ディープラーニングの仕組みを知ろう！

第 2 回　人工知能勉強会

Shion MORISHITA

July 18, 2024

目次

はじめに

目的

- 勾配降下法の概念を理解する
- ニューラルネットワークの各層の変数やパラメータの表記を理解する
- ニューラルネットワークのコスト関数に勾配降下法を適用する方法を理解する
- 勾配降下法を適用する上で発生する問題点について理解する

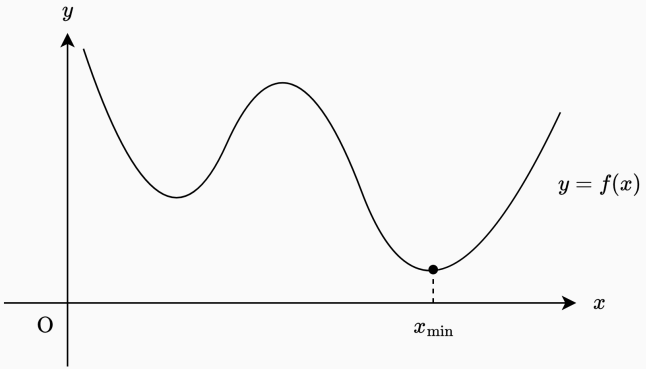
勾配降下法

勾配降下法

勾配降下法の基本概念

勾配降下法とその目的

- 機械学習や最適化の分野で広く用いられる最適化アルゴリズム
- 目的：最小化（または最大化）したい関数の最適なパラメータを見つけること



勾配降下法のアイデア

どのように関数が最小となるパラメータを見つけるか？

- 【重要】多変数関数の最小条件 を利用（第 1 回）
- 斜面を転がるボールのイメージ

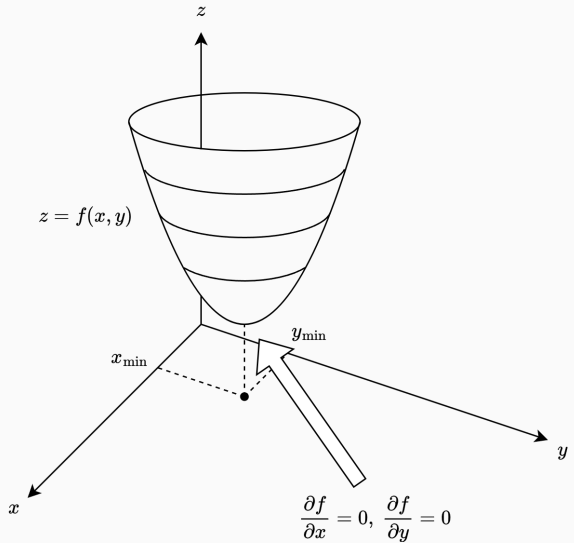
【重要】多変数関数の最小条件（第１回）

関数 $z = f(x, y)$ が最小になる必要条件は、$\dfrac{\partial f}{\partial x} = 0$ かつ $\dfrac{\partial f}{\partial y} = 0$

