YONSEI UNIVERSITY

Linear Regression

- Young Jo Choi
- Department of Digital Analytics

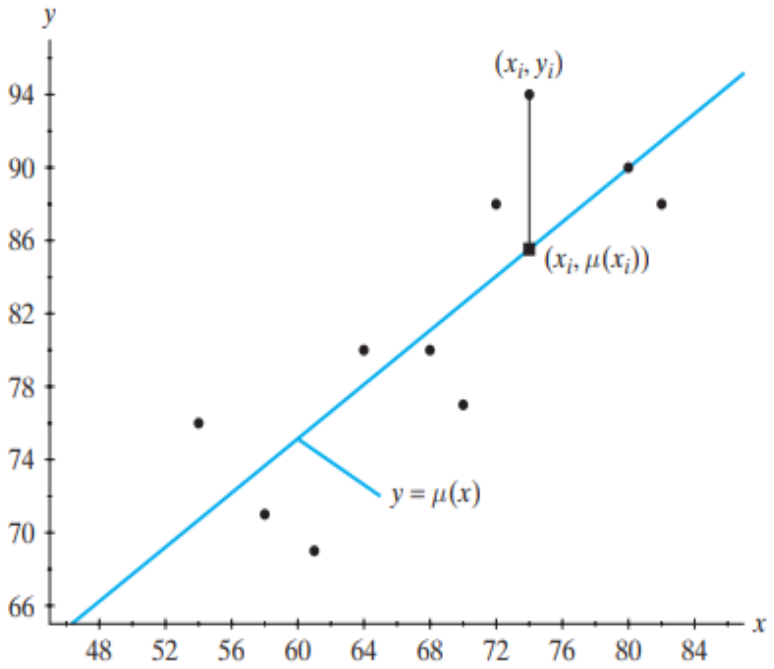
Severance

Simple regression problem

- $E(Y|x) = \mu(x)$ is a linear function of x.
- The data points are $(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)$
- We have linear model,

$$Y_i = \alpha_1 + \beta x_i + \varepsilon_i$$

- (it could be assumed to be equal to forms of $\alpha + \beta x,\ \alpha + \beta x + \gamma x^2, \alpha e^{\beta x}, ..$)



```python
import pandas as pd
import statsmodels.api as sm

iris_df = pd.DataFrame(iris.data, columns=['y','b','c','x'])
iris_df = iris_df.iloc[:30]
model = sm.OLS.from_formula('y~x', iris_df)
result = model.fit()

print(result.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                      y   R-squared:                       0.181
Model:                            OLS   Adj. R-squared:                  0.152
Method:                 Least Squares   F-statistic:                     6.178
Date:                Wed, 07 Jun 2023   Prob (F-statistic):             0.0192
Time:                        01:35:32   Log-Likelihood:                -9.4236
No. Observations:                  30   AIC:                             22.85
Df Residuals:                      28   BIC:                             25.65
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Intercept      4.6394      0.168     27.628      0.000       4.295       4.983
x              1.5701      0.632      2.486      0.019       0.276       2.864
==============================================================================
Omnibus:                        1.473   Durbin-Watson:                   2.015
Prob(Omnibus):                  0.479   Jarque-Bera (JB):                1.068
Skew:                           0.458   Prob(JB):                        0.586
Kurtosis:                       2.872   Cond. No.                         10.7
==============================================================================
```

- $Y_i = \alpha_1 + \beta x_i + \varepsilon_i, \quad \varepsilon \sim N(0, \sigma^2)$
- Let $\alpha_1 = \alpha - \beta\bar{x}$

$$\rightarrow Y_i = \alpha + \beta(x_i - \bar{x}) + \varepsilon_i \ ,$$
$$where\ \bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$$

