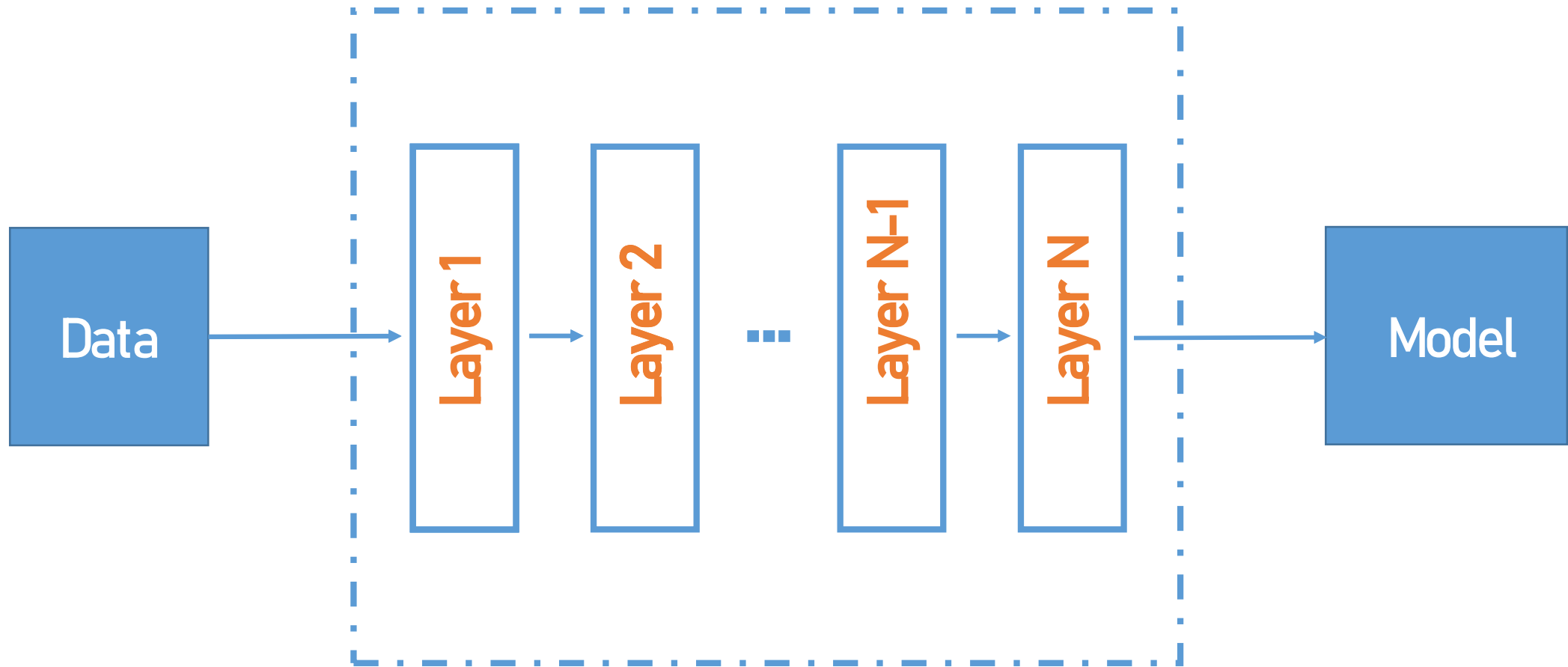


# Gradient Descent

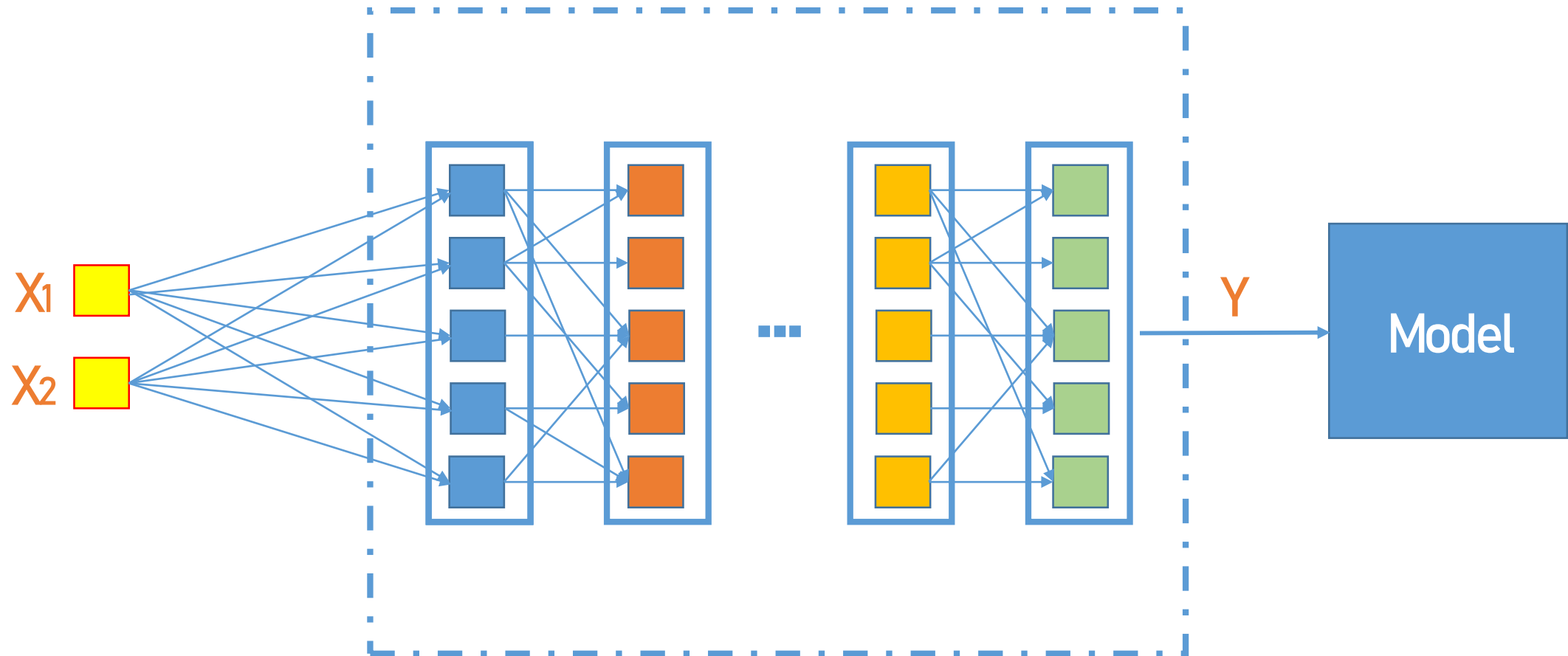
---



# Neural Networks Architecture



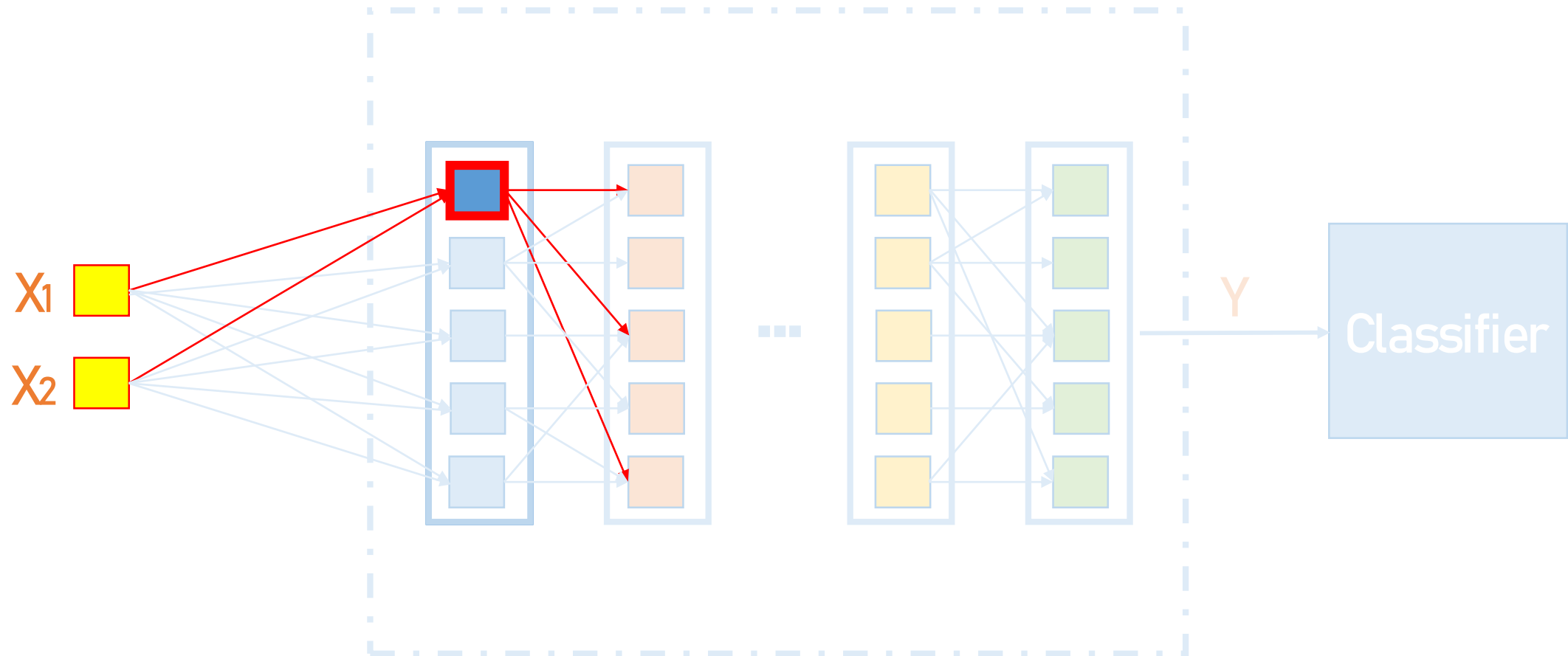
# Neural Networks Architecture



All **features** are connected to all neurons in the first layer



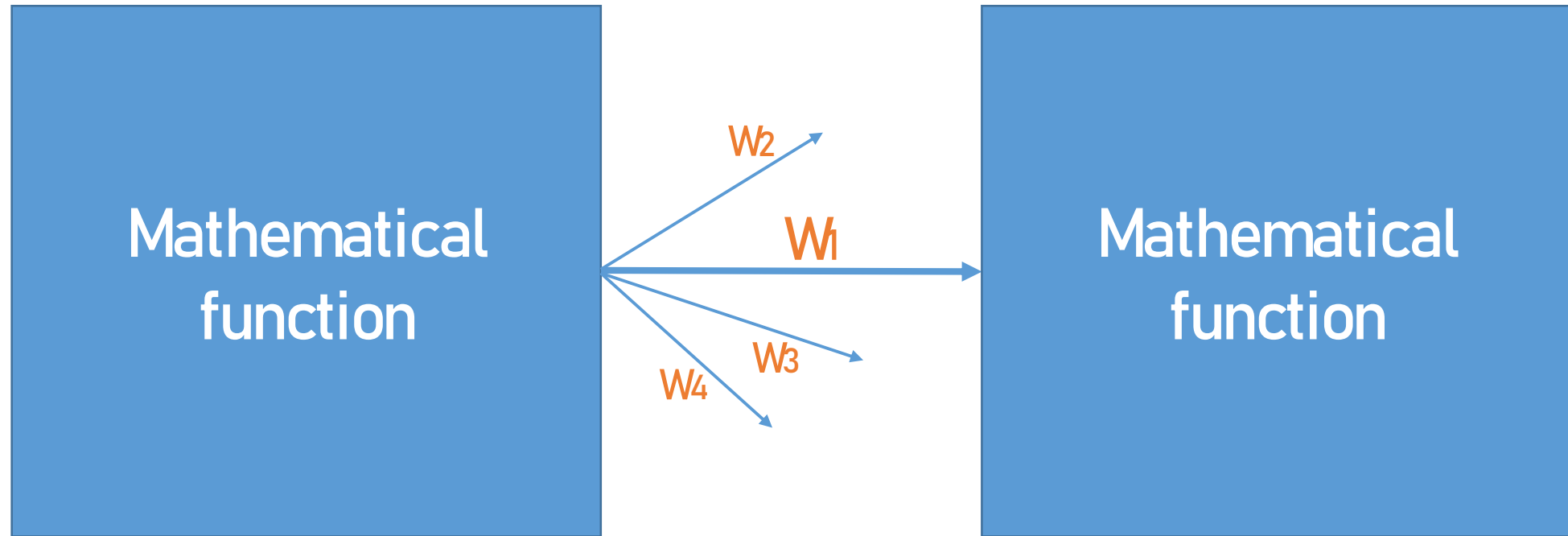
# Neural Networks Architecture



A **neuron** in the entire network



# Understanding a Single Neuron



**W** increases if the connection between two neurons is stronger

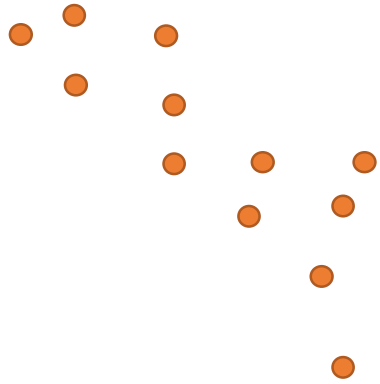


# GOAL OF THE TRAINING PROCESS

Find out the correct values for **weights** & **biases** of each neuron in the network