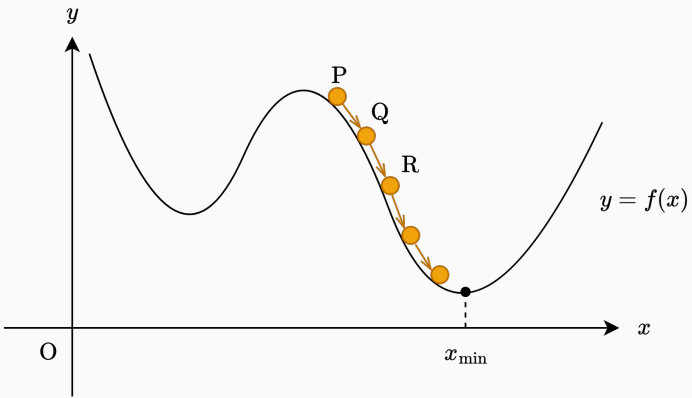
ポイント

どの成分から見ても傾きが 0 なら、最小値の可能性あり！

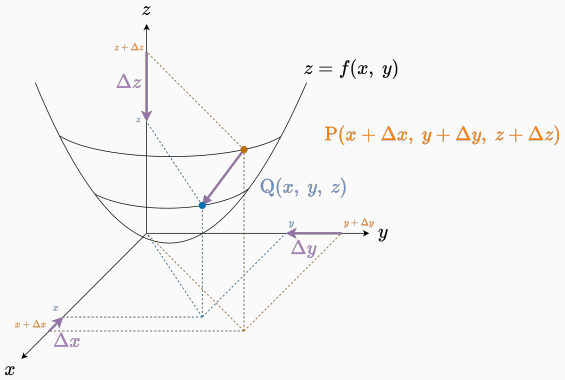
多変数関数の最小条件のイメージ



斜面を転がるボールのイメージ



斜面を転がるボール（多変数関数 ver.）



- 最速で転がる（Δz が最小になる）には $\Delta x,\ \Delta y$ をどう決める？

勾配降下法

勾配降下法の式

【重要】勾配降下法の基本式（2 変数関数）

η を正の小さな定数として、変数 $x,\ y$ が $x+\Delta x,\ y+\Delta y$ に変化するとき、関数 $z=f(x,y)$ が最も減少するのは次の関係を満たすときである：

$$\begin{bmatrix}\Delta x\\ \Delta y\end{bmatrix}=-\eta\begin{bmatrix}\dfrac{\partial z}{\partial x}\\ \dfrac{\partial z}{\partial y}\end{bmatrix}.$$

勾配降下法の基本式の導出 i

「関数の近似公式　簡潔 ver.」（第 1 回）より、

$$\Delta z \simeq \left\langle \begin{bmatrix} \dfrac{\partial z}{\partial x} \\ \dfrac{\partial z}{\partial y} \end{bmatrix}, \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} \right\rangle.$$

「ベクトルの基本公式」（第１回）の内積の式

$$\langle \boldsymbol{a}, \boldsymbol{b} \rangle \triangleq \|\boldsymbol{a}\| \|\boldsymbol{b}\| \cos\theta$$

および、コーシー・シュワルツの不等式

$$-\|\boldsymbol{a}\| \|\boldsymbol{b}\| \leq \langle \boldsymbol{a}, \boldsymbol{b} \rangle \leq \|\boldsymbol{a}\| \|\boldsymbol{b}\|$$

勾配降下法の基本式の導出 ii

より、内積が最小となるのは $\cos\theta = -1$ のとき、すなわち、ベクトルの向きが反対のとき（$\theta = 180°$）。

ベクトルの向きが反対というのは、ベクトルの符号が異なるという意味なので、

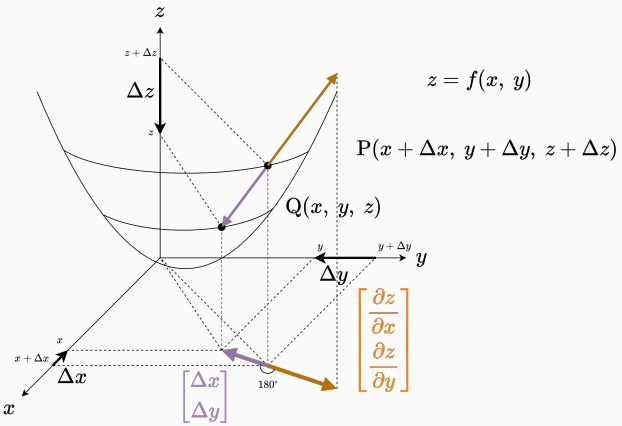
$$\boldsymbol{a} = -k\boldsymbol{b} \quad （k：正の定数）$$

と表せる。今回の表記に合わせれば、

$$\begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} = -\eta \begin{bmatrix} \dfrac{\partial z}{\partial x} \\ \dfrac{\partial z}{\partial y} \end{bmatrix} \quad （\eta：正の定数）$$

と導かれる。 □

結局どういうこと ??



