

# ニューラルネットワーク

## PRML 輪読会 第 1 回資料

西本 洋紀  
教養学部学際科学科 B 群 2 年

2019 年 2 月 10 日

### はじめに

本稿は平成 31 年 2 月 10 日開催の PRML 輪読会第 1 回資料として作成した。  
筆者の理解では、第 5 章は、大きく 5 つの部分に分けられていると考える。  
まずはじめに、ニューラルネットワークの中でも特にフィードフォワードネットワークに限定して話を進める。線形回帰／分類モデルとの比較において、フィードフォワードネットワークの関数の形を見る。  
次に、ネットワークパラメータを決定する問題を扱う。最尤推定の枠組みからこの問題を議論する。これには、非線形化最適化問題をどう解くかという問題も含まれる。その後「誤差逆伝播」について確認し、この方法がヤコビ行列、ヘッセ行列に対して拡張されることを見る。  
次に、ニューラルネットワークの正規化への様々なアプローチと互いの関係について確認する。ここまでニューラルネットワークの中でも特にフィードフォワードネットワークに限定した話であったが、次にフィードフォワードネットワーク以外のニューラルネットワークに話題を転じる。特に混合密度ネットワークという、条件付き確率分布をモデル化する枠組みについて扱う。  
最後に、ベイズ理論の立場からニューラルネットワークについて議論する。

### 用語についての確認

本題に入るまでに、本稿で用いられる用語の中で注意が必要ないくつかのものについて、意味を確認しておく。

- ニューラルネットワーク

本稿の表題となっているこの語は、生体システムにおける情報システムを数学的に表現しようという試みにその語源がある。しかし、本稿では、「パターン認識という実的な応用の観点からは、生物学的な現実性などは全くの不要な制約である」という、PRML 著者の見解に

倣い、統計的パターン認識のための効率的なモデルとしてのニューラルネットワークを扱うこととする。

- 多層パーセプトロン (Multilayer perceptron) 後述するフィードフォワードネットワークの別名としても知られ、実用面で最も高く評価されている特別なニューラルネットワークである。本稿の前半では、ニューラルネットワークの中で特にこのネットワークに限定して説明を進める。しかし、多層パーセプトロンは実際は (不連続な非線形性をもつ) 複数のパーセプトロンではなく、(連続的な非線形性を持つ) ロジスティック回帰モデルを多層にしたものなので、この呼び方は本当はあまり適切でない。以降、本稿では、フィードフォワードネットワークに呼び名を統一して説明を進めることとする。

## 1 フィードフォワードネットワーク関数

この章では、3 章、4 章で議論した線形回帰/分類モデルとの違いに着目しながら、フィードフォワードネットワークの関数の形を見ることとする。まず、3 章、4 章で議論した線形回帰/分類モデルは下のように定式化されるのであった。

$$y(\mathbf{x}, \mathbf{y}) = f(\mathbf{w}^T \phi(\mathbf{x})) \quad (1)$$

ここで、 $\phi(\mathbf{x})$  は基底関数である。回帰問題では  $f$  は恒等関数、分類問題では  $f$  は非線形活性化関数である。学習の際は、固定された基底関数  $\phi(\mathbf{x})$  に対し、最適な  $\mathbf{w}$  (の確率分布) を決める問題を解くのであった。

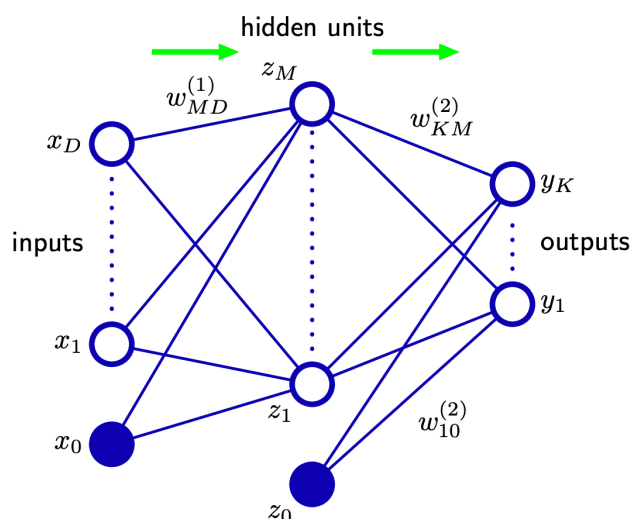
これらのモデルは、解析や計算においては有用な性質を持つが、次元の呪いのために実際的な応用可能性は限られる。このようなモデルを大規模な問題に適用するためには基底関数をデータに適応させる必要があるが、その 1 つのアプローチとして、基底関数に対してパラメトリック形を用い、そのパラメータ値を訓練中に適応させる、フィードフォワードネットワークが知られている。これは次のように定式化される。ただし、 $f$  は出力層の関数に相当し、回帰問題の時は恒等関数、2 クラス分類の時はロジスティック関数、他クラス分類の時は Softmax 関数を用いる。問題の形によって出力層の関数が増えるという点は、後で本稿 2 節で議論する。

$$y(\mathbf{x}, \mathbf{y}) = f\left(\sum_{j=0}^M w_{kj}^{(2)} h\left(\sum_{i=0}^D w_{ji}^{(1)} x_i\right)\right) = f\left(\sum_{j=0}^M w_{kj}^{(2)} \phi_j(\mathbf{x})\right) \quad (2)$$

