18회 ADP 기출 분석

1번 문항 – 다중클래스분류(40점)

데이터

- 독립변수: id, days, duration, count, amount
- 종속변수: grade(1~5 정수 이산형 다중분류)
- 파일 형식: csv

Index(?)	Id	Days	Duration	Count	Amount	grade
						1
						2
						3
						4
						5

백화점 고객 매출 데이터와 고객 등급(1~5, 이산형)으로 이루어진 데이터에 대해 아래 질문에 답하시오.

1번 문항 – 다중클래스분류(40점)

1-1) 데이터 전처리와 EDA를 실시하고 그 결과에 대해 논하시오

- 예상 풀이: 결측값 확인 및 처리, 이상치 탐지 및 처리, 사용할 수 없는 값 처리 등 기본적 전처리 수행 후 변수들의 기초통계량 확인, 시각화를 진행. EDA 결과에 대한 설명(결측치 1200개 정도 있었다 함)

- 관련 파이썬 라이브러리: pandas, matplotlib, seaborn

1번 문항 – 다중클래스분류(40점)

1-2) 고객 등급 예측에 도움이 될 파생 변수를 3개 생성하고, 생성한 타당한 이유를 함께 서술하시오.

- 예상 풀이: feature engineering 을 통한 파생 변수 생성.
  독립변수들을 잘 조합하여 새로운 변수를 3개 만들면 됨. 자세한 조합은 직접 데이터 값을 확인해봐야 알 수 있을 것 같음. 제약은 없으나 근거가 타당해야 할 듯.

- Ex) 구매 총액인 amount 변수를 구매 횟수인 count로 나누어 구매 당 평균 구매 금액 이라는 새로운 변수를 생성 가능

1번 문항 – 다중클래스분류(40점)

1-3) SOM을 활용하여 고객 등급 분류 모델을 학습시키고 F1 Score와 ROC 커브로 평가하시오.

예상 풀이: SOM(Self Organizing Map)은 비지도학습 알고리즘이며 클러스터링에 사용된다. K-mean 알고리즘보다 약간 복잡함. 파이썬 사이킷 런 에도 2021년 6월 추가됨.

독립변수와 종속변수를 분리, 데이터 스케일링 수행, 파라미터 설정(맵 사이즈, neighborhood의 반경, 학습률, 초기값 설정) 수행, 모델 학습, 예측 순으로 진행 후 필요하다면 시각화.

SOM의 예측 결과를 실제 레이블 값과 비교하여 F1 Score, ROC 커브를 도출하고 평가.

관련 파이썬 라이브러리: from sklearn_som.som import SOM, from som import Som, minisom?, from sklearn.metrics import f1_score, roc_curve

1번 문항 – 다중클래스분류(40점)

1-4) 추가로 랜덤 포레스트와 신경망을 포함한 4개의 고객 등급 분류 모델을 학습시키고, 각 모델을 F1 Score와 ROC 커브로 평가하여 최적의 모델을 선정하고, 그 모델을 튜닝하여 분류 성능을 향상시키시오.

예상 풀이: SOM 외 추가로 랜덤포레스트, 신경망을 포함한 4개 분류 모델을 만들라는 것 같음. 그러면 SOM까지 총 5개의 분류 모델을 만드는 것.

랜덤포레스트, 신경망(MLP, DNN), 의사결정나무, 로지스틱 회귀, K-NN, K-mean 등 비지도, 지도 학습 중 다중분류 문제 풀이가 가능한 모델을 선택하면 될 것 같음. 선형 모델도 사용 가능하지만 이 경우 일대다 방법을 사용해야 함.

4개의 모델 중 가장 성능이 좋은 것을 선택하여 랜덤 서치나 그리드 서치 등으로 파라미터 튜닝을 진행하면 됨.

관련 파이썬 라이브러리: ramdomforest, decisiontree, logistic regression, tensorflow, keras, k-nn, k-mean …

2번 문항 – 텍스트 마이닝(10점)

데이터: 영문 텍스트가 들어있는 .txt 파일

2-1) 주어진 텍스트를 전처리(형태소 분석, 불용어 처리 등) 하시오.

예상 풀이: 텍스트에서 명사 추출과 불용어 처리를 수행해야 한다고 함
데이터를 판다스로 읽어들여 데이터프레임에 저장, SpaCy 패키지가 제공되는지는 모르겠으나 있다면 scpacy.tokenizer로 토큰화를 진행, 이 때 정규표현식을 사용하여 소문자화, 특수문자 등 제거.