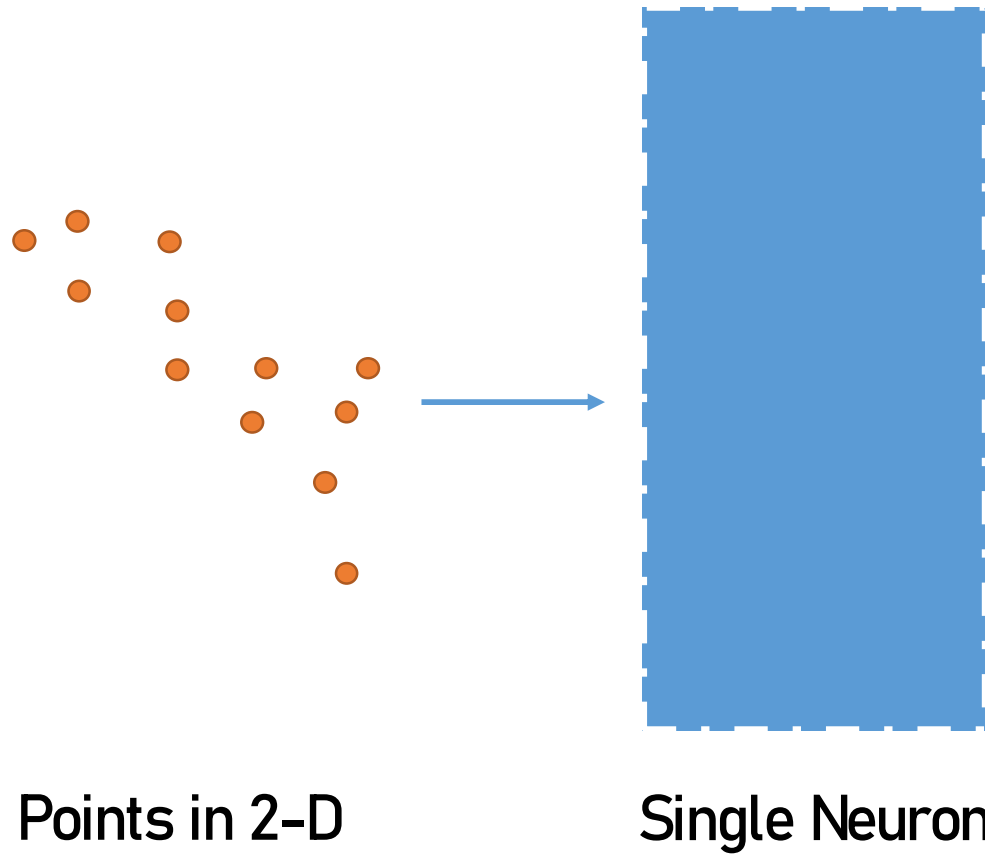
# Regression Problem



Points in 2-D

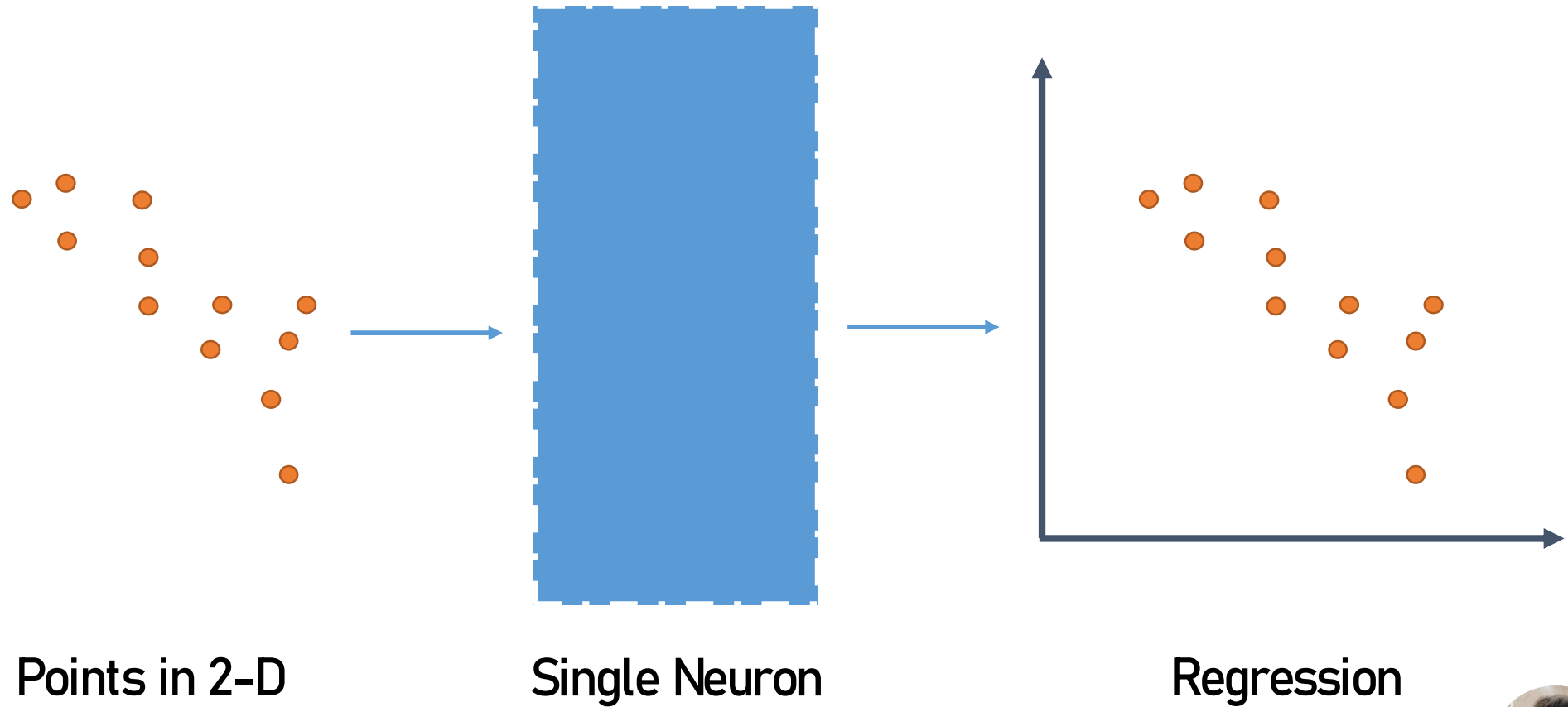


# Regression Problem

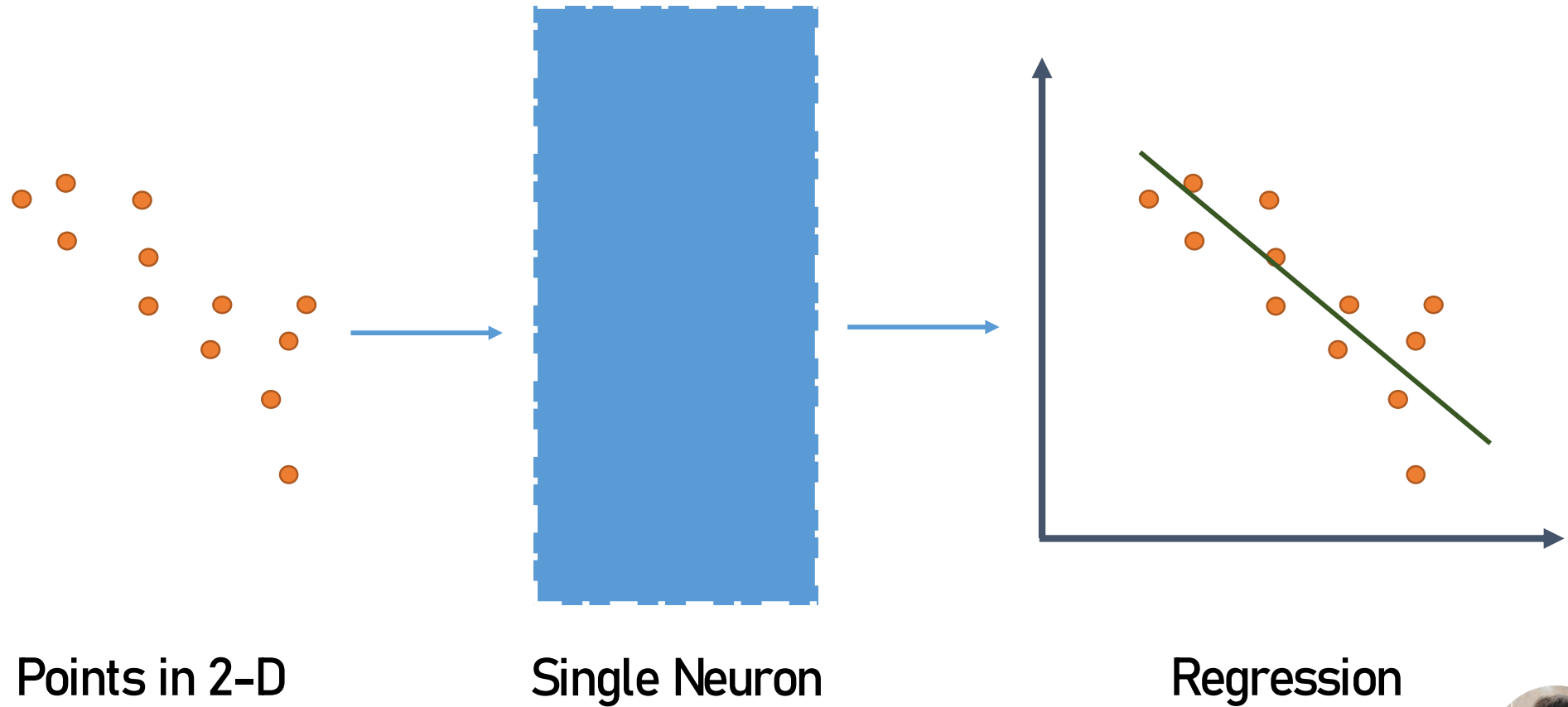




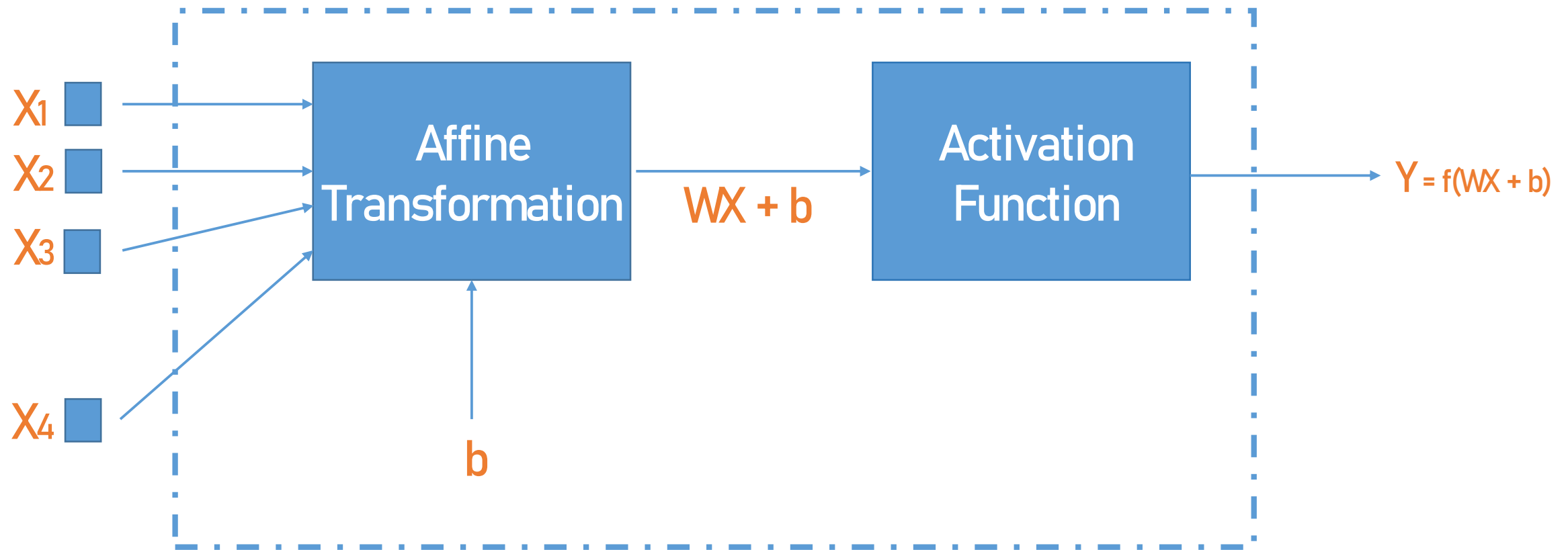