この式を見ると、基底関数に対してパラメトリック形が用いられている様子がよくわかるのではないだろうか。実際、学習の際は、基底関数の線型結合のパラメータのみならず、基底関数を決定するパラメータも決定する問題を解くことになる。

下図は、上式 (2) に相当する 2 層ニューラルネットワークの図である。入力変数、隠れ変数、出力変数はノードとして表され、重みパラメータは、入力変数  $x_0$  と隠れ変数  $z_0$  を追加し、そこからのリンクの重みとして表されている。この図において、基底関数の線型結合のパラメータは、出力層と隠

れユニットをつなぐリンクの重みづけに相当し、基底関数を決定するパラメータは、隠れユニットと入力層をつなぐリンクの重みづけに相当している. 上式 (2) に従って  $y$  を評価する過程は、ネットワークの情報の順伝播と解釈することができる.



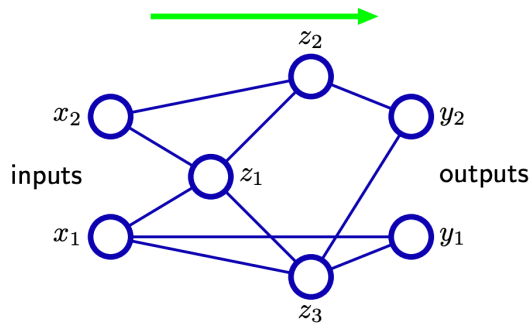
## ネットワーク構造の一般化

上図に示すネットワーク構造はごく普通に使われるものであるが、これは簡単に拡張される。例えば、

$$a_k = \sum_{j=1}^M w_{kj}^{(2)} z_j + w_{k0}^{(2)} \quad (k = 1, \dots, K) \quad (3)$$

のように重み付き和を計算し、それを非線形活性化関数を用いて変換するようなユニットからなる層を複数追加することなどが可能である.

もう1つのネットワークの構造の一般化は、層を飛び越えた結合である (下図参照). 下図ではわかりやすさのため省略されているが、隠れユニットと出力層はそれぞれバイアスパラメータを持つことに注意が必要である. ネットワーク図は数学的な関数と直接対応しているので、より複雑なネットワークを考えることで、より一般的な写像に発展させることができる. しかしながら、出力が入力の決定論的な関数であることを保証するために、有向閉路を持たないフィードフォワード構造である必要がある.



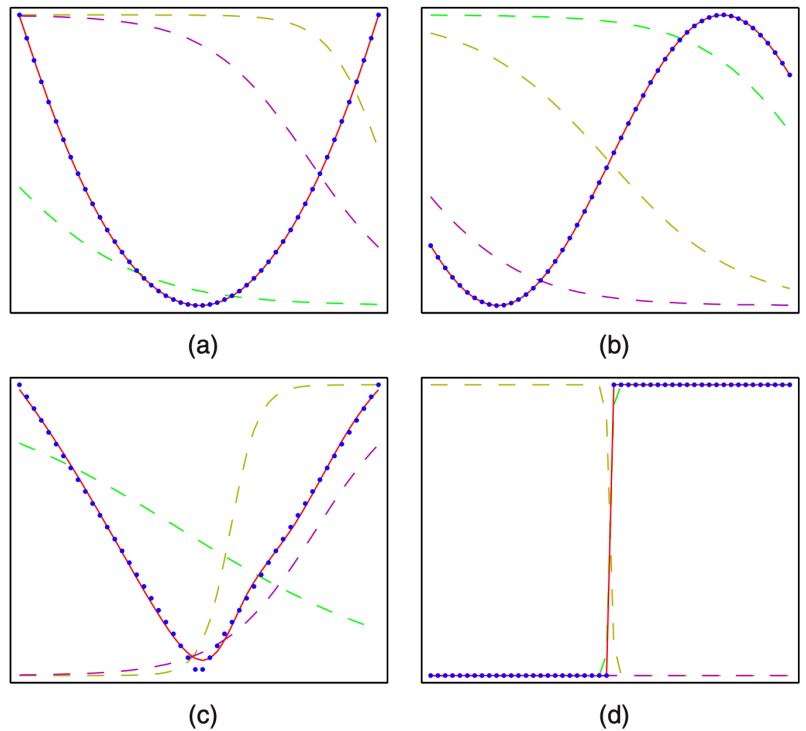
このようなネットワークの隠れユニットあるいは出力層は

$$z_k = h\left(\sum_j q_{kj} z_j\right) \quad (4)$$

で与えられる関数を計算する. ここで、和はユニット  $k$  へ接続を持つ全てのユニットに対してとる (バイアスパラメータも含む). 与えられたネットワークの入力に対して、式 (4) を順番に当てはめることで、ネットワークの活性を出力層を含めて評価することができる.

