実践機械学習

11.1.3.0-11.2.3

バッチ正規化のmotivation

Heの初期値とELU（またはReLUの変種）を使うことで訓練開始時点での勾配消失／爆発の問題は大幅に緩和された。

バッチ正規化のmotivation

Heの初期値とELU（またはReLUの変種）を使うことで訓練開始時点での勾配消失／爆発の問題は大幅に緩和された。

↓

初期値で対応しても訓練中にパラメータが変われば同様の問題を招来する可能性が残る

↓

バッチ正規化のmotivation

Heの初期値とELU（またはReLUの変種）を使うことで訓練開始時点での勾配消失／爆発の問題は大幅に緩和された。

↓

初期値で対応しても訓練中にパラメータが変われば同様の問題を招来する可能性が残る

↓

バッチ正規化（Batch Normalization）

バッチ正規化のmotivation

Heの初期値とELU（またはReLUの変種）を使うことで訓練開始時点での勾配消失／爆発の問題は大幅に緩和された。

↓

初期値で対応しても訓練中にパラメータが変われば同様の問題を招来する可能性が残る

↓

バッチ正規化（Batch Normalization）

バッチ正規化層は
各隠れ層の活性化関数の直前か直後に挿入される

バッチ正規化のアルゴリズム

アルゴリズム

$$1.\mu_B = \frac{1}{m_B}\sum_{i=1}^{m_B}\mathbf{x}^{(i)}$$

$$2.{\sigma_B}^2 = \frac{1}{m_B}\sum_{i=1}^{m_B}\left(\mathbf{x}^{(i)} - \mu_B\right)^2$$

$$3.\hat{\mathbf{x}}^{(i)} = \frac{\mathbf{x}^{(i)} - \mu_B}{\sqrt{{\sigma_B}^2 + \varepsilon}}$$

$$4.\mathbf{z}^{(i)} = \gamma \otimes \hat{\mathbf{x}}^{(i)} + \beta$$

①ミニバッチB全体で計算された入力の平均のベクトル

②ミニバッチB全体で計算された入力の標準偏差のベクトル

③平均0、分散1にスケーリング（正規化）

④スケーリング＆シフト
（ただしγとβは学習されるパラメータ）

バッチ正規化のアルゴリズム

アルゴリズム

$$1.\mu_B = \frac{1}{m_B}\sum_{i=1}^{m_B}\mathbf{x}^{(i)}$$

$$2.{\sigma_B}^2 = \frac{1}{m_B}\sum_{i=1}^{m_B}\left(\mathbf{x}^{(i)} - \mu_B\right)^2$$

$$3.\hat{\mathbf{x}}^{(i)} = \frac{\mathbf{x}^{(i)} - \mu_B}{\sqrt{{\sigma_B}^2 + \varepsilon}}$$

$$4.\mathbf{z}^{(i)} = \gamma \otimes \hat{\mathbf{x}}^{(i)} + \beta$$

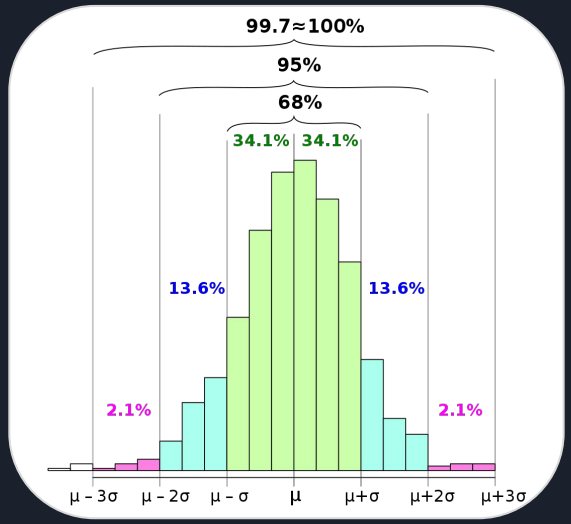
①ミニバッチB全体で計算された入力の平均のベクトル

②ミニバッチB全体で計算された入力の標準偏差のベクトル

③平均0、分散(ほぼ)1にスケーリング（正規化）

④スケーリング＆シフト
（ただしγとβは学習されるパラメータ）

バッチ正規化④の操作について



正規分布のグラフ

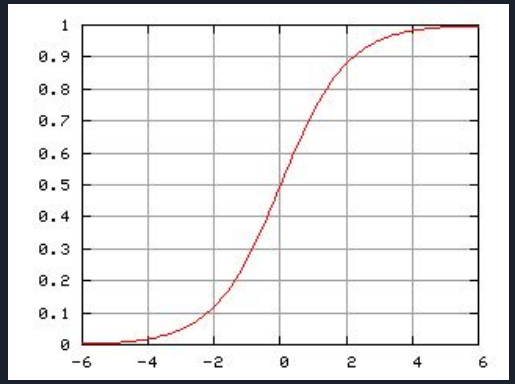
[https://ja.wikipedia.org/wiki/68%E2%80%9395%E2%80%9399.7%E5%89%87]

