

ニューラルネットワーク実習03

浅川伸一

損失関数

- 損失関数 `loss functions`
- 目的関数 `objective functions`
- 誤差関数 `error functions`

誤差, 損失, 目的, と呼び名が異なる場合があります。

ですが, 同じと考えて良い場合が多いです。

論文によっては, 誤差関数 と 制約項 との和 を損失関数と呼ぶ場合があります。

平均 2 乗損失 mean square loss

正解データ y とモデルによる予測 \hat{y} との差の 2 乗の総和をデータ数で割った値を **平均 2 乗誤差 mean square error** と呼びます。

MSEloss

$$E = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2 \quad (1)$$

2 値交差エントロピー binary cross entropy

エントロピーとは、熱力学 と 情報理論 の 両分野で用いられる概念です。

両分野で説明は異なりますが、式は同じで $H(x) = -p(x) \log p(x)$ と表されます。ここで、 $p(x)$ は x の確率、 \log は対数を表します。

交差エントロピー損失は、教師信号が 0 か 1 かを取る場合に用いられます。

この場合、平均 2 乗損失 MSE を用いるより学習が高速化します。

BCEloss

$$l = \sum_i (t_i \log y_i + (1 - t_i) \log(1 - y_i)) \quad (2)$$

```
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline

epsilon = 10 ** -4
print(epsilon)

x = np.linspace(0+epsilon, 1-epsilon)

plt.plot(x, - np.log(x))
plt.plot(x, - np.log(1-x))
plt.show()

bce = x * np.log(x) + (1-x) * np.log(1-x)
plt.plot(x, bce)
plt.show()
```

負の対数尤度損失

近似すべき目標が確率で与えら得ている場合に用いられる損失関数です。

対数尤度（「たいすうゆうど」と読む）を最大化することは、確率モデルを扱う場合頻用される概念です。

NLLLoss

$$l = - \sum_i \log p(x_i) \quad (3)$$

カルバック・ライブラー損失

2 つの確率密度関数の差をカルバック・ライブラー・ダイバージェンス (カルバック・ライブラー距離と呼ぶ場合もある) を損失関数として用います。

`KLDivLoss`

$$\begin{aligned} \text{KL}(x, y) &= - \int p(x) \log \frac{p(x)}{q(y)} dx \\ &= - \left(\int p(x) \log p(x) dx - \int p(x) \log q(x) \right) dx \end{aligned}$$

まとめ

頻用される 4 つの損失関数について、説明しました。

ニューラルネットワークの学習に限らず、関数近似、最適化、のためには、損失関数を定義することが行われます。

クイズ

以下の頭文字は損失関数を表します。それぞれ、何を指すでしょうか

- MSE
- BCE
- KL divergence