# 파이썬과 R에서 데이터로부터 수식을 찾아내는 기능

## 파이썬(Python)

### 1. Scipy의 curve\_fit 함수

- Scipy 라이브러리의 `curve\_fit` 함수를 사용하면 비선형 회귀를 통해 데이터를 모델링할 수 있음.  
- 원하는 함수 형태를 정의하고, 데이터에 가장 적합한 파라미터를 찾아낼 수 있음.

#### 사용 예시:

```python  
import numpy as np  
from scipy.optimize import curve\_fit  
import matplotlib.pyplot as plt  
  
# 모델 정의  
def model\_func(x, a, b):  
 return a \* np.sin(b \* x)  
  
# 데이터 생성  
x\_data = np.linspace(0, 2 \* np.pi, 50)  
y\_data = 2.5 \* np.sin(1.5 \* x\_data) + np.random.normal(size=x\_data.size)  
  
# 모델 피팅  
params, params\_covariance = curve\_fit(model\_func, x\_data, y\_data, p0=[2, 2])  
  
# 결과 출력  
print(params)  
  
# 피팅 결과 시각화  
plt.scatter(x\_data, y\_data, label='Data')  
plt.plot(x\_data, model\_func(x\_data, \*params), label='Fitted function', color='red')  
plt.legend()  
plt.show()  
```

### 2. Numpy의 polyfit 및 polyval

- Numpy의 `polyfit` 함수를 사용하여 다항식 피팅을 할 수 있음.  
- `polyval`을 통해 피팅된 다항식으로 새로운 데이터를 평가할 수 있음.

#### 사용 예시:

```python  
import numpy as np  
import matplotlib.pyplot as plt  
  
# 데이터 생성  
x = np.linspace(0, 10, 100)  
y = 3 \* x\*\*2 + 2 \* x + 1 + np.random.normal(0, 10, size=x.size)  
  
# 다항식 피팅  
coeffs = np.polyfit(x, y, 2)  
print(coeffs)  
  
# 피팅된 다항식의 결과 시각화  
p = np.poly1d(coeffs)  
plt.scatter(x, y, label='Data')  
plt.plot(x, p(x), label='Fitted Polynomial', color='red')  
plt.legend()  
plt.show()  
```

### 3. Statsmodels와 LinearRegression

- `statsmodels` 라이브러리에서 선형 회귀 모델을 만들고 데이터를 피팅할 수 있음.

#### 사용 예시:

```python  
import numpy as np  
import statsmodels.api as sm  
  
# 데이터 생성  
x = np.linspace(0, 10, 100)  
y = 3 \* x + 2 + np.random.normal(0, 1, size=x.size)  
  
# 선형 모델 피팅  
x = sm.add\_constant(x) # 절편을 추가  
model = sm.OLS(y, x)  
results = model.fit()  
  
# 결과 출력  
print(results.summary())  
```

## R

### 1. nls 함수 (비선형 회귀)

- R의 `nls` 함수는 비선형 최소제곱법을 사용하여 데이터를 모델링함.  
- 수식과 데이터를 제공하면 최적의 파라미터를 찾아줌.

#### 사용 예시:

```r  
# 데이터 생성  
x <- seq(0, 2\*pi, length.out = 50)  
y <- 2.5 \* sin(1.5 \* x) + rnorm(length(x))  
  
# 비선형 모델 피팅  
model <- nls(y ~ a \* sin(b \* x), start = list(a = 2, b = 2))  
  
# 결과 출력  
summary(model)  
```

### 2. lm 함수 (선형 회귀)

- R의 기본 함수인 `lm`을 사용하여 선형 회귀 모델을 피팅할 수 있음.

#### 사용 예시:

```r  
# 데이터 생성  
x <- 1:100  
y <- 3 \* x + 2 + rnorm(100)  
  
# 선형 모델 피팅  
model <- lm(y ~ x)  
  
# 결과 출력  
summary(model)  
```

### 3. poly 함수 (다항식 피팅)

- `lm` 함수와 `poly` 함수를 함께 사용하여 다항식 피팅을 할 수 있음.

#### 사용 예시:

```r  
# 데이터 생성  
x <- 1:100  
y <- 3 \* x^2 + 2 \* x + 1 + rnorm(100, sd = 50)  
  
# 2차 다항식 모델 피팅  
model <- lm(y ~ poly(x, 2))  
  
# 결과 출력  
summary(model)  
```

## 요약

- 파이썬: Scipy의 `curve\_fit`, Numpy의 `polyfit`, `statsmodels`의 `OLS`를 사용하여 데이터를 모델링하고 수식을 찾을 수 있음.  
- R: `nls`를 사용한 비선형 회귀, `lm`을 사용한 선형 회귀, `poly`를 사용한 다항식 피팅을 통해 수식을 찾아낼 수 있음.  
- 이 두 언어 모두 다양한 통계 모델과 머신러닝 방법을 지원하여 데이터로부터 복잡한 수식을 찾는 데 유용함.