

# Pseudocode

Algorithm: SimpleLinearRegression

Input:

X (independent variable)  
y (dependent variable)  
 $\alpha$  (learning rate)  
T (number of iterations)

Output:

m (slope)  
c (intercept)

Initialize  $m = 0$

Initialize  $c = 0$

for  $i = 1$  to  $T$  do

$y_{\text{hat}} = m * X + c$

$\text{error} = y_{\text{hat}} - y$

$dm = (1/n) * \text{sum}(\text{error} * X)$

$dc = (1/n) * \text{sum}(\text{error})$

$m = m - \alpha * dm$

$c = c - \alpha * dc$

end for

return  $m, c$

# Python Code

```
import numpy as np
```

```
def simple_linear_regression(X, y, alpha=0.01, iterations=1000):
```

```
    n = len(X)
```

```
    m = 0
```

```
    c = 0
```

```
    for _ in range(iterations):
```

```
         $y_{\text{hat}} = m * X + c$ 
```

```
error = y_hat - y

dm = (1 / n) * np.sum(error * X)
dc = (1 / n) * np.sum(error)

m = m - alpha * dm
c = c - alpha * dc

return m, c
```