ポイント　$\Delta x,\ \Delta y$ の値を、偏微分の値で決定できる！

【重要】勾配降下法の基本式（n 変数）

η を正の小さな定数として、変数 $x_1,\ x_2,\ \ldots,\ x_n$ が $x_1+\Delta x_1,\ x_2+\Delta x_2,\ \ldots,\ x_n+\Delta x_n$ に変化するとき、多変数関数 f が最も減少するのは次の関係を満たすときである：

$$\begin{bmatrix} \Delta x_1 \\ \vdots \\ \Delta x_n \end{bmatrix} = -\eta \begin{bmatrix} \dfrac{\partial f}{\partial x_1} \\ \vdots \\ \dfrac{\partial f}{\partial x_n} \end{bmatrix}. \tag{1}$$

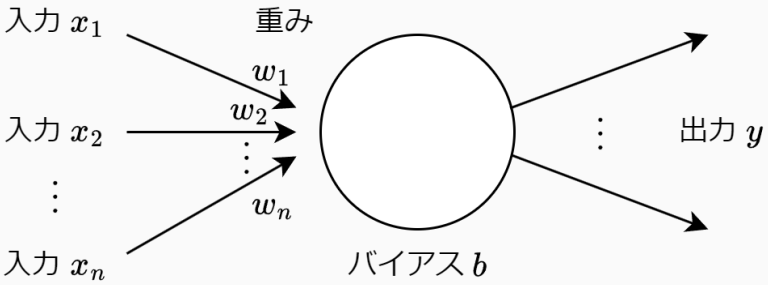
この式に従って点 $(x_1,\ \ldots,\ x_n)$ を次々と移動させることで、関数 f が最小になるパラメータを探索する方法を**勾配降下法**という。

