### [Futures pricing: KOSPI 200 example](https://colab.research.google.com/drive/1hvSGNp-qRYMW9DLJDnkQ-7eJ-eEOFwix)

앞서 배운바와 같이 구글 스프레드싯의 데이터를 가져올 수 있도록 환경을 설정하자.

!pip install --upgrade -q gspread

from google.colab import auth

auth.authenticate\_user()

import gspread

from oauth2client.client import GoogleCredentials

gc = gspread.authorize(GoogleCredentials.get\_application\_default())

항상쓰는 모듈들을 가져온다.

import pandas as pd

import numpy as np

필자가 구글스프레드싯에 저장해 놓은 데이터를 가져와서 df 라는 이름의 데이터프레임을 만든다. 참고로 데이터는 다음 링크에 저장이 되어 있다. <https://docs.google.com/spreadsheets/d/1WlOhyZzFE_CRd5coX3g5CDCEESWLx2n85nK4O7efpi8>

링크 URL로 데이터를 가져올 수도 있지만 여기서는 링크에서 하이라이트된 key를 이용해서 데이터를 가져와보자.

worksheet = gc.open\_by\_key('1WlOhyZzFE\_CRd5coX3g5CDCEESWLx2n85nK4O7efpi8').sheet1

rows = worksheet.get\_all\_values()

df = pd.DataFrame.from\_records(rows)

df.head()

0 1 2

0 Date 선물가격 지수

1 2017/09/01 308.4 307.78

2 2017/09/04 305.6 304.19

3 2017/09/05 304.85 304.06

4 2017/09/06 304.3 303.18

df.columns = df.iloc[0] # 첫번째 행을 열들의 이름으로

df = df.drop([0]) # 첫번째 행은 이제 필요없으니 삭제

df.head()

0 Date 선물가격 지수

1 2017/09/01 308.4 307.78

2 2017/09/04 305.6 304.19

3 2017/09/05 304.85 304.06

4 2017/09/06 304.3 303.18

5 2017/09/07 308.1 306.97

df = df.set\_index('Date') # 첫번째 열 정보를 행 이름으로

df.head()

0 선물가격 지수

Date

2017/09/01 308.4 307.78

2017/09/04 305.6 304.19

2017/09/05 304.85 304.06

2017/09/06 304.3 303.18

2017/09/07 308.1 306.97

또는 다음과 같이 읽어들일 수 있습니다.

!pip install gspread-dataframe

from gspread\_dataframe import get\_as\_dataframe, set\_with\_dataframe

worksheet = gc.open\_by\_url('https://docs.google.com/spreadsheets/d/1WlOhyZzFE\_CRd5coX3g5CDCEESWLx2n85nK4O7efpi8/edit?usp=sharing').sheet1

df = get\_as\_dataframe(worksheet)

df = df.set\_index('Date')

df = df.iloc[0:69,0:2]

print(df.head())

print(df.tail())

선물가격 지수

Date

2017/09/01 308.40 307.78

2017/09/04 305.60 304.19

2017/09/05 304.85 304.06

2017/09/06 304.30 303.18

2017/09/07 308.10 306.97

선물가격 지수

Date

2017/12/08 324.75 324.68

2017/12/11 324.85 325.09

2017/12/12 323.80 324.07

2017/12/13 326.20 326.18

2017/12/14 327.65 324.74

type(df.index) # 행이름이 날짜 정보인지 확인

pandas.core.indexes.base.Index

행이름이 날짜 정보가 아니므로 날짜 형식으로 변환하여 확인해보자.

df.index = pd.to\_datetime(df.index)

type(df.index)

pandas.core.indexes.datetimes.DatetimeIndex

이제 선물의 이론가를 계산하는 함수를 만들어보자. 편의상 연속복리와 배당를 가정하면 선물가격은 다음과 같다. *F, S, r, d, t* 는 각각 선물가격(이론가), 현물가격(spot price), 무위험이자율(연율), 배당(연율), 만기까지의 기간(연환산)이다.

[](https://www.codecogs.com/eqnedit.php?latex=F%20%3D%20S%20e%5E%7B(r-d)t%7D%0)

위 공식을 함수로 바꿔보자.

def futures\_price (S, r, d, T, t0):

t = np.busday\_count(t0.date(), T.date()) / 252

F = S \* np.exp((r-d)\*t)

return t

위에서 만든 선물이론가 함수를 데이터에 적용해서 이론가들을 날짜별로 계산해보자.

T = pd.to\_datetime('2017-12-14') # 최종거래일

r = 0.0225 # 이자율

d = 0.015 # 배당률

for n in (df.index):

t0 = n

S = df.loc[n, '지수']

F = futures\_price (S, r, d, T, t0)

f = '{:,.2f}'.format(F)

df.loc[n, '이론가1'] = f

df.dtypes

선물가격 float64

지수 float64

이론가1 object

dtype: object

df 라는 데이터프레임에 선물가격과 지수는 숫자형식인 실수(float)으로 되어 있는데 ‘이론가1’는 문자열로 되어 있다. 따라서 모두 실수로 바꿔주자.

df = df.apply(pd.to\_numeric, errors='coerce')

이제 모두 실수형식으로 바꾸었으니 데이터프레임내에서 자유롭게 연산을 할 수 있다. 관측된 선물가격과 이론가격과의 괴리를 계산하고 이를 ‘괴리’라는 열에 저장해보자.

df['괴리'] = df['선물가격']-df['이론가1']

마지막 다섯개행의 정보를 확인해보자.

print(df.tail())

선물가격 지수 이론가1 괴리

Date

2017-12-08 324.75 324.68 324.72 0.03

2017-12-11 324.85 325.09 325.12 -0.27

2017-12-12 323.80 324.07 324.09 -0.29

2017-12-13 326.20 326.18 326.19 0.01

2017-12-14 327.65 324.74 324.74 2.9