- In order to find proper α, β, σ^2, since $\boldsymbol{Y_i \sim N\left(\alpha + \beta(x_i - \bar{x}), \sigma^2\right)}$

$$L(\alpha, \beta, \sigma^2) = \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{[y_i - \alpha - \beta(x_i - \bar{x})]^2}{2\sigma^2}\right)$$
$$= \left(\frac{1}{\sqrt{2\pi\sigma^2}}\right)^{n/2} \exp\left\{-\frac{\sum_{i=1}^{n}[y_i - \alpha - \beta(x_i - \bar{x})]^2}{2\sigma^2}\right\}$$

- maximize $L(\alpha, \beta, \sigma^2)$

  $=$ minimize $-\ln L(\alpha, \beta, \sigma^2) = \frac{n}{2}\ln(2\pi\sigma^2) + \frac{\sum_{i=1}^{n}[y_i - \alpha - \beta(x_i - \bar{x})]^2}{2\sigma^2}$

- Select α and β to minimize, so we find the two first-order partial derivatives
- $H(\alpha,\beta)=\sum_{i=1}^{n}[y_i-\alpha-\beta(x_i-\bar{x})]^2$
- $\frac{\partial H(\alpha,\beta)}{\partial\alpha}=2\sum_{i=1}^{n}[y_i-\alpha-\beta(x_i-\bar{x})](-1)=0\rightarrow \boldsymbol{\hat{\alpha}=\bar{Y}}$
- $\frac{\partial H(\alpha,\beta)}{\partial\beta}=2\sum_{i=1}^{n}[y_i-\alpha-\beta(x_i-\bar{x})][-(x_i-\bar{x})]=0\rightarrow \boldsymbol{\hat{\beta}=\frac{\sum_{i=1}^{n}Y_i(x_i-\bar{x})}{\sum_{i=1}^{n}(x_i-\bar{x})^2}}$
- To find the maximum likelihood estimator of σ^2,

$$\frac{\partial[-\ln L(\alpha,\beta,\sigma^2)]}{\partial(\sigma^2)}=\frac{n}{2\sigma^2}-\frac{\sum_{i=1}^{n}[y_i-\alpha-\beta(x_i-\bar{x})]^2}{2(\sigma^2)^2}=0\rightarrow \boldsymbol{\widehat{\sigma^2}=\frac{1}{n}\sum_{i=1}^{n}[y_i-\hat{\alpha}-\hat{\beta}(x_i-\bar{x})]^2}$$

- $\Leftrightarrow \boldsymbol{\mu_{Y|X}=\mu_Y+\rho\frac{\sigma_Y}{\sigma_X}(X-\mu_X)}$
- $\hat{\alpha},\hat{\beta},\widehat{\sigma^2}$ are mutually independent.
- Maximum likelihood estimates of α,β are also called **least squares estimates.**

참고)

- $\frac{\partial H(\alpha,\beta)}{\partial\alpha}=0\rightarrow\sum_{i=1}^{n}y_i-n\alpha-\beta\sum_{i=1}^{n}(x_i-\bar{x})=0$
  since $\sum_{i=1}^{n}(x_i-\bar{x})=0\rightarrow\sum_{i=1}^{n}y_i-n\alpha=0$, thus $\hat{\alpha}=\bar{Y}$
- $\frac{\partial H(\alpha,\beta)}{\partial\beta}=0$ (With α replaced by $\bar{y}$) $\rightarrow$
  $\sum_{i=1}^{n}(y_i-\hat{y})(x_i-\bar{x})-\beta\sum_{i=1}^{n}(x_i-\bar{x})^2=0$

$$\hat{\beta}=\frac{\sum_{i=1}^{n}(y_i-\bar{y})(x_i-\bar{x})}{\sum_{i=1}^{n}(x_i-\bar{x})^2}=\frac{\sum_{i=1}^{n}y_i(x_i-\bar{x})}{\sum_{i=1}^{n}(x_i-\bar{x})^2}\ (\leftarrow\sum_{i=1}^{n}\bar{y}(x_i-\bar{x})=0)$$