ニューラルネットワークと勾配降下法

ニューラルネットワークと勾配降下法

ニューラルネットワークのパラメータと変数

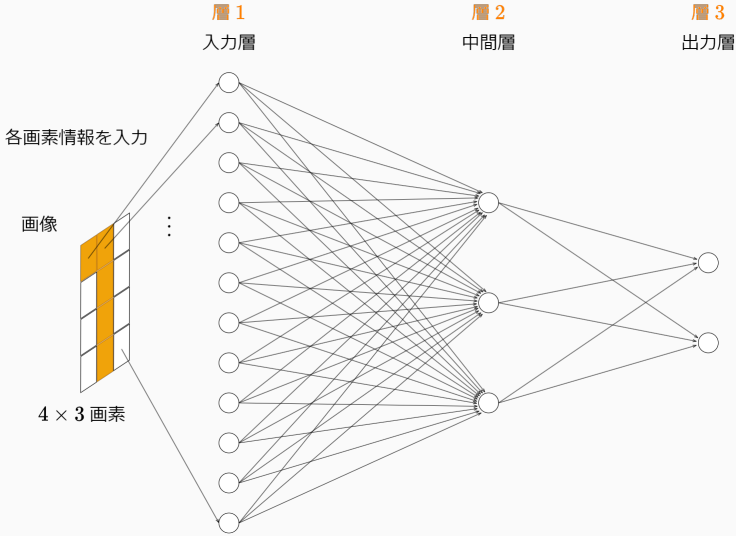
ユニットの復習



重み付き入力： $z = w_1x_1 + w_2x_2 \cdots + w_nx_n + b$

出力： $y = \sigma(z)$

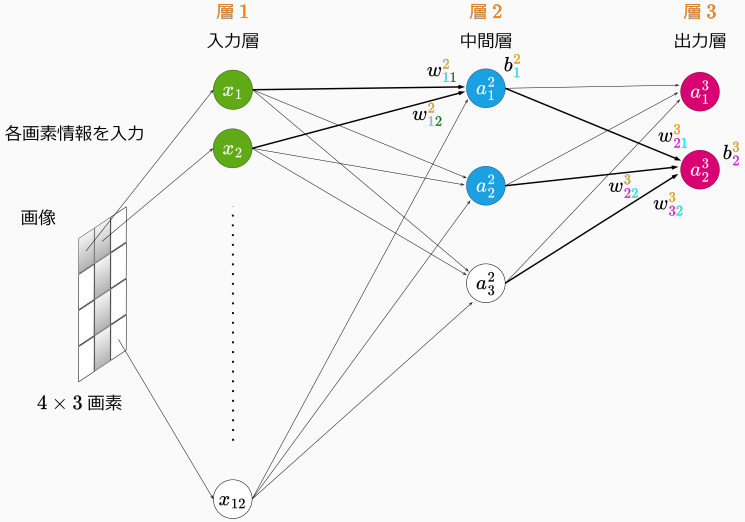
層に番号付けする



変数名・パラメータ名

- x_i
  - 入力層（層 1）にある i 番目のユニットの入力を表す変数。入力層では、出力と入力は同一値なので、出力の変数にもなる。また、該当するユニットの名称としても利用。
- w_{ji}^l
  - 層 $l-1$ の i 番目のユニットから層 l の j 番目のユニットに向けられた矢の重み。i と j の順序に注意。ニューラルネットワークを定めるパラメータ。
- z_j^l
  - 層 l の j 番目にあるユニットが処理する重み付き入力を表す変数。
- b_j^l
  - 層 l の j 番目にあるユニットのバイアス。ニューラルネットワークを定めるパラメータ。
- a_j^l
  - 層 l の j 番目にあるユニットの出力変数。また、そのユニットの名称としても利用。

変数名・パラメータ名の図示



ニューラルネットワークの変数の関係式：入力層 i

※必要な時に見返してください

入力層の i 番目のユニットの入力 x_i と出力 a_i^1 は同一値になる。

$$x_i = a_i^1 \quad (i = 1, \ldots, 12)$$

ニューラルネットワークの変数の関係式：中間層 i

※必要な時に見返してください

$a(\cdot)$ を活性化関数とする。

$$
\begin{cases}
z_1^2 = w_{11}^2 x_1 + w_{12}^2 x_2 + \cdots + w_{1,12}^2 x_{12} + b_1^2 \\
z_2^2 = w_{21}^2 x_1 + w_{22}^2 x_2 + \cdots + w_{2,12}^2 x_{12} + b_2^2 \\
z_3^2 = w_{31}^2 x_1 + w_{32}^2 x_2 + \cdots + w_{3,12}^2 x_{12} + b_3^2 \\
a_1^2 = a(z_1^2),\ a_2^2 = a(z_2^2),\ a_3^2 = a(z_3^2)
\end{cases}
$$

ニューラルネットワークの変数の関係式：中間層 ii

行列表現にすると、

$$\begin{bmatrix} z_1^2 \\ z_2^2 \\ z_3^2 \end{bmatrix} = \begin{bmatrix} w_{11}^2 & \cdots & w_{1,12}^2 \\ \vdots & \ddots & \vdots \\ w_{31}^2 & \cdots & w_{3,12}^2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_{12} \end{bmatrix} + \begin{bmatrix} b_1^2 \\ b_2^2 \\ b_3^2 \end{bmatrix}$$

$$\begin{bmatrix} a_1^2 \\ a_2^2 \\ a_3^2 \end{bmatrix} = a\left(\begin{bmatrix} z_1^2 \\ z_2^2 \\ z_3^2 \end{bmatrix} \right)$$

簡略化すると、

$$\boldsymbol{z}_2 = \mathbf{W}_2 \boldsymbol{x} + \boldsymbol{b}_2, \quad \boldsymbol{a}_2 = a(\boldsymbol{z}_2)$$