라이브러리가 제공하는 불용어 사전을 사용하여 불용어를 제거
이후 텍스트에 등장하는 단어의 빈도 막대그래프를 그리면 됨.
(워드클라우드 생성 문제가 있었으나 시험 도중 제외되었다 함)

관련 파이썬 라이브러리: pandas, spacy, spacy.tokenizer import Tokenizer, re, from collections import Counter(단어 빈도 구하기)

3번 문항 – 통계분석(50점)

데이터: 교통사고 관련 .csv 파일
계절성 시계열 데이터 분석

3-1) 제시된 시계열 데이터의 정상성을 판별하고, 필요하다면 데이터를 차분하시오.

예상 풀이: 시계열 데이터는 전처리가 상당히 많이 필요한 경우가 많다. 데이터가 어떻게 주어졌는지 모르겠으나 필요한 전처리를 모두 수행.
정상 시계열인지 보려면 평균, 분산이 일정한지 확인해야 한다. Rolling mean, rolling std 그래프를 그려 확인한다. 추세, 계절, 불규칙 요인등을 확인하려면 statsmodels.tsa 에서 seasonal_decompose를 불러와 그린다. 계절차분이 필요하다면 추가로 진행한다.
정량적인 방법을 사용하기 위해 ADF test를 수행한다. ADF test의 귀무가설은 해당 '시계열이 비정상 시계열이다' 이다. (p-value > 0.05 면 귀무가설을 기각할 수 없음). 비정상 시계열이라면 차분을 통해 변환한다.

관련 파이썬 라이브러리: pandas, statsmodel.tsa, adfuller

3번 문항 – 통계분석(50점)

3-2) ACF와 PACF를 참고하여 ARMA 모형을 3개 이상 제시하시오.

예상 풀이: 차분을 했기 때문에 ARIMA(p, 1, q) 모형을 설계하라는 것 같음. 데이터에 대한 ACF, PACF 그래프를 그리고 그래프 개형을 보면서 AR term, MA term을 하나씩 정해서 모델을 만들어 보는 것이 좋아보임. (2, 1, 0), (0, 1, 1) 이런 식.

3개의 초기 모델을 완성한다.

계절성이 있는 데이터라는 언급이 있었기 때문에 경우에 따라 SARIMA 모델을 사용하면 될 것 같음.

관련 파이썬 라이브러리: statsmodels.graphics.tsaplots import plot_acf, plot_pacf, statsmodels.tsa.statespace.sarimax import SARIMAX

3번 문항 – 통계분석(50점)

3-3) 제시한 모형들을 비교하여, 타당한 이유를 들어 가장 적합한 모델을 선택하시오.

예상 풀이: auto_arima 함수 직접 코딩하여 AIC 점수를 근거로 최적 모델을 선택하는것이 가장 효율적이라고 생각.

최종 모델을 선택했다면 해당 ARIMA(p, d, q) 조합이 타당한지 검증을 수행. 잔차 그래프를 그려보고, acf, pacf 분석을 통해 잔차가 정상성을 가지는지 확인. Q-Q plot, 샤피로 테스트, 노말테스트 등으로 잔차가 정규 분포 형태를 가지는지 확인. 더빈 왓슨, 포트만토 검정(융-박스 테스트)을 통해 잔차의 자기상관성 여부를 확인.

관련 파이썬 라이브러리: from scipy.stats import shapiro, normaltest
from statsmodels.stats.stattools import durbin_watson, statsmodels.stats.diagnostic.acorr_ljungbox

3번 문항 – 통계분석(50점)

3-4) 해당 모형으로 교통사고 데이터를 예측하여 정확도를 평가하고 필요하다면 근거와 함께 제시하시오.

예상 풀이: 예측 기간에 대한 레이블 값이 주어지지 않은 듯 하다. 따라서 데이터를 train, test 데이터로 분리하고 train 데이터에 대해서 최적 모델을 학습, 예측 결과를 test 데이터셋의 레이블과 비교한다.

예측 평가 방법은 Error, RMSE, MAPE, sMAPE, mMAPE 등을 사용한다.

참고 자료

시계열

https://github.com/bluemumin/ADP_certificate_preperation
https://otexts.com/fppkr/index.html