**Advance Data Mining**

**경희대학교 산업경영공학과**

Reading Assignment

**Chapter 1~2**



고급데이터마이닝

진창호

산업경영공학과

2016100937

김성수

과목명

담당교수

학과

학번

이름

**Chapter1**

1. 머신러닝의 정의와 특징

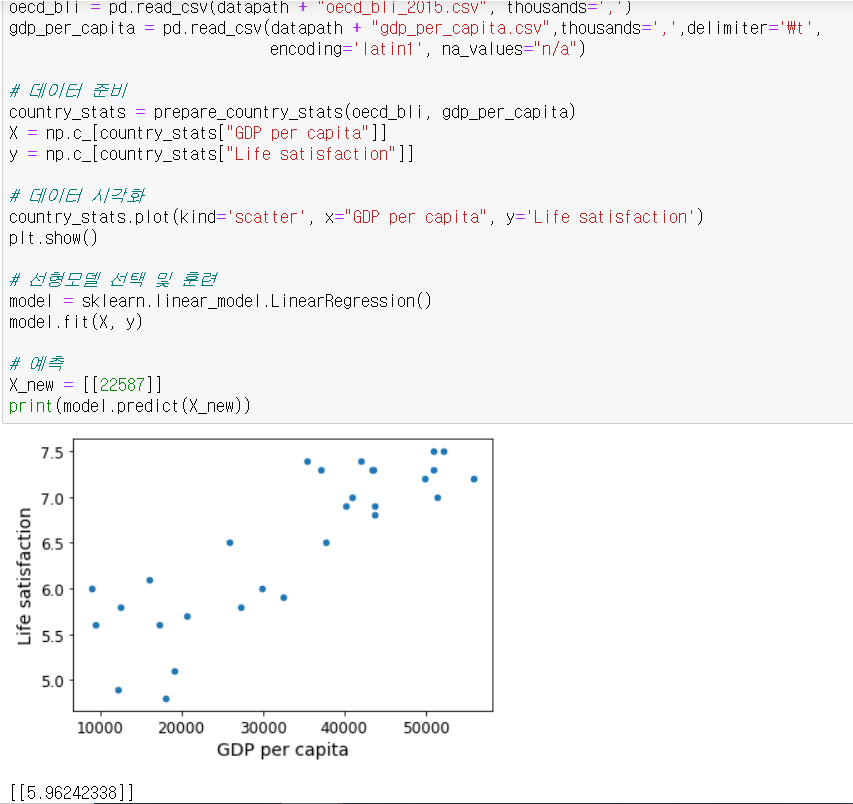
* 어떤 작업 T에 대한 컴퓨터 프로그램의 성능이 P일 때, 경험 E로 인해 성능이 향상됐다면, 이 컴퓨터 프로그램은 작업 T와 성능 측정 P에 대해 경험 E로 학습한 것
* 새로운 데이터에 잘 적응하며 대량의 복잡한 데이터에서 통찰을 얻기가 수월

1. 머신러닝의 분류

* 기준1: 사람의 감독하에 훈련하는가 여부
  + 지도학습: 알고리즘에 주입하는 훈련데이터에 레이블이라는 답이 포함되며, 크게 분류와 회귀로 나뉘고 KNN, SVM, 로지스틱 회귀모형 등이 포함
  + 비지도학습: 훈련데이터에 레이블이 없으며, 군집분석, 시각화 및 차원축소, 연관 규칙 탐색 등이 포함
  + 준지도학습: 훈련데이터 일부에만 레이블이 있으며 지도학습과 비지도학습의 조합으로 이루어져 있고 심층 신뢰 신경망 등이 포함
  + 강화학습: 시스템이 환경을 관찰하여 행동을 실행하고 그 결과로 보상이나 벌점을 받아 최적의 결과를 얻기 위해 스스로 학습하며 알파고 등이 포함
* 기준2: 실시간으로 점진적인 학습을 하는가 여부
  + 배치학습: 점진적으로 학습할 수 없으며 새로운 데이터에 대해 학습하려면 전체 데이터를 사용하여 처음부터 다시 훈련하기에 많은 시간과 컴퓨팅 자원이 소모
  + 온라인학습: 데이터를 순차적으로 한 개씩 또는 미니배치 단위로 주입하여 학습률(Learning rate)에 따라 훈련하기에 매 학습단계가 빠르고 비용이 적게 들며 제한된 컴퓨팅 자원 환경에 적합
* 기준3: 단순히 기존 데이터와 새 데이터를 비교하는지, 예측모델을 만드는지 여부
  + 사례기반학습: 시스템이 훈련 샘플을 기억하여 학습하고 유사도를 측정하여 새로운 데이터화 학습한 샘플을 비교하는 식으로 일반화
  + 모델기반학습: 기존 샘플들로 모델을 만들어 예측에 사용하는 식으로 일반화