ニューラルネットワークの変数の関係式：出力層 i

※必要な時に見返してください

$a(\cdot)$ を活性化関数とする。

$$
\begin{cases}
z_1^3 = w_{11}^3 a_1^2 + w_{12}^3 a_2^2 + w_{13}^3 a_3^2 + b_1^3 \\
z_2^3 = w_{21}^3 a_1^2 + w_{22}^3 a_2^2 + w_{23}^3 a_3^2 + b_2^3 \\
a_1^3 = a(z_1^3),\ a_2^3 = a(z_2^3)
\end{cases}
$$

ニューラルネットワークの変数の関係式：出力層 ii

行列表現にすると、

$$\begin{bmatrix} z_1^3 \\ z_2^3 \end{bmatrix} = \begin{bmatrix} w_{11}^3 & w_{12}^3 & w_{13}^3 \\ w_{21}^3 & w_{22}^3 & w_{23}^3 \end{bmatrix} \begin{bmatrix} a_1^2 \\ a_2^2 \\ a_3^2 \end{bmatrix} + \begin{bmatrix} b_1^3 \\ b_2^3 \end{bmatrix}$$

$$\begin{bmatrix} a_1^3 \\ a_2^3 \end{bmatrix} = a\left(\begin{bmatrix} z_1^3 \\ z_2^3 \end{bmatrix} \right)$$

簡略化すると、

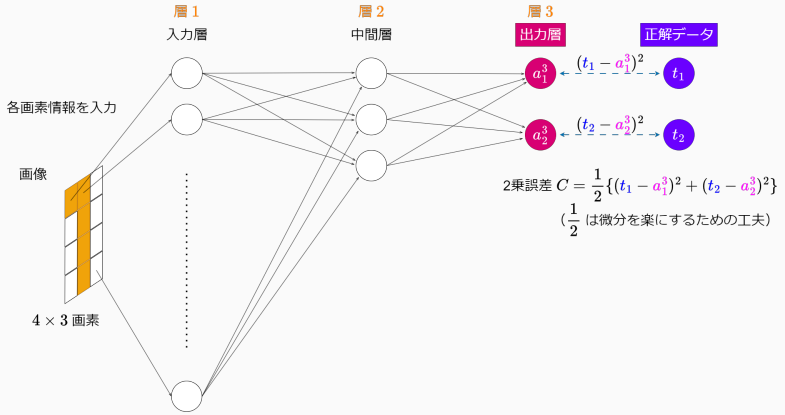
$$\boldsymbol{z}_3 = \mathbf{W}_3 \boldsymbol{a}_3 + \boldsymbol{b}_3, \quad \boldsymbol{a}_3 = a(\boldsymbol{z}_3)$$

ニューラルネットワークと勾配降下法

ニューラルネットワークのコスト関数

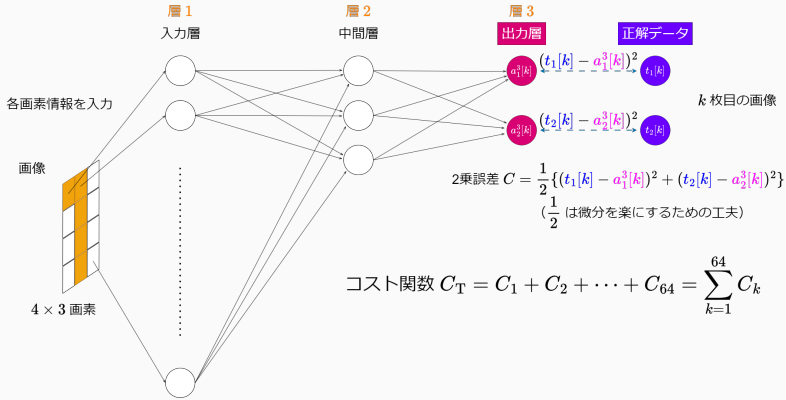
コスト関数とは

出力データと正解データとの 2 乗誤差の合計を表す関数



コスト関数とは

出力と正解データとの 2 乗誤差の合計を表す関数



コスト関数は重みとバイアスの多変数関数

- コスト関数：$C_{\mathrm{T}} = \sum_{k=1}^{64} C_k$
- 2 乗誤差：$C_k = \frac{1}{2}\left\{(t_1[k] - a_1^3[k])^2 + (t_2[k] - a_2^3[k])^2\right\}$
- 出力層のユニットの出力 $a_j^3[k]$：入力層のユニットから始まり、ニューラルネットワークを通して重みとバイアスで処理されたもの