# Regression Problem



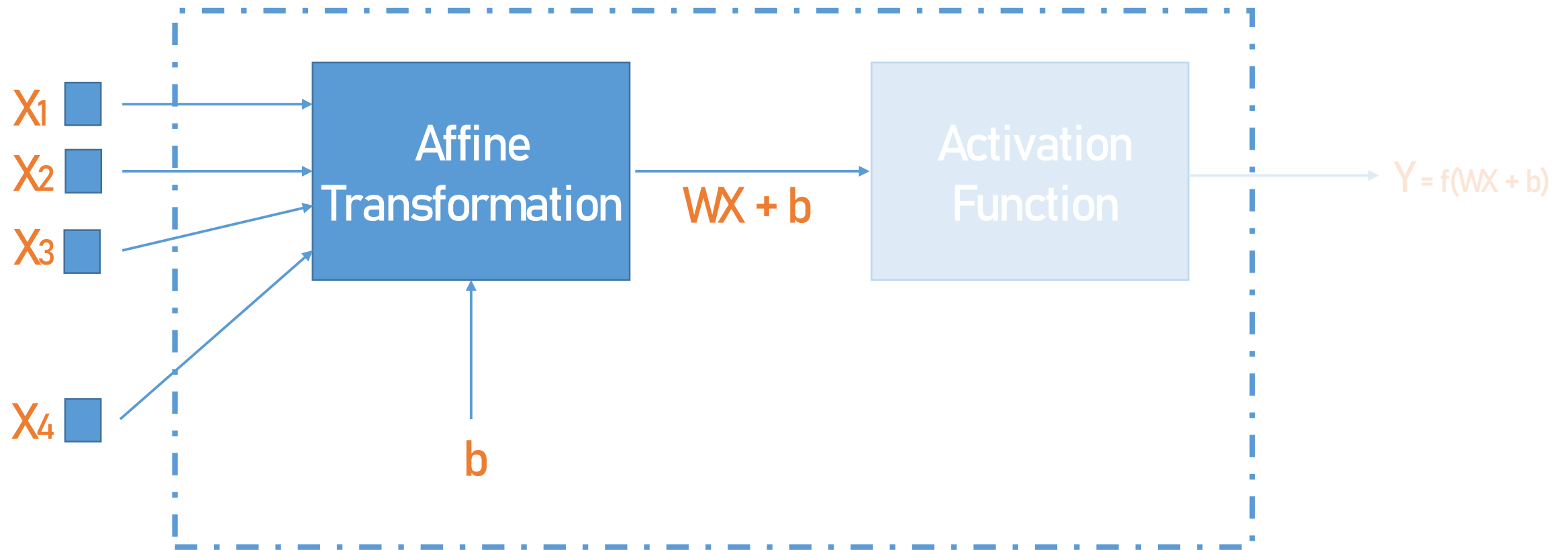
# Regression Problem



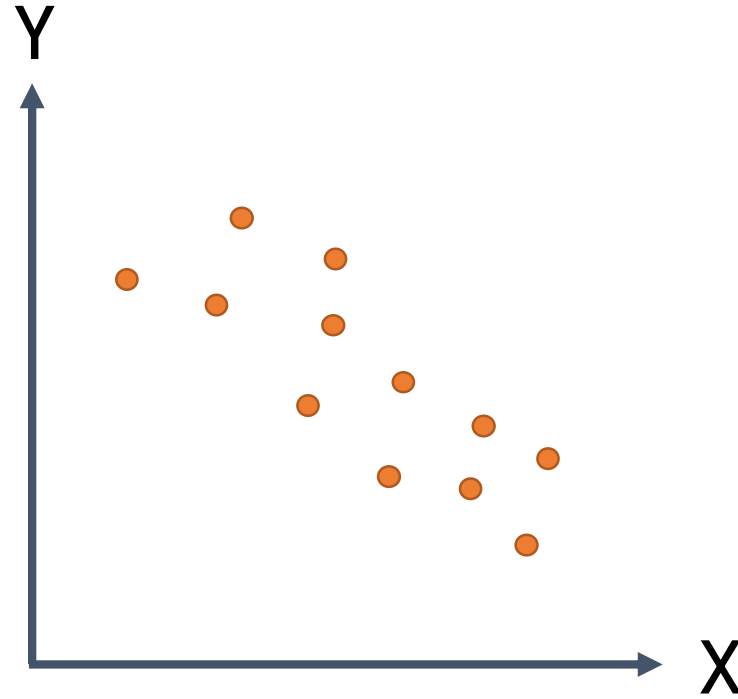
# A Single Neuron



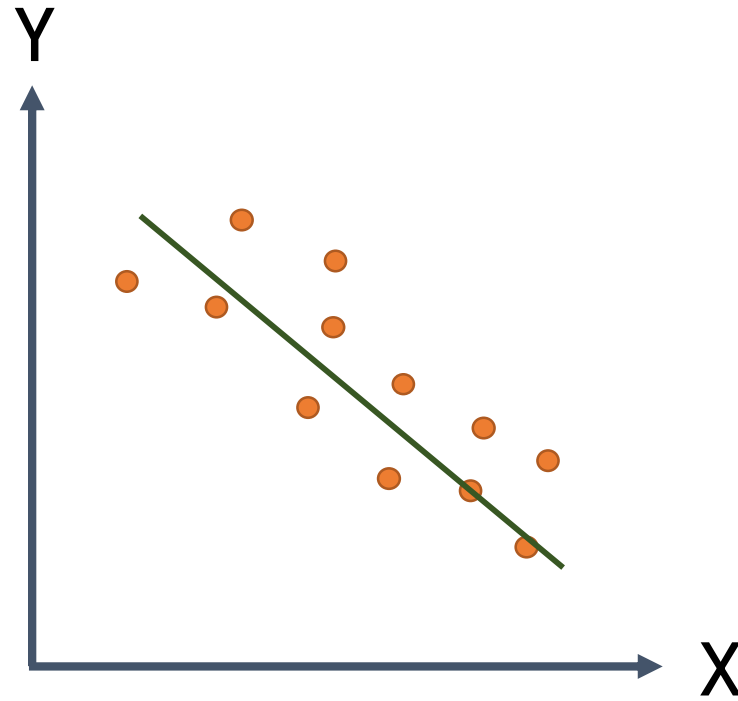
# Zoom into a neuron



# Regression Problem



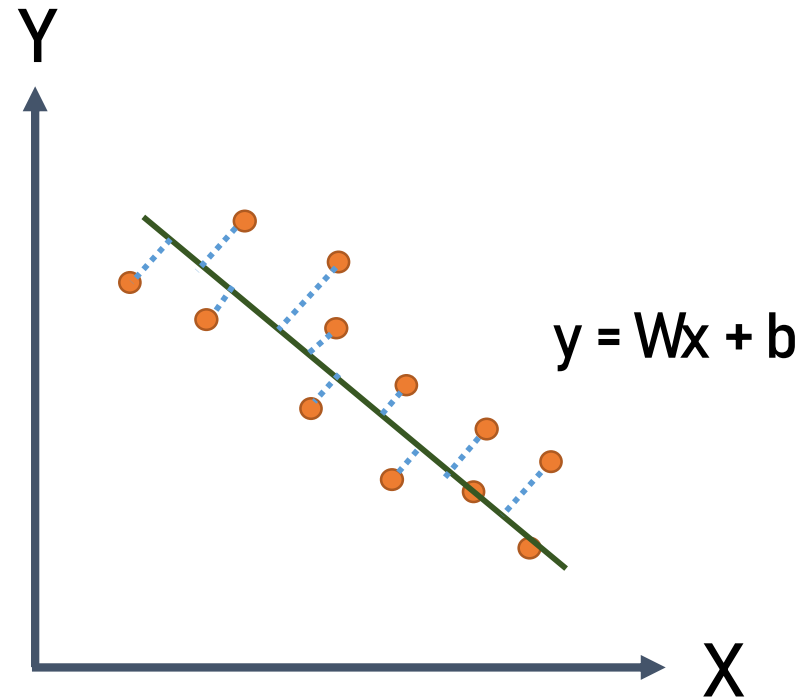
