AI 부트캠프 06기
Section 2 Project

[Movies on streaming platform]

강지호

Movies on Streaming Platform : Rotten Tomatoes 사이트의 영화 정보 데이터를 활용하여 target 예측

시나리오1

- 영화 제작사에서 만든 영화가 해당 사이트에서 어떤 평점을 받을지 예측해보고자 한다. Rotten Tomatoes에 등록되어 있는 영화의 정보(방영년도, 평점 등)와 target값인 Rotten Tomatoes(평점) 데이터가 있다.

- target: Rotten Tomatoes

- 평가지표: r2

- 회귀 모델

시나리오2

- OTT플랫폼인 회사에서 어떤 작품과 계약을 맺어야할지 알아보고자 한다. Rotten Tomatoes에 등록되어 있는 데이터는 영화의 정보(방영년도, 평점) 뿐만 아니라 Netflix와 같은 경쟁사가 해당 작품과 계약했는지의 여부까지 확인할 수 있다. 이를 활용하여 target값인 Recommend를 만들 수 있다.

- target: 새로운 Recommend feature 생성, IMDb와 Rotten Tomatoes가 특정 기준 이상일 경우 1로 코딩

- 평가지표: roc auc score

- 분류 모델

시나리오1 : 만들어진 영화가 Rotten Tomatoes에서 어떤 평점을 받을까?

- **데이터 설명 및 전처리**

1) **Age** : 결측치 43%, 특성 제거
2) **IMDb** : 문자 제거 후 숫자형 데이터로 변환 (7.8)
3) **Rotten Tomatoes** : target 이므로 결측치 행 삭제, 문자 제거 후 숫자형 데이터로 변환 (98)
4) **Type**: 0만 있으므로 특성 제거

	Title	Year	Age	IMDb	Rotten Tomatoes	Netflix	Hulu	Prime Video	Disney+	Type
0	The Irishman	2019	18+	7.8/10	98/100	1	0	0	0	0
1	Dangal	2016	7+	8.4/10	97/100	1	0	0	0	0
2	David Attenborough: A Life on Our Planet	2020	7+	9.0/10	95/100	1	0	0	0	0
3	Lagaan: Once Upon a Time in India	2001	7+	8.1/10	94/100	1	0	0	0	0

시나리오1 : 만들어진 영화가 Rotten Tomatoes에서 어떤 평점을 받을까?

- **데이터 설명 및 전처리**

1) **Age** : 결측치 43%, 특성 제거
2) **IMDb** : 문자 제거 후 숫자형 데이터로 변환 (7.8)
3) **Rotten Tomatoes** : target 이므로 결측치 행 삭제, 문자 제거 후 숫자형 데이터로 변환 (98)
4) **Type**: 0만 있으므로 특성 제거
5) **Directors**(7214), **Genres**(27), **Country**(130), **Language**(153) : IMDb 기준 Top 10 or 5 가 포함되어 있으면 1 else 0
   빈도수 기준 Top 10 or 5 가 포함되어 있으면 1 else 0

Directors	Genres	Country	Language	Runtime
Martin Scorsese	Biography,Crime,Drama	United States	English,Italian,Latin,Spanish,German	209.0
Nitesh Tiwari	Action,Biography,Drama,Sport	India,United States,United Kingdom,Australia,K...	Hindi,English	161.0
Alastair Fothergill,Jonathan Hughes,Keith Scholey	Documentary,Biography	United Kingdom	English	83.0
Ashutosh Gowariker	Drama,Musical,Sport	India,United Kingdom	Hindi,English	224.0

시나리오1 : 만들어진 영화가 Rotten Tomatoes에서 어떤 평점을 받을까?

- **데이터 설명 및 전처리**

1) **Age** : 결측치 43%, 특성 제거
2) **IMDb** : 문자 제거 후 숫자형 데이터로 변환 (7.8)
3) **Rotten Tomatoes** : target 이므로 결측치 행 삭제, 문자 제거 후 숫자형 데이터로 변환 (98)
4) **Type**: 0만 있으므로 특성 제거
5) **Directors**(7214), **Genres**(27), **Country**(130), **Language**(153) : IMDb 기준 Top 10 or 5 가 포함되어 있으면 1 else 0
   빈도수 기준 Top 10 or 5 가 포함되어 있으면 1 else 0

genre_imdb_top5	genre_count_top5	director_imdb_top10	director_count_top10	country_imdb_top5	country_count_top5	language_imdb_top5
0	0	0	0	0	1	0
0	1	0	0	0	1	0
1	0	0	0	0	1	0

시나리오1 : 만들어진 영화가 Rotten Tomatoes에서 어떤 평점을 받을까?

- **가설**

1) 여러가지 회귀모델 중, 랜덤포레스트와 XGB가 성능이 가장 좋을 것이다.

- 랜덤포레스트: 원본 데이터셋에서 랜덤하게 복원추출된 여러 데이터셋을 만들고 각각 독립적인 트리를 만든다. 각 트리의 예측 결과를 평균내어 예측하는 방법.
- XGB: 랜덤포레스트처럼 여러 데이터셋과 트리를 만드는데, 이전에 만들어진 트리가 다음 트리에 영향을 준다.

