R-Py 컴퓨팅 : Homework1

Part 1. 네이버 뉴스 크롤러 만들기 및 검색

- Q1. 네이버 뉴스에서 금리를 키워드로 링크가 https://news.naver.com/main/read.nhn로 시작하는 모든 링크 수집

먼저 Import 해주기

```
import urllib.request
import urllib.parse
from bs4 import BeautifulSoup
import re
```

Q1의 코드

```
keywords=urllib.parse.quote("금리")
url ='https://search.naver.com/search.naver?where=news&query='+ keywords
+'&sm=tab_opt&sort=0&photo=0&field=0&reporter_article=&pd=3&ds=2020.04.13&de=2020.
04.14'
req=urllib.request.urlopen(url)
data=req.read()
soup=BeautifulSoup(data,'html.parser')
anchor_set=soup.findAll('a')
news_link=[]
for link in anchor_set:
    if(link['href'].startswith('https://news.naver.com/main/read.nhn')):
        news_link.append(link['href'])
```

news_link 결과값

```
['https://news.naver.com/main/read.nhn?
mode=LSD&mid=sec&sid1=101&oid=016&aid=0001661790',
'https://news.naver.com/main/read.nhn?
mode=LSD&mid=sec&sid1=101&oid=011&aid=0003724354',
'https://news.naver.com/main/read.nhn?
mode=LSD&mid=sec&sid1=101&oid=277&aid=0004661182',
'https://news.naver.com/main/read.nhn?
mode=LSD&mid=sec&sid1=102&oid=277&aid=0004660975',
'https://news.naver.com/main/read.nhn?
mode=LSD&mid=sec&sid1=101&oid=215&aid=0000864138',
'https://news.naver.com/main/read.nhn?
mode=LSD&mid=sec&sid1=102&oid=016&aid=0001661681',
'https://news.naver.com/main/read.nhn?
mode=LSD&mid=sec&sid1=101&oid=029&aid=0002593371',
'https://news.naver.com/main/read.nhn?
```

```
  mode=LSD&mid=sec&sid1=101&oid=015&aid=0004323634',
  'https://news.naver.com/main/read.nhn?
  mode=LSD&mid=sec&sid1=101&oid=016&aid=0001661201',
  'https://news.naver.com/main/read.nhn?
  mode=LSD&mid=sec&sid1=101&oid=011&aid=0003723666',
  'https://news.naver.com/main/read.nhn?
  mode=LSD&mid=sec&sid1=101&oid=028&aid=0002493428',
  'https://news.naver.com/main/read.nhn?
  mode=LSD&mid=sec&sid1=101&oid=421&aid=0004583057',
  'https://news.naver.com/main/read.nhn?
  mode=LSD&mid=sec&sid1=101&oid=018&aid=0004619322',
  'https://news.naver.com/main/read.nhn?
  mode=LSD&mid=sec&sid1=004&oid=215&aid=0000863829',
  'https://news.naver.com/main/read.nhn?
  mode=LSD&mid=sec&sid1=101&oid=033&aid=0000040753',
  'https://news.naver.com/main/read.nhn?
  mode=LSD&mid=sec&sid1=101&oid=001&aid=0011544540',
  'https://news.naver.com/main/read.nhn?
  mode=LSD&mid=sec&sid1=101&oid=016&aid=0001661456',
  'https://news.naver.com/main/read.nhn?
  mode=LSD&mid=sec&sid1=105&oid=092&aid=0002186007',
  'https://news.naver.com/main/read.nhn?
  mode=LSD&mid=sec&sid1=101&oid=277&aid=0004660417',
  'https://news.naver.com/main/read.nhn?
  mode=LSD&mid=sec&sid1=101&oid=277&aid=0004660713',
  'https://news.naver.com/main/read.nhn?
  mode=LSD&mid=sec&sid1=101&oid=277&aid=0004660541',
  'https://news.naver.com/main/read.nhn?
  mode=LSD&mid=sec&sid1=101&oid=366&aid=0000508733']
```

뉴스 검색과 연관된 총 기사수 확인(검색어:금리)

```
  count_tag=soup.find('div',{'class','title_desc all_my'})
  count_text=count_tag.find('span').get_text().split()
  total_num=count_text[-1][0:-1].replace(",","")
```