Calculation

OLS Regression Results

Dep. Variable:	y	R-squared:	0.181
Model:	OLS	Adj. R-squared:	0.152
Method:	Least Squares	F-statistic:	6.178
Date:	Wed, 07 Jun 2023	Prob (F-statistic):	0.0192
Time:	01:30:51	Log-Likelihood:	-9.4236
No. Observations:	30	AIC:	22.85
Df Residuals:	28	BIC:	25.65
Df Model:	1		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	4.6394	0.168	27.628	0.000	4.295	4.983
x	1.5701	0.632	2.486	0.019	0.276	2.864

Omnibus:	1.473	Durbin-Watson:	2.015
Prob(Omnibus):	0.479	Jarque-Bera (JB):	1.068
Skew:	0.458	Prob(JB):	0.586
Kurtosis:	2.872	Cond. No.	10.7

```
n = 30
p = 1
df = n-p-1

# beta
upper = 0
lower = 0
for i in range(n):
    iris_df_i = iris_df.iloc[i,:]
    upper += (iris_df_i['x']-iris_df['x'].mean())*iris_df_i['y']
    lower += (iris_df_i['x']-iris_df['x'].mean())**2
beta = upper/lower
print('beta : ',beta)

alpha = iris_df['y'].mean()
alpha_1 = alpha-beta*iris_df['x'].mean()
print('intercept : ' ,alpha_1)
```

```
sigma_square = 0
for i in range(30):
    iris_df_i = iris_df.iloc[i,:]
    sigma_square_i = (iris_df_i['y'] - alpha - ㅐ
                      beta*(iris_df_i['x']-iris_df['x'].mean()))
    sigma_square += sigma_square_i**2
sigma_square /= 30
print('sigma_square : ' ,sigma_square)

summ = 0
for i in range(n):
    iris_df_i = iris_df.iloc[i,:]
    summ+=(iris_df_i['x']-iris_df['x'].mean())**2
std_beta = (sigma_square / summ) **(1/2)
std_beta_amend = std_beta * ((n/df)**(1/2))
print('std_beta : ', std_beta_amend)
```

```
beta :  1.5701357466063322
intercept :  4.639366515837106
sigma_square :  0.10974057315233794
std_beta :  0.6316837699752129
```

Details

OLS Regression Results

Dep. Variable:	y	R-squared:	0.181			
Model:	OLS	Adj. R-squared:	0.152			
Method:	Least Squares	F-statistic:	6.178			
Date:	Wed, 07 Jun 2023	Prob (F-statistic):	0.0192			
Time:	01:30:51	Log-Likelihood:	-9.4236			
No. Observations:	30	AIC:	22.85			
Df Residuals:	28	BIC:	25.65			
Df Model:	1					
Covariance Type:	nonrobust					

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	4.6394	0.168	27.628	0.000	4.295	4.983
x	1.5701	0.632	2.486	0.019	0.276	2.864

Omnibus:	1.473	Durbin-Watson:	2.015
Prob(Omnibus):	0.479	Jarque-Bera (JB):	1.068
Skew:	0.458	Prob(JB):	0.586
Kurtosis:	2.872	Cond. No.	10.7

- P-value

  $H_0: \beta_j = 0 \quad \text{vs} \quad H_1: \beta_j \neq 0$

  (귀무가설 : Y에 끼치는 변수 X_j의 영향이 없다. (↔ 대립가설 : 영향이 있다.))
- 95% confidence interval 확인

$$\widehat{\boldsymbol{\alpha}} = \overline{\boldsymbol{Y}},\ \widehat{\boldsymbol{\beta}} = \frac{\sum_{i=1}^{n} \boldsymbol{Y_i}(\boldsymbol{x_i}-\overline{\boldsymbol{x}})}{\sum_{i=1}^{n}(\boldsymbol{x_i}-\overline{\boldsymbol{x}})^2}$$