2) Target Encoder가 Ordinal Encoder보다 성능이 좋을 것이다.

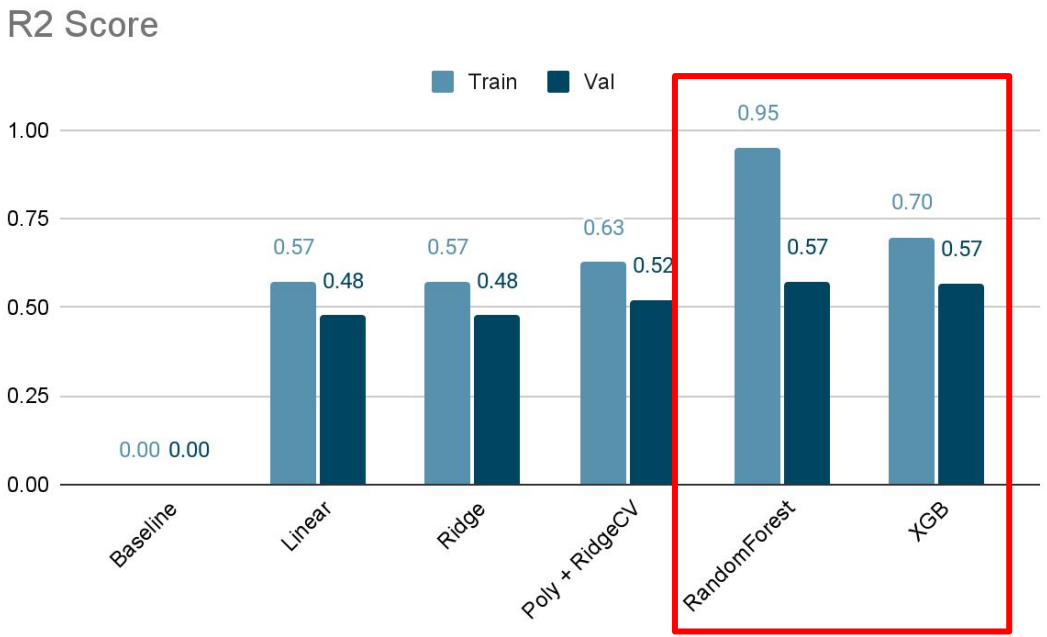
- Target Encoder: target값을 사용해서 인코딩
- Ordinal Encoder: 범주형 자료를 1부터 숫자로 변환하여 인코딩

3) XGB가 Randomforest보다 성능이 좋을 것이다.

시나리오1 : 만들어진 영화가 Rotten Tomatoes에서 어떤 평점을 받을까?

가설 1) 여러 모델 중 RandomForest와 XGB가 성능이 가장 좋을 것이다.

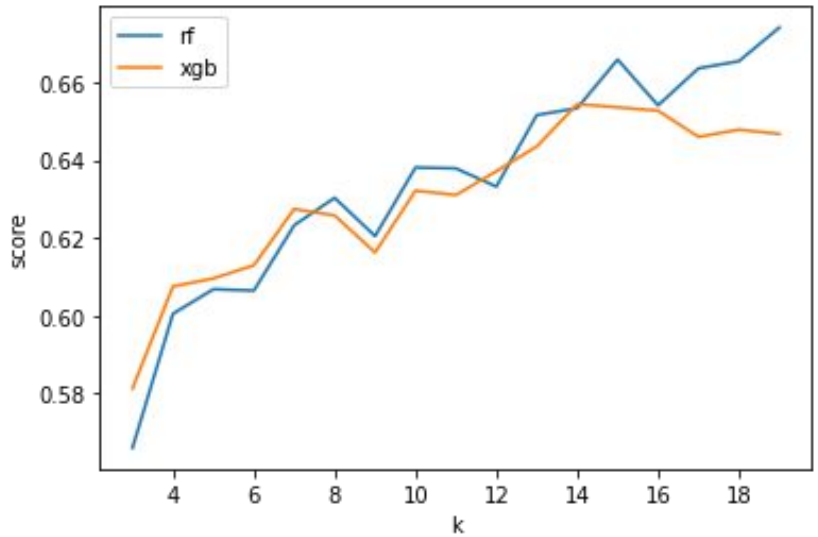
- RandomForest와 XGB의 성능이 가장 좋은 것으로 나타남.
- 다만, RandomForest의 경우 train set에서 과적합이 일어나므로, 하이퍼파라미터 조정이 필요해보임.



시나리오1 : 만들어진 영화가 Rotten Tomatoes에서 어떤 평점을 받을까?

가설 1) 여러 모델 중 RandomForest와 XGB가 성능이 가장 좋을 것이다. ✔

- CV를 통해 일반화될 가능성 확인: k가 증가할수록 r2 또한 증가.



시나리오1 : 만들어진 영화가 Rotten Tomatoes에서 어떤 평점을 받을까?