결과값

```
print(count_tag)
```

< div class="title_desc all_my">< span>1-10 / 1,296건< /span>< /div >

```
print(count_text)
```

['1-10', '/', '1,296건']

```
print(total_num)
```

'1296'

- Q2. 이자율 검색 관련 모든 링크를 찾아서 new_link에 담기

Q2코드

new_link를 `List`로 할 경우에는 중복을 허용하므로 `Set`의 자료구조를 이용해야한다.

```
keywords=urllib.parse.quote('이자율')
url='https://search.naver.com/search.naver?where=news&query=' + keywords
+'&sm=tab_opt&sort=0&photo=0&field=0&reporter_article=&pd=3&ds=2020.04.13&de=2020.
04.14'
data=urllib.request.urlopen(url).read()
soup=BeautifulSoup(data,'html.parser')
count_tag=soup.find('div',{'class','title_desc all_my'})
count_text=count_tag.find('span').get_text().split()
total_num=count_text[-1][0:-1].replace(",","")
new_link=set()
for val in range(int(total_num)//10):
    start_val=str(val*10+1)
    url_sample ='https://search.naver.com/search.naver?
where=news&query='+keywords+'&sm=tab_opt&sort=0&photo=0&field=0&reporter_article=&
pd=3&ds=2020.04.13&de=2020.04.14&docid=&nso=so:r,p:from20200413to20200414,a:all&my
news=0&cluster_rank=26&start='+start_val+'&refresh_start=0'

soup_sample=BeautifulSoup(urllib.request.urlopen(url_sample).read(),'html.parser')
    anchor_set_sample=soup_sample.findAll('a')
    for link in anchor_set_sample:
        if(link['href'].startswith('https://news.naver.com/main/read.nhn')):
            new_link.add(link['href'])
```

(**참고** 여기에서 total_num은 **57**)

new_link의 결과값 확인하기(크기 21)

```
{'https://news.naver.com/main/read.nhn?
mode=LSD&mid=sec&sid1=101&oid=277&aid=0004661669',
'https://news.naver.com/main/read.nhn?
mode=LSD&mid=sec&sid1=101&oid=014&aid=0004408522',
'https://news.naver.com/main/read.nhn?
mode=LSD&mid=sec&sid1=101&oid=079&aid=0003349322',
'https://news.naver.com/main/read.nhn?
mode=LSD&mid=sec&sid1=101&oid=629&aid=0000022807',
'https://news.naver.com/main/read.nhn?
mode=LSD&mid=sec&sid1=101&oid=421&aid=0004581969',
'https://news.naver.com/main/read.nhn?
mode=LSD&mid=sec&sid1=103&oid=008&aid=0004393435',
'https://news.naver.com/main/read.nhn?
mode=LSD&mid=sec&sid1=101&oid=008&aid=0004393599',
'https://news.naver.com/main/read.nhn?
mode=LSD&mid=sec&sid1=101&oid=020&aid=0003280942',
'https://news.naver.com/main/read.nhn?
mode=LSD&mid=sec&sid1=101&oid=009&aid=0004555904',
'https://news.naver.com/main/read.nhn?
mode=LSD&mid=sec&sid1=101&oid=018&aid=0004619946',
```

```
'https://news.naver.com/main/read.nhn?
mode=LSD&mid=sec&sid1=101&oid=421&aid=0004585585',
'https://news.naver.com/main/read.nhn?
mode=LSD&mid=sec&sid1=101&oid=008&aid=0004394409',
'https://news.naver.com/main/read.nhn?
mode=LSD&mid=sec&sid1=101&oid=023&aid=0003523129',
'https://news.naver.com/main/read.nhn?
mode=LSD&mid=sec&sid1=101&oid=008&aid=0004393317',
'https://news.naver.com/main/read.nhn?
mode=LSD&mid=sec&sid1=101&oid=277&aid=0004660505',
'https://news.naver.com/main/read.nhn?
mode=LSD&mid=sec&sid1=101&oid=015&aid=0004323348',
'https://news.naver.com/main/read.nhn?
mode=LSD&mid=sec&sid1=103&oid=016&aid=0001662032',
'https://news.naver.com/main/read.nhn?
mode=LSD&mid=sec&sid1=101&oid=018&aid=0004619016',
'https://news.naver.com/main/read.nhn?
mode=LSD&mid=sec&sid1=101&oid=015&aid=0004323403',
'https://news.naver.com/main/read.nhn?
mode=LSD&mid=sec&sid1=101&oid=018&aid=0004620165'}
```