# Regression Problem



Finding the **best-fit** line for the given data



# Regression Problem



Minimize the least square error

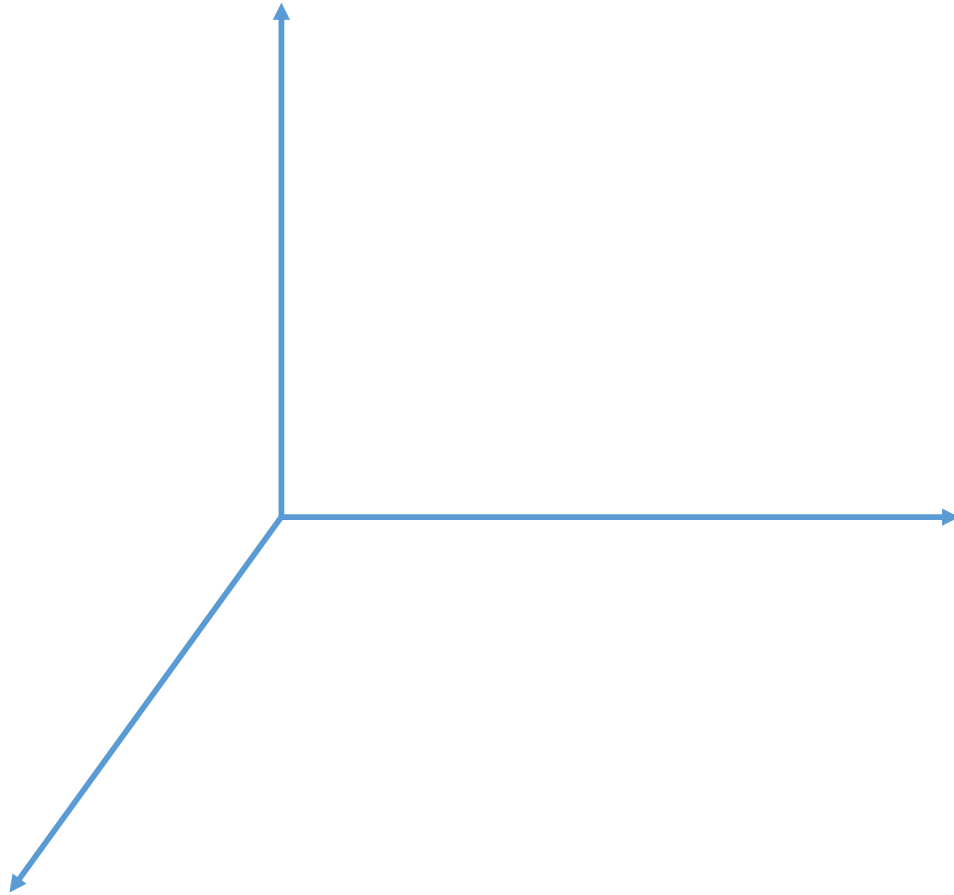


**Gradient Descent Optimization is the process in which the weights & biases are changed & network