가설 2) Target Encoder가 Ordinal Encoder보다 성능이 좋을 것이다. ✔

- 하이퍼파라마미터 default 값으로 비교한 결과, Target Encoder가 성능이 더 좋음.

R2	RandomForestRegressor	XGBRegressor
Target Encoder	0.58	0.57
Ordinal Encoder	0.51	0.49

시나리오1 : 만들어진 영화가 Rotten Tomatoes에서 어떤 평점을 받을까?

가설 3) XGB가 RandomForest보다 성능이 좋을 것이다.

- RandomForest 모델 성능 개선

1. Randomized Search CV | n_iter=100, cv=20 val: 0.577 test: 0.601

- Target Encoder
  smoothing: [2.,4., 6., 8., 10., 20.,50.,60.,100.] -> 20
  min samples leaf: randint(1, 50) -> 37

- Random Forest Regressor
  n_estimators: randint(100, 1000) -> 478
  min_samples_leaf: randint(1, 100) -> 1
  max_depth: randint(5, 25) -> 13
  max_features: uniform(0, 1) -> 0.4930108

시나리오1 : 만들어진 영화가 Rotten Tomatoes에서 어떤 평점을 받을까?

가설 3) XGB가 RandomForest보다 성능이 좋을 것이다.

- XGB 모델 성능 개선

1. Randomized Search CV | n_iter=100, cv=20 **val: 0.597, test: 0.622**

- Target Encoder
  smoothing: [2.,4., 6., 8., 10., 20.,50.,60.,100.] -> 50
  min samples leaf: randint(1, 50) -> 2

- XGBRegressor
  max_depth: randint(5, 20) -> 6
  learning_rate: list(np.arange(0, 1, 0.001)) -> 0.115
  gamma: list(np.arange(0,5,0.5)) -> 1
  reg_alpha:: list(np.arange(0,1,0.001)) -> 0.114
  reg_lambda: list(np.arange(0,1,0.001)) -> 0.947

시나리오1 : 만들어진 영화가 Rotten Tomatoes에서 어떤 평점을 받을까?

가설 3) XGB가 RandomForest보다 성능이 좋을 것이다.

- XGB 모델 성능 개선

2. Randomized Search CV로 찾은 최적의 파라미터 + early stopping으로 n estimator 조절

val: 0.595, test: 0.624

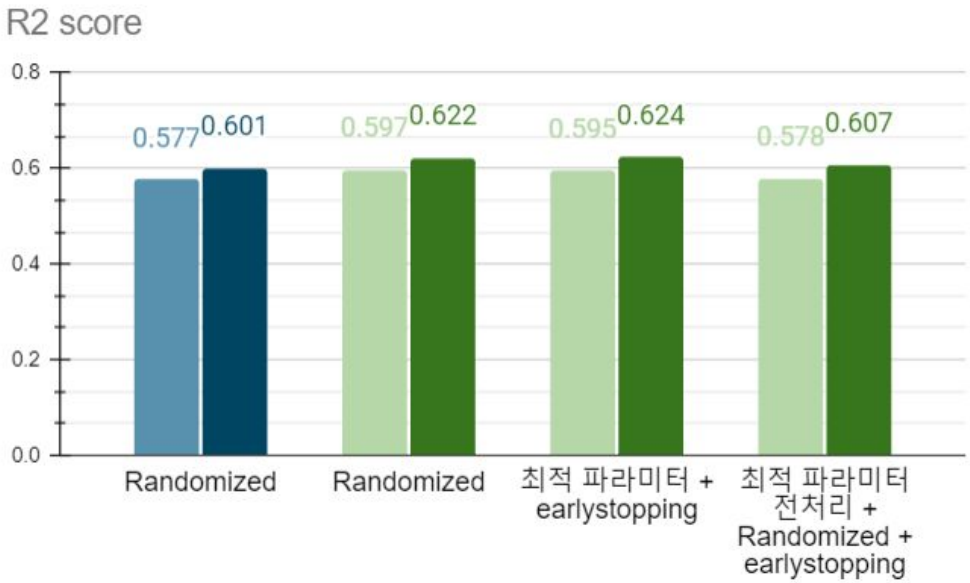
3. Randomized Search CV로 찾은 최적의 파라미터 중 Encoder만 사용
+ Randomized Search CV + early stopping

val: 0.578, test: 0.607

시나리오1 : 만들어진 영화가 Rotten Tomatoes에서 어떤 평점을 받을까?

가설 3) XGB가 RandomForest보다 성능이 좋을 것이다.