$$3.\hat{\mathbf{x}}^{(i)} = \frac{\mathbf{x}^{(i)} - \mu_B}{\sqrt{{\sigma_B}^2 + \varepsilon}}$$

$$4.\mathbf{z}^{(i)} = \gamma \otimes \hat{\mathbf{x}}^{(i)} + \beta$$

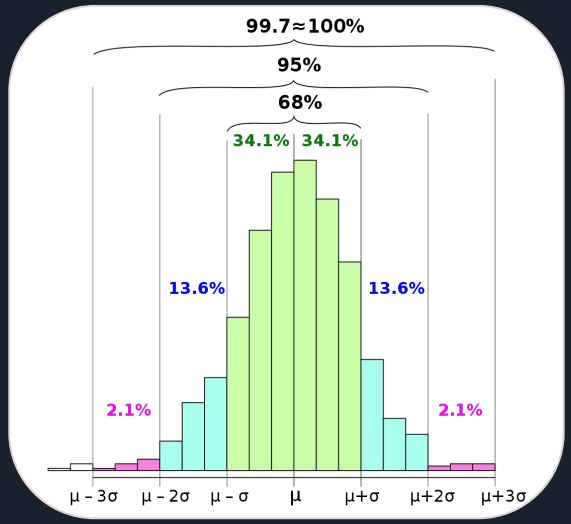
③の左辺において、平均0、分散(ほぼ)1になり、-1から1の範囲の値が多くなる（仮に正規分布であれば約68%）。
たとえば、シグモイド関数において、-1から1の範囲は線形に近い傾向にあり、非線形の活性化関数を使うメリットが損なわれる可能性もある。

→



シグモイド関数

バッチ正規化④の操作について



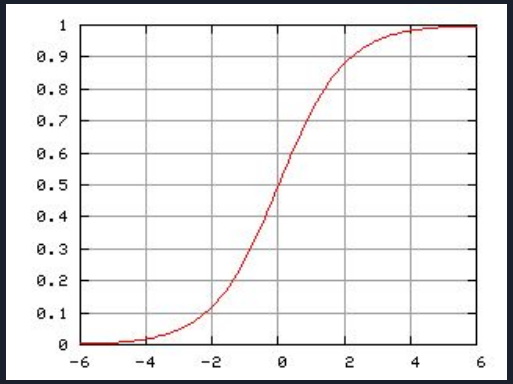
正規分布のグラフ

[https://ja.wikipedia.org/wiki/68%E2%80%9395%E2%80%9399.7%E5%89%87]

$$3.\hat{\mathbf{x}}^{(i)} = \frac{\mathbf{x}^{(i)} - \mu_B}{\sqrt{{\sigma_B}^2 + \varepsilon}}$$

$$4.\mathbf{z}^{(i)} = \gamma \otimes \hat{\mathbf{x}}^{(i)} + \beta$$

③の左辺において、平均0、分散(ほぼ)1になり、-1から1の範囲の値が多くなる（仮に正規分布であれば約68%）。
たとえば、シグモイド関数において、-1から1の範囲は線形に近い傾向にあり、非線形の活性化関数を使うメリットが損なわれる可能性もある。
→分布を、学習するパラメータγとβで調整



シグモイド関数

推論時のバッチ正規化

学習時はバッチごとに計算ができるが、推論時は以下の問題が発生する。

・テストデータが1つしかない場合

→バッチの平均や標準偏差を考えられない

・テストデータが少量で学習データと分布が異なる場合

→平均や標準偏差を計算しても信頼性に乏しい

↓

予測時に使うため、訓練中にレイヤの入力平均と標準偏差の指数移動平均を用いて、最終的な平均（μ）と標準偏差（σ）を推計する。

BatchNormalization層のパラメータはγ、β、μ、σの4つ。学習時と異なり、予測時にはこの4つを固定された値として使用する。

バッチ正規化の指数移動平均

予測時に使うため、訓練中にレイヤの入力平均と標準偏差の指数移動平均を用いて、最終的な平均（μ）と標準偏差（σ）を推計する。

指数移動平均の更新式

$$\mu := \mu * momentum + \mu' (1 - momentum)$$

$$\sigma := \sigma * momentum + \sigma' (1 - momentum)$$

μ’やσ’はそれぞれ最新のミニバッチの平均と標準偏差を指す。

momentumは0～1のあいだの値を取るハイパーパラメータ（平滑化係数）

一般に、momentumの適切な値は、0.9、0.99、0.999などと1に近い数。データセットが大きくなるか、ミニバッチが小さくなれば9の数を増やしていく。