1. 머신러닝 프로젝트의 형태

* 데이터 분석 🡪 모델 선택 🡪 모델 훈련(비용함수를 최소화하는 파라미터 찾기) 🡪 새로운 데이터로 예측 실시



<그림1: 데이터 준비부터 모델 생성 및 예측 후 결과 및 시각화 코딩>

1. 머신러닝의 주요과제

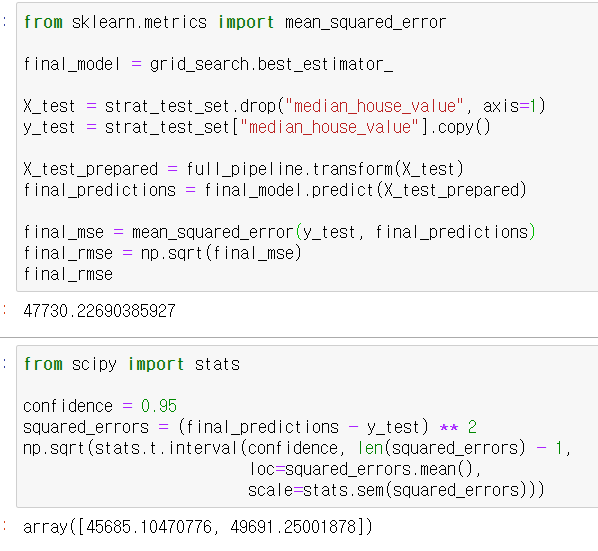
* 데이터: 충분한 양, 대표성이 있는, 좋은 품질의 그리고 좋은 특성의 데이터
* 알고리즘: 과대적합과 과소적합을 방지하는 최적의 하이퍼파라미터의 설정
  + 과대적합: 훈련 데이터에는 잘 맞으나 테스트 데이터에는 적합하지 않아 일반성이 떨어지는 현상
  + 과소적합: 모델이 너무 단순하여 데이터의 내재된 구조를 학습하지 못하는 현상

1. 모델의 테스트와 검증

* 일반적으로 훈련세트와 테스트 세트를 나누어 일반화 오차와 훈련 오차를 측정
* 홀드아웃 검증: 훈련세트의 일부를 검증세트로 분류 🡪 훈련 세트로 다양한 하이퍼파라미터값의 모델을 훈련 후 검증세트로 최적의 모델을 선택 🡪 전체 훈련 세트로 다시 훈련 후 최종 모델 생성 🡪 최종 모델을 테스트 세트로 평가 및 일반화 오차 추정
* 교차검증: 검증 세트마다 나머지 데이터에서 훈련한 모델을 해당 검증 세트에서 평가하여, 모든 모델의 성능을 평균화하여 평가
* 훈련-개발세트: 훈련세트의 일부를 훈련-개발세트로 나누고 🡪 훈련세트로 훈련 후 훈련-개발세트와 검증세트로 평가

