[데이콘] 1회 : 상점 신용카드 매출 예측 경진대회_1등

Description

- 2016.03.01~2019.02.28 까지의 카드 거래 데이터를 이용해 2019.03.01~2019.05.31 까지의 **상점별 3개월 총 매출**을 예측하자.

Data field

- store_id : 상점의 고유 아이디
- card_id : 사용한 카드의 고유 아이디
- card_company : 비식별화된 카드 회사
- trasacted_date : 거래 날짜
- transacted_time : 거래 시간(시:분)
- installment_term : 할부 개월 수(포인트 사용 시 할부 개월수 = 60개월 + 실제할부개월)
- region : 상점의 지역
- type_of_business : 상점의 업종
- amount : 거래액(단위는 원이 아니다)

Index

step1. Data Load & Packages

- R 패키지 사용을 위해 python에 rpy2 패키지 설치
- 단위근 검정을 위해 pmdarima의 ADF_Test 사용
- AR값을 정하기 위해 tsaplots의 plot_acf와 plot_pcaf 사용
- 시계열 모델링을 위해 R forecast, forecastHybrid 패키지 사용

step2. Data Cleansing & Pre-Processing

전체적인 데이터, 결측치 확인

- 결측치 확인 및 처리

9개 변수 중 2개의 변수에서 결측치 발생

1) region(상점의 지역) : 31.15% 결측치 확인
2) type_of_business(상점의 업종) : 60.28% 결측치 확인
=> 결측치 비율이 높아서, region, type_of_business 두 변수 제거

각 변수 하나씩 뜯어서 이상치 확인, 파생변수 생성, 변수 제거 등

=> 시계열 분석 위해 date와 amount 변수만 확인하고 전처리 진행하였다

- datetime 파생변수 생성

datetime = trasacted_date(거래 날짜) + transacted_tim(거래 시간)

- amount(거래액) 변수 확인

amount에서 음수 값이 존재 하는 것을 확인 => 환불 금액으로 예상

- def return_remove() : 환불 노이즈 제거하는 함수 생성

1) 거래액이 0보다 작은지 큰지에 따라 refund, non_refund 집단 생성
2) 각 스토어 아이디별로 refund와 non_refund 데이터를 뽑아 순차적으로 진행
3) refund를 하나씩 돌리면서, non_refund 에서 환불 시간 이전의 데이터 중 카드 아이디와 환불액이 같은 후보 리스트가 있다면 뽑는다
4) 후보 리스트가 있다면 가장 최근 시간의 데이터를 non_refund에서 제거한다
5) 최종적으로, 환불한 데이터는 삭제되고 환불 안하고 팔린 데이터만 얻을 수 있다
=> 6555613 행에서 6406073 행으로 축소

- def month_resamplign() : 월별로 다운 샘플링해주는 함수 생성

1) 스토어 아이디와 년월로 그룹 지어서 amount 합하기
2) 매출이 없는 월은 2로 채운다 => 나중에 로그 변환시 무한대를 방지하기 위해
3) 최종적으로 모델링 하기에 적합한, 상점별 월별 amount 데이터 프레임으로 변형시켜준거다
=> 나중에 발표할 때 월별 말고 다른 시기로 잡아서 해보는 것도 추천한다고 말했다

- def time_series(df,i) : store_id 입력 시 그 store의 amount 데이터를 시계열로 변환

step2. EDA

- 각 상점마다 매출액 시계열, 분포 그래프 그려본 결과, 매출 특성과 분포가 각각 다르다

=> 개별적인 시계열 모델링을 진행해야 한다

step3. Modling - Time Series

- def adf_test() : 시계열 자료의 정상성을 여부를 확인해 차분을 결정하기 위한 단위근 검정

1) store_id별로 ADFTest한 p-value값을 본 결과, 평균 0.4로 0.05 이상이므로 차분이 필요하다
2) acf(자기상관), pcaf 그래프를 보면 대부분 상점의 적정 ar값이 2이하로 보인다

* 단위근 검정 :시계열 데이터가 stationary *정상성인지 테스트 하는 방법
귀무가설 : 자료에 단위근이 존재한다
대립가설 : 자료에 단위근이 존재하지 않는다, 정상성/추세 정상성을 만족한다
* 단위근은 시계열 자료에서 예측할 수 없는 결과를 초래한다

- 변수 선택

1) 연도-월
2) 매출액

- 모델 학습

1) hybridModel 진행
2) ndiffs() 함수 통해 몇 번 차분 해야 할지 parameter 값 구하기
3) auto_arima 파라미터 튜닝시, ar값이 2이하 이기에 max.p=2, d값=ndiffs값

4) 매출액의 변동계수(표준편차/평균)가 0.3이하인 경우에는 변수 log 정규화 진행 => 변동성이 커서 잡음 제거 하기 위해
5) store_id별로 3개월 매출 예측 후 합산