

# ニューラルネットが「分かる」とは 暴力的ではないアプローチ

伊達 章<sup>1,a)</sup>

## 概要：

人工知能技術への期待は大きい。Github の出現、それを利用したオープンな開発環境、とくに Google 社、Facebook 社、Preferred Networks 社などから deep learning 関連のフレームワークとよばれる環境の無償提供、これにより現実社会の問題に対し機械学習を容易に試すことができるようになった。「容易に」とは言っても、いろいろなことを「学ぶ」必要はある。ここ数年、大量のソフトウェアエンジニアが機械学習の勉強をはじめたと思われる。ここでは、人工知能技術の核として、今後利用されそうなニューラルネットについて、それを学ぶ際の分かりにくさを考える。社会の人工知能技術への期待とは別に、研究者としては危機感もある。それについても最後に述べたい。

キーワード：連想記憶モデル、自己組織化モデル、数理脳科学、構成性システム

## 1. はじめに

Deep Learning が注目を集めている。2016年11月現在、人工知能・機械学習関連書籍がひっきりなしに出版されている。Github 上には、Python や Javascript など書かれた機械学習関連のソースコードが大量に置かれており、大勢のソフトウェアエンジニアが、ニューラルネット（神経回路モデル）や機械学習技術を学習し利用している。

私は第2次ニューロブーム後期（1987年～1996年）に学生時代を過ごし、神経回路モデルを卒業論文、博士論文の研究テーマとした。ニューラルネットには、教師あり学習、教師なし学習、連想記憶モデルなど、いくつかの典型的な要素モデルがある。私は各数理モデルを体系的に学んだ方だと思うが、自己組織（教師なし学習）のモデルを理解

するには時間がかかった、ある程度、納得するまで2年はかかったのではないだろうか。ニューラルネットのような並列分散型システムの挙動は、数式だけで理解することは困難である。たとえば、連想記憶モデルを知らない人に、一定の理解を短時間で要領よく伝えるのは至難の技である。脳の記憶の書き込み・読み出しをモデル化した数理モデルの振る舞いにもかかわらず、日常生活で、よく仕組みのわかっている別のものに例えることができればよいが、そういうものもない。他のニューラルネットも、少なくとも、私自身は、ゼロからコンピュータプログラムを書いて走らせて、ようやく納得することが多かった。実際に走らせて、モデルのパラメータをいろいろ変え、何度か実験して、ようやく「分かった」という実感が得られる。最近では、やさしそうに記述された解説書が出版されているが、学習者が納得のいく理解にたどり

<sup>1</sup> 宮崎大学工学部情報システム工学科

<sup>a)</sup> date@cs.miyazaki-u.ac.jp

着くのは、簡単ではないはずである。

夏のプログラミング・シンポジウム 2016 は、テーマ「教育・学習」として開催された。本稿では、ニューラルネットの分かりにくさについて考えてみる。まずはじめに、いくつかの典型的な要素モデルについて紹介し、大学で実施しているニューラルネットの講義、およびその演習課題に対する学生の反応を紹介する。

サブタイトルに「暴力的でないアプローチ」と、発表申込時に手が滑って書いてしまった。気にしてもらえないものではない。個人研究者が、Google に所属する人達と同じ場で競争しても到底かなわない。一方、たとえば Google の Tensorflow, Preferred Networks の Chainer などを利用し、研究者でなくても、機械学習を現実社会の問題解決に簡単に応用できるようになった。こういった現状を単に「暴力的」と言ったままである。個人研究者が、計算機パワーを使わない、大規模データも使わない、古典的な手法で、太刀打ちできることは何かを最後に議論したい。