- Q3. new_link 이용하여 뉴스의 본문내용과 제목을 list에 넣기

Q3코드

코드에서 제목을 뽑아내기 위해 `find('h3',{'id':'articleTitle'}).get_text()`를 이용했고, 내용을 뽑아내기위해 `find('div',{'id':'articleBodyContents'}).get_text()`를 이용했다.

```
keywords = urllib.parse.quote("금리")
url = 'https://search.naver.com/search.naver?where=news&query=' + keywords +
'&sm=tab_opt&sort=0&photo=0&field=0&reporter_article=&pd=3&ds=2020.04.13&de=2020.0
4.14'
req = urllib.request.urlopen(url)
data = req.read()
soup = BeautifulSoup(data, 'html.parser')
anchor_set = soup.findAll('a')
news_link = []
for link in anchor_set:
   if(link['href'].startswith('https://news.naver.com/main/read.nhn')):
       news_link.append(link['href'])

title_list=[]
text_list=[]
for url in news_link:
   data=urllib.request.urlopen(url).read()
   soup=BeautifulSoup(data,'html.parser')
   title_tag=soup.find('h3',{'id':'articleTitle'}).get_text()
   text_tag=soup.find('div',{'id':'articleBodyContents'}).get_text()
   title_list.append(title_tag)
   text_list.append(text_tag)
```

title_list와 text_list의 결과값

```
pd.DataFrame(text_list)
```

(앞의 \n은 제거해주었다)

```
0    5% 적금은 1만좌 한정예금금리도 최고 2.3%[헤럴드경제=박자연...
1    모바일정기적금 1만좌 한정판매[서울경제] 애큐온저축은행은 '애큐온...
2    [아시아경제 김민영 기자] 애큐온저축은행이 모바일 전용 예·적금 ...
3    코로나19 장기화에 따른 민생·경제 대책...561개 업체 950...
4    [한국경제TV 신인규 기자]서울 용산구가 중소기업·소상공인·청년기...
5    561개 업체 9502만원 상당 이자비용 낮춰줘용산구청 전경 이미...
6    제로금리 시대에 3%, 10%?…한국투자증권 발행어음 ...
7    고객 5000명에 30만원 한도한국투자증권은 비대면계좌 개설 고객...
8    [헤럴드경제=이태형 기자]한국금융지주 자회사 한국투자증...
9    [서울경제] 한국투자증권은 뱅키스 계좌개설 고객과 금융상품권 등록...
10   자본시장연구원 보고서 주요국, 매입 총액·일정 공표 한은은 시장 ...
11   "국채금리 하향 안정화 유도해야"© 뉴스1(서울=뉴스1) 송상현 ...
12   "매수 여력없는데 발행 증가 불가피한 '수급불균형'이 원인""국채...
13   [한국경제TV 정희형 기자]국채금리의 하향 안정화를 위해 적극적인...
14   겨울이 지나고 지천에 꽃이 피었지만 몸도, 마음도 춥다...
15   (서울=연합뉴스) 성서호 기자 = KB국민은행은 스마트...
16   [헤럴드경제=이승환 기자] KB국민은행은 스마트 공장 보급 확산과...
17   SKT PASS 앱 가입 고객 전용 상품(지디넷코리아=손예술  기...
18   [아시아경제 김민영 기자] Sh수협은행은 SK텔레콤과 패스(PAS...
19   [아시아경제 김민영 기자] 금리빙하기에 연 5%에 달하는 이자를 ...
20   초저금리 시대 추가 금리인하 예상 속 고금리 상품 러시[아시아경제...
21   신종 코로나 바이러스 감염증(코로나 19)이 확산되면서...
```

```
pd.DataFrame(title_list)
```

```
0            '또 나왔다' 5% 금리 정기적금... 한도는 240만원
1     애큐온저축銀, 모바일 전용 상품 3종 출시···최대 연 5.0% 금리
2     애큐온저축은행, 모바일 전용 예·적금 상품 3종 출시…최대 5% 금리
3            용산구, 중소·소상공인·청년기업 융자금리 0%대로 인하
4    용산구, 코로나 피해 중소상공인·청년기업 살린다…융자금리 0%대로 인하
5                용산구, 중소·청년기업 융자금리 0%대로 인하
6       제로금리 시대에 3%, 10%?…한국투자증권 발행어음 특판 이벤트
7               한투證 발행어음 특판…최대 연 3%·10% 금리
8       제로금리 시대에 3%, 10%…한국투자증권 발행어음 특판 이벤트
9    한국투자증권, 고객 대상 발행어음 특판 이벤트···최대 연 10% 금리
10               "한은, 국채 매입 사전공표로 금리 낮춰야"
11           자본硏 "국채매입제도 도입·기준금리 추가 인하 필요"
12         자본硏 "국채금리 높은 수준…국채매입제도 도입해 사들여야"
13      자본시장연구원 "국채금리 안정화 위해 국채매입·금리인하 필요"
14               [우정이야기]우체국보험 약관 대출 금리 인하
15          국민은행, 스마트 공장·규제자유특구 금리우대 대출 출시
16               국민은행, 스마트 공장 금리우대 대출 출시
17     수협은행, 1만명 선착순 최대 3.5% 금리주는 6개월 적금 출시
18          수협은행, 패스 앱 전용 적금 출시…금리 연 2.8%
19       초저금리 시대 5% 금리 '감지덕지'…고금리 적금 눈길(종합)
```

```
20              금리 빙하기에 5%라니…"고금리 적금으로 갈아타자"
21            코로나, 나이롱환자·주택대출한도 줄이고 저축銀 금리높였다
```