**Chapter2**

1. 머신러닝 프로젝트의 진행단계
2. 큰 그림 보기
   1. 문제정의: 비즈니스의 목적을 파악하고, 주어진 문제를 해결하기 위해 사용하고자 하는 모델 및 학습 방법 선택
   2. 성능 측정 지표의 선택: 모델이 예측한 결과값에 대해 어떻게 평가할 것인지 정하는 단계로, 회귀분석에는 RMSE, MAE 등을 사용
   3. 가정검사: 위 2단계에서 정한 내용이 주어진 문제 해결에 적합한지 재확인
3. 데이터 가져오기
   1. 데이터 파일의 형태 확인: 데이터 파일이 어떤 형식(csv, sas7bdat 등)인지 확인하고 함수를 이용해 데이터 불러오기
   2. 데이터 형태 살펴보기: head, info, value\_counts, describe 등을 이용하여 데이터가 가지고 있는 전체적인 개요 파악
   3. 테스트 세트 만들기: 기존 데이터에서 무작위로 20% 정도를 테스트 세트로 분류하고 이를 일반화 오차 추정을 위해 사용. 만약 불균형 데이터의 경우 층화 샘플링으로 테스트 세트 만들기가 가능
4. 데이터 이해를 위한 탐색과 시각화
   1. Matplotlib 라이브러리 등을 활용하여 시각화로 데이터의 분포를 파악
   2. 시각화 자료 또는 상관행렬을 활용하여 상관관계로 유의미한 변수를 파악 가능
   3. 데이터의 조합으로 유의미한 변수를 직접 만들어 분석에 활용
5. 머신러닝 알고리즘을 위한 데이터 준비
   1. 결측치는 제거하거나 평균, 중간값 등으로 대치
   2. 범주형 데이터의 경우 텍스트를 숫자로 변형하거나, 원-핫-인코딩을 활용하여 카테고리별 이진 특성을 생성하여 해결
   3. Min-max 스케일링, 표준화 스케일링을 활용하여 모든 특성의 범위를 같도록 함
   4. 변환 파이프라인을 활용하여 변환기 및 추정기를 지정해 column별 데이터 타입에 따른 전처리를 자동화 가능
6. 모델 선택과 훈련
   1. 위에서 정제한 데이터의 훈련 데이터와 선택한 모델을 활용하여 모델 훈련 실시
   2. 모델 평가: 1단계에서 선택한 성능 측정 지표를 활용하거나 K-fold 교차검증을 활용하여 훈련 데이터를 k개의 부분으로 나눈 후 k번의 검증을 시행하여 평가
7. 모델의 세부 튜닝
   1. 만족할 만한 성능을 위해 최적의 하이퍼파라미터 조합을 수동으로 조정 시 시간과 비용이 많이 소모
   2. 그리드 탐색: 비교하고자 하는 하이퍼파라미터를 인자로 넣은 후 교차검증을 이용한 자동화된 방식으로 최적의 하이퍼파라미터 도출 가능
   3. 랜덤탐색: 비교하고자 하는 하이퍼파라미터의 수가 많은 경우 하이퍼파라미터의 모든 조합에 대해 무작위로 각기 다른 값을 대입한 후 지정한 횟수만큼 평가, 그리고 최적의 파라미터를 도출
   4. 앙상블 방법: 1가지의 모델을 튜닝하기 보다, 여러 개의 모델을 연결하여 활용
   5. 모델의 세부 튜닝이 완료되었다면 테스트 데이터를 이용하여 시스템을 평가
8. 론칭, 모니터링, 시스템 유지 및 보수
   1. 배포: REST API 나 구글 클라우드 AI 플랫폼 등을 활용하여 배포
   2. 모니터링: 하위 시스템의 지표를 활용하여 주기적으로 모델의 성능을 추정하는 등의 일정 간격으로 시스템의 실시간 성능 측정의 필요성
   3. 시스템 유지 및 보수: 정기적으로 새로운 데이터를 넣은 새로운 모델을 만들어 기존 모델과 성능을 비교하여 성능이 좋아지면 새로운 모델을 제품에 배포

  
**<그림2: 위 모든 과정을 실행한 최종 결과>**