## 2. ニューラルネットの要素モデル

ニューラルネットには、いくつかの典型的な要素モデルがある [1]。一つは、パーセプトロンを代表とする教師あり学習。これは、入力とそれに対する望ましい出力のペアが多数与えられることから、「例からの学習」[27] とよばれている。Deep Learning の問題設定もこれにあたる。次に、教師なし学習。これは教師信号が与えられない。外界の信号だけを見て、その時間・空間的な規則性を抽出、内部（脳）に写し取ることが目的である。これは自己組織ともいう。三番目は、連想記憶とよばれるモデルである。これは学習後の、興奮ダイナミクスが研究対象とされる場合が多い。ほかにもボルツマンマシンや強化学習がある。

大部分のニューラルネットについては、以下のようなイメージを持っておけばよいだろう。

- (1) 活動のダイナミクス。これは「思考」に対応。個々の素子（ニューロン）は、他の素子の活動を重み付きの和で受け取り、自分が興奮するか否かを決定する。

- (2) 結合のダイナミクス。これは「学習」に対応。「重み」は（活動のダイナミクスとは比較にならないくらい）ゆっくり変化する。学習のダイナミクスとも言う。学習がローカル（局所的）とは、 $j$  番目から  $i$  番目の素子への結合荷重  $w_{ij}$  の変化分が、 $j$  番目の素子の活動と  $i$  番目の素子の活動にのみ依存する学習法則のこと。誤差逆伝搬法は、ローカルな学習とは考えられていないが、「活動」をそれぞれの素子がなんらかの形で持つ「信号」と思えば、誤差逆伝搬法 [1,27] もローカルな学習である。

- (3) 大量の素子が並列に動作する。

別の言い方をすれば、単純な Yes, No の意思決定をする人間が多数存在する世界、とイメージしておけばよい。

ニューラルネットの各モデルの問題設定は、入学試験問題のような短い文章では表現できない。これが分かりにくさの原因の一つになっている。以下では、いくつかのモデルの問題設定を記述してみる。これがなかなか難しい。細かい条件や、どんな回路を使うかも省いてしまっているものもある。最大・最小化問題に帰着できれば分かりやすいが、そのように記述されるモデルは少ない。問題設定をきちんと記述するとこの文章の 3 倍は必要だろう。そうすると、今度は問題設定を理解するのが難しくなる。こういうジレンマがある。

### 2.1 例からの学習（教師あり学習、パーセプトロンなど） [1,27]

入力とそれに対する望ましい出力（教師信号）のペア  $(\mathbf{x}^1, \mathbf{y}^1), (\mathbf{x}^2, \mathbf{y}^2), \dots, (\mathbf{x}^m, \mathbf{y}^m)$  が多数 ( $m$  個) 与えられているとする。ここで  $(\mathbf{x}, \mathbf{y})$  はある確率分布  $\Pr(\mathbf{x}, \mathbf{y})$  にしたがっているとする。このとき  $\mathbf{x}$  を入力すると、 $\mathbf{y}$  を出力するような回路を作りたい。目的は、与えられた例題に対してだけではなく、望ましい出力が与えられていない未知の入力  $\mathbf{x}$  に対しても、正しい答え  $\mathbf{y}$  を出力できる回路を作ることである。「問題の本質」がどこにあるのか分からない場合に使える。

## 2.2 自己想起型の連想記憶モデル（物事の断片から全体を想起）[1, 15, 16]

あらかじめ  $m$  個の記憶パターン  $\mathbf{x}^1, \mathbf{x}^2, \dots, \mathbf{x}^m$  が与えられているとする。ここで  $\mathbf{x}$  は各要素が  $1, -1$  の値をとる  $n$  次元ベクトルとする。あるパターン  $\mathbf{x}_0$  を回路の初期状態として与え、状態更新を繰り返すと、あらかじめ記憶しておいた  $m$  個のパターンのうち、 $\mathbf{x}_0$  にもっとも近い記憶パターンが想起されるようにしたい。