is trained**

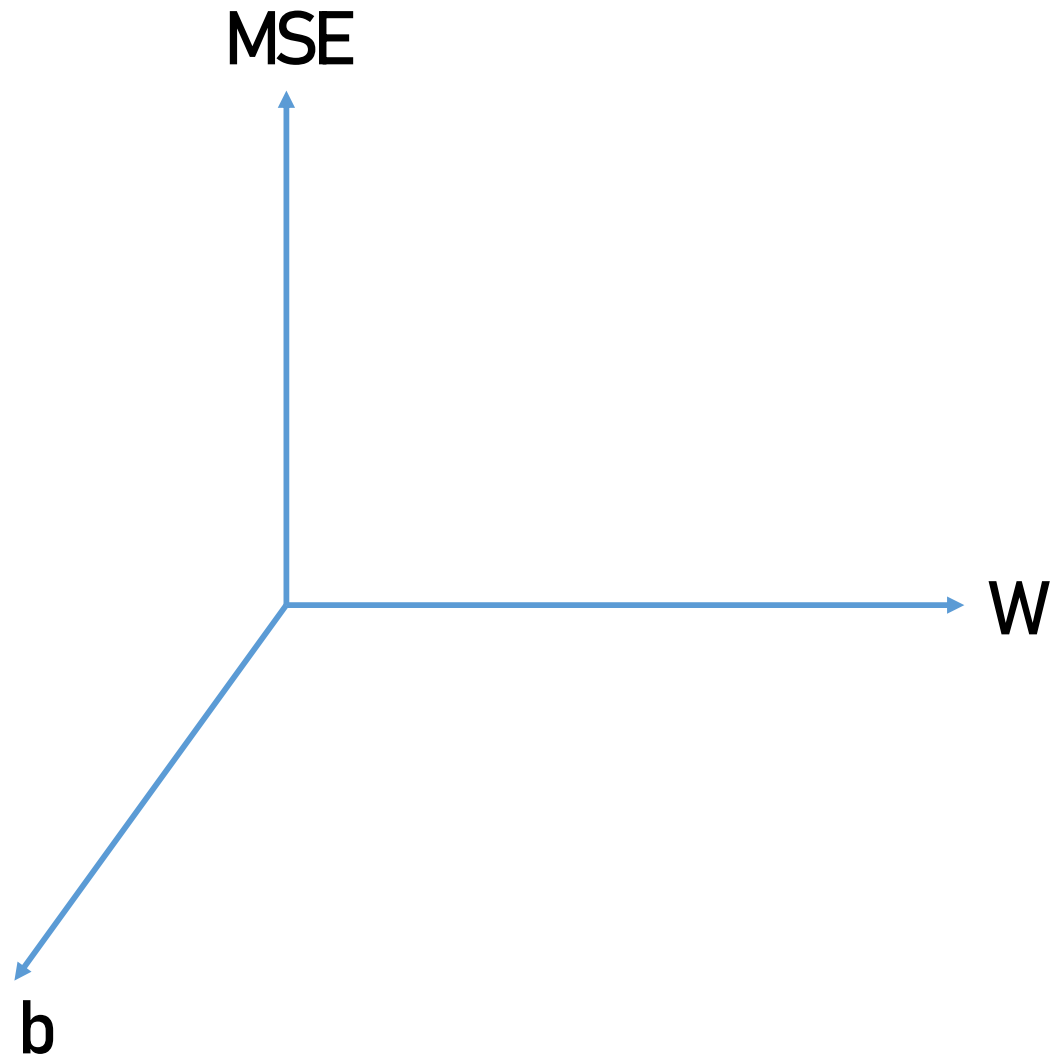




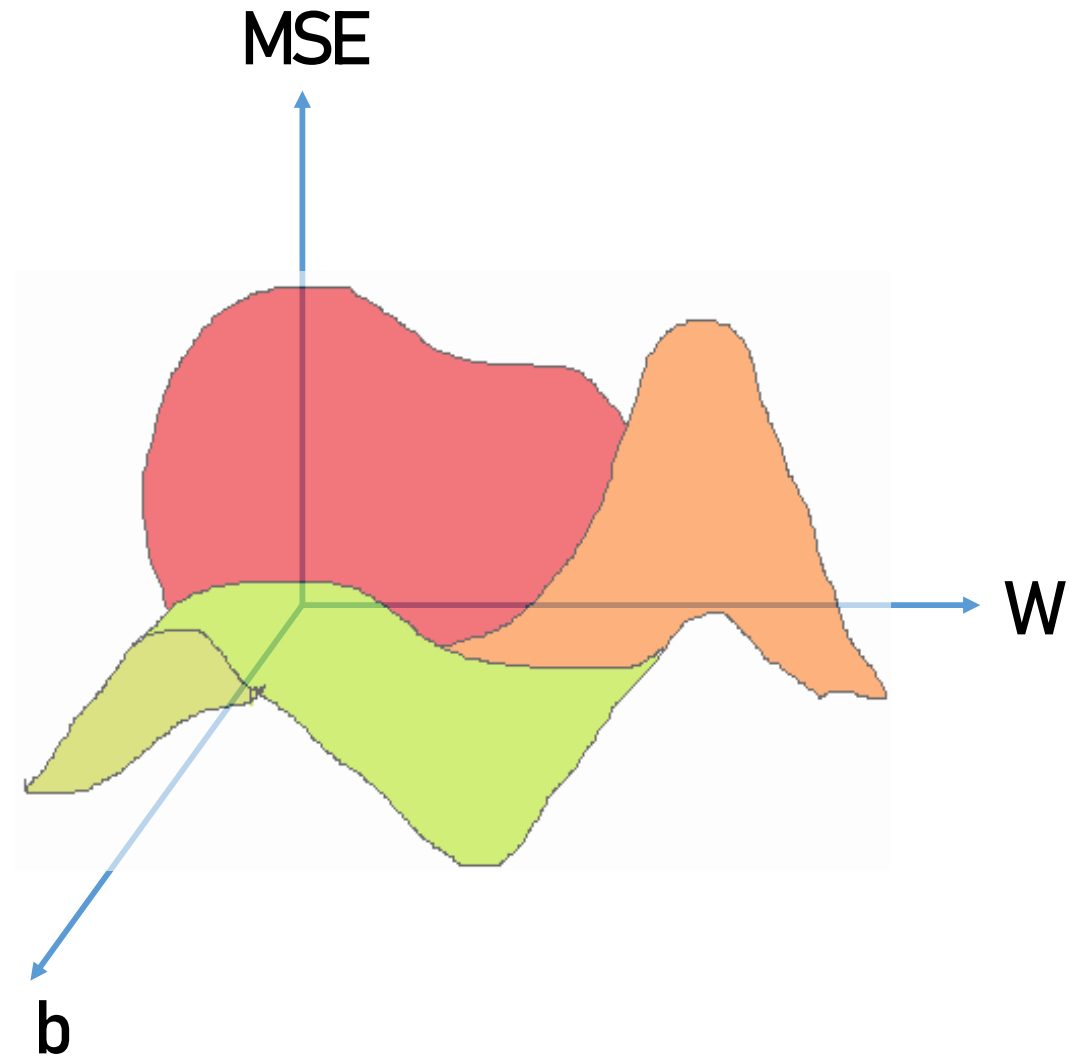
# Visualizing Gradient Descent



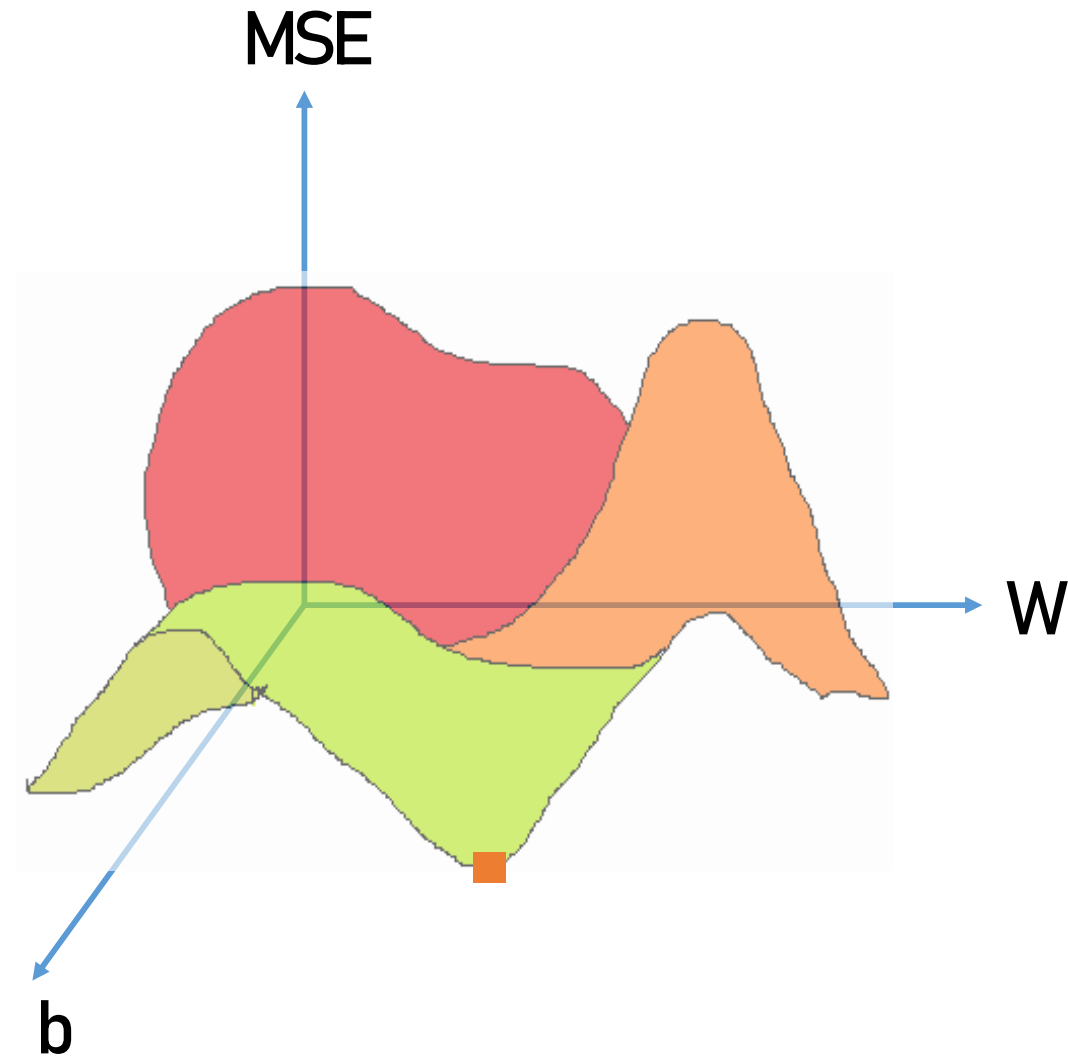
# Visualizing Gradient Descent



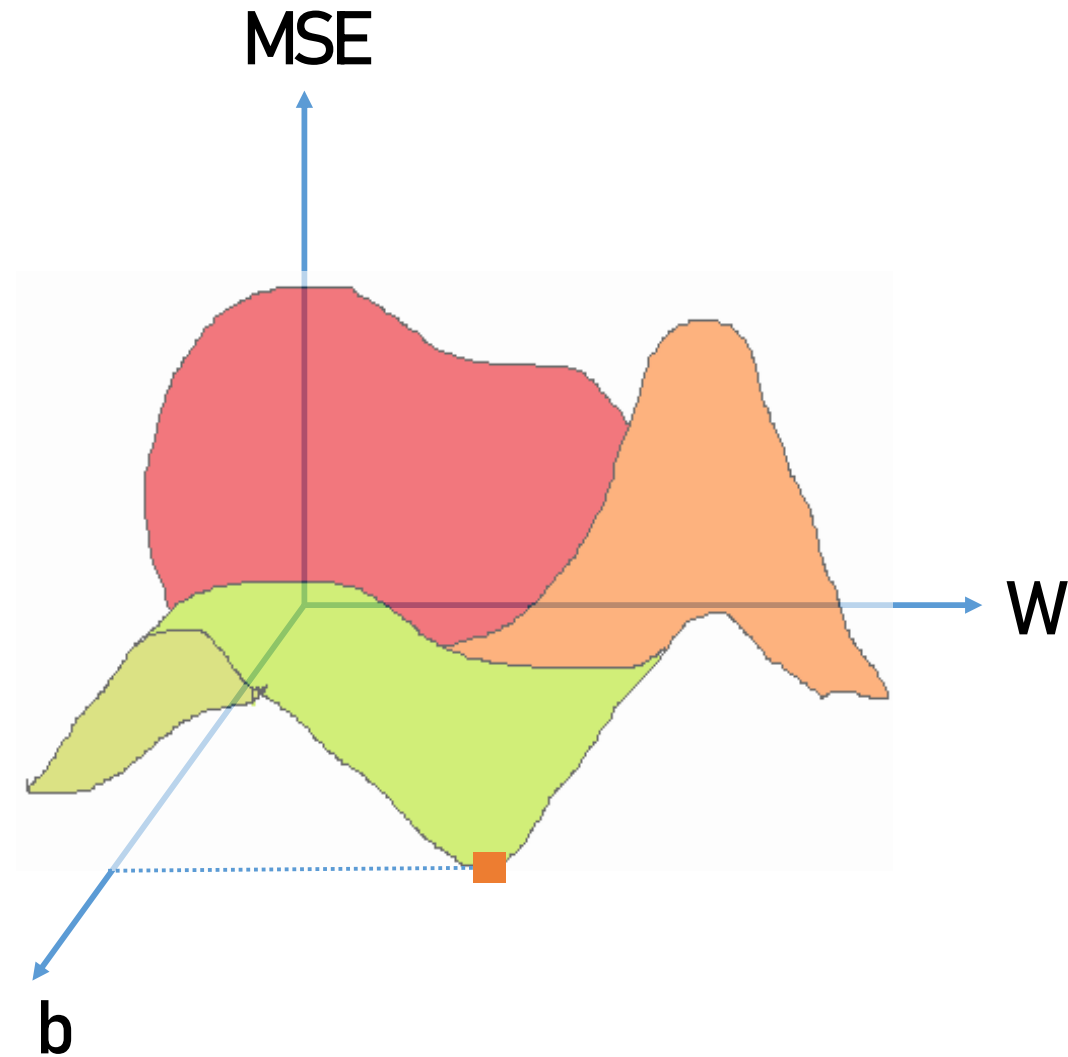
