


ML_01_2. CODE 전반적인 코드 진행

🕒 생성일	@2022년 6월 6일 오후 12:15
🏷️ 유형	머신러닝/딥러닝
👤 작성자	 동훈 오

HandsOn ML(2nd) - 머신러닝 프로젝트 처음부터 끝까지


handson-ml2/02_end_to_end_machine_learning_project.ipynb at master · rickiepark/handson-ml2

핸즈온 머신러닝 2/E의 주피터 노트북. Contribute to rickiepark/handson-ml2 development by creating an account on GitHub.

https://github.com/rickiepark/handson-ml2/blob/master/02_end_to_end_machine_learning_project.ipynb

Machine Learning with Scikit-Learn, Keras & TensorFlow

핸즈온 머신러닝 2판



핸즈온 머신러닝 2장에서는 캘리포니아 인구 조사 데이터를 바탕으로 주택 가격을 예측하는 지도학습 - 다중 회귀 모델을 보인다.

기초 EDA(데이터분석)을 보여주기 위한 코드 진행이 절반 분량을 차지하고 모델 선택과 훈련, 검증 및 평가 부분이 나머지 분량을 차지한다.

라이브러리 세팅 및 os를 통한 데이터로드 부분은 이후 코랩에서 개발 환경을 세팅하는데 중요하고, 데이터 탐색부분은 마지막 부분 - 일부 특성 필드를 삭제 및 null 값 대체 하는 부분 - 을 제외하고는 빠르게 넘겨도 괜찮을 것 같다. 결국 모델을 통해 훈련시킬 데이터, 테스트할 데이터만 확보하면 된다.

모델은 간단한 LinearRegression을 사용했다. 실제로 딥러닝 파트에서도 모델의 코드는 생각보다 많지 않다.

```
from sklearn.linear_model import LinearRegression

lin_reg= LinearRegression()
lin_reg.fit(housing_prepared, housing_labels)
```

두 번째 모델로 DecisionTreeRegressor를 사용했다. 결정 트리에 기반한 회귀 모델로 생각하면 된다.

```
from sklearn.tree import DecisionTreeRegressor

tree_reg= DecisionTreeRegressor(random_state=42)
tree_reg.fit(housing_prepared, housing_labels)
```

(실제로 ml, 딥러닝에서 대부분의 코드 진행이 데이터 로드 및 전처리, configuration 관리, logging 및 디버깅을 위한 코드 관리에 주력한다.

이 챕터에서는 교차 검증 방법(cross-validation)을 결정 트리 모델을 평가하는 방법으로 제안한다. 앞서 'ML_01. Intro'에서 언급한, validation set을 나누어서 사용하는 방식을 의미한다. 이후 결정 트리 부분에서 더 자세히 다룰 예정이다.

마지막 모델로 앙상블 기법의 랜덤 포레스트를 사용했다.

```
from sklearn.ensemble import RandomForestRegressor

forest_reg= RandomForestRegressor(n_estimators=100, random_state=42)
forest_reg.fit(housing_prepared, housing_labels)
```

파라미터 튜닝 방식으로 그리드 탐색(GridSearchCV) 와 랜덤 탐색(RandomizedSearchCV) 를 제안한다. 이 방식들은 적절한 파라미터 값을 찾기 위해 나온 아이디어이며, 주로 트리 기반 앙상블 모델, 랜덤포레스트 모델과 같이 활용된다. 딥러닝 파트로 넘어가면 잘 사용하지 않는데, 이유는 시간이 너무 오래 걸리기 때문이다.