- $\hat{\alpha}, \hat{\beta}$: normal distribution
  (linear function of independent and normally distributed random variables $Y_1, Y_2, \ldots, Y_n$)
- $\boldsymbol{E(\widehat{\alpha})} = E\left(\frac{1}{n}\sum_{i=1}^{n} Y_i\right) = \frac{1}{n}\sum_{i=1}^{n} E(Y_i) = \frac{1}{n}\sum_{i=1}^{n}[\alpha + \beta(x_i - \bar{x})] = \boldsymbol{\alpha}$
- $\boldsymbol{Var(\widehat{\alpha})} = (\frac{1}{n})^2 \sum_{i=1}^{n} Var(Y_i) = \frac{\boldsymbol{\sigma^2}}{\boldsymbol{n}}$
- $\boldsymbol{E(\widehat{\beta})} = \frac{\sum_{i=1}^{n}(x_i-\bar{x})E(Y_i)}{\sum_{i=1}^{n}(x_i-\bar{x})^2} = \frac{\sum_{i=1}^{n}(x_i-\bar{x})[\alpha+\beta(x_i-\bar{x})]}{\sum_{i=1}^{n}(x_i-\bar{x})^2} = \frac{\cancel{\alpha \sum_{i=1}^{n}(x_i-\bar{x})}+\beta \sum_{i=1}^{n}(x_i-\bar{x})^2}{\sum_{i=1}^{n}(x_i-\bar{x})^2} = \boldsymbol{\beta}$
- $\boldsymbol{Var(\widehat{\beta})} = \sum_{i=1}^{n}\left[\frac{x_i-\bar{x}}{\sum_{j=1}^{n}(x_j-\bar{x})}\right]^2 Var(Y_i) = \frac{\sum_{i=1}^{n}(x_i-\bar{x})^2}{[\sum_{i=1}^{n}(x_i-\bar{x})^2]^2}\sigma^2 = \frac{\boldsymbol{\sigma^2}}{\boldsymbol{\sum_{i=1}^{n}(x_i-\bar{x})^2}}$

$\hat{\alpha}, \hat{\beta}$는 α, β의 unbiased estimators (maximum likelihood estimator)

- Let $\hat{Y}_i = \hat{\alpha} + \hat{\beta}(x_i - \bar{x})$

→ $Y_i - \hat{Y}_i = Y_i - \hat{\alpha} - \hat{\beta}(x_i - \bar{x})$: i-th residual (=ε_i)

- Square of normal distribution = chi-square with degree of freedom 1

→ $\sum_{i=1}^{n}\left\{\frac{\varepsilon_i}{\sigma}\right\}^2 = \sum_{i=1}^{n}\left\{\frac{Y_i-\alpha-\beta(x_i-\bar{x})}{\sigma}\right\}^2 \left(\sim\chi^2(n)\text{: }\chi^2(1)\text{의 sum}\right)$

$= \frac{1}{\sigma^2}\sum_{i=1}^{n}[(\hat{\alpha}-\alpha) + (\hat{\beta}-\beta)(x_i-\bar{x}) + \{Y_i - \hat{\alpha} - \hat{\beta}(x_i-\bar{x})\}]^2$

$= \frac{n(\hat{\alpha}-\alpha)^2}{\sigma^2} + \frac{(\hat{\beta}-\beta)^2\sum_{i=1}^{n}(x_i-\bar{x})^2}{\sigma^2} + \sum_{i=1}^{n}\frac{\{Y_i-\hat{\alpha}-\hat{\beta}(x_i-\bar{x})\}^2}{\sigma^2}$ (나머지 항들은 독립)

Since $\frac{n(\hat{\alpha}-\alpha)^2}{\sigma^2}$ and $\frac{(\hat{\beta}-\beta)^2\sum_{i=1}^{n}(x_i-\bar{x})^2}{\sigma^2}$ have $\chi^2(1)$ (표준정규분포의 제곱)