# Visualizing Gradient Descent



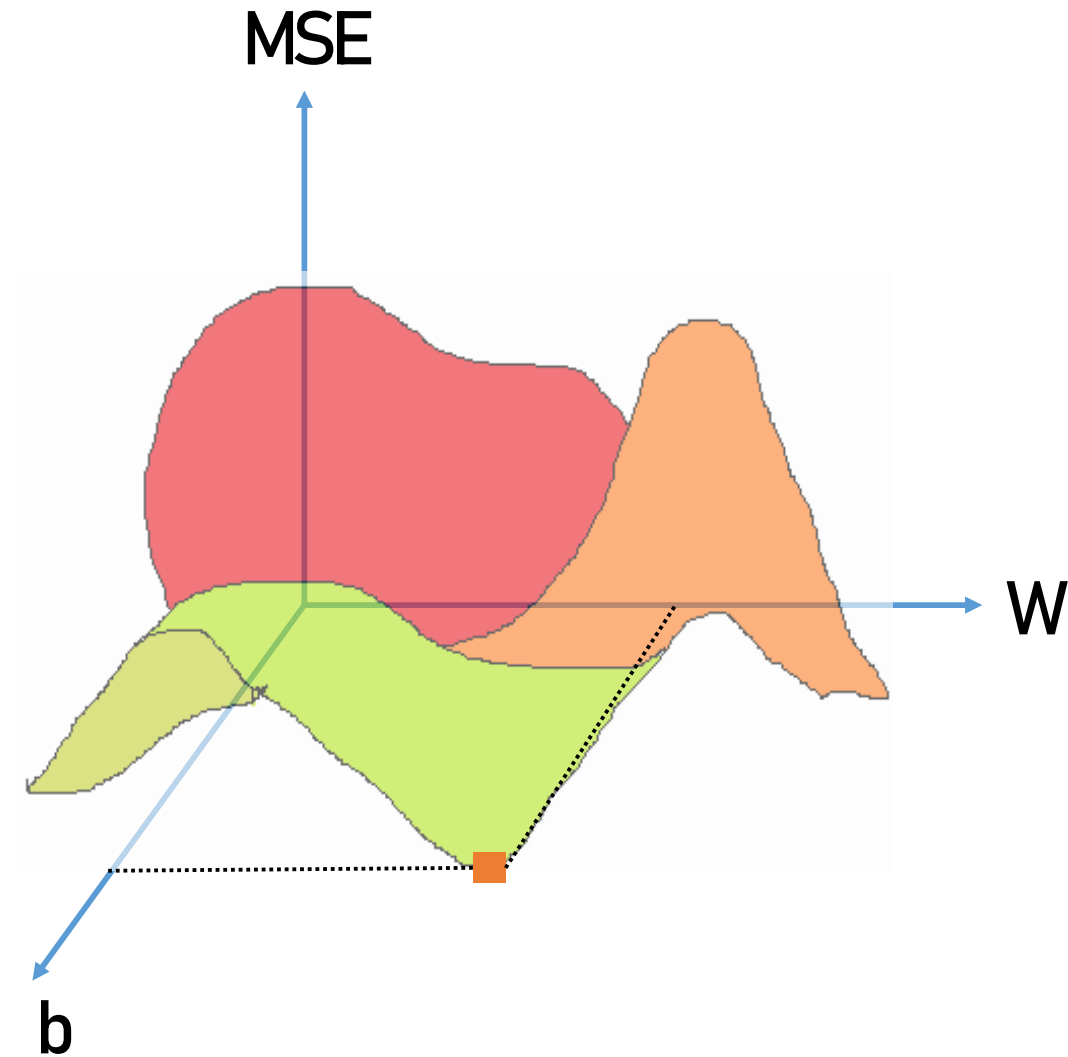
# Visualizing Gradient Descent



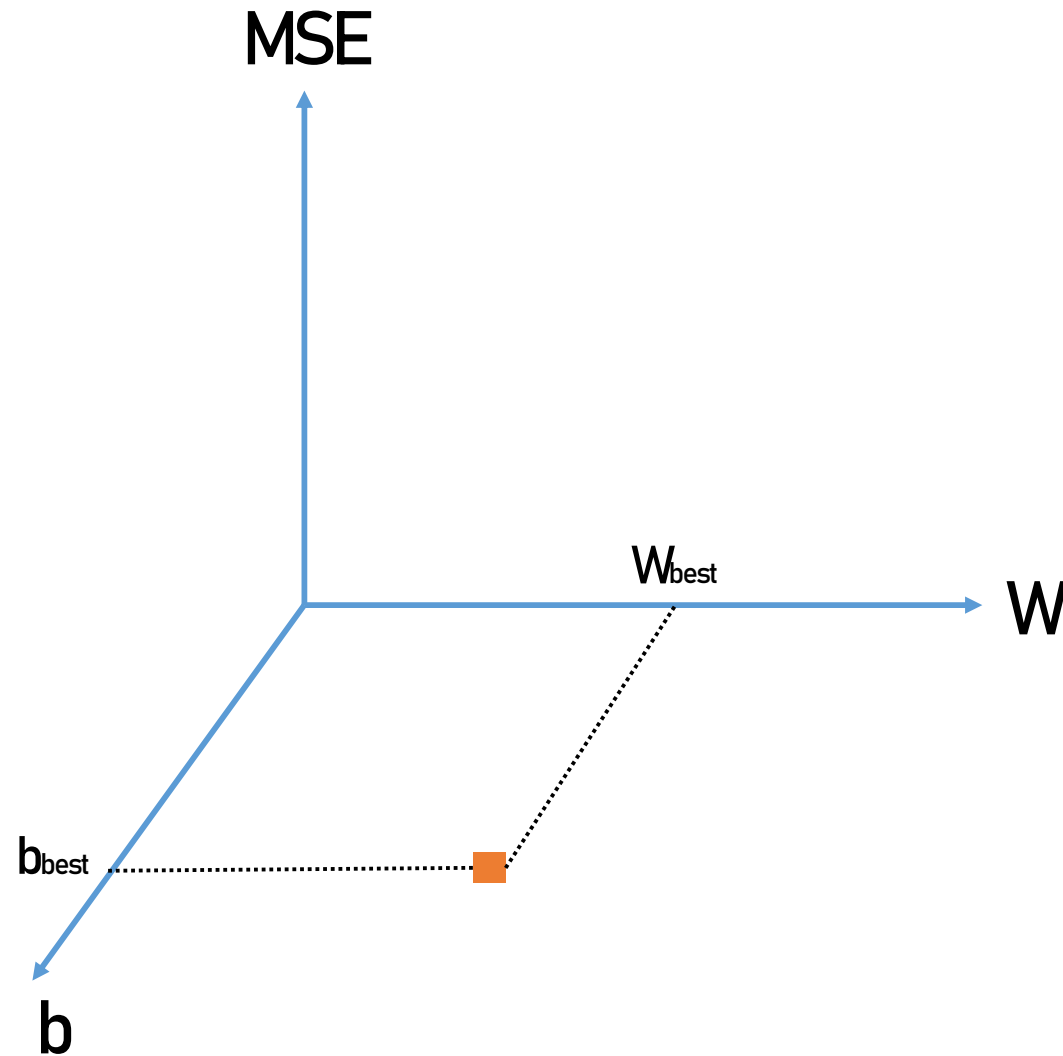
# Visualizing Gradient Descent



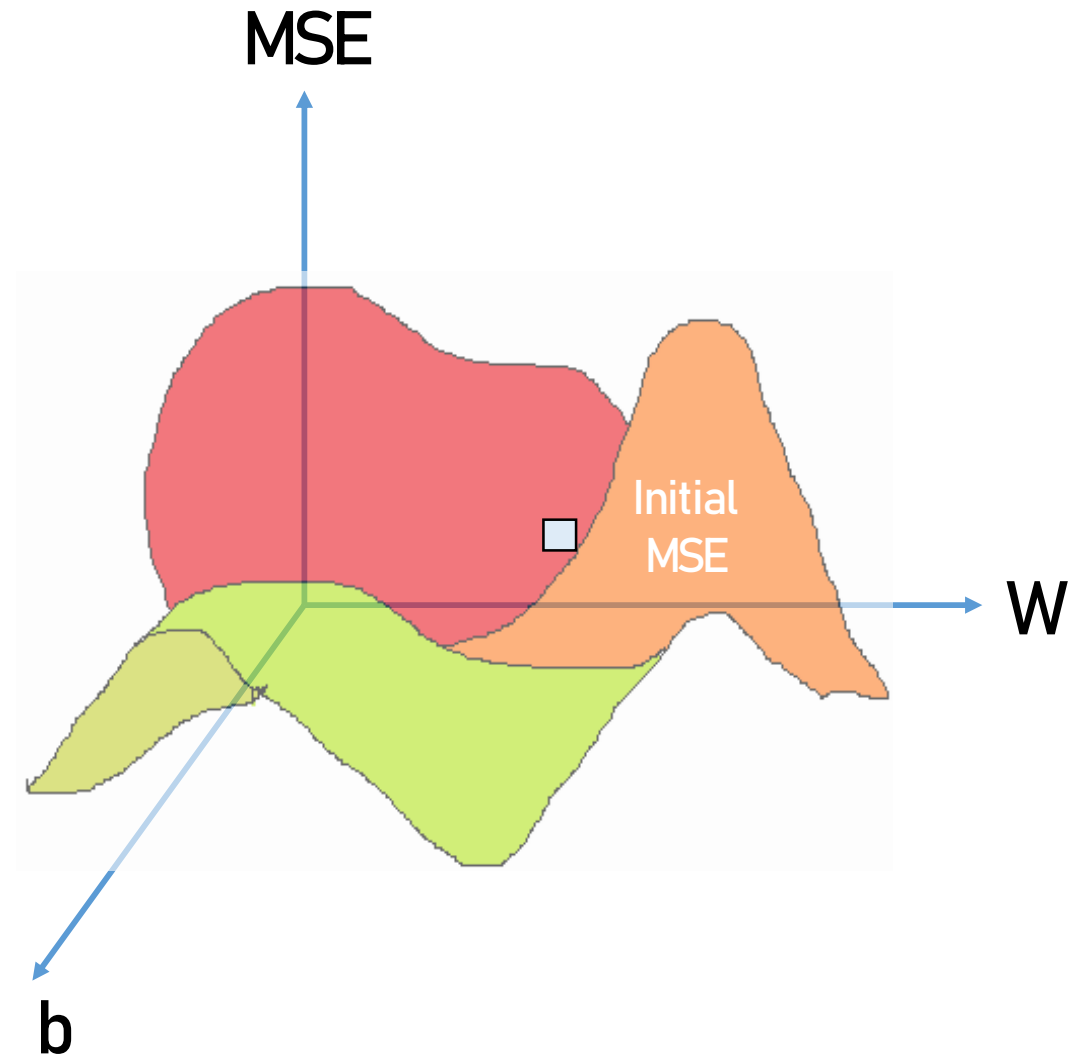
# Visualizing Gradient Descent



# Visualizing