주제분석 1주차 패키지

- 분석 툴은 R/Python 둘 다 가능합니다. 주제분석 1-3주차 패키지 문제의 조건 및 힌트는 Python을 기준으로 하지만, R을 사용해도 무방합니다.
- 제출형식은 HTML, PDF 모두 가능합니다. **.ipynb 이나 .R 등의 소스코드 파일은 불가능합니다.** 파일은 psat2009@naver.com으로 보내주세요.
- 패키지 과제 발표는 세미나 쉬는 시간 후에 하게 되며, 랜덤으로 16시 00분에 발표됩니다.
- 제출기한은 **목요일 자정까지** 이고, 지각 시 벌금 5000원, 미제출시 10000원입니다. 패키지 무단 미제출 2회 시 퇴출이니 유의해주세요. 또한 불성실한 제출로 인한 경고를 2회 받으실시도 퇴출이니 유의해주세요.

Chapter 1 : Data Preprocessing

주제분석 1,2주차 패키지는 전국에 매장을 갖고 있는 회사의 데이터분석가가 됐다는 생각으로 진행해보겠습니다. 2014년 7월 1일부터 2017년 8월 15일까지의 데이터에는 날짜 별 각 매장의 sales와 promotion, 각 날짜의 oil 가격 등이 있습니다. 이를 통해 여러분은 회사의 의사결정에 도움이 될만한 인사이트를 도출하고 sales를 예측하는 모델을 세워야 합니다.

우선 주제분석 1주차에는 데이터 전처리에 집중해보겠습니다.

문제0. 분석에 사용할 라이브러리를 설치해주세요. 목록은 다음과 같습니다.

pandas, matplotlib.pyplot, seaborn, missingno, numpy,
tslearn, scalecast, pmdarima, statsmodels

문제1. Pandas를 이용해 sales, oil, stores, holidays_events 데이터를 불러와주세요.

문제2. 데이터의 구조를 자유롭게 파악해주세요.

문제 3. Sales 를 기준으로 oil, stores 를 결합시켜 하나의 data frame 을 만들어주세요.

(HINT) pandas의 merge 함수를 이용하면 쉽게 결합할 수 있습니다.

- 해당 Data frame 을 data 라고 저장해주세요.

문제4. Holiday 데이터를 이용해 아래와 같이 dummy variable을 만들어 data에 넣어주세요.

- Holiday에 판매량이 증가한다는 가정을 해봅시다. 어떤 holiday인지에 따라 증가폭이 달라질 수 있으나 모두 같은 영향을 미친다는 가정 하에 holiday라는 dummy variable을 만들어보겠습니다.
- 우선 transferred가 True인 행은 삭제해주세요.
- Holiday 적용 범위(locale)에 따라 다음과 같이 해당 날짜, 도시에만 1, 그렇지 않은 경우에는 0을 넣어주면 됩니다.
  - locale = National : 전체 도시
  - locale = Regional : 해당 state
  - locale = Local : 해당 city

문제5. data에서 id 변수를 제거해주세요.

문제6. data의 변수 타입을 다음과 같이 바꿔주세요.

- date 변수 타입을 datetime으로, 범주형 변수를 object로 바꿔주세요.

(HINT) pandas의 to_datetime, astype을 사용하면 됩니다.

문제7. data의 date 변수를 이용해서 year, month, weekday 변수를 만들어주세요.

(HINT) pandas의 dt.year, dt.month, dt.weekday를 사용하면 쉽게 만들 수 있습니다.

문제8. data의 마지막 15일을 기준으로 train set과 validation set을 만들겠습니다.

- 각 set을 train, valid로 저장해주세요.
- 아래 모든 과정은 train을 활용합니다.

문제9. 일일 판매량이 적은 store는 관심 대상이 아닙니다. Sales가 3000보다 낮은 날의 비율이 10% 이상인 store를 제거해주세요.

