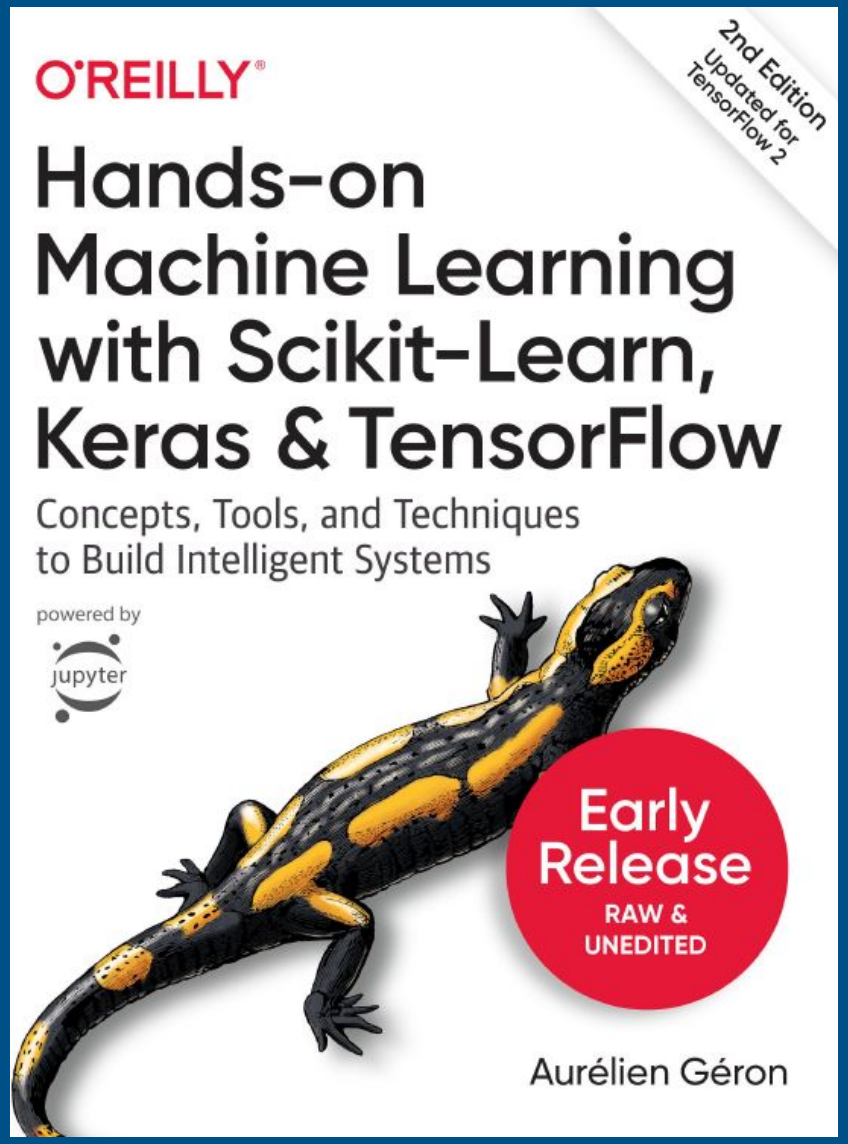
10장: Keras를 사용한 인공 신경망 소개

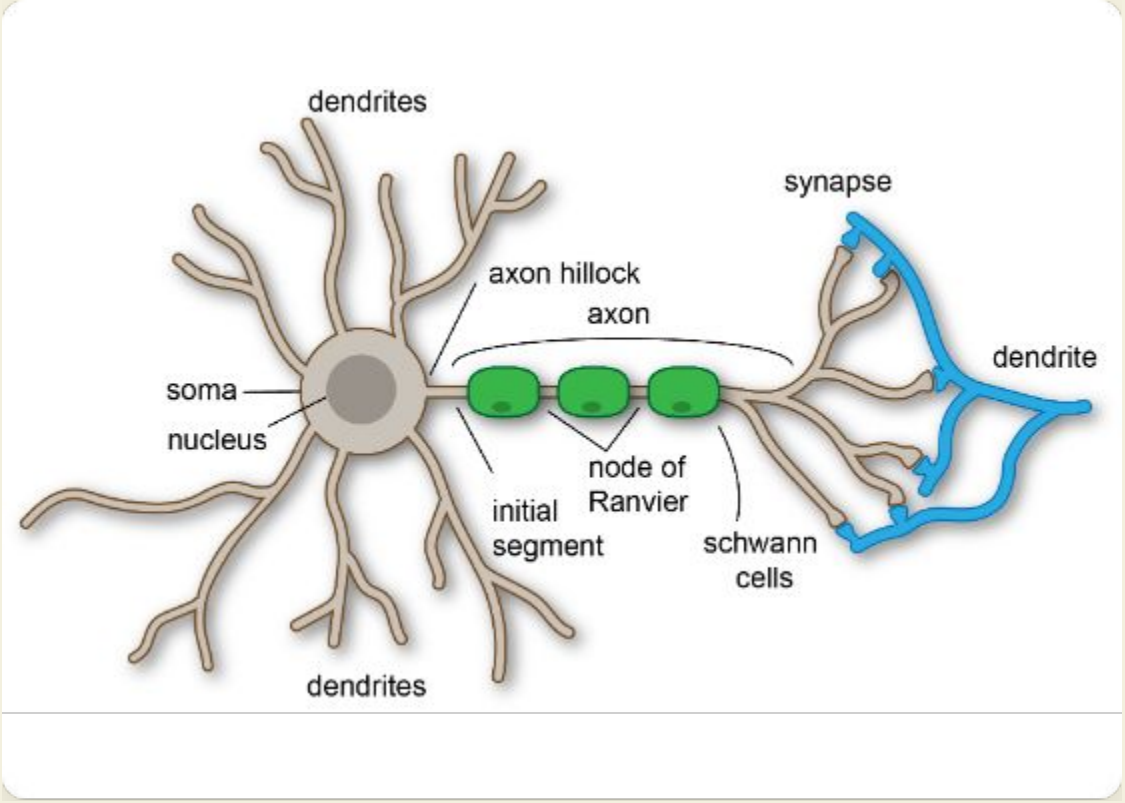
역사적 배경부터 Keras 구현 및 하이퍼파라미터 튜닝까지

I. 인공 신경망(ANN)의 기원 및 기본 개념



인공 신경망(ANN)의 탄생과 부활

- **기원 (1943):** McCulloch & Pitts가 제안. 생물학적 뉴런이 명제 논리를 수행하는 방식에서 영감을 받음.
- **생물학적 구조:** 세포체(핵), 수상돌기(수신), 축삭돌기(전송), 시냅스(연결)로 구성되어 전기 신호(AP)를 전달.
- **최근의 부활:** 컴퓨팅 성능 향상(GPU), 방대한 데이터(Big Data), 훈련 알고리즘의 개선으로 다시 주목받음.



II. 퍼셉트론과 다층 퍼셉트론(MLP)