Gradient Descent

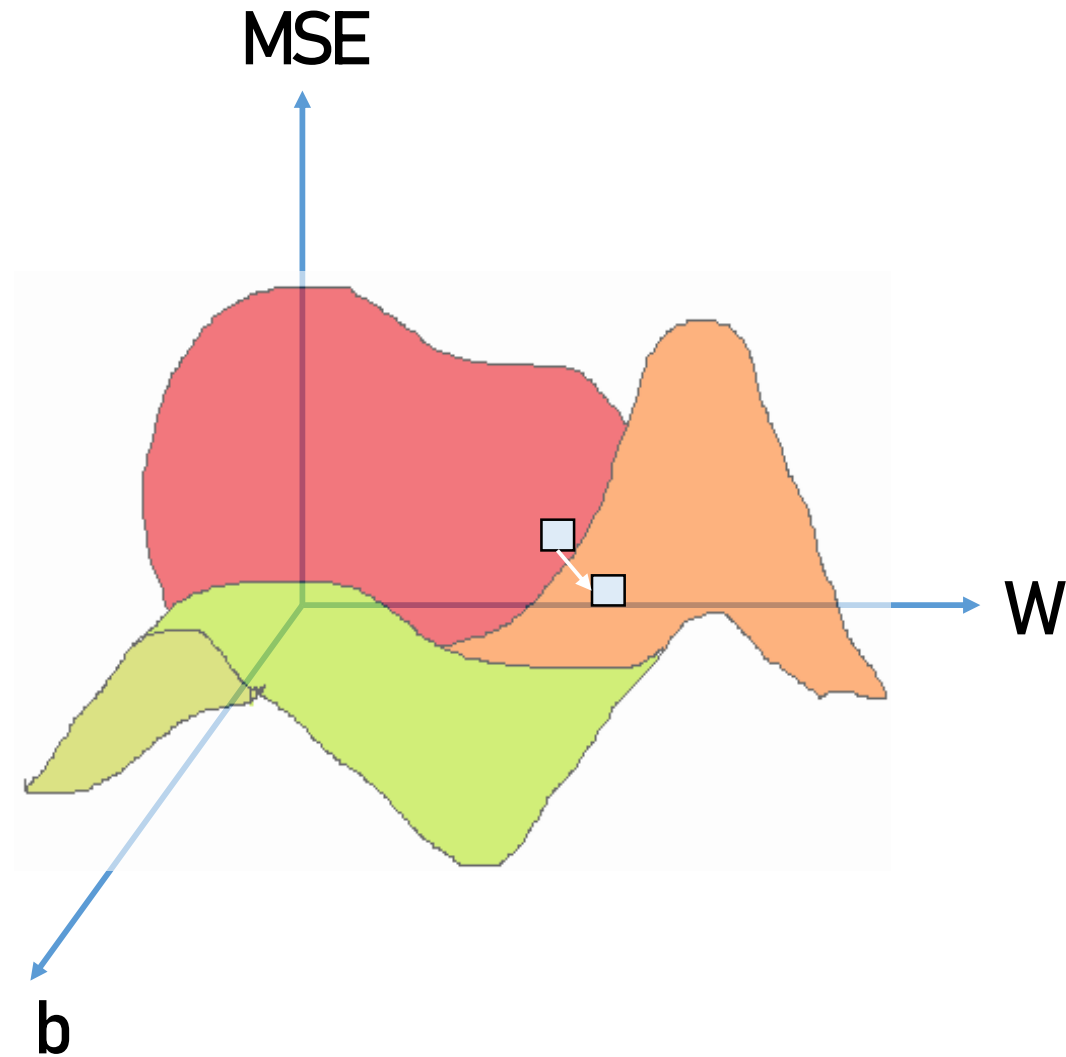


# Visualizing Gradient Descent

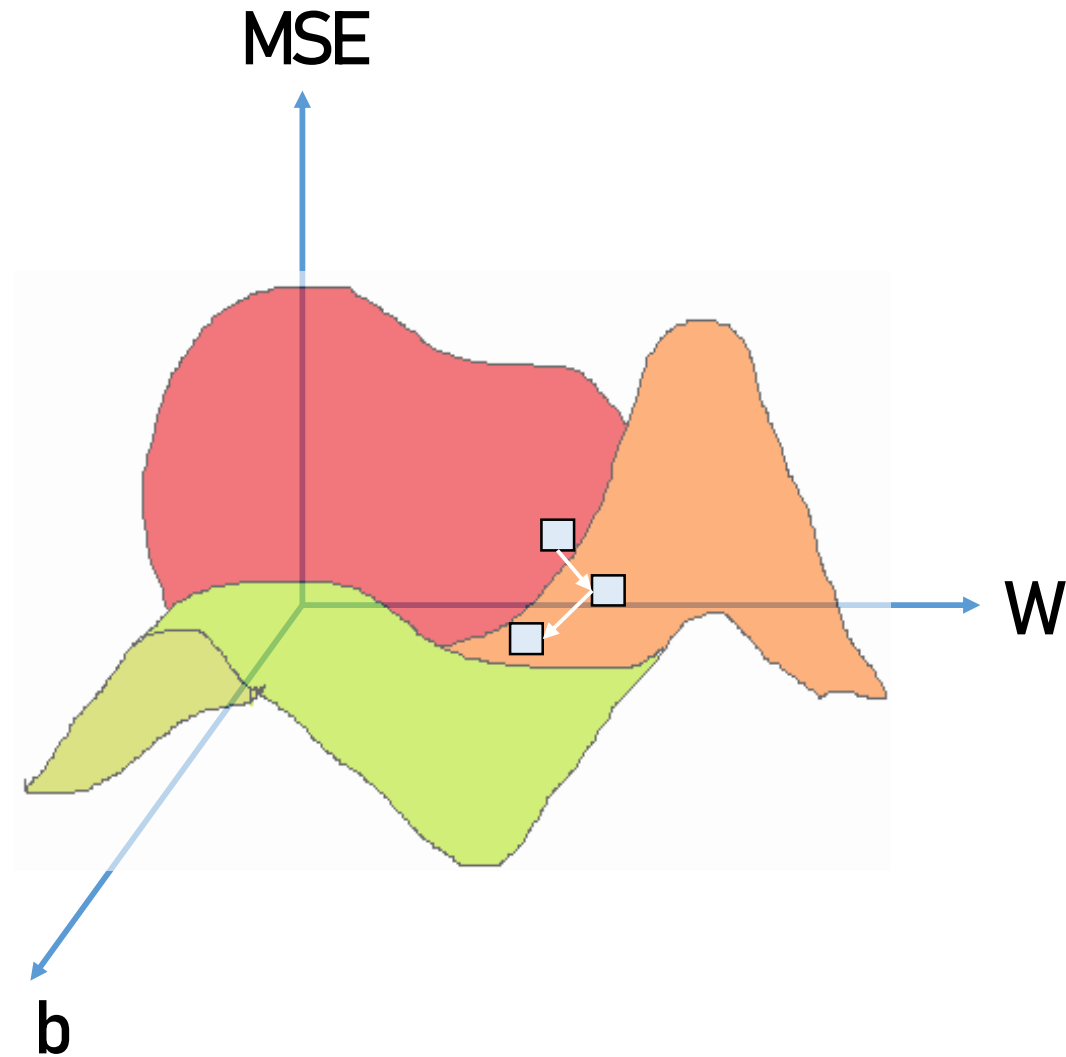




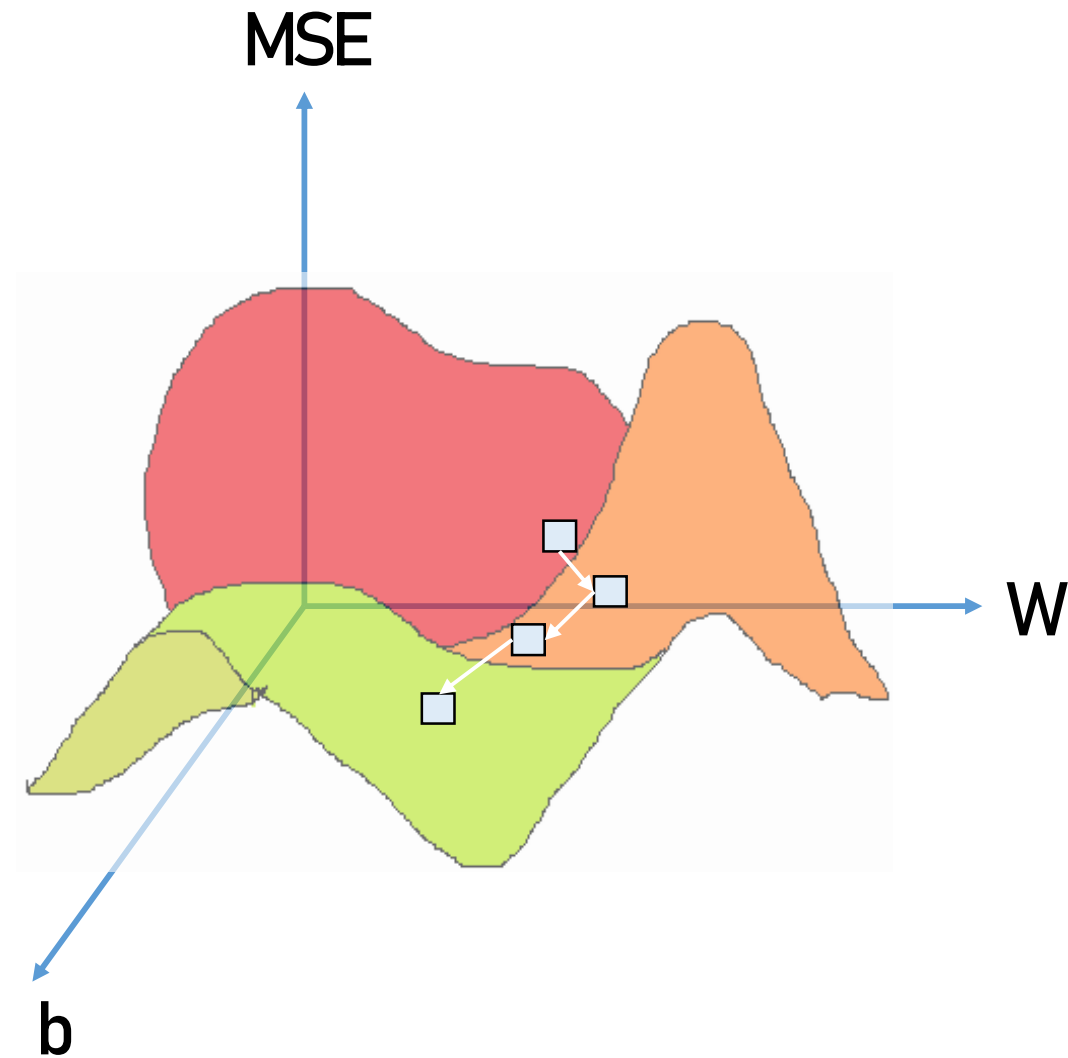
# Visualizing Gradient Descent



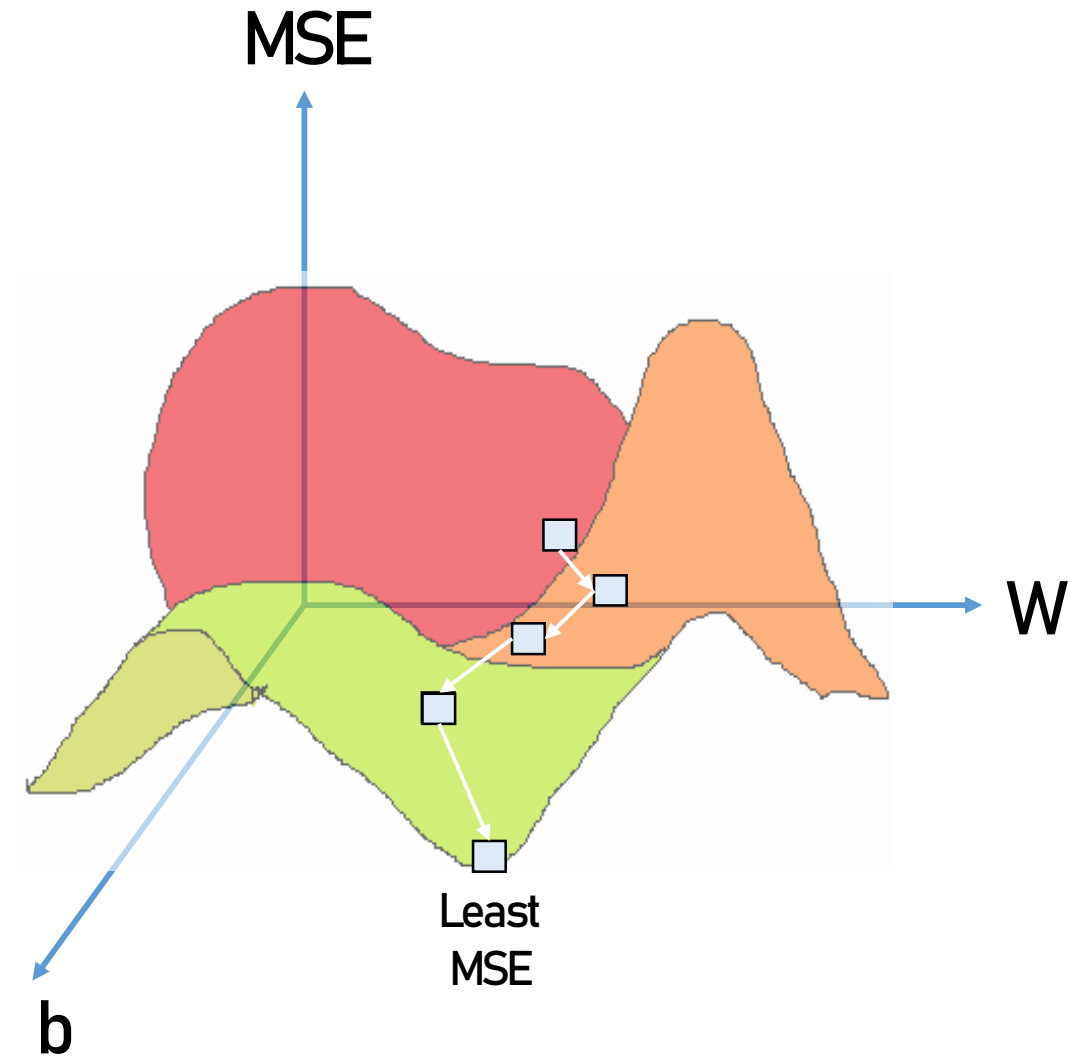
