다중선형회귀

독립변수가 두 개 이상인 경우의 회귀분석

분석 정확도를 높이기 위해 적절하지 않은 변수를 추려내는 과정을 반복적으로 수행하여 최적의 독립변수 그룹을 찾아내는 것을 목표로 한다.

패키지 참조

```
from pandas import read_excel
from statsmodels.formula.api import ols
from matplotlib import pyplot as plt
import seaborn as sb
import sys
import os
```

```
sys.path.append(os.path.dirname(os.path.dirname(os.getcwd())))
from helper import ext_ols
```

데이터 가져오기

필드	설명
CRIM	범죄율

필드	설명
ZN	25,000 평방피트를 초과 거주지역 비율
INDUS	비소매상업지역 면적 비율
CHAS	찰스강의 경계에 위치한 경우는 1, 아니면 0
NOX	일산화질소 농도
RM	주택당 방 수
AGE	1940년 이전에 건축된 주택의 비율
DIS	직업센터의 거리
RAD	방사형 고속도로까지의 거리
TAX	재산세율
PTRATIO	학생/교사 비율
B	인구 중 흑인 비율
LSTAT	인구 중 하위 계층 비율
MEDV	집값
CAT.MEDV	$3000 이상 여부

```
df = read_excel("https://data.hossam.kr/E04/boston.xlsx")
df
```

	CRIM	ZN	INDUS	CHAS	NOX	RM	AGE	DIS	RAD	TAX
0	0.00632	18.0	2.31	0	0.538	6.575	65.2	4.0900	1	296
1	0.02731	0.0	7.07	0	0.469	6.421	78.9	4.9671	2	242
2	0.02729	0.0	7.07	0	0.469	7.185	61.1	4.9671	2	242
3	0.03237	0.0	2.18	0	0.458	6.998	45.8	6.0622	3	222
4	0.06905	0.0	2.18	0	0.458	7.147	54.2	6.0622	3	222
...	...	...	...	...	...	...	...	...	...	...
501	0.06263	0.0	11.93	0	0.573	6.593	69.1	2.4786	1	273
502	0.04527	0.0	11.93	0	0.573	6.120	76.7	2.2875	1	273
503	0.06076	0.0	11.93	0	0.573	6.976	91.0	2.1675	1	273
504	0.10959	0.0	11.93	0	0.573	6.794	89.3	2.3889	1	273
505	0.04741	0.0	11.93	0	0.573	6.030	80.8	2.5050	1	273

506 rows × 15 columns

파이썬의 ols 객체로 분석

```
model = ols("MEDV ~ CRIM + INDUS", data=df)
fit = model.fit()
fit.summary()
```

다중선형회귀

패키지 참조

데이터 가져오기

파이썬의 ols 객체로 분석

모듈화 한 기능을 활용

OLS Regression Results

Dep. Variable:	MEDV	**R-squared:**	0.278
Model:	OLS	**Adj. R-squared:**	0.275
Method:	Least Squares	**F-statistic:**	96.83
Date:	Tue, 25 Jul 2023	**Prob (F-statistic):**	2.66e-36
Time:	14:52:54	**Log-Likelihood:**	-1757.8
No. Observations:	506	**AIC:**	3522.
Df Residuals:	503	**BIC:**	3534.
Df Model:	2		
Covariance Type:	nonrobust		

	coef	**std err**	**t**	**P>|t|**	**[0.025**	**0.975]**
Intercept	29.2483	0.670	43.624	0.000	27.931	30.566
CRIM	-0.2455	0.044	-5.536	0.000	-0.333	-0.158
INDUS	-0.5234	0.056	-9.414	0.000	-0.633	-0.414

Omnibus:	193.751	**Durbin-Watson:**	0.739
Prob(Omnibus):	0.000	**Jarque-Bera (JB):**	653.883
Skew:	1.800	**Prob(JB):**	1.03e-142
Kurtosis:	7.248	**Cond. No.**	27.7

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.

모듈화 한 기능을 활용

```
cls = list(df.columns)
cls.remove("MEDV")
cls.remove("CAT. MEDV")
cls
```

```
['CRIM',
 'ZN',
 'INDUS',
 'CHAS',
 'NOX',
 'RM',
 'AGE',
 'DIS',
 'RAD',
 'TAX',
 'PTRATIO',
 'B',
 'LSTAT']
```

다중선형회귀

패키지 참조

데이터 가져오기

파이썬의 ols 객체로 분석

모듈화 한 기능을 활용

```
model, fit, summary, table, result, goodness, varstr = ext_ols(df, x=cls
```

```
summary
```

OLS Regression Results

Dep. Variable:	MEDV	**R-squared:**	0.741
Model:	OLS	**Adj. R-squared:**	0.734
Method:	Least Squares	**F-statistic:**	108.1
Date:	Tue, 25 Jul 2023	**Prob (F-statistic):**	6.72e-135
Time:	14:55:00	**Log-Likelihood:**	-1498.8
No. Observations:	506	**AIC:**	3026.
Df Residuals:	492	**BIC:**	3085.
Df Model:	13		
Covariance Type:	nonrobust		

	coef	**std err**	**t**	**P>|t|**	**[0.025**	**0.975]**
Intercept	36.4595	5.103	7.144	0.000	26.432	46.487
CRIM	-0.1080	0.033	-3.287	0.001	-0.173	-0.043
ZN	0.0464	0.014	3.382	0.001	0.019	0.073