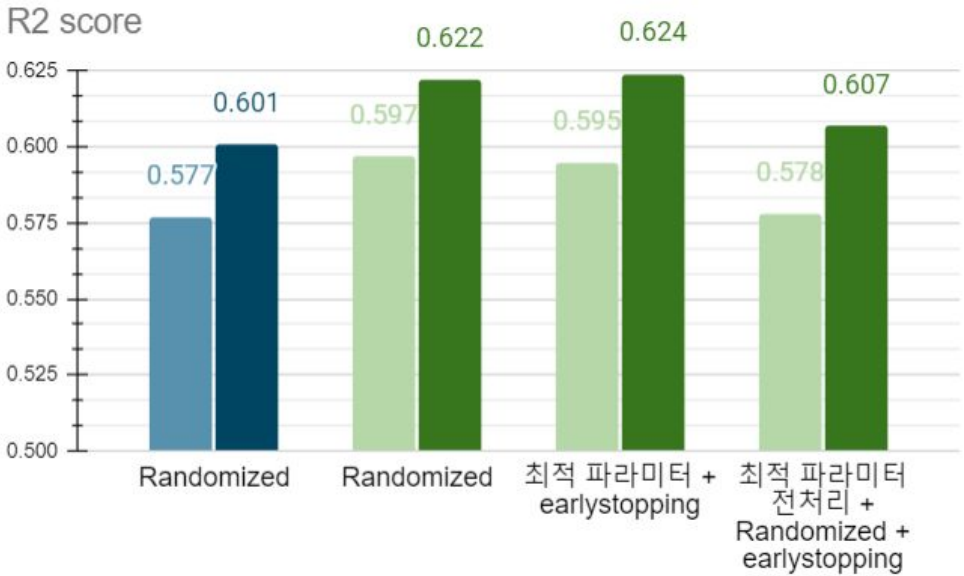
- **RandomForest** vs **XGB**



시나리오1 : 만들어진 영화가 Rotten Tomatoes에서 어떤 평점을 받을까?

가설 3) XGB가 RandomForest보다 성능이 좋을 것이다. ✔

- **RandomForest vs XGB** 다만, **test data**에 대한 모델의 성능이 **0.62**정도로 좋지 않은 편이다.



시나리오1 : 만들어진 영화가 Rotten Tomatoes에서 어떤 평점을 받을까?

모델 해석

- **Permutation Importances**

  각 특성마다 한 번씩 무작위로 노이즈를 주어 기존에 특성이 하던 역할을 하지 못하게 하여 특성의 중요도(영향력의 크기)를 평가

Weight	Feature
[illegible]	[illegible]
0.0885 ± 0.0049	Runtime
0.0802 ± 0.0121	Genres
0.0617 ± 0.0085	Country
0.0547 ± 0.0138	Directors
0.0372 ± 0.0087	Prime Video
0.0302 ± 0.0080	Year
0.0258 ± 0.0074	Language
0.0157 ± 0.0067	genre_imdb_top5
0.0103 ± 0.0018	Hulu
0.0103 ± 0.0029	Netflix
0.0046 ± 0.0020	language_count_top5
0.0039 ± 0.0029	genre_count_top5
0.0032 ± 0.0029	country_count_top5
0.0032 ± 0.0021	Disney+
0.0002 ± 0.0003	director_count_top10
0 ± 0.0000	language_imdb_top5
0 ± 0.0000	director_imdb_top10
0 ± 0.0000	country_imdb_top5
0 ± 0.0000	Title

- IMDb 외의 특성은 영향력의 크기가 작음

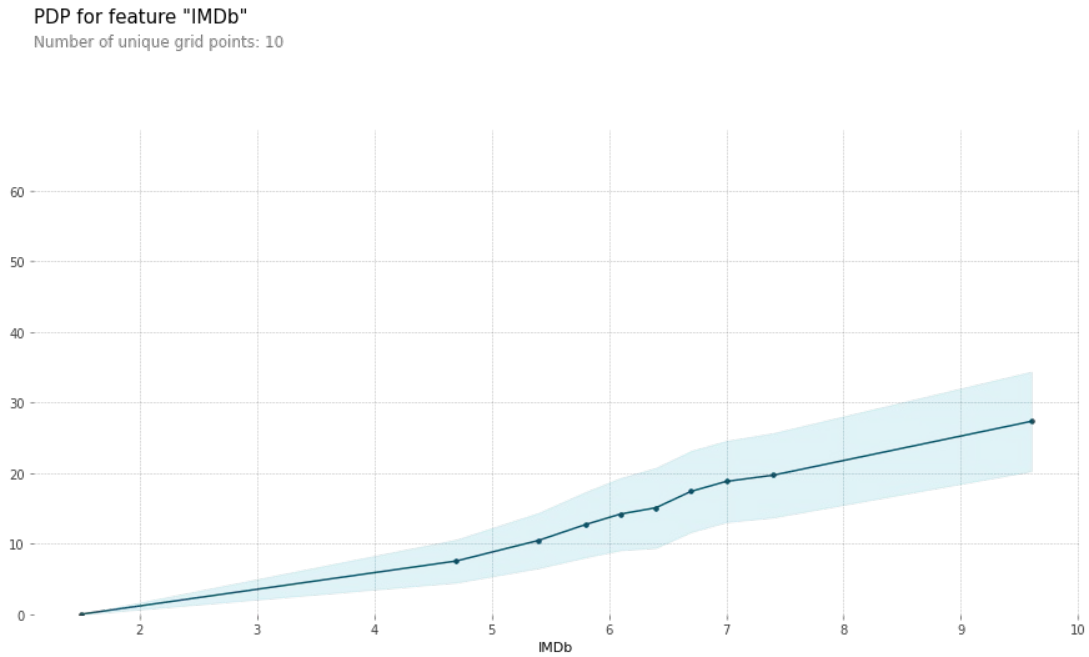
시나리오1 : 만들어진 영화가 Rotten Tomatoes에서 어떤 평점을 받을까?

