



#4 - Linear Regression Practice


주제 : Linear Regression Practice

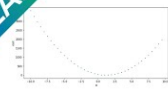

링크 :

[#4.Lab] Linear Regression Practice - 딥러닝 홀로서기

실습자료 링크 : https://github.com/heartcored98/Standalone-DeepLearning/blob/master/Lec1/Lab1_linear_regression.ipynb

b자료 저장소 링크 : <https://github.com/heartcored98/St...>

 <https://www.youtube.com/watch?v=nQqMT0wbthw&feature=youtu.be>



**From scratch
Gradient
Descent**

Gradient Descent Algorithm

이 레슨에서는 Gradient Descent 알고리즘을 사용하여 선형 회귀 모델을 훈련하는 방법을 알아보겠습니다. 이 알고리즘은 손실 함수를 최소화하는 방향으로 파라미터를 업데이트하는 반복적인 과정입니다.

Complete this approach in order to complete Gradient Descent (Analytic, Formula)

```
def gradient_descent(X, y, initial_weights, learning_rate, iterations):  
    # Initialize weights and bias  
    w = initial_weights[0]  
    b = initial_weights[1]  
    # Loop for iterations  
    for i in range(iterations):  
        # Calculate the current loss  
        loss = calculate_loss(X, y, w, b)  
        # Calculate the gradient of the loss with respect to w and b  
        dw, db = calculate_gradient(X, y, w, b)  
        # Update weights and bias using the gradient and learning rate  
        w = w - learning_rate * dw  
        b = b - learning_rate * db  
    return w, b
```



**From scratch
Gradient
Descent**

Gradient Descent Algorithm

이 레슨에서는 Gradient Descent 알고리즘을 사용하여 선형 회귀 모델을 훈련하는 방법을 알아보겠습니다. 이 알고리즘은 손실 함수를 최소화하는 방향으로 파라미터를 업데이트하는 반복적인 과정입니다.

Complete this approach in order to complete Gradient Descent (Analytic, Formula)

```
def gradient_descent(X, y, initial_weights, learning_rate, iterations):  
    # Initialize weights and bias  
    w = initial_weights[0]  
    b = initial_weights[1]  
    # Loop for iterations  
    for i in range(iterations):  
        # Calculate the current loss  
        loss = calculate_loss(X, y, w, b)  
        # Calculate the gradient of the loss with respect to w and b  
        dw, db = calculate_gradient(X, y, w, b)  
        # Update weights and bias using the gradient and learning rate  
        w = w - learning_rate * dw  
        b = b - learning_rate * db  
    return w, b
```

Tool : 구글 콜라보레이터 (주피터 노트북 클라우드 버전)

<https://colab.research.google.com/notebooks/intro.ipynb#recent=true>

Source Code : [Standalone-DeepLearning/Lab1_linear_regression.ipynb](https://github.com/heartcored98/Standalone-DeepLearning/blob/master/Lab1_linear_regression.ipynb) at
[master](https://github.com/heartcored98/Standalone-DeepLearning/blob/master/Lab1_linear_regression.ipynb) ·

[heartcored98/Standalone-DeepLearning](https://github.com/heartcored98/Standalone-DeepLearning/blob/master/Lab1_linear_regression.ipynb) (github.com).