Sklearn

import sklearn.linear_model as skl_lm

오늘은 sklearn.linear_model.LinearRegression 에 대해서 알아볼 것이다.

형태는 **class** sklearn.linear_model.**LinearRegression(fit_intercept=True, normalize_False, copy_X=True, n_jobs=1)**이다.

우선 parameter들을 살펴보면 다음과 같다.

-**fit_intercept**: boolean type이며, 선택적이고, default는 True이다. 여기서 intercept는 y절편으로, 선형회귀식에서는 β_0이다. 기본적으로는 구하게 되어있지만, 예를 들어 data가 이미 centered 되어 있는 경우에는 False라고 설정해주면 된다.

-**normalize**: 역시 boolean type이고, 선택적인데, default는 False이다. 이 파라미터의 경우에는 위의 fit_intercept가 False인 경우에 무시된다. 왜냐하면, centered data이기 때문이다. centered를 한 것은 이미 자신이 조치를 취한 것이기 때문이다. 만약에 True라면, X들은 정규화(normalize)가 되는데, 이는 X값에서 평균을 빼고 이를 L2-norm으로 나눔으로써 실행된다. 그런데, 만약에 표준화(standardize)를 하고 싶으면, normalize = True를 하기 전에, sklearn.preprocessing.StandardScaler을 사용하길 바란다. 보통은 표준화를 더 많이 하므로, False로 설정하면 될 것 같다.

*이때, normalization은 요소값-최솟값/최댓값-최솟값 으로 0과 1사이의 값으로 나타내는 것이고,

standardization은 특정한 분포들의 평균과 분산 혹은 표준편차를 이용해, 특정값-평균/표준편차로 해당분포에서의 이 값이 평균으로부터의 위치를 표준편차 단위로 옮겨서 나타낸 것이다.

-**copy_X**: boolean type, 선택적, default는 True이다. True이면, 계속 지정해줘야한다.

-**n_jobs**: 정수형태고, 선택적이며, default는 1이다. 이는 computation과 연관이 있다. n_jobs가 1이라면, 모든 CPU가 사용된다는 뜻이다. 선형회귀를 하는데, computation문제는 잘 일어나지 않을 거 같아서, 그냥 1로 설정하면 좋을 것 같다.

-

Statsmodels.

import statsmodels.api as sm

형태는 **class** statsmodels.regression.linear_model.**OLS(endog, exog=None, missing='none',hasconst=None, **kwargs)**

parameter들은 다음과 같다.

-**endog**: array 형태이다. 1-d endogenous 반응변수이다. 종속변수이다.

-**exog**: array 형태이다. nobs x k array 형태로 nobs는 관측치의 개수이고 k는 regressor의 개수이다. y절편은 default값으로 주어지지 않는다. 구하고 싶으면, statsmodels.tools.add.constant를 쓰면 된다.(→독립변수라고 보면 된다.)

*이때, endog와 exog의 matrix size는 같아야한다.

-**missing**: 문자열 형태이다. 가능한 문자열은 'none', 'drop', 'raise'이다. 'none'이면, nan값 확인이 되지 않는다. 'drop'이면, nan값은 삭제된다. 'raise'이면, error가 뜬다.(If 'raise', an error is raised.) default는 'none'이다. nan값은 따로 처리하는 방법이 좋은 거 같아서 그런 것 같다.

-**hasconst**: none 또는 bool이다. 사용자가 상수를 지정했는지 여부를 나타낸다. True라면 상수는 체크되지 않고, k_constant가 1로 설정되어, 모든 통계량이 constant가 있는 것처럼 계산된다. 만약에 False라면 상수는 체크되지 않고 k_constant가 0으로 설정된다.