모델 해석

- **PDP**

  특성의 영향력의 방향을 파악

  - IMDb: 양(+)의 영향력



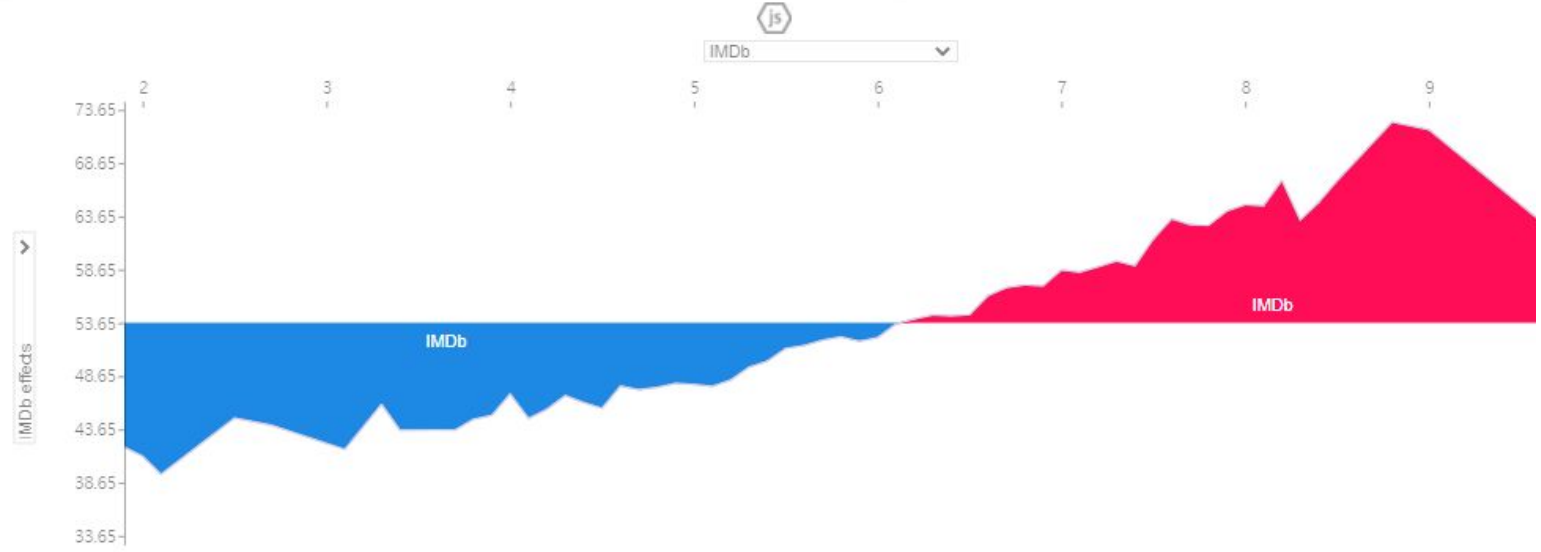
시나리오1 : 만들어진 영화가 Rotten Tomatoes에서 어떤 평점을 받을까?

모델 해석

- **SHAP**

  n개의 샘플이 갖는 특성들의 기여도 ex) IMDb: 6 이상일 경우 +, 6 이하일 경우 - 영향력

```
shap.initjs()
shap_values = explainer.shap_values(X_test_ready.iloc[:500])
shap.force_plot(explainer.expected_value, shap_values, X_test_ready.iloc[:500])
```



시나리오1 : 만들어진 영화가 Rotten Tomatoes에서 어떤 평점을 받을까?

- 정리

[가설]

1) 여러가지 회귀모델 중, 랜덤포레스트와 **XGB**가 성능이 가장 좋을 것이다.
2) **Target Encoder**가 **Ordinal Encoder**보다 성능이 좋을 것이다.
3) **XGB**가 **Randomforest**보다 성능이 좋을 것이다.

[해석]

- 최종 모델인 **XGB**의 **test data** 성능: **0.62**
- **Permutation Importance**로 확인한 결과, **IMDb** 특성 이외에는 영향력이 적기 때문에 성능이 좋지 않았음.

시나리오2 : 어떤 작품을 들여와야 할까?

- **데이터 설명 및 전처리**

1) **Age** : 결측치 43%, 특성 제거
2) **IMDb** : 문자 제거 후 숫자형 데이터로 변환 (7.8)
3) **Rotten Tomatoes** : 문자 제거 후 숫자형 데이터로 변환 (98)
4) **Type** : 0만 있으므로 특성 제거

	Title	Year	Age	IMDb	Rotten Tomatoes	Netflix	Hulu	Prime Video	Disney+	Type
0	The Irishman	2019	18+	7.8/10	98/100	1	0	0	0	0
1	Dangal	2016	7+	8.4/10	97/100	1	0	0	0	0
2	David Attenborough: A Life on Our Planet	2020	7+	9.0/10	95/100	1	0	0	0	0
3	Lagaan: Once Upon a Time in India	2001	7+	8.1/10	94/100	1	0	0	0	0

시나리오2 : 어떤 작품을 들여와야 할까?