- Q4. title_list에서 **금리**로 시작하고 **인하**로 끝내는 문자열이 있는지 찾고, 제목을 출력

코드

```
for title in title_list:
    if re.search('.*금리.*인하',title):
        print(title)
```

결과값

```
용산구, 중소·소상공인·청년기업 융자금리 0%대로 인하
용산구, 코로나 피해 중소상공인·청년기업 살린다…융자금리 0%대로 인하
용산구, 중소·청년기업 융자금리 0%대로 인하
자본硏 "국채매입제도 도입·기준금리 추가 인하 필요"
자본시장연구원 “국채금리 안정화 위해 국채매입·금리인하 필요”
[우정이야기]우체국보험 약관 대출 금리 인하
```

Part2. 보스턴 주택가격 데이터 분석하기

- Q5. 데이터 전처리 시행

Import 먼저 하기

```
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
import seaborn as sns
```

결측치 코드인 **na**와 **NaN** 모두 실제 결측치가 되도록하고 **제거**하기
(**주의** csv를 불러오는데 컴퓨터마다 다를 수 있음)

코드

```
missing_value=['na','NaN']
data=pd.read_csv('C:\\Users\\rlqja\\OneDrive\\바탕 화면
\\Rpython\\boston_csv.csv',na_values=missing_value)
data.dropna(inplace=True)
```

- Q6. 요약 통계구하기

describe()를 적용하여 요약통계구하기
코드

```
summarize=data.describe()
```

```
print(summarize)
```

```
              CRIM          ZN       INDUS  ...       LSTAT        MEDV   CAT. MEDV
count   502.000000  502.000000  502.000000  ...  502.000000  502.000000  502.000000
mean      3.641708   11.418327   11.163765  ...   12.681514   22.564343    0.167331
std       8.629979   23.396912    6.873538  ...    7.155966    9.217580    0.373643
min       0.009060    0.000000    0.460000  ...    1.730000    5.000000    0.000000
25%       0.082492    0.000000    5.190000  ...    6.950000   17.100000    0.000000
50%       0.262660    0.000000    9.690000  ...   11.395000   21.200000    0.000000
75%       3.689387   12.500000   18.100000  ...   17.057500   25.000000    0.000000
max      88.976200  100.000000   27.740000  ...   37.970000   50.000000    1.000000
```