퍼셉트론 (Perceptron)

구조 및 원리

TLU(Threshold Logic Unit)를 기반으로 하며,
입력의 가중치 합(z)에 스텝 함수(Heaviside 등)를
적용하여 결과를 출력

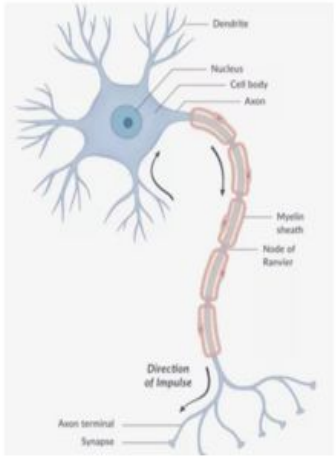
학습 규칙

Hebb의 규칙을 변형하여, 오차가 발생할 때마다
연결 가중치를 조정하여 학습

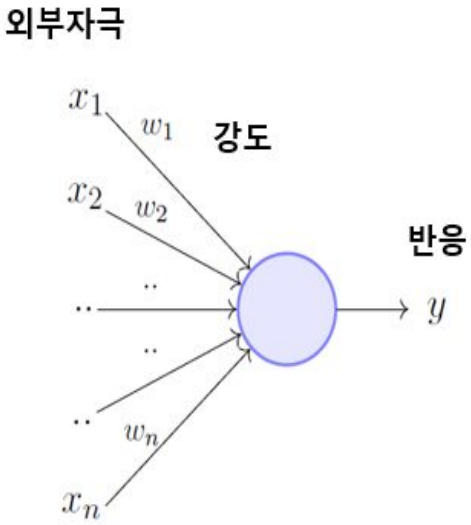
한계점

단순 선형 이진 분류만 가능하며, **XOR 문제**와 같은
비선형 문제를 해결할 수 없다는 한계 존재

- 인간의 뉴런과 축삭돌기를 모방한 인공신경망 구조
- 퍼셉트론(perceptron) : 외부자극과 강도의 총합이 특정한 역치(threshold, θ)를 넘으면 반응(y)을 하는 단일 뉴런