- **데이터 설명 및 전처리**

1) **Age** : 결측치 43%, 특성 제거
2) **IMDb** : 문자 제거 후 숫자형 데이터로 변환 (7.8)
3) **Rotten Tomatoes** : 문자 제거 후 숫자형 데이터로 변환 (98)
4) **Type** : 0만 있으므로 특성 제거
5) **Directors, Genres, Country, Language** : 결측치 'missing'으로 변환
6) **Recommend** : (중위값 사용) IMDb >= 6.2 & Rotten Tomatoes >= 42 일 경우, 추천 1 else 0

Directors	Genres	Country	Language	Runtime	Recommend
Martin Scorsese	Biography,Crime,Drama	United States	English,Italian,Latin,Spanish,German	209.0	1
Nitesh Tiwari	Action,Biography,Drama,Sport	India,United States,United Kingdom,Australia,K...	Hindi,English	161.0	1
Alastair Fothergill,Jonathan Hughes,Keith Scholey	Documentary,Biography	United Kingdom	English	83.0	1
Ashutosh Gowariker	Drama,Musical,Sport	India,United Kingdom	Hindi,English	224.0	1

시나리오2 : 어떤 작품을 들여와야 할까?

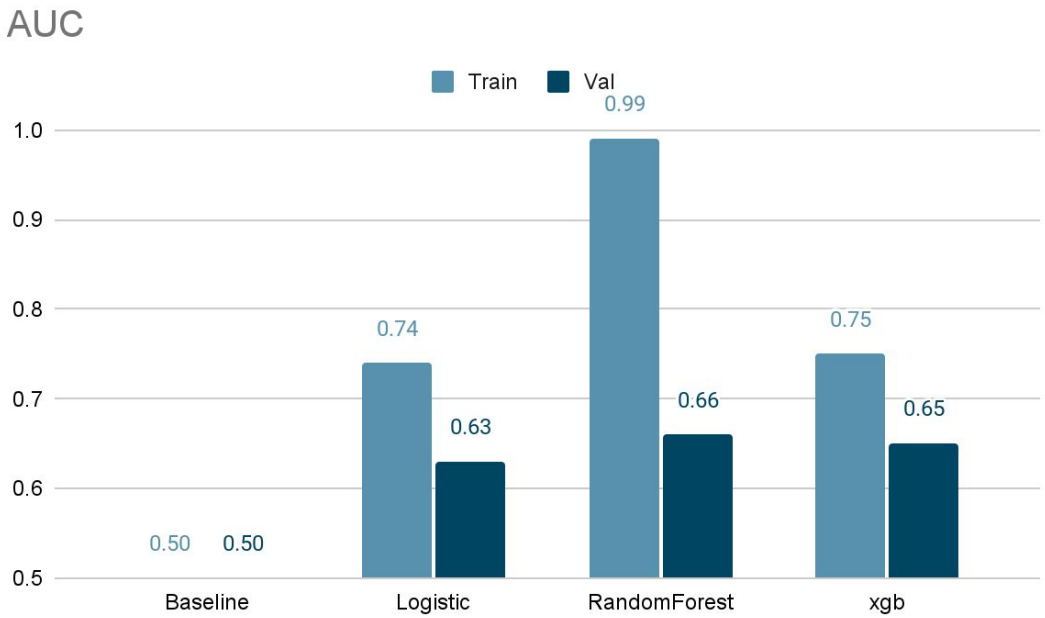
- **가설**

1) 분류모델 분석 방법 중, 랜덤포레스트와 XGB가 성능이 가장 좋을 것이다.
2) Target Encoder가 Ordinal Encoder보다 성능이 좋을 것이다.
3) XGB가 Randomforest보다 성능이 좋을 것이다.

시나리오2 : 어떤 작품을 들여와야 할까?

가설 1) 여러 모델 중 RandomForest와 XGB가 성능이 가장 좋을 것이다.