→ $\sum_{i=1}^{n}\frac{\{Y_i-\hat{\alpha}-\hat{\beta}(x_i-\bar{x})\}^2}{\sigma^2} = \frac{n\widehat{\sigma^2}}{\sigma^2} \sim \chi^2(n-2)$

Variance 의 정의

- $E(\hat{\beta}) = \beta$ • $\mathrm{Var}(\hat{\beta}) = \frac{\sigma^2}{\sum_{i=1}^{n}(x_i - \bar{x})^2}$

- $\frac{n\widehat{\sigma^2}}{\sigma^2} \sim \chi^2(n-2)$

$\rightarrow \frac{(\hat{\beta} - \mathrm{E}(\hat{\beta}))/\mathrm{s.e}(\hat{\beta})}{\sqrt{\frac{n\widehat{\sigma^2}}{\sigma^2} \times \frac{1}{(n-2)}}} = \frac{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}(\frac{\hat{\beta} - \beta}{\sigma})}{\sqrt{\frac{n\widehat{\sigma^2}}{\sigma^2(n-2)}}} \sim T$ with n-2 degrees of freedom

→ p-value 계산하고 각 변수에 대한 귀무가설 검증 가능

(Z: 표준정규분포, V : 자유도가 ν인 chi-square 분포 → $\frac{Z}{\sqrt{V/\nu}}$ ~자유도가 ν인 T 분포)

$\rightarrow [\hat{\beta} - t_{\gamma/2}\sqrt{\frac{n\widehat{\sigma^2}}{(n-2)\sum_{i=1}^{n}(x_i - \bar{x})^2}}, \hat{\beta} + t_{\gamma/2}\sqrt{\frac{n\widehat{\sigma^2}}{(n-2)\sum_{i=1}^{n}(x_i - \bar{x})^2}}]$

($\hat{\beta} \pm t_{\gamma/2}\, s.e(\hat{\beta})\sqrt{\frac{n}{(n-2)}}$ 로 대체 가능)

: 100(1- γ)% confidence interval for β.

- $E(\hat{\alpha}) = \alpha$ • $\mathrm{Var}(\hat{\alpha}) = \frac{\sigma^2}{n}$
- $\frac{(\hat{\alpha}-\mathrm{E}(\hat{\alpha}))/\mathrm{s.e}(\hat{\alpha})}{\sqrt{\frac{n\widehat{\sigma^2}}{\sigma^2}\times\frac{1}{(n-2)}}} = \frac{\frac{\sqrt{n}(\hat{\alpha}-\alpha)}{\sigma}}{\sqrt{\frac{n\widehat{\sigma^2}}{\sigma^2(n-2)}}} = \frac{\hat{\alpha}-\alpha}{\sqrt{\frac{\widehat{\sigma^2}}{(n-2)}}} \sim T$ with n-2 degrees of freedom

$\rightarrow [\hat{\alpha} - t_{\frac{\theta}{2}}\sqrt{\frac{\widehat{\sigma^2}}{(n-2)}}, \hat{\alpha} + t_{\frac{\theta}{2}}\sqrt{\frac{\widehat{\sigma^2}}{(n-2)}}]$ $(\hat{\alpha} \pm t_{\gamma/2}\, s.e(\hat{\alpha})\sqrt{\frac{n}{(n-2)}}$ 로 대체 가능)

: 100(1-θ)% confidence interval for α

- $\frac{n\widehat{\sigma^2}}{\sigma^2} \sim \chi^2(n-2) \rightarrow [\frac{n\widehat{\sigma^2}}{\chi^2_{\frac{\theta}{2}}(n-2)}, \frac{n\widehat{\sigma^2}}{\chi^2_{1-\frac{\theta}{2}}(n-2)}]$: 100(1-θ)% confidence interval for σ^2

- 단순회귀이기 때문에 자유도가 n-2이지만 변수가 p개인 다항회귀라면 각 계수들은 자유도가 n-p-1인 T 분포를 따르게 됨
- T값에 각 $\hat{\alpha}$와 $\hat{\beta}$의 표준편차(*$\sqrt{n/(n-p-1)}$)를 곱하여 confidence interval 계산