つまり、コスト関数は大量の重みと大量のバイアスをパラメータとする多変数関数！

ディープラーニングの仕組みは、、、

- 出力データと正解データを比べて、そのズレがなくなるようにアップデートすることを繰り返す
- コスト関数という多変数関数 $C_{\mathrm{T}} = f(w_{11}^{2}, \ldots,\ w_{11}^{3}, \ldots,\ b_{1}^{2}, \ldots,\ b_{1}^{3}, \ldots)$ の最小化
  - 最小化には式 (1) で勾配降下法を適用すればよい

勾配降下法をニューラルネットワークに適用

$$\begin{bmatrix} \Delta w_{11}^2 \\ \vdots \\ \Delta w_{11}^3 \\ \vdots \\ \Delta b_1^2 \\ \vdots \\ \Delta b_1^3 \\ \vdots \end{bmatrix} = -\eta \begin{bmatrix} \dfrac{\partial C_{\mathrm{T}}}{\partial w_{11}^2} \\ \vdots \\ \dfrac{\partial C_{\mathrm{T}}}{\partial w_{11}^3} \\ \vdots \\ \dfrac{\partial C_{\mathrm{T}}}{\partial b_1^2} \\ \vdots \\ \dfrac{\partial C_{\mathrm{T}}}{\partial b_1^3} \\ \vdots \end{bmatrix}$$

を用いて、

$$\begin{bmatrix} w_{11}^2 \\ \vdots \\ w_{11}^3 \\ \vdots \\ b_1^2 \\ \vdots \\ b_1^3 \\ \vdots \end{bmatrix}$$

を

$$\begin{bmatrix} w_{11}^2 + \Delta w_{11}^2 \\ \vdots \\ w_{11}^3 + \Delta w_{11}^3 \\ \vdots \\ b_1^2 + \Delta b_1^2 \\ \vdots \\ b_1^3 + \Delta b_1^3 \\ \vdots \end{bmatrix}$$

へ更新を繰り返す。

※正の小さな定数 η を学習係数といい、モデル作成者が自由に設定する。

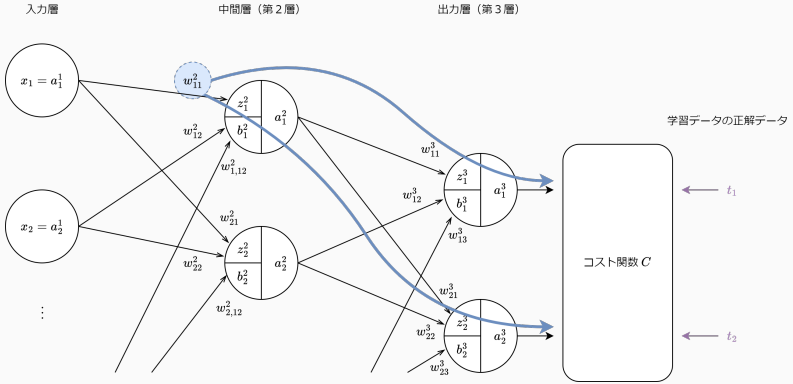
ニューラルネットワークと勾配降下法

今回の勾配降下法の問題点

実際に計算するのは大変

試しに $\dfrac{\partial C_{\mathrm{T}}}{\partial w_{11}^{2}}$ を計算してみよう！

w_{11}^{2} で偏微分するので、通り道すべてで偏微分するイメージ



実際に計算してみる i

$C_{\mathrm{T}} = \sum_{k=1}^{64} C_k$ なので、まずは $\dfrac{\partial C_k}{\partial w_{11}^2}$ を計算する。