# Visualizing Gradient Descent



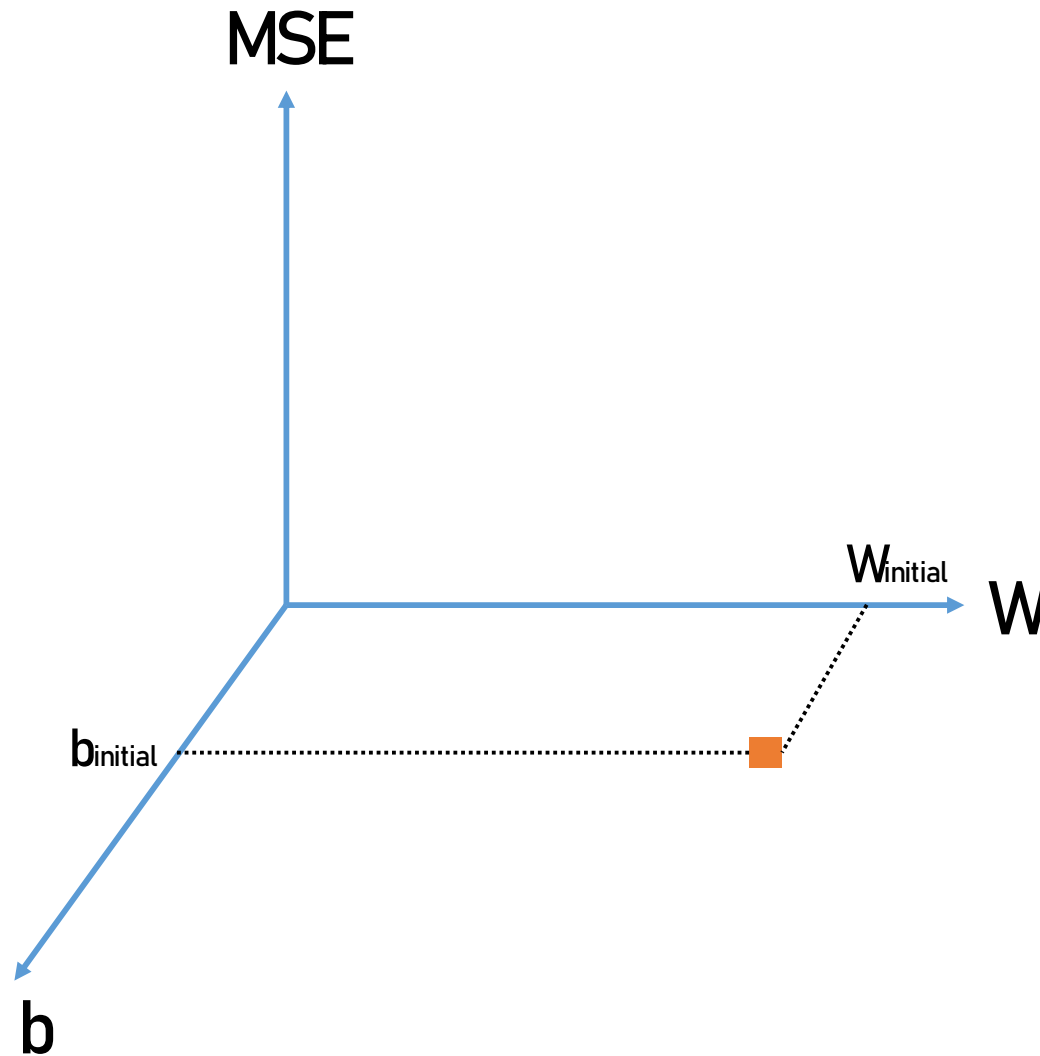
# Visualizing Gradient Descent



# Visualizing Gradient Descent



# Visualizing Gradient Descent



Change  $W$  and  $b$  to reduce the MSE to find best values



# Visualizing Gradient Descent