Calculation

OLS Regression Results

Dep. Variable:	y	R-squared:	0.181
Model:	OLS	Adj. R-squared:	0.152
Method:	Least Squares	F-statistic:	6.178
Date:	Wed, 07 Jun 2023	Prob (F-statistic):	0.0192
Time:	01:30:51	Log-Likelihood:	-9.4236
No. Observations:	30	AIC:	22.85
Df Residuals:	28	BIC:	25.65
Df Model:	1		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	4.6394	0.168	27.628	0.000	4.295	4.983
x	1.5701	0.632	2.486	0.019	0.276	2.864

Omnibus:	1.473	Durbin-Watson:	2.015
Prob(Omnibus):	0.479	Jarque-Bera (JB):	1.068
Skew:	0.458	Prob(JB):	0.586
Kurtosis:	2.872	Cond. No.	10.7

```
from scipy.stats import t
n = 30
p = 1
df = n-p-1
mean = beta
rv = t(df)

t_val = (beta / std_beta) / (n/df)**(1/2)
print('t값 : ',t_val)

p_value = (1-rv.cdf(t_val))*2
print('p-value : ',p_value)

interval_length = rv.ppf(0.975) * (std_beta * (n/df)**(1/2))

print(f'confidence interval : [{mean-interval_length},{mean+interval_length}]')
```

```
t값 :  2.4856357266673696
p-value :  0.01917203697222236
confidence interval : [0.27619020083295154,2.8640812923796926]
```

Prediction interval

- $(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n) \rightarrow$ estimate α, β / $Y_{n+1} = \alpha + \beta(x_{n+1} - \bar{x}) + \varepsilon_{n+1}$
- Let $W = Y_{n+1} - \hat{\alpha} - \hat{\beta}(x_{n+1} - \bar{x}) \rightarrow E[W] = E[Y_{n+1}] - \alpha - \beta(x_{n+1} - \bar{x}) = 0$
- $Var(W) = \sigma^2 + \frac{\sigma^2}{n} + \frac{\sigma^2}{\sum_{i=1}^{n}(x_i - \bar{x})^2}(x_{n+1} - \bar{x})^2 = \sigma^2[1 + \frac{1}{n} + \frac{(x_{n+1} - \bar{x})^2}{\sum_{i=1}^{n}(x_i - \bar{x})^2}]$
  ($Y_{n+1}, \hat{\alpha}, \hat{\beta}$ are independent)

- $T = \frac{Y_{n+1} - \hat{\alpha} - \hat{\beta}(x_{n+1} - \bar{x}) \Big/ \sigma \sqrt{[1 + \frac{1}{n} + \frac{(x_{n+1} - \bar{x})^2}{\sum_{i=1}^{n}(x_i - \bar{x})^2}]}}{\sqrt{\frac{n\widehat{\sigma^2}}{(n-2)\sigma^2}}}$ with n-2 degrees of freedom,

$(d = \sqrt{\frac{n\widehat{\sigma^2}}{(n-2)}} \sqrt{[1 + \frac{1}{n} + \frac{(x_{n+1} - \bar{x})^2}{\sum_{i=1}^{n}(x_i - \bar{x})^2}]})$

- $P\left[\hat{\alpha} + \hat{\beta}(x_{n+1} - \bar{x}) - dt_{\gamma/2} \leq Y_{n+1} \leq \hat{\alpha} + \hat{\beta}(x_{n+1} - \bar{x}) + dt_{\gamma/2}\right] = 1 - \gamma$