[이진분류 뉴런]



$$y = 1 \quad if \sum_{i=1}^{n} w_i * x_i \geq \theta$$
$$= 0 \quad if \sum_{i=1}^{n} w_i * x_i < \theta$$

Rewriting the above,

$$y = 1 \quad if \sum_{i=1}^{n} w_i * x_i - \theta \geq 0$$
$$= 0 \quad if \sum_{i=1}^{n} w_i * x_i - \theta < 0$$

Scikit learn
퍼셉트론 (Perceptron)

$$z = w_1 x_1 + w_2 x_2 + b$$

```python
import numpy as np
from sklearn.datasets import load_iris
from sklearn.linear_model import Perceptron

iris = load_iris(as_frame=True)
X = iris.data[["petal length (cm)",
               "petal width (cm)"]].values
y = (iris.target == 0)  # Iris setosa

per_clf = Perceptron(random_state=42)
per_clf.fit(X, y)

X_new = [[2, 0.5], [3, 1]]
y_pred = per_clf.predict(X_new)
```

- x_1: 꽃잎 길이 (petal length)
- x_2: 꽃잎 너비 (petal width)
- w_1, w_2: 각 특성에 할당된 **가중치 (Weights)**
- b: **편향 (Bias)**
- z: 최종 점수 (Logit)

$$z = (-0.7) \cdot \text{length} + (-1.2) \cdot \text{width} + 1.3$$

Scikit learn
퍼셉트론 (Perceptron)

3. 활성화 함수 (Heaviside Step Function)

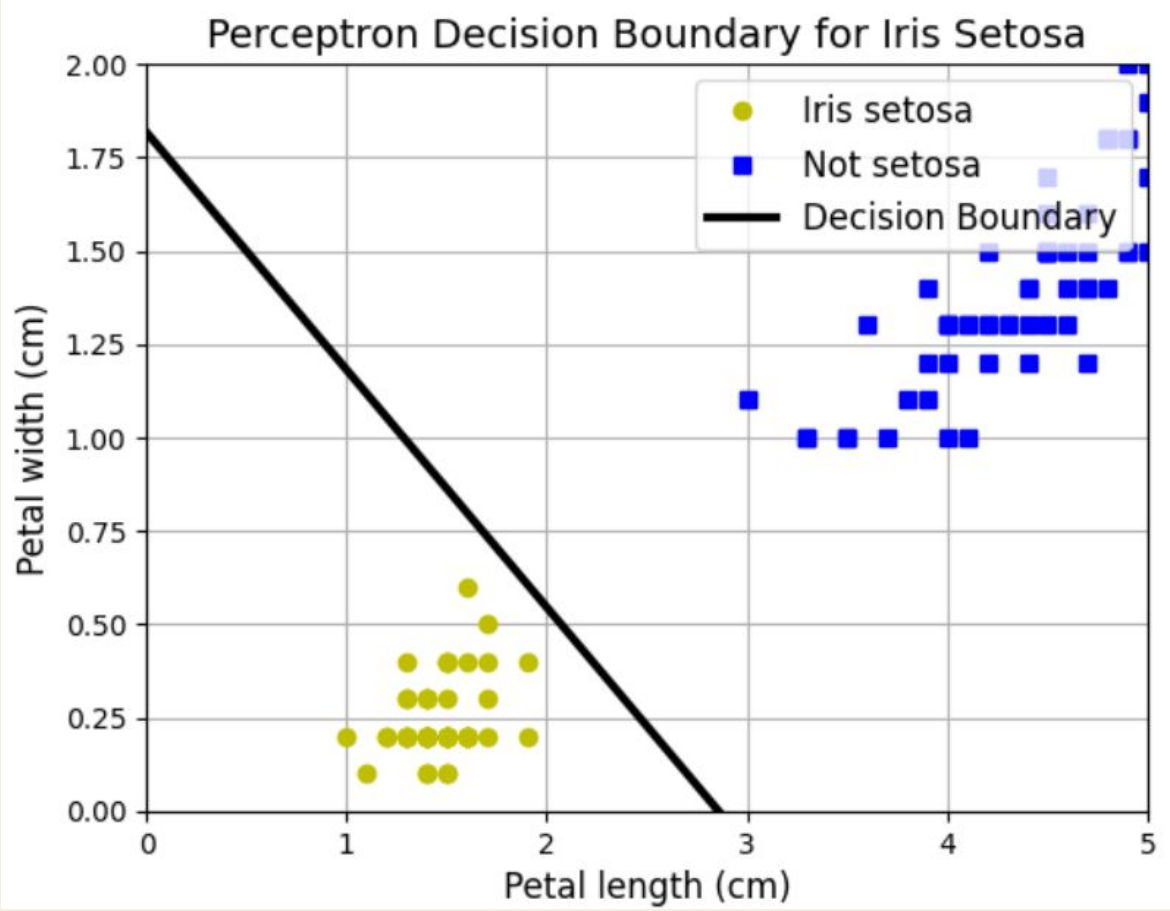
퍼셉트론은 위에서 계산된 z 값을 바탕으로 최종 클래스($\hat{y}$)를 결정합니다.