### 1.1 フィードフォワードネットワークの優れた関数近似特性を示す例

**Figure 5.3** Illustration of the capability of a multilayer perceptron to approximate four different functions comprising (a)  $f(x) = x^2$ , (b)  $f(x) = \sin(x)$ , (c),  $f(x) = |x|$ , and (d)  $f(x) = H(x)$  where  $H(x)$  is the Heaviside step function. In each case,  $N = 50$  data points, shown as blue dots, have been sampled uniformly in  $x$  over the interval  $(-1, 1)$  and the corresponding values of  $f(x)$  evaluated. These data points are then used to train a two-layer network having 3 hidden units with 'tanh' activation functions and linear output units. The resulting network functions are shown by the red curves, and the outputs of the three hidden units are shown by the three dashed curves.



## 2 ネットワーク訓練

ここまで、ニューラルネットワークを入力変数  $\mathbf{x}$  から出力変数のベクトル  $\mathbf{y}$  へのパラメトリックな非線形関数の一般的なクラスとしてみてきた。次に、隠れユニットのパラメータや、出力層のパラメータの最適化をどのように行うかについて議論する。

### 2.1 最尤推定

#### ネットワーク出力の確率的な解釈

最尤推定によるネットワークパラメータの最適化を考える前に、ネットワーク出力の確率的な解釈を考える。これにより、出力層の非線形活性化関数の選択と誤差関数の選択に、明確な誤差関数の選択に、明確な理由づけができるようになる。

- 回帰問題

任意の実数値を取れる 1 つの目標変数  $t$  を考える。  $t$  は  $\mathbf{x}$  に依存する平均を持つガウス分布に従うことにする。ただしその平均は

$$p(t|\mathbf{x}, \mathbf{w}) = N(t|y(\mathbf{x}, \mathbf{w}), \beta^{-1}) \quad (5)$$

のように、ニューラルネットワークの出力で与えられる。式 (5) で与えられる条件付き分布に対しては、出力層の活性化関数は恒等写像をとれば十分である。というのは、そのようなネットワークで任意の  $\mathbf{x}$  から  $y$  への連続関数を近似できるからである。 $N$  個の独立同分布に従う観測値  $\mathbf{X} = \{x_1, x_2, \dots, x_N\}$  とそれぞれに対応する目標値  $\mathbf{t} = \{t_1, t_2, \dots, t_N\}$  からなるデータ集合が与えられた時、尤度関数は

$$p(\mathbf{y}|\mathbf{X}, \mathbf{w}, \beta) = \prod_{n=1}^N p(t_n|\mathbf{x}_n, \mathbf{w}, \beta) \quad (6)$$

と表される。負の対数を取ると誤差関数

$$\frac{\beta}{2} \sum_{n=1}^N \{y(\mathbf{x}_n) - t_n\}^2 - \frac{N}{2} \ln \beta + \frac{N}{2} \ln(2\pi) \quad (7)$$

が得られ、これを用いてパラメータ  $\mathbf{w}$  および  $\beta$  を学習することができる。 $\mathbf{w}$  について最大化する。尤度関数を最大化することは、加法および乗法定理の違いを無視すると

$$E(\mathbf{w}) = \frac{1}{2} \sum_{n=1}^N \{y(\mathbf{x}_n, \mathbf{w}) - t_n\}^2 \quad (8)$$

で与えられる二乗和誤差関数の最小化と等価である。 $E(\mathbf{w})$  を最小化する  $\mathbf{w}$  は最尤推定の会に相当するので、 $\mathbf{w}_{ML}$  と表すことにする。現実には得られるのは尤度の極大点 (誤差関数の極

小点に相当)である。このことは次節で議論する。 $\mathbf{w}_{ML}$ が求まれば、 $\beta$ の値は負の対数尤度を最小化することで得られ、

$$\frac{1}{\beta_{ML}} = \frac{1}{N} \sum_{n=1}^N \{y(\mathbf{x}_n, \mathbf{w}_{ML}) - t_n\}^2 \quad (9)$$

と与えられる。

回帰関数の場合、ネットワークは恒等写像の出力活性化関数  $y_k = a_k$  を持つとみなせ、それに対応する二乗和誤差関数には

$$\frac{\partial E}{\partial a_k} = y_k - t_k \quad (10)$$

という性質がある。これは誤差逆伝播を議論するときに利用される。

- 2クラス分類問題
- 多クラス分類問題
- まとめ出力層の活性化関数および対応する誤差関数は、解くべき問題の方によって自然に選択される。回帰問題においては、線形出力関数および二乗和誤差を使い、2クラス分類問題では、ロジスティックシグモイド関数および交差エントロピー誤差関数を使い、そして他クラス分類問題においては、ソフトマックス関数およびそれに対応する他クラス交差エントロピー誤差関数を使う。2クラス分類問題においては、単一のロジスティックシグモイド関数を使うこともできるし、ソフトマックス活性化関数を持つ2出力のネットワークを使うこともできる。

## パラメータ最適化

### 局所2次近似

### 誤差逆伝播

### ヤコビ行列

### ヘッセ行列

## 3 ニューラルネットワークの正則化

## 4 混合密度ネットワーク

## 5 ベイズニューラルネットワーク