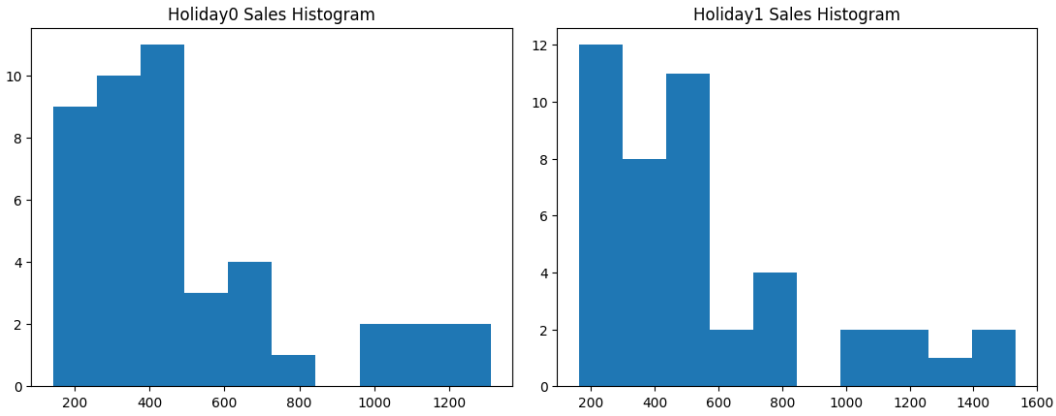
문제10. 마찬가지로 일일 판매량이 적은 품목(family)은 관심 대상이 아닙니다. Sales가 10보다 낮은 날의 비율이 10% 이상인 family를 제거해주세요.

문제11. 문제4에서 만든 holiday 여부에 따른 store의 sales 평균을 비교해보겠습니다.

(HINT) groupby를 이용해 holiday, store_nbr에 따른 sales의 평균을 구하세요.

- store의 sales 평균에 유의한 차이가 없다면 예측에 도움이 되는 변수라고 보기 힘들 것입니다.
- 문제11-1. 일반적으로 샘플 사이즈가 30보다 크면 정규성 가정을 하고 t-test를 진행합니다.
  각 집단의 sales 평균으로 히스토그램을 그려보세요.

  (HINT) matplotlib.pyplot의 hist를 이용하면 쉽게 그릴 수 있습니다.



- 문제11-2. Bootstrap을 이용하면 분포 가정 없이도 집단 간 평균 비교가 가능합니다.
  Bootstrap을 이용한 가설검정 방법을 찾아보고 간단히 기술해주세요.
  (*분산이 같지 않은 두 집단의 평균 차이 검정)
- 문제11-3. Bootstrap을 이용해 holiday 0, holiday 1 집단의 sales 평균을 비교해보세요.
  - $H_0: \mu_0 = \mu_1 \ vs. \ H_1: \mu_0 \neq \mu_1 \ (when \ \sigma_0^2 \neq \sigma_1^2), \ \alpha = 0.05$
  - Bootstrap sample 10000개를 이용해서 검정을 진행하세요.
  - 검정 통계량 T는 다음과 같습니다.
    두 집단을 x, y라고 하고 sample size가 각각 n, m이라고 했을 때,

$$T = \frac{\bar{x} - \bar{y}}{\sqrt{\frac{S_x^2}{n} + \frac{S_y^2}{m}}}$$

  - Bootstrap sample과 T_b는 다음의 식을 통해 얻을 수 있습니다.
    i) $v_i = x_i - \bar{x} + \bar{z}, \ where \ \bar{z}: \ pooled \ sample \ mean$
    $w_i = x_i - \bar{x} + \bar{z}, \ where \ \bar{z}: \ pooled \ sample \ mean$
    ii) v_i, w_i에서 각각 sample size 만큼 복원추출 후 bootstrap test statistic T_b 계산
    iii) $b = 1, \dots, 10000$에 대해,