それには記憶パターンが回路の安定平衡状態になるように回路のパラメータ（結合係数）を設定すればいい。たとえば、

$$W = \frac{1}{n} \sum_{\alpha=1}^m \mathbf{x}^{\alpha} (\mathbf{x}^{\alpha})^{\top} \quad (1)$$

のように設定しておけば、 $m$  が小さければ、とりあえずはよい（ $\mathbf{x}$  は縦ベクトルで、 $\top$  は転置を意味する。 $W$  は  $\{w_{ij}\}$  を要素とする行列）。

## 2.3 自己組織のモデル、ニューロンが必要に応じて外界の環境に適応

入力  $\mathbf{x}$  が、ある確率分布  $\Pr(\mathbf{x})$  にしたがって与えられる。各素子は  $\mathbf{x}$  および他の素子の状態に依存して、興奮するか否か定まる。各素子が興奮する入力信号の集合を受容野とよぶ。学習は、素朴な Hebb 学習（2つの素子が同時に興奮する頻度に応じて、それらを結ぶ結合の重みが大きくなる）がよく使われる。これを続けた場合、各入力  $\mathbf{x}$  に対し、どのような表現が自己組織的に形成されるだろうか [1]。この問題は、目的が分かりにくい。以下では、目的の分かりやすいモデルを紹介する。

### 2.3.1 Kohonen の自己組織化マップ

各入力  $\mathbf{x}$  に対し、出力素子（ノード）のうち一つだけが興奮するような回路を設計する。もちろん、単にどれかが興奮すればいいというものではない。入力  $\mathbf{x}$  に対し、興奮する素子を勝者素子と呼ぶ。 $c$  番目の素子が興奮したとすると、この回路は、入力に対し勝者素子に対応づける写像  $f: \mathbf{x} \rightarrow c$  を実現する。各素子は、入力空間に参照ベクトル  $\mathbf{m}_i$  を持っており（ $\mathbf{m}_i$  は  $i$  番目の素子の参照ベクトル）、それが変化するつれて  $f$  も変化する。 $f$  がトポロジカルマップになるよう、参照ベクトルを修

正（学習）することが、学習の目的である。トポロジカルなマップとは：

- (1) 1つの入力  $\mathbf{x}$  に対し、1つのノードが対応する。
- (2) 1つのノードに対し、信号空間内の1つの狭い領域が対応する。
- (3) ノード上で隣り合ったノードに対応する領域は（おおむね）信号空間内で隣り合っている。

### 2.3.2 ボルツマンマシン [4, 9, 20, 21, 24, 28]

一口に言うと、外の世界のコピーを自分（脳）の中に作成することが目的である。入力（「外の世界」に対応）を  $\mathbf{x}$ 、出力（「脳内」に対応、隠れ素子ともよばれる）を  $\mathbf{h}$  としよう。ボルツマンマシンは、同時確率分布  $p(\mathbf{x}, \mathbf{h}; W)$  のことと思えば話がはやい。外からは、ある確率分布  $q(\mathbf{x})$  にしたがって、次々に信号  $\mathbf{x}$  がモデルに与えられる。ボルツマンマシンは自分で自分（ $W$ ）を変えていく。これを学習という。どういう状態になったら、学習が完了したと言えるのか。ボルツマンマシンは確率的に動作するので、入力が与えられなくても動作（状態を遷移）し続ける。このとき、ボルツマンマシンが入力層に実現する確率分布

$$p(\mathbf{x}; W) = \sum_{\mathbf{h}} p(\mathbf{x}, \mathbf{h}; W) \quad (2)$$

がある。これを  $q(\mathbf{x})$  と一致させることが学習の目的である。 $W$  を一つ発見できればよい。

学習則も記述しておこう。具体的に  $p(\mathbf{x}; W)$  を  $q(\mathbf{x})$  に近づけていく学習則が知られており

$$w_{ij} := w_{ij} + \epsilon(f_{ij} - \tilde{f}_{ij}) \quad (3)$$

で与えられる。ここで  $\epsilon$  は微小な正の定数、 $:=$  は更新、 $\tilde{f}_{ij}$  はすべてのニューロンを「自由に」動作させたときの  $i$  番目と  $j$  番目の素子の活動度の積の平均値、 $f_{ij}$  は入力層を  $q(\mathbf{x})$  にしたがって選ばれたパターン  $\mathbf{x}$  に固定し、隠れ層のニューロン  $\mathbf{h}$  を自由に動作させたときの平均値である。