$$\hat{y} = \begin{cases} 1(\text{Setosa}) & \text{if } z > 0 \\ 0(\text{Not Setosa}) & \text{if } z \leq 0 \end{cases}$$

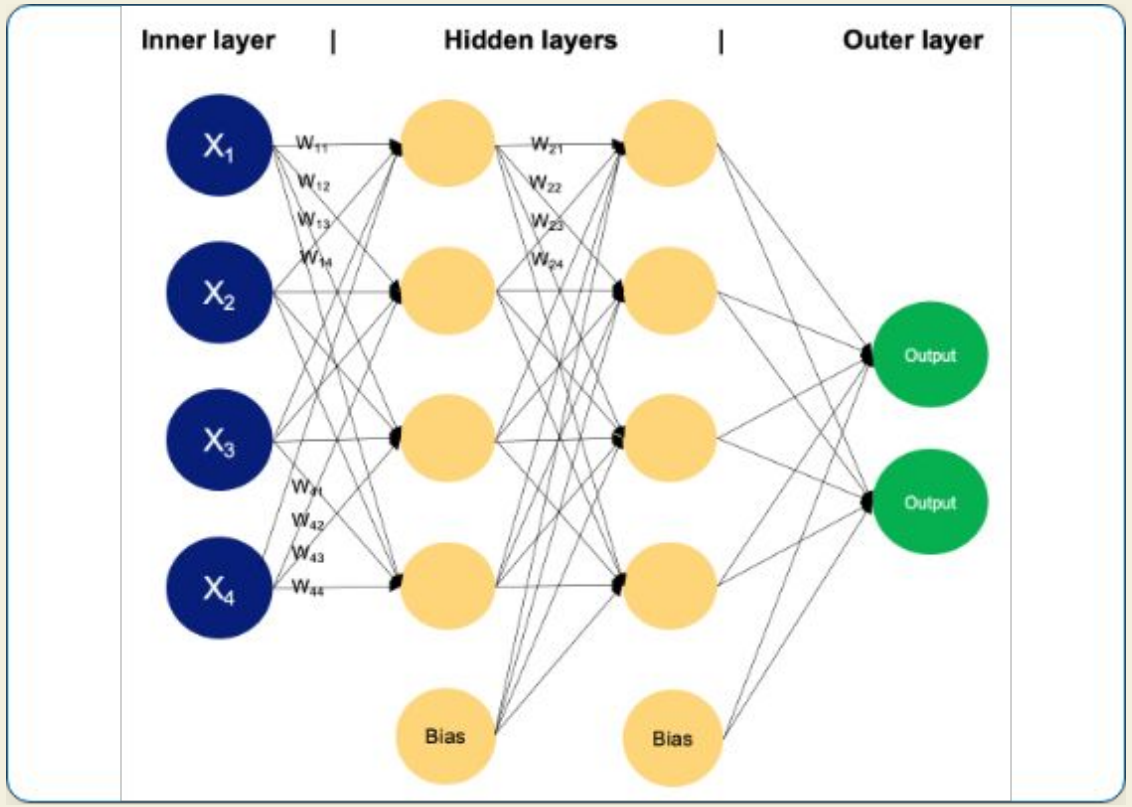
4. 기하학적 의미 (Decision Boundary)