$$ASL = \frac{\#\ of\ (T_b > |T| \ or \ T_b < -|T|)}{10000}$$

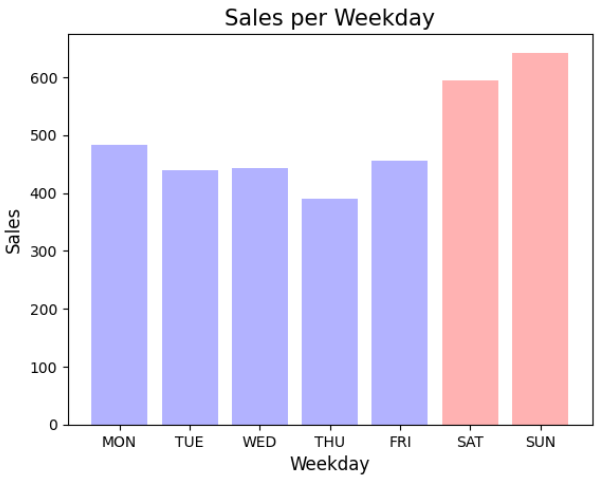
    $ASL < \alpha$면, H_0 기각

문제12. 요일 별 sales의 평균을 아래 그림과 같이 시각화해주세요.

(HINT) matplotlib.pyplot의 bar를 사용하면 bar plot을 그릴 수 있습니다.

- barplot의 color는 “blue”와 “red”이고 alpha는 0.3입니다.

- fontsize : {title: 15, xlabel: 12, ylabel: 12, xticks: 12, yticks: 12}



문제13. 위 plot을 통해 주말과 평일 sales에 차이가 존재한다는 것을 확인했습니다. 토요일, 일요일, holiday 인 경우와 그렇지 않은 경우로 나누어 두 집단의 store sales 평균을 비교해주세요.

- 문제11-3과 같은 방법으로 비교하면 됩니다.

문제14. 문제11에서 구한 sales의 평균에 계절 차분을 진행한다면, holiday 0, holiday 1 집단의 sales 평균에 유의한 차이가 있을지 시계열의 비정상성과 관련 지어 설명해주세요.

- 문제11과 문제13의 결과를 참고해주세요.

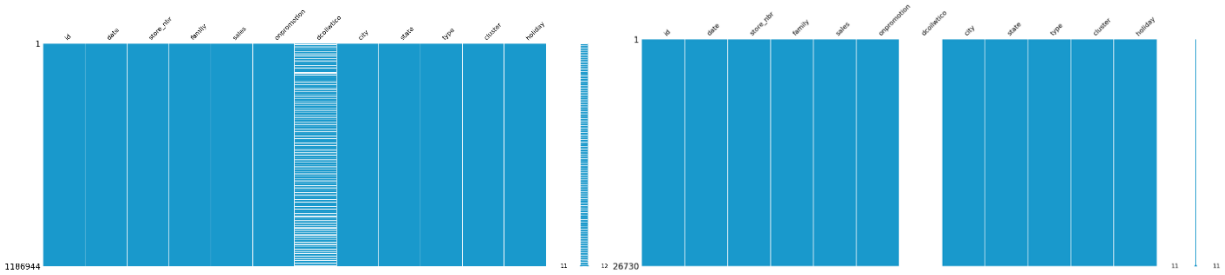
문제15. Valid에 train에 했던 전처리 과정을 똑같이 진행해주세요.

Chapter 2 : NA Imputation

NA 값이 존재하는 경우 상황에 따라 제거하거나 대체해야 합니다. 특히 결측치 보간의 경우 평균이나 중앙값으로 대체하는 single imputation부터 KNN, MICE, 딥러닝을 이용하는 다양한 imputation 방법이 있습니다. 이번 패키지에서는 시계열 데이터의 NA imputation에 대해 공부해봅시다.

문제1. Train과 Valid 데이터에 결측치가 존재하는지 확인한 후 아래 그림과 같이 시각화해주세요.

(HINT) missingno 패키지를 사용하여 결측치를 시각화 할 수 있습니다.



