|  |  |  |
| --- | --- | --- |
| **| 학습주제** | | * Python에서 선형 회귀 분석을 통해 교통량을 예측하는 방법 |
| **| 학습목표** | | * Python에서 선형 회귀 모델을 학습하는 방법을 이해한다. * 교통량 데이터와 같은 시계열 데이터를 처리하고 예측하는 과정을 학습한다. * train\_test\_split 함수를 사용하여 데이터를 훈련 세트와 테스트 세트로 분리하는 방법을 학습한다. * 선형 회귀 모델을 사용하여 훈련 데이터로 학습하고, 테스트 데이터를 예측하는 과정을 배운다. |
| **| 학습 개념** | 선형 회귀는 두 변수 간의 선형 관계를 기반으로 데이터를 예측하는 모델입니다. 이 예제에서는 각 날의 8시 교통량 데이터를 독립 변수로 하여 모델을 학습하고, 마지막 날의 교통량을 예측합니다. 데이터는 훈련 세트와 테스트 세트로 나누어져 있으며, 모델은 훈련 세트에서 학습되고, 테스트 세트로 성능을 검증합니다. LinearRegression 모델을 사용하여 이 관계를 추정하며, train\_test\_split 함수로 데이터를 나누어 적절한 학습 및 테스트 환경을 조성합니다. 이 모델은 각 날을 인덱스(독립 변수)로, 해당 날의 교통량(종속 변수)을 예측하는 방식으로 학습됩니다. | |
| **| 학습 방향** | 이번 예제에서는 특정 시간대의 교통량 데이터를 사용해 모델을 학습하고, 마지막 날의 교통량을 예측하는 과정입니다. 이를 위해 먼저 데이터를 불러와 특정 열의 값을 선택하고, 훈련 데이터와 테스트 데이터를 나눈 후, 선형 회귀 모델을 사용하여 학습을 진행합니다. 이 과정을 통해 실제 값과 예측된 값 간의 차이를 비교해 볼 수 있습니다. | |
| **| 요구 사항** | 1. 데이터 준비 및 전처리  - 데이터를 불러오고, 각 날을 나타내는 인덱스를 독립 변수로 설정하세요.  - 8시 교통량 데이터를 종속 변수로 설정하세요.  2. 데이터 분리  - train\_test\_split 함수를 사용하여 데이터를 훈련 세트(80%)와 테스트 세트(20%)로 분리하세요.  3. 모델 생성 및 학습  - LinearRegression 객체를 생성하고, 훈련 데이터로 모델을 학습하세요.  4. 예측 및 비교  - 마지막 날의 교통량을 예측하고, 실제 교통량과 비교하여 예측 성능을 확인하세요.  . | |