

TensorFlow Playground (텐서플로우 플레이그라운드)

1. 목적

- **신경망 시각화 실험장:** 복잡한 코드 없이 웹 브라우저에서 직접 신경망(Neural Network)이 어떻게 학습하는지 눈으로 볼 수 있도록 만든 도구입니다.
- **교육·연구용 학습 도구:** 딥러닝의 핵심 개념(은닉층 수, 뉴런 개수, 활성화 함수, 학습률, 정규화 등)이 모델 성능에 어떤 영향을 주는지 즉각적으로 시각화하여 보여줍니다.
- **범위 제한:** CNN, RNN, Transformer 같은 복잡한 구조는 지원하지 않으며, 단순한 다층 퍼셉트론(MLP)을 2차원 데이터에 적용하는 것에 집중합니다.

2. 사용법 (상세 설명)

A. 화면 구성

왼쪽 패널: 데이터 & 특징 (Features)

- **데이터셋 선택:** 두 개의 점 분포(Gaussians), 원형(Circle), XOR, 나선형(Spiral) 등 다양한 데이터셋을 선택할 수 있습니다.
- **노이즈 (Noise):** 데이터 라벨을 일부 섞어 학습의 난이도를 높입니다.
- **학습/검증 비율:** Train 데이터와 Test 데이터의 비율을 조절합니다.
- **특징 선택 (Features):** 기본 특징인 x_1, x_2 뿐만 아니라 $x_1^2, x_2^2, x_1x_2, \sin(x_1), \sin(x_2)$ 와 같은 파생 특징을 추가할 수 있습니다. 이를 통해 '특징 공학(Feature Engineering)'을 체험해볼 수 있습니다.

중앙: 신경망 구조

- **은닉층 추가/삭제:** + 또는 - 버튼으로 층(Layer)을 늘리거나 줄일 수 있습니다.
- **뉴런 개수 조정:** 각 층마다 뉴런(Neuron)의 수를 선택합니다.
- **활성화 함수 (Activation):** Tanh, ReLU, Sigmoid, Linear 중에서 설정할 수 있습니다.
- **가중치 시각화:** 뉴런을 연결하는 선의 색깔과 두께로 가중치(Weight)를 시각화합니다. (파랑: 양수, 주황: 음수, 두께: 가중치의 크기)

오른쪽 패널: 학습 설정

- **실행/일시정지/초기화**: 학습을 시작(▶), 일시 정지(■ ■), 초기화(↺) 할 수 있습니다.
- **학습률 (Learning rate)**: 값이 크면 학습이 빠르지만 불안정해질 수 있고, 작으면 안정적이지만 느립니다.
- **정규화 (Regularization)**: L1, L2 정규화를 선택하고 강도(λ)를 조절하여 과적합(Overfitting)을 방지합니다.
- **배치 크기 (Batch size)**: 한 번의 학습 단계에서 사용하는 데이터의 개수를 정합니다.
- **손실/정확도 그래프**: Train loss와 Test loss를 그래프로 비교하며 모델의 성능을 확인할 수 있습니다.

B. 실습 예시

- **XOR 문제:**

- Linear 활성화 함수만 사용하면 문제를 해결하지 못하지만, Tanh나 ReLU 같은 비선형 함수로 바꾸면 성공합니다.
- **학습 목표:** 비선형 함수의 필요성을 이해합니다.

- **Circle 데이터:**

- 기본 특징(x_1, x_2)만으로는 학습이 매우 어렵습니다.
- x_{12}, x_{22} 특징을 추가하면 학습이 훨씬 빠르고 정확해집니다.
- **학습 목표:** 특징 공학의 효과를 확인합니다.

- **정규화 (Regularization):**

- 데이터에 노이즈를 추가한 뒤 정규화 없이($\lambda=0$) 학습하면 Train 성능은 높지만 Test 성능은 낮은 과적합 현상이 발생합니다.
- λ 값을 높이면 결정 경계가 단순해지면서 Test 성능이 개선됩니다.
- **학습 목표:** 정규화를 통한 과적합 방지 원리를 이해합니다.

- **학습률 (Learning Rate) 실험:**

- 학습률이 너무 크면 손실 값이 발산하거나 진동합니다.
- 너무 작으면 학습 속도가 매우 느립니다.
- 적절한 값(예: 0.03)에서 안정적으로 학습이 진행됩니다.
- **학습 목표:** 적절한 학습률 설정의 중요성을 배웁니다.

- **깊이 vs 너비:**

- 뉴런 8개를 한 층에 배치하는 것(넓은 모델)과 2~3개의 층으로 나누어 배치하는 것(깊은 모델)의 성능을 비교할 수 있습니다.
- 나선형(Spiral)과 같이 복잡한 패턴에서는 깊은 모델의 성능이 더 좋은 경향을 보입니다.
- **학습 목표:** 신경망의 깊이와 너비가 모델 성능에 미치는 영향을 탐구합니다.