

# 회귀 분석 - 실습

---

Kyungsik Han

# 본 영상에서 다룰 내용

- 회귀 알고리즘 실습을 통한 회귀에 대한 이해
- sklearn에서 제공하는 대표적인 데이터 셋인 당뇨병 데이터를 활용

# 당뇨 진행 상황 예측 예제

# 사용되는 Libraries

```
In [28]: import sklearn.datasets
import pandas as pd
from matplotlib import pyplot as plt
from sklearn.cross_validation import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.preprocessing import normalize
from sklearn.metrics import r2_score
```

# 데이터 처리

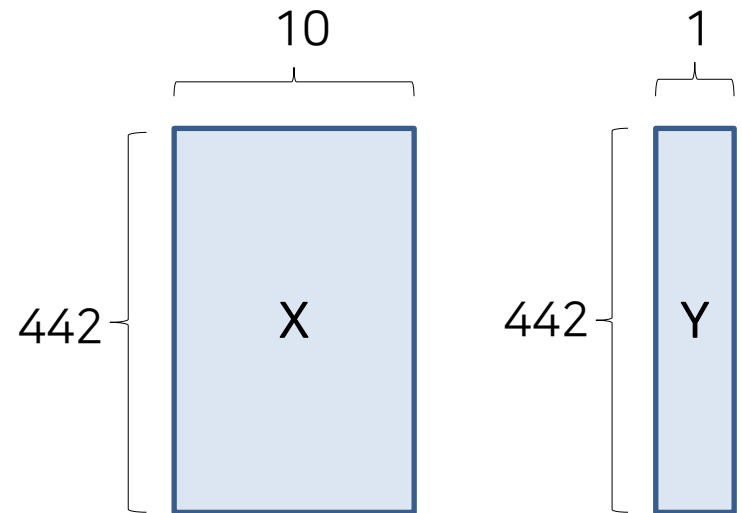
sklearn에서 기본 제공하고 있는 데이터를 활용

```
In [4]: diabetes = sklearn.datasets.load_diabetes()  
X, Y = normalize(diabetes['data']), diabetes['target']
```

normalize()를 통해서 데이터 정규화 진행

데이터 형태가 어떤지 확인

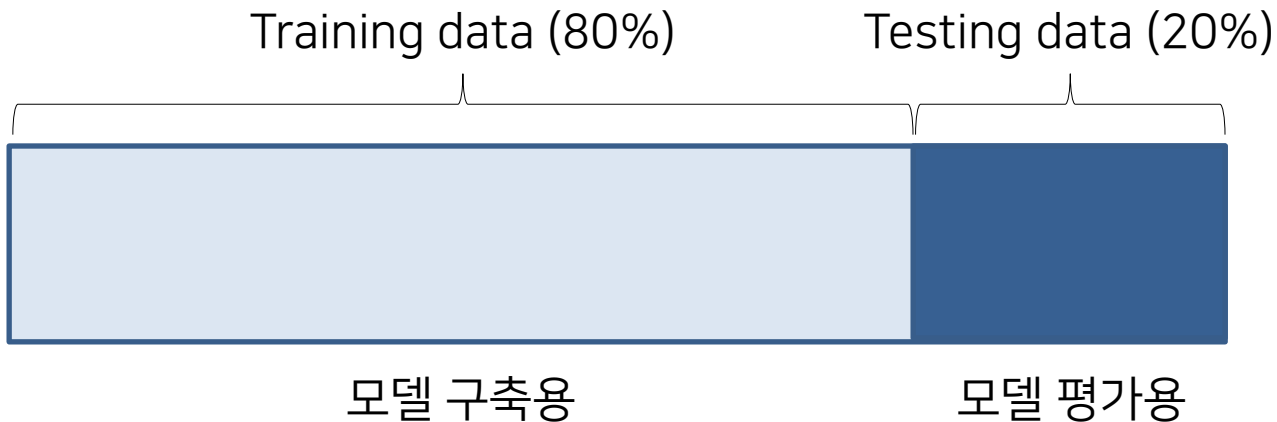
```
In [29]: print(X.shape, Y.shape)  
(442, 10) (442,)
```



# Training/Test 데이터

Training과 Test Dataset으로 데이터 분할

```
In [22]: # train 과 test dataset으로 데이터 분할  
X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size=.2)
```



# 선형 회귀 모델링

X\_train, Y\_train 값을 기준으로 모델을 fit 함

```
In [23]: # linear regression 실행  
linear = LinearRegression()  
# train data로 학습 진행  
linear.fit(X_train, Y_train)
```

```
Out[23]: LinearRegression(copy_X=True, fit_intercept=True, n_jobs=1, normalize=False)
```

linear = linear.fit() 와 같이 안하고, 바로 linear.fit()으로 진행

# 모델 성능 평가

```
In [24]: # test data로 결과 확인  
preds_linear = linear.predict(X_test)
```

X\_test 값을 기반으로 모델을 통한 예측된 Y 결과 (pred\_linear)를 얻어냄

```
In [58]: # correlation  
corr_linear = pd.Series(preds_linear).corr(pd.Series(Y_test))  
# r2_score 을 통해서 모델의 성능 확인  
rsquared_linear = r2_score(Y_test, preds_linear)
```

pred\_linear의 값과 실제 Y\_test 값과 비교하여 실제로 어느정도 차이 나는지 확인함

- correlation을 통한 상관관계
- r2\_score() 함수를 통한 성능 비교



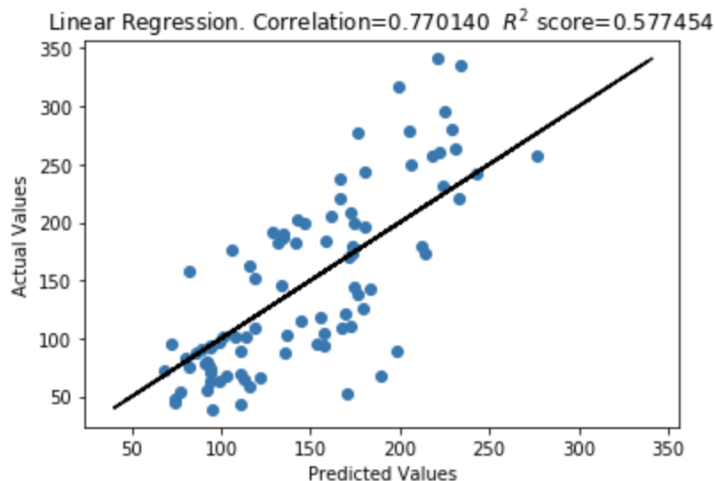
# 결과 확인

```
In [26]: print("선형 계수:")  
print(linear.coef_)
```

선형 계수 확인을 통해서 10개의 features의 모델에 대한 영향력 확인 가능

```
In [13]: # 결과를 그림으로 표현  
plt.scatter(preds_linear, Y_test)  
plt.title("Linear Regression. Correlation=%f  $R^2$ score=%f" % (corr_linear, rsquared_linear))  
plt.xlabel("Predicted Values")  
plt.ylabel("Actual Values")  
# 비교를 위해 x=y 라인 추가  
plt.plot(Y_test, Y_test, 'k--')  
plt.show()
```

결과를 그림으로 표현



practice\_step\_by\_step 실습

# 다음 영상에서 배울 내용

- 지도학습 분류(classification) 개념 및 실습

수고하셨습니다