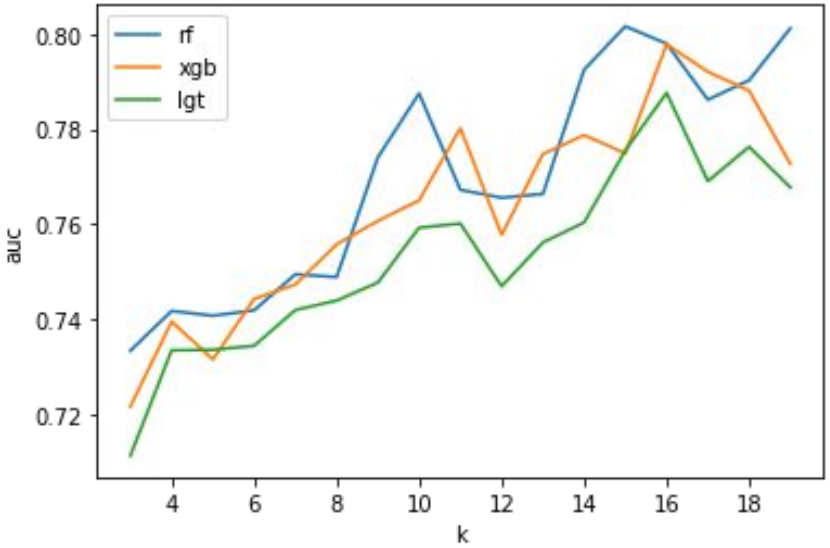
- 기본 파라미터의 경우, 세 모델 모두 val data에서 비슷한 점수를 보인다.
- 세 모델 모두 train set에서 과적합이 일어나므로, 하이퍼파라미터 조정이 필요해보임.



시나리오2 : 어떤 작품을 들여와야 할까?

가설 1) 여러 모델 중 RandomForest와 XGB가 성능이 가장 좋을 것이다.✔

- CV를 통해 일반화될 가능성 확인: k가 증가할수록 r2 또한 증가.
- RandomForest와 XGB가 Logistic보다 성능이 높은 것을 알 수 있다.



시나리오2 : 어떤 작품을 들여와야 할까?

가설 2) Target Encoder가 Ordinal Encoder보다 성능이 좋을 것이다. ✔

- 하이퍼파라마미터 default 값으로 비교한 결과, Target Encoder가 성능이 더 좋음.

AUC	RandomForestClassifier	XGBClassifier
Target Encoder	0.66	0.66
Ordinal Encoder	0.55	0.61

가설 3) XGB가 RandomForest보다 성능이 좋을 것이다.

- RandomForest 모델 성능 개선

1. Randomized Search CV val: 0.647 test: 0.626

- Target Encoder
  smoothing: [2.,4., 6., 8., 10., 20.,50.,60.,100.] -> 7
  min samples leaf: randint(1, 50) -> 4

- RandomForest Classifier
  n_estimators: randint(100, 1000) -> 546
  min_samples_leaf: randint(1, 100) -> 14
  min_samples_split: randint(1, 100) -> 39
  max_depth: randint(5, 25) -> 17
  max_features: uniform(0, 1) -> 0.106367

시나리오2 : 어떤 작품을 들여와야 할까?

가설 3) XGB가 RandomForest보다 성능이 좋을 것이다.

- XGB 모델 성능 개선

1. Randomized Search CV **val: 0.662, test: 0.672**

- Target Encoder
  smoothing: [2.,4., 6., 8., 10., 20.,50.,60.,100.] -> 10
  min samples leaf: randint(1, 50) -> 17

- XGBRegressor
  max_depth: randint(5, 20) -> 10
  learning_rate: list(np.arange(0, 1, 0.001)) -> 0.211
  gamma: list(np.arange(0,5,0.5)) -> 3
  reg_alpha:: list(np.arange(0,1,0.001)) -> 0.705
  reg_lambda: list(np.arange(0,1,0.001)) -> 0.701

시나리오2 : 어떤 작품을 들여와야 할까?

가설 3) XGB가 RandomForest보다 성능이 좋을 것이다.

- XGB 모델 성능 개선

2. Randomized Search CV로 찾은 최적의 파라미터 + early stopping으로 n estimator 조절

val: 0.672, test: 0.671

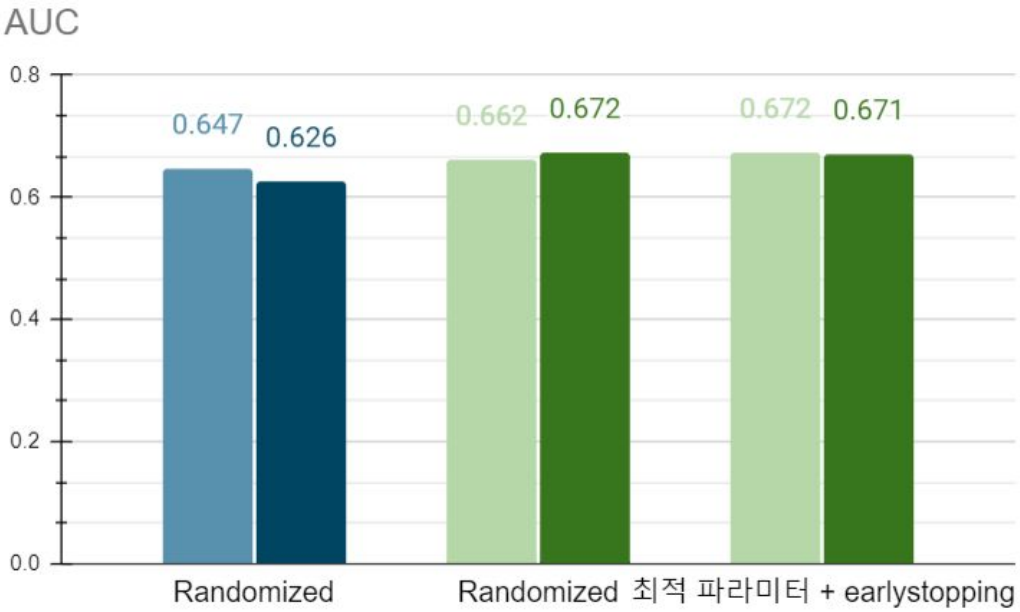
3. Randomized Search CV로 찾은 최적의 파라미터 중 Encoder만 사용
+ Randomized Search CV + early stopping

앞선 시나리오에서 **2**번이 제일 높은 성능을 보였으므로 **2**번까지만 진행.