*이게 무슨 소리인지 몰라서 돌려보니, hasconst = True라고 하면, Prob(F-statistic) = 1로 설정되었다. none인 경우와 False인 경우에는 차이가 없었다.

```
est=sm.regression.linear_model.OLS(endog=y,exog=x,missing='raise',hasconst=True).fit()
est.summary()
```

```
C:\ProgramData\Anaconda3\lib\site-packages\scipy\stats\stats.py:1334: UserWarning: kurtosistest only valid for n>=2
  "anyway, n=%i" % int(n))
```

OLS Regression Results

Dep. Variable:	y	**R-squared:**	-0.079
Model:	OLS	**Adj. R-squared:**	-0.214
Method:	Least Squares	**F-statistic:**	-0.5847
Date:	Fri, 16 Mar 2018	**Prob (F-statistic):**	1.00
Time:	21:24:53	**Log-Likelihood:**	-25.120
No. Observations:	10	**AIC:**	54.24
Df Residuals:	8	**BIC:**	54.85
Df Model:	1		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
x1	0.1385	0.097	1.428	0.191	-0.085	0.362
x2	0.1498	0.091	1.647	0.138	-0.060	0.360

Omnibus:	2.821	**Durbin-Watson:**	0.401
Prob(Omnibus):	0.244	**Jarque-Bera (JB):**	1.683
Skew:	0.965	**Prob(JB):**	0.431
Kurtosis:	2.438	**Cond. No.**	3.21

```
est=sm.regression.linear_model.OLS(endog=y,exog=x,missing='raise',hasconst=None).fit()
est.summary()
```

```
C:\ProgramData\Anaconda3\lib\site-packages\scipy\stats\stats.py:1334: UserWarning: kurtosistest only valid for n>=20
  "anyway, n=%i" % int(n))
```

OLS Regression Results

Dep. Variable:	y	**R-squared:**	0.769
Model:	OLS	**Adj. R-squared:**	0.711
Method:	Least Squares	**F-statistic:**	13.30
Date:	Fri, 16 Mar 2018	**Prob (F-statistic):**	0.00286
Time:	21:25:25	**Log-Likelihood:**	-25.120
No. Observations:	10	**AIC:**	54.24
Df Residuals:	8	**BIC:**	54.85
Df Model:	2		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
x1	0.1385	0.097	1.428	0.191	-0.085	0.362
x2	0.1498	0.091	1.647	0.138	-0.060	0.360

Omnibus:	2.821	**Durbin-Watson:**	0.401

Table 3.4 & 3.6 - Statsmodels

In [17]:
```
est = smf.ols('Sales ~ TV + Radio + Newspaper', advertising).fit()
est.summary()
```

OLS Regression Results

Dep. Variable:	Sales	**R-squared:**	0.897
Model:	OLS	**Adj. R-squared:**	0.896
Method:	Least Squares	**F-statistic:**	570.3
Date:	Tue, 09 Jan 2018	**Prob (F-statistic):**	1.58e-96
Time:	23:14:15	**Log-Likelihood:**	-386.18
No. Observations:	200	**AIC:**	780.4
Df Residuals:	196	**BIC:**	793.6
Df Model:	3		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	2.9389	0.312	9.422	0.000	2.324	3.554
TV	0.0458	0.001	32.809	0.000	0.043	0.049
Radio	0.1885	0.009	21.893	0.000	0.172	0.206
Newspaper	-0.0010	0.006	-0.177	0.860	-0.013	0.011

Omnibus:	60.414	**Durbin-Watson:**	2.084
Prob(Omnibus):	0.000	**Jarque-Bera (JB):**	151.241
Skew:	-1.327	**Prob(JB):**	1.44e-33
Kurtosis:	6.332	**Cond. No.**	454.

statsmodel을 사용하면 method, r-squared값, F 검정통계량, 상관계수, p-value, 왜도, 첨도 등의 데이터에 대해서 상세하게 나온다.

이를 바탕으로 각각의 변수들이 유의한지, 신뢰할 수 있는지, 우리의 데이터가 잘 적합됐는지를 확인할 수 있다.