<Chapter 2> 머신러닝 프로젝트 처음부터 끝까지 내용 정리

프로젝트 진행 과정

1. 실제 데이터셋으로 작업: 여러 데이터셋 저장소에서 실제 데이터 사용
2. 주택 가격 모델 만들기

➔ 데이터로 모델 학습시키기

➔ 다른 측정 데이터가 주어졌을 때 예측.

1) 문제정의:

➔ 비즈니스 목적 정의하기 (수익과 연결)

*파이프라인: 데이터 처리 컴퍼넌트들의 연속

➔ 현재 솔루션 참고

➔현재 시스템 : 레이블된 훈련 샘플 (지도학습)/ 값 예측, 사용하는 특성 여러개 (다중 회귀) / 구역마다 여러 값 예측하면 (다변량 회귀) / 데이터 연속 흐름 x (배치 학습)

2) 성능 측정 지표 선택:

➔여기는 회귀문제니까 평균제곱근오차 rmse (오차가 커질수록 이 값이 커짐) or 평균절대오차(mae)

3) 가정검사

3. 데이터 가져오기

1) 작업환경

2) 데이터다운로드

3) 데이터 구조 보기

- Head(): 처음 다섯행 확인
- Info(): 데이터 설명, 전체 행수, 데이터 타입, non-null 값 확인가능
- ~.value_counts(): 범주형 카테고리 확인
- Describe(): 숫자형 특성 요약 정보
- Hist(): 히스토그램 출력

4) 테스트세트 만들기

- 데이터 스누핑 편향
- 테스트 세트 생성-> 약 20% 떼어놓기.

행번호(row number)로 선택하는 방법 (.iloc)

label이나 조건표현으로 선택하는 방법 (.loc)

df.iloc[[행],[열]] # Data의 행 번호 활용, integer만 가능

df.loc[[행],[열]] # DataFrame index 활용, 아무 것이나 활용 가능

출처: https://mellowlee.tistory.com/entry/Pandas-데이터-추출행-Row-iloc-loc [잡토의 잠망경]

- Np.random.seed(42): 초깃값 지정
- Index를 id로 사용-> ~.reset_index()
- 무작위 샘플링 하면 편향 → 계층적 샘플링 필요 (데이터셋이 충분히 커야됨)
- 계층샘플링:stratifiedshuffleplit (sklearn에서 import)
- Inplace=true(원본 보존안하고 바꾸기) .drop(행/열 삭제)

4. 데이터 이해를 위한 탐색 및 시각화
   - .copy() 복사본

     1) 지리적 데이터 시각화
   - .plot(kind="scatter(산점도), x=, y=)

     *alpha 옵션 주면 밀집된 영역 잘 보여줌 → 투명도

     2) 상관관계 조사

     - 표준 상관계수 corr_matrix= ~.corr()

     1에 가까울수록 양의 상관관계, -1에 가까우면 음의 상관관계, 0은 관계없음

     -상관관계 확인→ 판다스에서 scatter_matrix 함수import

     Attributes=[]

     Scatter_matrix()

대각성 방향(왼->오) 그냥 직선은 유효x

*plt.axis([]) 축

3) 특성 조합으로 실험

Ascending=false 내림차순 정렬

5. 머신러닝 알고리즘을 위한 데이터 준비

- 원래 훈련세트 복사 /복원
- 예측변수랑 레이블 분리

1) 데이터 정제

- Dropna():제거 drop():삭제 fillna():어떤 값으로 채움
- 누락된 값 채우기 (SimpleImputer import)
- 텍스트 특성 제외하기

2) 범주형, 텍스트 다루기 -> 숫자로 변환

- Sklearn Ordinal Encoder 사용 하고 econder.categories_해서 카테고리 목록 얻을 수 있음. +원-핫벡터로 바꾸기 OneHotEncoder

* 원-핫 인코딩을 위해서 먼저 해야할 일은 단어 집합을 만드는 일입니다. 텍스트의 모든 단어를 중복을 허용하지 않고 모아놓으면 이를 단어 집합이라고 합니다. 그리고 이 단어 집합에 고유한 숫자를 부여하는 정수 인코딩을 진행합니다. 텍스트에 단어가 총 5,000개가 존재한다면, 단어 집합의 크기는 5,000입니다. 5,000개의 단어가 있는 이 단어 집합의 단어들마다 1번부터 5,000번까지 인덱스를 부여한다고 해보겠습니다. 가령, book은 150번, dog는 171번, love는 192번, books는 212번과 같이 부여할 수 있습니다. 원-핫 벡터는 표현하고 싶은 단어의 인덱스에 1의 값을 부여하고, 다른 인덱스에는 0을 부여하는 단어의 벡터 표현 방식. (구글링)

3) 나만의 변환기

파라미터: 매개변수 하이퍼파라미터: 모델링할때 직접 세팅해주는 값

- TransformerMixin상속. BaseEstimator → get_params() set_params 얻을 수 있음

4) 특성 스케일링

- 정규화 (min-max스케일링) MinMaxScaler sklearn
- 표준화 이상치 영향 덜 받음 StandardScaler

5) 변환 파이프라인

Import Pipeline

연속된 단계를 나타내는 이름추정기 쌍의 목록을 입력으로 받음.

Fit_transform()메서드 호출 마지막에 fit()만 호출 → ColumnTransformer

*희소 행렬: 행렬값이 대부분 0 이 많은 것*밀집행렬

6. 모델 선택과 훈련

1) 훈련세트에서 훈련 하고 평가하기

- 선형회귀 모델 LinearRegression 적용
- Mean_square_error → rmse 측정 가능
- 의사결정나무 DecisionTreeRegressor → 과대적합

2) 교차 검증

- K-겹 교차 검증 cross_val_score
- Neg_mean_squared_error함수 사용
- RandomForestRegressor 모델 : 앙상블 학습

7. 세부 튜닝

1) 그리드 탐색 GridSearchCV

2) 랜덤탐색 RandomizedSearchCV

3) 최상의 모델 + 오차분석

- 덜 중요한 특성 제외 가능

4) 테스트 세트로 시스템 평가

- Full_pipeline 사용해서 데이터 변환 하기
- 신뢰구간: scipy.stats.t.interval()

8. 론칭, 모니터링, 유지보수

- 전처리 파이프라인이랑 예측 파이프라인이 포함된 사이킷 훈련 모델 저장
- Predict ()사용해서 예측 하기

- 구글 클라우드 AI플랫폼에 배포
- 성능 모니터링: 하위 시스템 지표로 추정 가능 + 모니터링 시스템 준비
- 모델 백업