$$\frac{\partial C_k}{\partial w_{11}^2} = \frac{\partial C_k}{\partial a_1^3[k]} \frac{\partial a_1^3[k]}{\partial z_1^3[k]} \frac{\partial z_1^3[k]}{\partial a_1^2[k]} \frac{\partial a_1^2[k]}{\partial z_1^2[k]} \frac{\partial z_1^2[k]}{\partial w_{11}^2} + \frac{\partial C_k}{\partial a_2^3[k]} \frac{\partial a_2^3[k]}{\partial z_2^3[k]} \frac{\partial z_2^3[k]}{\partial a_1^2[k]} \frac{\partial a_1^2[k]}{\partial z_1^2[k]} \frac{\partial a_1^2[k]}{\partial w_{11}^2}.$$

よって、

$$\begin{aligned}\frac{\partial C_{\mathrm{T}}}{\partial w_{11}^2} &= \sum_{k=1}^{64} \frac{\partial C_k}{\partial w_{11}^2} \\ &= \sum_{k=1}^{64} \left\{ \frac{\partial C_k}{\partial a_1^3[k]} \frac{\partial a_1^3[k]}{\partial z_1^3[k]} \frac{\partial z_1^3[k]}{\partial a_1^2[k]} \frac{\partial a_1^2[k]}{\partial z_1^2[k]} \frac{\partial z_1^2[k]}{\partial w_{11}^2} + \frac{\partial C_k}{\partial a_2^3[k]} \frac{\partial a_2^3[k]}{\partial z_2^3[k]} \frac{\partial z_2^3[k]}{\partial a_1^2[k]} \frac{\partial a_1^2[k]}{\partial z_1^2[k]} \frac{\partial a_1^2[k]}{\partial w_{11}^2} \right\}.\end{aligned}$$

実際に計算してみる ii

あとは、「ニューラルネットワークの変数の関係式」のスライドの式を見て実際に計算すれば（かなり面倒だが）計算できる。

ここで計算した $\dfrac{\partial C_{\mathrm{T}}}{\partial w_{11}^2}$ の値を用いて、Δw_{11}^2 の値を

$$\Delta w_{11}^2 \simeq -\eta \frac{\partial C_{\mathrm{T}}}{\partial w_{11}^2}$$

と設定して、

$$w_{11}^2 \longleftarrow w_{11}^2 + \Delta w_{11}^2$$

と変更することになる。 □

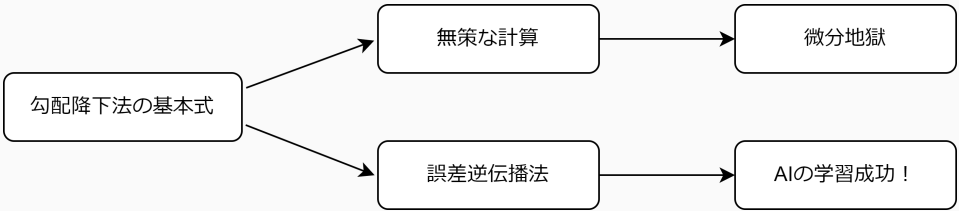
実際に計算してみる iii

ポイント　勾配降下法をそのまま適用すると、煩雑な微分地獄になる

まとめ

- 多変数関数の最小値を探す問題には勾配降下法が有効
- 一方で、ニューラルネットワークの世界では、変数・パラメータと関数が複雑に絡み合い、勾配降下法をそのままでは利用できない

この状況を打開するのが誤差逆伝播法



To Be Continued...

【参考】ギリシャ文字一覧

文字	名称	文字	名称
α	アルファ	ν	ニュー
β	ベータ	ξ	グザイ
γ	ガンマ	o	オミクロン
δ	デルタ	π	パイ
ϵ	イプシロン	ρ	ロー
ζ	ゼータ	σ	シグマ
η	イータ	τ	タウ
θ	シータ	υ	ウプシロン
ι	イオタ	ϕ	ファイ
κ	カッパ	χ	カイ
λ	ラムダ	ψ	プサイ
μ	ミュー	ω	オメガ