문제2. 시계열 데이터의 결측치 처리 방법 및 유의할 부분을 알아보고 간단히 서술하세요. 또한 우리 데이터에 맞는 결측치 처리 방법이 무엇일지 이유와 함께 설명해주세요.

- 시계열 결측치 처리 방법에는 LOCF, NOCB와 같은 간단한 방법부터 선형, 비선형 보간, 시계열 모델링 등이 있습니다.

문제3. Train의 dcoilwtico 결측치는 간단히 선형 보간법을 사용해서 imputation해주세요.

(HINT) pandas의 interpolate 함수를 사용하면 됩니다.

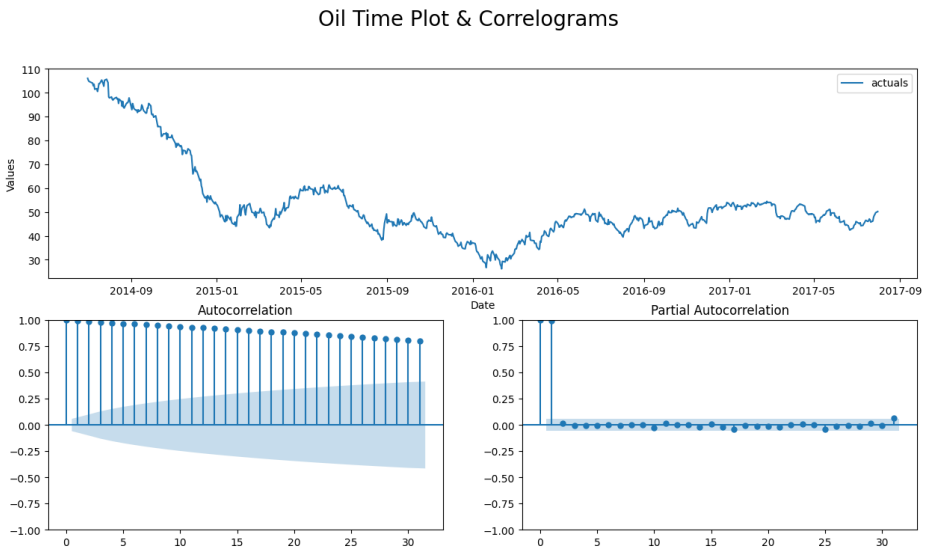
문제4. Valid의 dcoilwtico 결측치는 시계열 모델링을 통해 imputation 하겠습니다. 먼저 dcoilwtico 데이터의 time plot과 correlogram을 통해 특징을 파악하겠습니다.

- 모델링에 사용할 데이터는 train의 dcoilwtico입니다. (2014-07-01 ~ 2017-07-31 데이터)
- 문제4-1. 아래 그림과 같이 time plot과 correlogram을 그린 후 특징을 간단히 서술해주세요.

  (HINT) scalecast의 Forecaster 함수를 사용하면 time plot, acf, pacf를 쉽게 그릴 수 있습니다.

  matplotlib.pyplot의 subplots을 사용하면 여러 plot을 동시에 그릴 수 있습니다.
- 문제4-2. 1차 차분한 데이터의 time plot과 correlogram을 그린 후 특징을 서술해주세요.

  (HINT) pandas의 diff를 사용하면 쉽게 차분할 수 있습니다.



문제5. ARIMA(0,1,0)으로 fitting한 후 residual plot을 확인하고 ADF test를 진행해주세요.

(HINT1) pmdarima의 ARIMA를 import 해주세요.
ARIMA 모델에 fit, plot_diagnostics를 사용하면 fitting 및 잔차 plot 확인을 쉽게 할 수 있습니다.