다중선형회귀

패키지 참조

데이터 가져오기

파이썬의 ols 객체로 분석

모듈화 한 기능을 활용

INDUS	0.0206	0.061	0.334	0.738	-0.100	0.141
CHAS	2.6867	0.862	3.118	0.002	0.994	4.380
NOX	-17.7666	3.820	-4.651	0.000	-25.272	-10.262
RM	3.8099	0.418	9.116	0.000	2.989	4.631
AGE	0.0007	0.013	0.052	0.958	-0.025	0.027
DIS	-1.4756	0.199	-7.398	0.000	-1.867	-1.084
RAD	0.3060	0.066	4.613	0.000	0.176	0.436
TAX	-0.0123	0.004	-3.280	0.001	-0.020	-0.005
PTRATIO	-0.9527	0.131	-7.283	0.000	-1.210	-0.696
B	0.0093	0.003	3.467	0.001	0.004	0.015
LSTAT	-0.5248	0.051	-10.347	0.000	-0.624	-0.425

Omnibus:	178.041	**Durbin-Watson:**	1.078
Prob(Omnibus):	0.000	**Jarque-Bera (JB):**	783.126
Skew:	1.521	**Prob(JB):**	8.84e-171
Kurtosis:	8.281	**Cond. No.**	1.51e+04

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.

다중선형회귀

패키지 참조

데이터 가져오기

파이썬의 ols 객체로 분석

모듈화 한 기능을 활용

[2] The condition number is large, 1.51e+04. This might indicate that there are strong multicollinearity or other numerical problems.

```
table
```

		B	표준오차	β	t	유의확률	VIF
종속변수	**독립변수**						
MEDV	**CRIM**	-0.1080	0.033	0	-3.287*	0.001	1.792192
	ZN	0.0464	0.014	0	3.382*	0.001	2.298758
	INDUS	0.0206	0.061	0	0.334*	0.738	3.991596
	CHAS	2.6867	0.862	0	3.118*	0.002	1.073995
	NOX	-17.7666	3.820	0	-4.651*	0.000	4.393720
	RM	3.8099	0.418	0	9.116*	0.000	1.933744
	AGE	0.0007	0.013	0	0.052*	0.958	3.100826
	DIS	-1.4756	0.199	0	-7.398*	0.000	3.955945
	RAD	0.3060	0.066	0	4.613*	0.000	7.484496
	TAX	-0.0123	0.004	0	-3.280*	0.001	9.008554
	PTRATIO	-0.9527	0.131	0	-7.283*	0.000	1.799084
	B	0.0093	0.003	0	3.467*	0.001	1.348521
	LSTAT	-0.5248	0.051	0	-10.347*	0.000	2.941491

```
result
```

```
'R(0.741), R^2(0.734), F(108.1), 유의확률(6.72e-135), Durbin-Watson(1.078
```

```
goodness
```

```
'MEDV에 대하여 CRIM,ZN,INDUS,CHAS,NOX,RM,AGE,DIS,RAD,TAX,PTRATIO,B,LSTAT로
```

```
varstr
```

```
['CRIM의 회귀계수는 -0.1080(p<0.05)로, MEDV에 대하여 유의미한 예측변인인 것으로
 'ZN의 회귀계수는 0.0464(p<0.05)로, MEDV에 대하여 유의미한 예측변인인 것으로 나타
 'INDUS의 회귀계수는 0.0206(p>0.05)로, MEDV에 대하여 유의하지 않은 예측변인인 것
 'CHAS의 회귀계수는 2.6867(p<0.05)로, MEDV에 대하여 유의미한 예측변인인 것으로 나
 'NOX의 회귀계수는 -17.7666(p<0.05)로, MEDV에 대하여 유의미한 예측변인인 것으로
 'RM의 회귀계수는 3.8099(p<0.05)로, MEDV에 대하여 유의미한 예측변인인 것으로 나타
 'AGE의 회귀계수는 0.0007(p>0.05)로, MEDV에 대하여 유의하지 않은 예측변인인 것으
 'DIS의 회귀계수는 -1.4756(p<0.05)로, MEDV에 대하여 유의미한 예측변인인 것으로 나
 'RAD의 회귀계수는 0.3060(p<0.05)로, MEDV에 대하여 유의미한 예측변인인 것으로 나
 'TAX의 회귀계수는 -0.0123(p<0.05)로, MEDV에 대하여 유의미한 예측변인인 것으로 나
 'PTRATIO의 회귀계수는 -0.9527(p<0.05)로, MEDV에 대하여 유의미한 예측변인인 것으
```

다중선형회귀

패키지 참조

데이터 가져오기

파이썬의 ols 객체로 분석

모듈화 한 기능을 활용

```
'B의 회귀계수는 0.0093(p<0.05)로, MEDV에 대하여 유의미한 예측변인인 것으로 나타
'LSTAT의 회귀계수는 -0.5248(p<0.05)로, MEDV에 대하여 유의미한 예측변인인 것으로
```