이 식을 x_2(너비)에 대해 정리하면, 데이터 공간을 두 영역으로 나누는 **직선의 방정식**

$$x_2 = -\frac{w_1}{w_2}x_1 - \frac{b}{w_2}$$



다층 퍼셉트론 (MLP)



구조적 특징

- **구층:** 입력층, 하나 이상의 은닉층(Hidden Layer), 출력층으로 구성됩니다. 은닉층이 깊어지면 DNN(Deep Neural Network)이라 부릅니다.
- **역전파 (Backpropagation):** 경사 하강법과 역방향 자동 미분(Reverse-mode Autodiff)을 사용하여 효율적으로 학습합니다.
- **활성화 함수:** 미분 가능한 함수(Sigmoid, Tanh, ReLU)를 사용하여 비선형 문제를 해결합니다.

MLP의 활용: 회귀 vs 분류

하이퍼파라미터	회귀 (Regression)	이진 분류 (Binary)	다중 분류 (Multiclass)
출력 뉴런 수	예측 차원당 1개	1개	클래스 수만큼
출력 활성화 함수	없음 (또는 ReLU/Softplus)	로지스틱 (Sigmoid)	소프트맥스 (Softmax)
손실 함수	MSE, MAE	Binary Cross-Entropy	Categorical Cross-Entropy

III. Keras API
구현 방식

Keras 모델 구현을 위한 3가지 API

Sequential API

가장 간단한 구조로, 레이어를 순차적으로 쌓는 방식입니다. `Flatten`과 `Dense` 레이어를 주로 사용하며, 빠르게 프로토타입을 만들 때 유용합니다.

Functional API

다중 입력/출력이나 비순차적 연결이 필요한 복잡한 모델(Wide & Deep 등)을 설계할 때 사용합니다. 유연성이 높습니다.

Subclassing API

명령형 프로그래밍 방식입니다. `call()` 메서드 내에 조건문이나 루프 등 동적인 동작을 정의할 수 있으나, 모델 분석이 어렵습니다.

IV. 모델 관리 및 하이퍼파라미터 튜닝

모델 훈련 및 관리 도구

- **컴파일 (Compile):** 손실 함수, 옵티마이저(SGD, Adam 등), 평가 지표(Accuracy)를 설정하는 단계입니다.
- **훈련 및 모니터링 :** `fit()` 메서드로 훈련하며, `History` 객체를 통해 학습 곡선을 시각화하여 과대적합을 감지합니다.
- **콜백 (Callbacks):**
  - **ModelCheckpoint:** 훈련 중 최상의 모델을 자동으로 저장합니다.
  - **EarlyStopping:** 성능 향상이 멈추면(patience) 훈련을 조기 종료합니다.
  - **TensorBoard:** 인터랙티브한 학습 시각화 도구를 제공합니다.

하이퍼파라미터 튜닝

탐색 방법: RandomizedSearchCV 등을 사용하여 최적의 조합을 찾습니다.

주요 파라미터:

- **은닉층 수:** 깊은 층이 파라미터 효율성이 높고 전이 학습에 유리함.
- **뉴런 수:** 과하게 설정 후 조기 종료(Early Stopping)로 제어.
- **학습률 (Learning Rate):** 가장 중요한 요소. 발산 직전 최대값의 절반 정도가 적절.
- **배치 크기:** GPU 메모리가 허용하는 최대 크기 권장.

