작은 종양 크기(radius1↓)가 어떻게 해석되는가?"를 보기 위함이었다.
- 두 방식 모두 작은 크기(radius)와 낮은 concavity → 양성으로 해석.
- RandomSearchCV는 극단적 확신, Optuna는 완화된 확신 → 임상적 해석에서 Optuna가 더 현실적일 수 있음.

#### • 결론

- 236 & 297 (확신 vs 불확실 샘플)에서 모델이 어떻게 극단적 확신/애매한 상황을 처리하는지 확인했고 radius·concavity 관련 피쳐가 강하게 양수였던 RandomSearchCV와 달리 Optuna는 불확실성을 더 보수적으로 다뤄서(radius2·compactness2는 악성 방향, radius1·concavity3는 양성 방향) 최종 결과로는 양성으로 기운 해석을 제공했다.
- 0 (SHAP 총합 큰 샘플)은 모델이 가장 많은 변수 상호작용을 보이는 복잡 샘플이었는데 확인 결과 RandomSearchCV는 강한 확신, Optuna는 완화된 해석을 보여서 optuna가 좀더 현실적인 해석을 내놓는다고 생각된다.
- 101 (도메인 특이 케이스)로 작은 종양 크기의 양성 패턴 학습 여부 확인했는데 두 모델 모두 양성이었지만 optuna가 완화된 확신을 보여서 마찬가지로 좀더 현실적인 해석을 내놓는다고 생각된다.

## ▼ 9. Final Report - SHAP 결과 해석 및 임상 활용 방안

### 1. Bar Summary Plot 기반 - 전역적 중요도 분석

- 두 모델 모두 radius1(종양 크기)와 concavity(모양 불규칙성)를 핵심 지표로 식별하였고, RandomSearchCV는 texture(조직 패턴)까지 균형 있게 반영, Optuna는 concavity를 더 중점적으로 반영.
- 임상 활용 방안
  - 병리 진단 지원 시 “크기(radius)”와 “모양 불규칙성(concavity)”이 암 진단의 주요 기준과 일치한다면 SHAP 기반 해당 전역 해석은 의사·연구자에게 모델이 어떤 병리학적 지표를 중시하는지 투명하게 제시 가능할 것으로 생각됨.

### 2. Beeswarm Plot 기반 - 분포와 방향성 분석

- radius와 concavity 값이 클수록 악성 확률 ↑, 작을수록 ↓ 하다는 것은 병리학적 일반 지식과 일치한다.
- 임상 활용 방안
  - “크기가 크고 모양이 불규칙한 환자군”을 조기에 탐지하여 우선 검사 대상으로 분류하는 등 병원 데이터베이스에서 환자군별 위험 패턴 분석 시 활용 가능할 것으로 생각된다.

### 3. Force / Waterfall Plot 기반 개별 사례 분석

- 두가지 다른 하이퍼파라미터 튜닝 방식의 차이를 보면 극단적 확신 샘플(236)은 두 모델이 명확히 악성으로 분류했지만 불확실 샘플(297)은 RandomSearchCV는 악성 기울임, Optuna는 양성 쪽 보수적 해석을 했다
- 임상 활용 방안
  - 확신의 악성 샘플(236)과 불확실 샘플(297)의 특성을 잘 활용하면?
    - 236 환자는 진단 신뢰성 높아서 다른 분석을 할때도 샘플 데이터로 쓰기 좋을듯.
    - 두가지 튜닝 방식의 서로다른접근이 297와 같이 경계선에있는 환자 판단에 차이를 준것같다. 튜닝 특성 측면에서 도메인 반영 추가 분석을 하면 297 환자 뿐만아니라 경계 환자의 특성을 뽑아낼수있을것같다.
  - 전체적으로 완화된 예측을 제공하는 Optuna를 잘 활용하면?
    - “양성 판정”을 설명하면서도 확신 완화된 예측을 제공하면 불필요한 수술·치료를 줄이는 데 기여할수있을것으로 생각된다.

### 4. 결론 - SHAP 결과 3 plots 해석

- 병리학적으로 중요한 지표(radius, concavity)를 SHAP가 재확인해주기때문에 의사나 연구진에게 모델을 설명할때 모델 타당성 설명 가능.
- radius와 concavity 값이 클수록 악성 확률 ↑, 작을수록 ↓ 하다 같은 척도를 뽑았으니 이를 활용해서 환자군별 위험 패턴을 시각화하여 선별검사 및 예방의학에 활용 가능.
- 튜닝 방식(RandomSearchCV vs Optuna)에 따라 예측 방향과 강도가 달라지는 경계선 환자에 대한 도메인 및 튜닝 특성 반영 downstream 분석 가능.
- Optuna는 예측 기여도의 절대값을 완화시켜 불확실성을 반영하는 보수적 해석을 제공하는듯 하므로 Optuna로 하이퍼파라미터 최적화된 모델은 실제 임상에서 불필요한 치료를 줄이는 의사결정 지원 도구로 활용 가능할 것으로 생각됨.