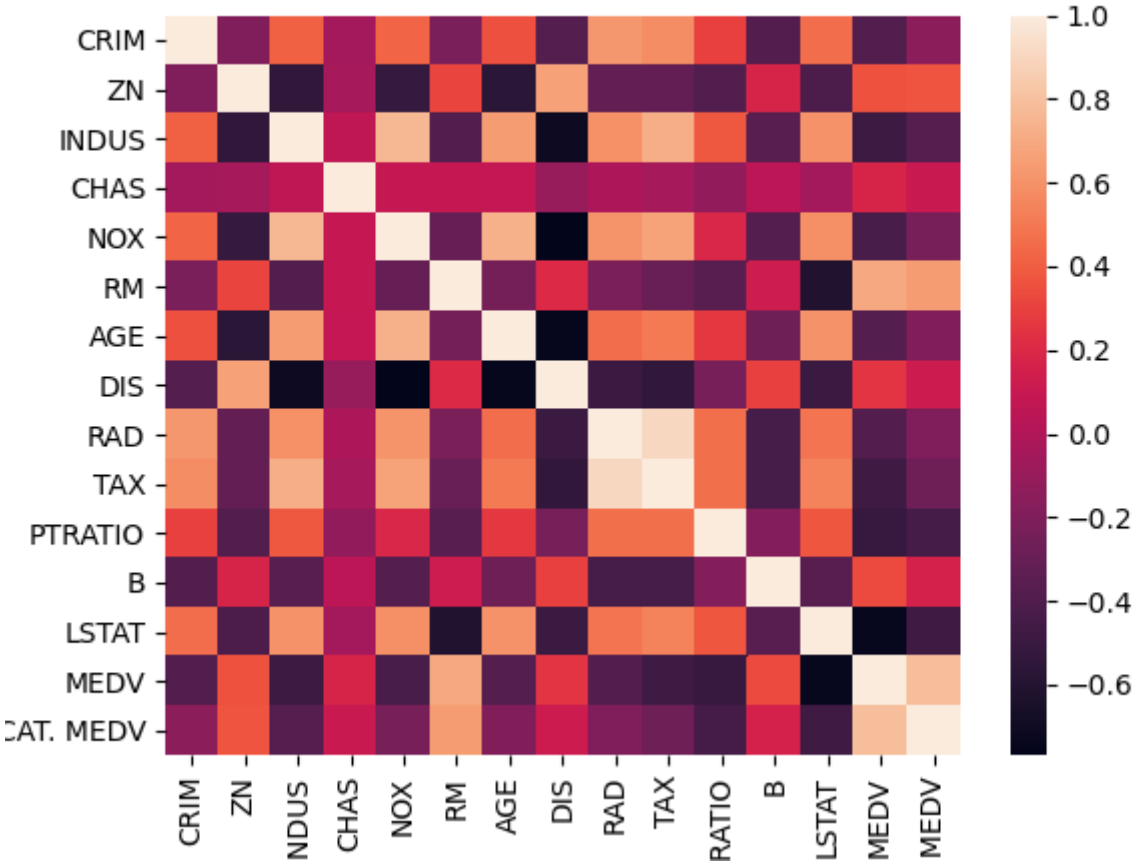
상관관계를 구한후 seaborn의 heatmap 구한하기
코드

```
corr=data.corr(method='pearson')
heat=sns.heatmap(corr)
plt.show()
```

상관계수(corr출력)

```
                CRIM        ZN     INDUS  ...     LSTAT      MEDV  CAT. MEDV
CRIM        1.000000 -0.201718  0.406051  ...  0.455111 -0.390650  -0.153676
ZN         -0.201718  1.000000 -0.535297  ... -0.414400  0.359445   0.364826
INDUS       0.406051 -0.535297  1.000000  ...  0.603096 -0.485291  -0.369083
CHAS       -0.056841 -0.043384  0.062010  ... -0.055137  0.174682   0.107779
NOX         0.421132 -0.517580  0.764556  ...  0.591440 -0.427771  -0.233252
RM         -0.219579  0.311633 -0.391454  ... -0.614499  0.695529   0.641968
AGE         0.354022 -0.569038  0.646623  ...  0.604278 -0.376197  -0.190483
DIS        -0.380747  0.664917 -0.709355  ... -0.498160  0.248532   0.118635
RAD         0.625027 -0.314729  0.594547  ...  0.486908 -0.386034  -0.201537
TAX         0.582237 -0.316928  0.720994  ...  0.542725 -0.472305  -0.276752
PTRATIO     0.290985 -0.391435  0.381237  ...  0.374125 -0.507027  -0.445634
B          -0.384175  0.176832 -0.356247  ... -0.365301  0.335823   0.156951
LSTAT       0.455111 -0.414400  0.603096  ...  1.000000 -0.741372  -0.472958
MEDV       -0.390650  0.359445 -0.485291  ... -0.741372  1.000000   0.790267
CAT. MEDV  -0.153676  0.364826 -0.369083  ... -0.472958  0.790267   1.000000
```

