|  |  |  |
| --- | --- | --- |
| **| 학습주제** | | * Python에서 선형 회귀와 다항 회귀를 사용하여 교통량 예측 비교 |
| **| 학습목표** | | * Python에서 선형 회귀와 다항 회귀 모델을 학습하고 성능을 비교하는 방법을 이해한다. * 교통량 데이터와 같은 시계열 데이터를 처리한 후, 각 회귀 모델의 성능을 비교한다. * 두 모델의 예측 성능을 평균 제곱 오차(MSE) 및 루트 평균 제곱 오차(RMSE)로 평가하고 어떤 모델이 더 적합한지 비교한다. |
| **| 학습 개념** | 선형 회귀는 독립 변수와 종속 변수 간의 선형 관계를 기반으로 데이터를 분석하는 기법입니다.  다항 회귀는 독립 변수를 다항식으로 변환하여 비선형 관계를 반영하는 모델입니다.  이 예제에서는 각 날의 8시 교통량 데이터를 독립 변수로 사용하여 두 모델을 학습한 후, 두 모델의 예측 성능을 비교합니다. MSE, RMSE를 통해 어떤 모델이 해당 데이터에 대해서 더 나은 성능을 보이는지 확인합니다. | |
| **| 학습 방향** | 이번 예제에서는 날짜별 8시 교통량 데이터를 불러와서 날짜를 독립 변수, 8시 교통량을 종속 변수로 설정합니다. 그 후, train\_test\_split 함수를 사용하여 데이터를 훈련 세트와 테스트 세트로 분리합니다. 선형 회귀 모델을 학습한 뒤, 다항 회귀 모델을 학습하고, 각각의 모델에 대해 예측을 수행합니다. 마지막으로, MSE, RMSE 값을 통해 두 모델의 성능을 비교하고, 어느 모델이 더 나은 성능을 보였는지 확인합니다. | |
| **| 요구 사항** | 1. 데이터 준비 및 전처리  - pandas를 사용하여 데이터를 불러오고, 날짜를 나타내는 인덱스를 독립 변수로, 8시 교통량 데이터를 종속 변수로 설정하세요.  2. 데이터 분리  - train\_test\_split 함수를 사용하여 데이터를 훈련 데이터(80%)와 테스트 데이터(20%)로 분리하세요.  3. 선형 회귀 및 다항 회귀 모델 학습  - 선형 회귀 모델을 사용하여 데이터를 학습한 후, 예측을 수행하세요.  - PolynomialFeatures를 사용하여 다항 특성을 생성하고, 다항 회귀 모델을 학습시키세요.  4. 성능 비교 및 평가  - 두 모델 모두에 대해 평균 제곱 오차(MSE)와 루트 평균 제곱 오차(RMSE)를 계산하여 성능을 비교하세요.  - 두 모델 중 어느 모델의 성능이 더 나은지 확인하세요. | |