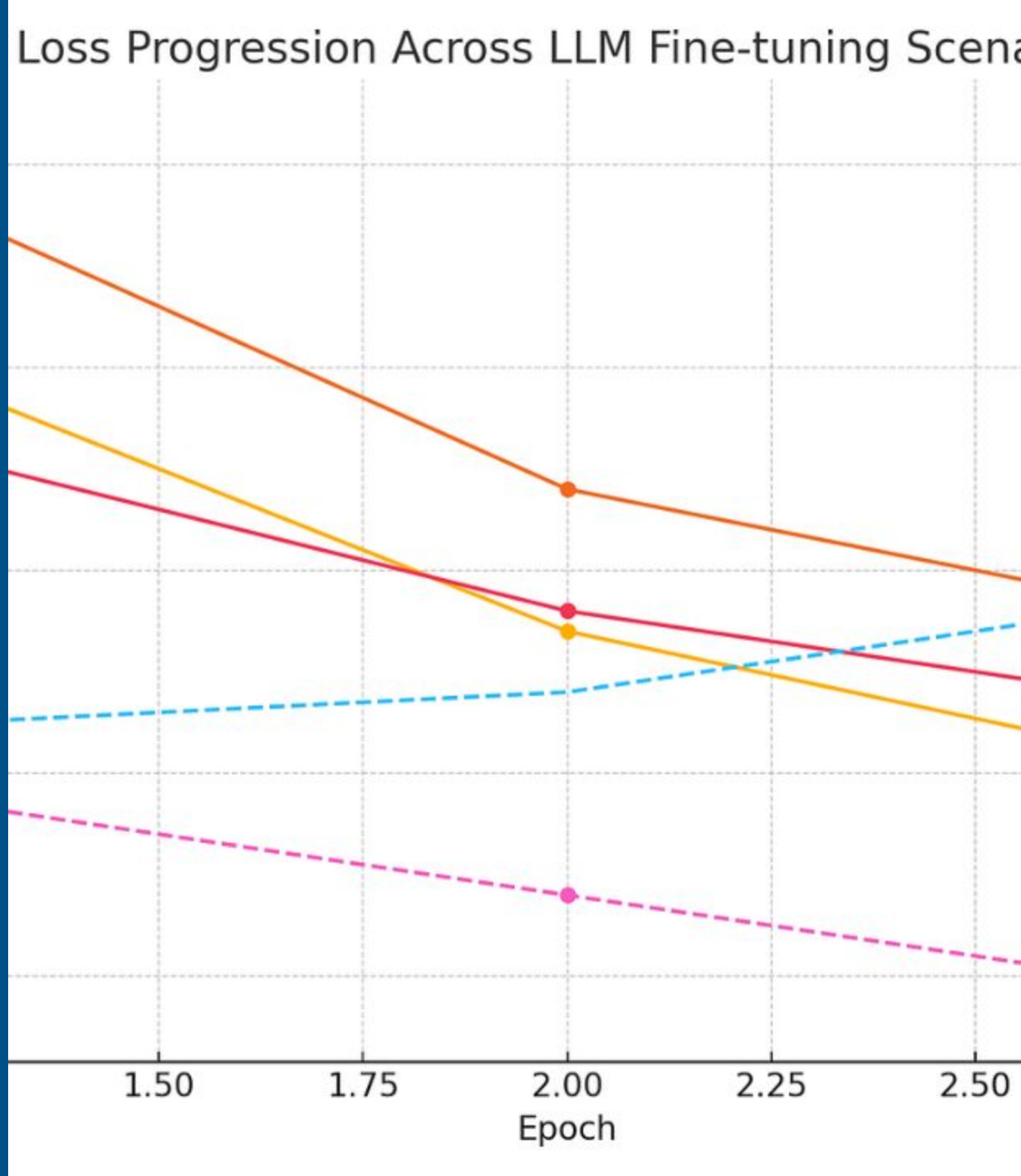
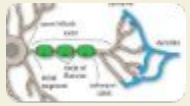
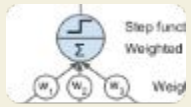


Image Sources



https://upload.wikimedia.org/wikipedia/commons/9/9e/Anatomy_of_neuron.png

Source: en.wikipedia.org

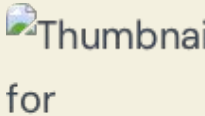


https://www.googleapis.com/download/storage/v1/b/kaggle-forum-message-attachments/o/inbox%2F11399696%2Fab97840b7b39b9e615cbef0a756253a7%2Fmlst_1004.png?generation=1708861648029441&alt=media

Source: www.kaggle.com

https://miro.medium.com/v2/resize:fit:1400/1*k_qarynPR-7Lb-jVUpq_Qw.png

Source: medium.com



https://miro.medium.com/v2/resize:fit:1400/1*Yt2o4eGhitQpHwYEga75Ug.png

Source: medium.com

https://img.freepik.com/free-photo/polygonal-blue-abstract-background-shapes-network-neural-connections-big-data-neural-concept_90220-514.jpg

Source: www.freepik.com