시나리오2 : 어떤 작품을 들여와야 할까?

가설 3) XGB가 RandomForest보다 성능이 좋을 것이다.

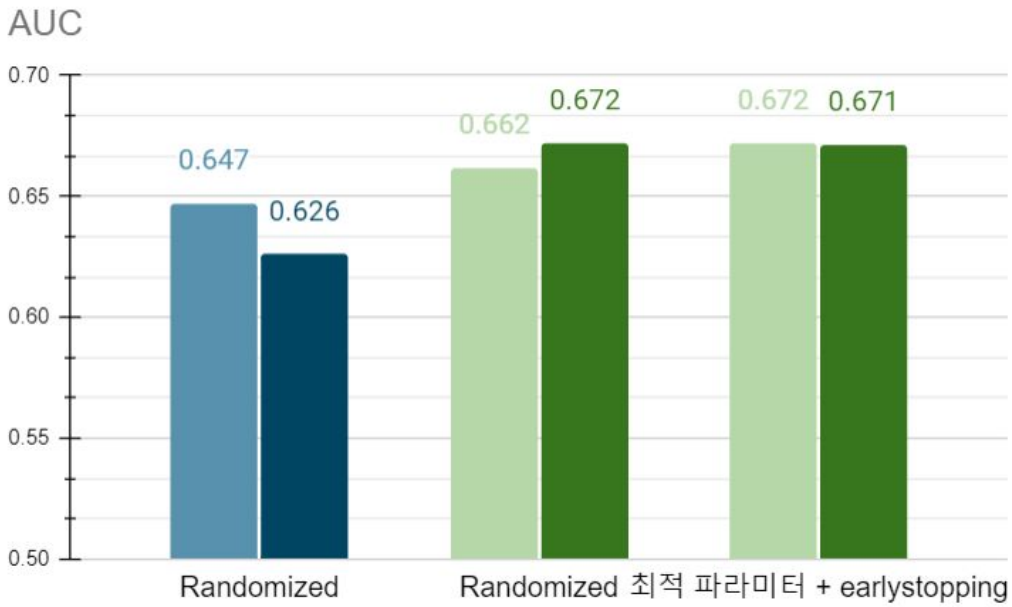
- RandomForest vs XGB



시나리오2 : 어떤 작품을 들여와야 할까?

가설 3) XGB가 RandomForest보다 성능이 좋을 것이다. ✔

- **RandomForest vs XGB** 다만, **test data**에 대한 모델의 성능이 **0.67**정도로 좋지 않은 편이다.

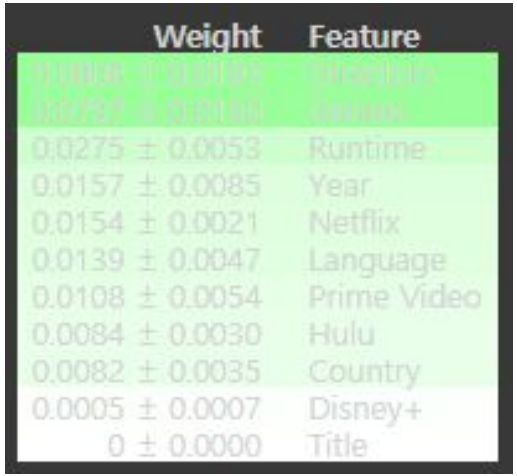


시나리오2 : 어떤 작품을 들여와야 할까?

모델 해석

- **Permutation Importances**

  특성마다 한 번씩 무작위로 노이즈를 주어
  기존에 특성이 하던 역할을 하지 못하게 하여
  특성의 중요도(영향력의 크기)를 평가



- 전체적으로 영향력이 약함을 알 수 있다.

- 특성의 영향력이 크지 않기 때문에,
  PDP와 Shap으로 살펴볼 의미가 없음.

시나리오2 : 어떤 작품을 들여와야 할까?

- 정리

[가설]

1) 여러가지 분석 모델 중, 랜덤포레스트와 **XGB**가 성능이 가장 좋을 것이다.
2) **Target Encoder**가 **Ordinal Encoder**보다 성능이 좋을 것이다.
3) **XGB**가 **Randomforest**보다 성능이 좋을 것이다.

[해석]

- 최종 모델인 **XGB**의 **test data** 성능: **0.67**
- **Permutation Importance**로 확인한 결과, 대부분의 특성이 영향력이 적었다.