heatmap 결과값



- Q7. 단순회귀분석 모형을 Training Set과 Test Set을 통해 구현하기

train set(75)과 test set(25) 구분하고 fitting 시키기
train set이 표본의 75%차지하기 위해서 index 0번째부터 502 x 0.75까지 설정하고 index가 502*0.75이상은 **test set**으로 설정했다. **Yhat_train**은 fitting 시켰을 때 x_train이 **independent variable**로 들어갔을 때 나오는 **y 결과값**이다.

```
x_train=data[['LSTAT']].loc[:502*0.75]
x_test=data[['LSTAT']].loc[502*0.75:]
y_train=data['MEDV'].loc[:502*0.75]
y_test=data['MEDV'].loc[502*0.75:]
lm=LinearRegression()
lm.fit(x_train,y_train)
#y^으로서 예측값 구하기
Yhat_train=lm.predict(x_train)
```

Training Set에 대한 MSE, 계수(기울기), 상수항, R^2

```
print(mean_squared_error(y_train,Yhat_train))   #MSE
print(lm.coef_) #coefficient(계수)
print(lm.intercept_) #y절편(상수)
print(lm.score(x_train,y_train))  #R square
```

결과값-위 코드에서 순서대로 시행
44.17485048854666(MSE)
[-0.94064425](coefficient)
35.05730880252354(상수항)
0.42853775675518746(R^2)

Training Set에 대한 회귀분석 추정 계수 값을 이용하여 Test Set에서 예측 후 MSE 구하기

```
Yhat_test=pd.DataFrame(lm.coef_[0]*x_test + lm.intercept_) # 회귀분석 추정계수를 바탕으로 예측
print(mean_squared_error(y_test,Yhat_test))
```

결과값
21.812477828486838(Test set예측한 후의 MSE)

- Q8. 다중회귀분석 모형을 Training Set과 Test Set을 통해 구현하기

train set(75)과 test set(25) 구분하고 fitting 시키기
Q7과 같이 train set과 test set을 잡음. 대신 x_train대신 **x_train2** 사용
코드

```
lm=LinearRegression()
x_train2 = data[['LSTAT', 'TAX']].loc[:502*0.75]
x_test2 = data[['LSTAT', 'TAX']].loc[502*0.75:]
y_train2 = data['MEDV'].loc[:502*0.75]
y_test2 = data['MEDV'].loc[502*0.75:]
lm=LinearRegression()
lm.fit(x_train2,y_train2)
```

Training Set에 대한 MSE, 계수(기울기), 상수항, R^2
코드

```
print(lm.coef_) #독립변수 LSTAT와 TAX의 계수
print(lm.intercept_) #상수
print(lm.score(x_train2,y_train2))#R square
Yhat2=lm.predict(x_train2)
print(mean_squared_error(y_train2,Yhat2))
```

결과값-위 순서대로 시행
[-0.94613249 0.00143417]([0]은 LSTAT 독립변수의 계수, [1]은 TAX 독립변수의 계수)
34.64066662940464(상수항)
0.42881615940338896(R^2)
44.153329564801844(MSE)

Training Set에 대한 회귀분석 추정 계수 값을 이용하여 Test Set에서 예측 후 MSE 구하기
코드

```
Yhat2_test =
pd.DataFrame(lm.coef_[0]*x_test2['LSTAT']+lm.coef_[1]*x_test2['TAX']+lm.intercept_
) #회귀분석 추정계수 값을 바탕으로 예측
print(mean_squared_error(y_test2, Yhat2_test)) #mean squared error 구하기
```

결과값

23.86708872467574(MSE)