## 2.4 確率的生成モデル [19]

ベイズ推論では、あらかじめ対象の規則性を反映した確率分布を構築しておく。観測データ  $\mathbf{y}$  に対する解釈用の内部（隠れ）変数を  $\mathbf{x}$  とすると、これは確率分布  $\Pr(\mathbf{x})$  と  $\Pr(\mathbf{y}|\mathbf{x})$  の設計にあたる。

$\Pr(\mathbf{y}|\mathbf{x})$  をデータモデルと呼ぶ。例えば、音声認識の場合では  $\mathbf{y}$  は音声信号の生データ、 $\mathbf{x}$  は音素や単語などの各概念である。このようなモデルは確率的にデータ  $\mathbf{y}$  を生成する能力を持つため確率的生成モデルと呼ばれる。このモデルはデータ  $\mathbf{y}$  を観測し、 $\mathbf{x}$  を推定することに利用できる。例えば、入力  $\mathbf{y}$  に対し事後確率最大推定量

$$\mathbf{x}_{\text{MAP}} = \arg \max_{\mathbf{x}} \Pr(\mathbf{x}|\mathbf{y}) \quad (4)$$

$$= \arg \max_{\mathbf{x}} \frac{\Pr(\mathbf{y}|\mathbf{x})\Pr(\mathbf{x})}{\Pr(\mathbf{y})} \quad (5)$$

$$= \arg \max_{\mathbf{x}} \Pr(\mathbf{y}|\mathbf{x})\Pr(\mathbf{x}) \quad (6)$$

を求める最適化問題として定式化できる。ベイズ推論の本質は事後確率分布  $\Pr(\mathbf{x}|\mathbf{y})$  の利用にある。

### 3. ニューラルネットの講義

脳はどのような風に動いているのか。その気分を味わうことができるのが神経回路網モデルの醍醐味である。数理モデルを学習する際、はじめにその振る舞いを計算機シミュレーションにより大雑把に掴んでおくと、数学的な性質についても理解が進みやすい。神経回路モデルの場合、数年前までは Matlab が教育に限らず研究においても使いやすいため利用されていたが、価格が高いため誰でも利用できるわけではなかった [7]。最近では Python (Numpy, Scipy, Matplotlib) を使う環境が整い、この問題はほとんど解消されたように思う。とは言っても、学生が大学のプログラミング演習で習っているのが C 言語であるので、Python は、大部分の学生にとっては、敷居が高いようである。

#### 3.1 講義の進め方

著者が現在、大学院生向けにおこなっている講義「数理脳科学」では、自己組織化マップ、連想記憶モデルに続き、ボルツマンマシン、隠れマルコフモデルによる情報処理（確率的生成モデル）の話をしている。一つの要素モデルにつき、おおよそ講義3回分を使っている。まずは、板書しながらの講義。次に、ゼロからプログラムを書いている\*1（ゼロから書いていく様子を見せて

もいる。「問題設定」を理解していないと、「私は何をしているのだろうか」状態になるため、これを回避するため、問題設定の理解度を口頭試問している（この作業は学生に嫌がられる）。

以前は、テンプレート（C 言語のコード）を用意し、いろいろパラメータを変えて走らせてもらっていた [7]。時間が無い場合には仕方がないが、ゼロからコードを書いてもらうと、あたりまえであるが、よく分かった気になってもらえる。