: prediction interval for Y_{n+1}

Polynomial regression problem

Boston 집값 예측 문제
# of observations : 506
# of variables : 13

INDUS 변수
$\mathrm{E}(\hat{\beta})$: 0.0206, $\mathrm{S.E}(\hat{\beta})$: 0.061

```
from scipy.stats import t
n = 506
p = 13
df = n-p-1
mean = 0.0206
std = 0.061
t_val = 0.334

rv = t(df)
interval_length = rv.ppf(0.975) * std # * (n/df)**(1/2)

print('p-value :',(1-rv.cdf(t_val))*2)
print(f'confidence interval :[{mean-interval_length},{mean+interval_length}]')
```

```
p-value : 0.7385218727795606
confidence interval :[-0.09925263869488882,0.1404526386948888]
```

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, 06 Jun 2023	Prob (F-statistic):	6.72e-135
Time:	08:46:57	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
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

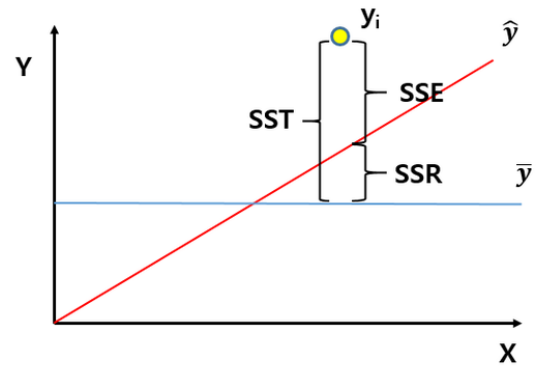
Polynomial regression problem

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, 06 Jun 2023	Prob (F-statistic):	6.72e-135
Time:	08:46:57	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
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



SST(Y의 전체 변동) : $\sum(y_i - \bar{y})^2$
SSR(모형에 의해 설명되는 변동) : $\sum(\hat{y}_i - \bar{y})^2$
SSE(모형에 의해 설명이 되지 않는 변동) : $\sum(y_i - \hat{y}_i)^2$

- $R^2 = \frac{SSR}{SST} = 1 - \frac{SSE}{SST}$
- $R^2_{adj} = 1 - \frac{SSE/(n-p-1)}{SST/(n-1)}$
- F-statistics : 모델 유의성 검정

Polynomial regression problem

- F-statistics : 모델 유의성 검정
  $H_0: \beta_1 = \cdots = \beta_p = 0 \ (\Leftrightarrow Y = \alpha + \varepsilon)$ vs $H_1: not\ H_0$
  귀무가설 : 독립변수들이 종속변수를 설명하는데 효과가 없다.
  대립가설 : 독립변수들 중 적어도 하나 이상은 종속변수를 설명하는데 효과가 있다.
- $F = \frac{SSR/p}{SSE/(n-p-1)} = \frac{\sum_{i=1}^{n}(\hat{y}_i-\bar{y})^2/\sigma^2 p}{\sum_{i=1}^{n}(y_i-\hat{y}_i)^2/\sigma^2(n-p-1)}$
  with degree of freedom (p, n-p-1)
  $\sum_{i=1}^{n}\frac{(y_i-\hat{y}_i)^2}{\sigma^2} \sim \chi^2(n-p-1)$ (앞에서 이미 보임)
  $\frac{SST}{\sigma^2} = \frac{\sum_{i=1}^{n}(y_i-\bar{y})^2}{\sigma^2}\ is\ the\ form\ of\ \frac{(n-1)S^2}{\sigma^2} \sim \chi^2(n-1)$
  (S는 표본표준편차)
  $\rightarrow \frac{SSR}{\sigma^2} \sim \chi^2(p)$

```
from scipy.stats import f
n,p = df.drop('MEDV',axis=1).shape

y_pred = result.predict(df.iloc[:,:-1]).values
y_real = df['MEDV'].values
y_mean = df['MEDV'].mean()

upper = np.sum([(y_pred[i]-y_mean)**2 for i in range(n)])
lower = np.sum([(y_real[i]-y_pred[i])**2 for i in range(n)])

F = (upper/p)/(lower/(n-p-1))
print('F값 : ',F)
```

```
F값 :  108.07666617432566
```