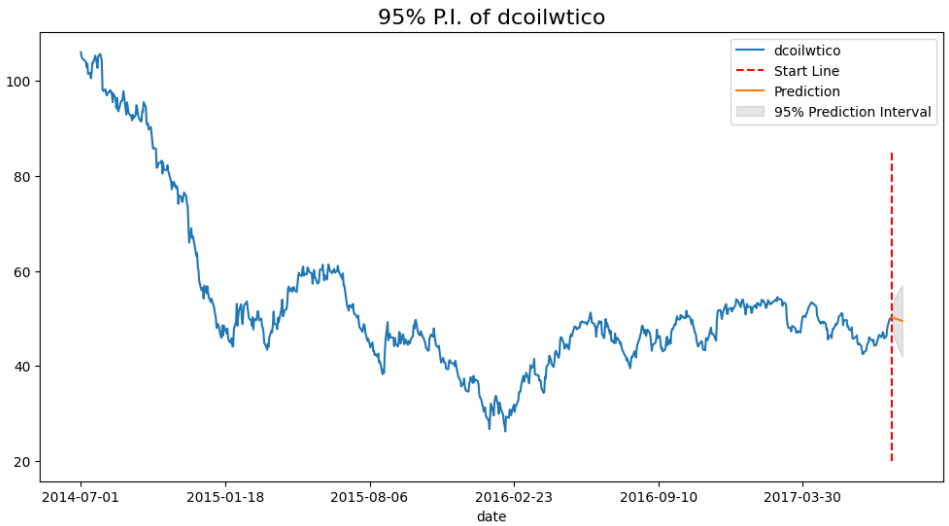
(HINT2) statsmodels의 adfuller 함수를 사용하면 ADF Test를 진행할 수 있습니다.

문제6. 모델을 활용하여 15일 동안의 dcoilwtico를 예측하고, 그 결과를 아래와 같이 시각화해주세요.

(HINT1) ARIMA 모델에 predict를 사용하면 평균, upper bound, lower bound를 얻을 수 있습니다.

(HINT2) matplotlib.pyplot의 vlines, fill_between 함수를 사용하면 됩니다.

- Start Line의 linestyle은 "--", color는 "r"입니다.
- 95% Prediction Interval의 color는 "k"이고 alpha는 0.1입니다.



문제7. 예측값을 이용해 valid의 결측치를 대체하고 missingno 패키지를 사용해 이상이 없는지 확인하세요.

보너스문제1. 차분만으로 잔차 패턴이 사라지는 경우는 거의 없습니다. 따라서 다른 데이터를 활용해 패턴이 사라지지 않는 경우 어떻게 모델링을 해야 하는지 알아보겠습니다. 먼저 family가 BEVERAGES, store_nbr가 1인 data의 date와 sales를 새로운 데이터프레임 beverages1로 저장해주세요.

(HINT) pandas의 loc를 사용하면 쉽게 다중 조건으로 필터링할 수 있습니다.

보너스문제2. 문제4에서 했던 것처럼 beverages1 sales의 time plot과 correlograms를 통해 특징을 파악해주세요. 이때 차분을 통해 ACF, PACF의 패턴을 최대한 제거해보세요.

- 일반적으로 ACF의 한 cycle을 주기로 볼 수 있습니다.

보너스문제3. ARMA 모형의 경우 ACF, PACF를 통해 정확한 차수를 파악하기 힘듭니다. 따라서 AIC나 BIC 같은 Information Criterion을 사용해 모형을 결정합니다. IC를 기준으로 최적의 (S)ARIMA 모형을 찾아주는 auto_arima를 사용해보겠습니다. (colab 기준 3분 정도 소요)

- auto_arima는 여러 모형을 적합하여 IC를 구한 후 가장 IC가 작은 모형을 추천해주기 때문에 각 차수의 max값이 클수록 많은 시간이 걸립니다. 따라서 ACF, PACF를 통해 적절한 max 값을 정해주는 것이 중요합니다.

(HINT1) pmdarima의 auto_arima를 import 해주세요.

(HINT2) 파라미터: max_p=3, max_q=3, d=0, max_P=2, max_Q=1, m=7, seasonal=True

보너스문제4. auto_arima를 통해 구한 모델의 잔차를 이용해 ADF test를 진행하고 15일 동안의 sales를 예측하여 그 결과를 아래와 같이 시각화해주세요. (문제5, 6 참고)