

17

21회 기출 분석 (2)

[수업 목표]

이번 시간에는 21회 기출 문제 중 모델링 및 평가 문제를 풀어보겠습니다

[데이터 설명]

0:00

17.2 21회 기출문제 풀이 (2)

17.2.1 머신 러닝 (50점)

학생 성적에 관한 데이터셋은 제공 394행짜리 소규모 데이터.

1.4. 데이터 분할 방법을 2가지 쓰고 적절한 데이터 분할을 적용. 선택한 이유 설명.
 1.5. svm, xgboost, randomforest 3개의 알고리즘 공통점을 쓰고 이 예측 분석에 적합한 알고리즘인지 설명.
 1.6. 세 가지 모델 모두 모델링 해보고 가장 적합한 알고리즘 선택하고 이유 설명. 한계점 설명하고 보완 가능한 부분 설명.
 현업에서 사용시 주의할 점 등에 대해 기술.

```
In [ ]: import pandas as pd
import numpy as np
df = pd.read_csv("../student_data_2.csv")
df
```

```
In [ ]: df.info()
```

17.2.2 1.4. 데이터 분할 방법을 2가지 쓰고 적절한 데이터 분할을 적용. 선택한 이유 설명.

1) 랜덤 분할

- train test 데이터셋을 나누어서 학습된 데이터를 검증할 수 있음
- 분할 시에 무작위로 사용자가 지정하여 비율로 분할 함
- 전체 분석 데이터 중 머신러닝 모델을 학습시키기 위한 학습용 데이터와 테스트 데이터를 나누어서 적용시키는 이유는 모델 결과와 다른 데이터에도 적용 가능한지, 일반화가 가능한지를 검증하기 위함이다.

2) 층화 추출 기법

- 종속변수의 클래스의 비율이 학습용 데이터와 테스트용 데이터에 비율이 같게 분할함
- 클래스의 편향을 막을 수 있음

[데이터 분할 방법 및 적용]

02:20

17.2.2 1.4. 데이터 분할 방법을 2가지 쓰고 적절한 데이터 분할을 적용. 선택한 이유 설명.

1) 랜덤 분할

- train test 데이터셋을 나누어서 학습된 데이터를 검증할 수 있음
- 분할 시에 무작위로 사용자가 지정하여 비율로 분할 함
- 전체 분석 데이터 중 머신러닝 모델을 학습시키기 위한 학습용 데이터와 테스트 데이터를 나누어서 적용시키는 이유는 모델 결과와 다른 데이터에도 적용 가능한지, 일반화가 가능한지를 검증하기 위함이다.

2) 층화 추출 기법

- 종속변수의 클래스의 비율이 학습용 데이터와 테스트용 데이터에 비율이 같게 분할함
- 클래스의 편향을 막을 수 있음
- 종속변수가 범주형 변수인 분류분석에 사용

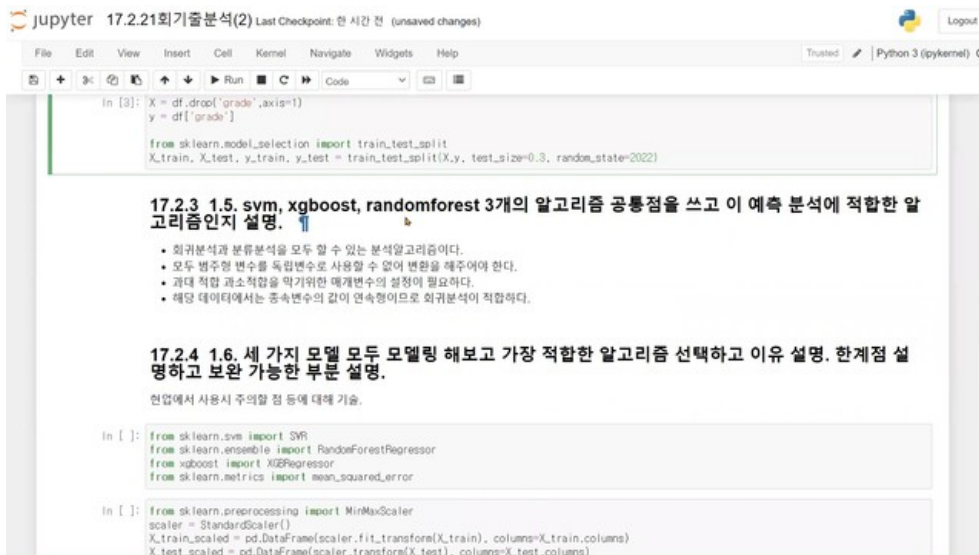
해당 데이터는 종속변수가 연속형이므로, 회귀분석을 사용한다. 그러므로 층화추출기법을 사용한 분할이 아닌 랜덤 샘플링을 통한 분할을 사용하며, 7:3 비율로 분할하였다.

```
In [ ]: X = df.drop('grade', axis=1)
```

[머신러닝 모델 선택 및 이유 서술]

03:50

#모델들의 장단점을 기억해주시는 것이 중요합니다.



Jupyter 17.2.21회기출분석(2) Last Checkpoint: 한 시간 전 (unsaved changes)

File Edit View Insert Cell Kernel Navigate Widgets Help

Python 3 (ipykernel)

```
In [3]: X = df.drop('grade',axis=1)
y = df['grade']

from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=2022)
```

17.2.3 1.5. svm, xgboost, randomforest 3개의 알고리즘 공통점을 쓰고 이 예측 분석에 적합한 알고리즘인지 설명.

- 회귀분석과 분류분석을 모두 할 수 있는 분석알고리즘이다.
- 모두 범주형 변수를 독립변수로 사용할 수 없어 변환을 해주어야 한다.
- 과대 적합 과소적합을 막기위한 매개변수의 설정이 필요하다.
- 해당 데이터에서는 종속변수의 값이 연속형이므로 회귀분석이 적합하다.

17.2.4 1.6. 세 가지 모델 모두 모델링 해보고 가장 적합한 알고리즘 선택하고 이유 설명. 한계점 설명하고 보완 가능한 부분 설명.

현업에서 사용시 주의할 점 등에 대해 기술.

```
In [ ]: from sklearn.svm import SVC
from sklearn.ensemble import RandomForestRegressor
from xgboost import XGBRegressor
from sklearn.metrics import mean_squared_error

In [ ]: from sklearn.preprocessing import MinMaxScaler
scaler = StandardScaler()
X_train_scaled = pd.DataFrame(scaler.fit_transform(X_train), columns=X_train.columns)
X_test_scaled = pd.DataFrame(scaler.transform(X_test), columns=X_test.columns)
```

[다음 수업 예고]

다음 시간에는 통계분석 파트 문제를 풀어보겠습니다