#### 3.2 提出されたレポートから

以下、レポートに記述されていた感想をいくつか抜粋して紹介する。

##### 3.2.1 連想記憶モデル

- (1) 正直、先生の授業を聞くよりも課題を解いている時のほうが「へえ…連想記憶モデルってこういうものなのか」と思えました。
- (2) 最初こそまったく理解が追いつきませんでした。C 言語でプログラムを組んだ回で一気に理解することができました。
- (3) はじめて連想記憶モデルというものに触れたが、仕組みが分かればアルゴリズムも簡単であり、この分野について非常に興味を持って学習できた。しかし、初めて説明を受けた段階では、プログラムに起こすためのイメージがほとんどわからなかった。特にモデルの入出力が分からず、学習の結果何が出てくるのかが分からなかった。

##### 3.2.2 自己組織化マップ

- (1) 単純な学習更新式のみで、ここまで分かりやすく学習が進むのはとてもおもしろいと感じた。
- (2) 今回の自己組織化モデルは非常に興味深いと感じた。学習が進むに連れて自己組織化の過程を観察するのはとてもおもしろかった。

##### 3.2.3 ボルツマンマシン

- (1) ボルツマン機械とは何なのか、何をするための機械なのか、この機械は何をすることができるのか。最初はまったくわかりませんでした。講義での話と、実際に C 言語のプログラムで実装して動かしてみた結果、少し理解が深まったかなと感じた。（中略）また、条件

\*1 [https://github.com/date33cs/mathematical\\_neuroscience](https://github.com/date33cs/mathematical_neuroscience)

付き確率を知らず知らずのうちに学習していることにも驚かされた。ダイバージェナスが正常に収束するようにバグをつぶしながら修正する作業がとて長く（5～6時間かかりました）感じましたが、その苦勞の成果はあったと思います。

- (2) 今までボルツマンマシンについて、説明を見ても何をしているのかよく分からなかったのが、今回実際に実装してみて、やっと理解できました。

個々の神経回路網モデルの振る舞いを理解するための手っとり早い方法は、自分で短いプログラムを書いて、走らせてもらうことである。モデルを記述するパラメータの変化にしたがい、モデルの振る舞いがどう変化するのも楽しめる。

### 3.3 ボルツマンマシンの不思議さ

まだ問題がある。数ヶ月の間にこのような内容を次々に講義するため、それぞれが、どう関わっているのか関わっていないのか、学習者にとっては全体像を捕えるのが難しい。これは例えば、次のようなことである。自己組織化モデルは、入力信号の例題を何度もモデルに提示することにより、モデル内部の結合係数などのパラメータを変化させていくものである。一方、連想記憶モデルの醍醐味は想起のダイナミクスである。ここで結合係数もしくは記憶させたいパターン群は、とりあえず与えられたものとして考える。したがって、講義では、自己組織化モデルと連想記憶モデルは別のものとして扱われる。ところがボルツマンマシンの場合、もっとも単純なものは対称結合回路であるので、回路自身の構造は自己想起型連想記憶と同一であり、当然、連想記憶モデルとしても使えるが、ボルツマンマシンは外界の（条件付き）確率分布を近似する自己組織化モデルと言える [21]。それに加えて、（全結合の）マルコフ確率場でもある。その上、実はボルツマンマシンは単純パーセプトロンの拡張であるとも見ることができる [20]。

こうなると何がなんだかわからなくなる。無用の混乱を避けるため、講義では、神経回路モデルのダイナミクスには、速さの違い、素子の活動

のダイナミクスと結合のダイナミクスがあることを強調するにとどめ、個々の神経回路モデルは別物という扱いにしている。では、その2つのダイナミクスは、どういう関係にあるのか。対称結合回路の場合はポテンシャル関数を通じて結合係数と活動状態が結び付いている。一般的な神経回路モデルについてはどう考えたらいだろうか。個々の神経回路モデルの学習の様子を幾何学的に理解することに成功している「ニューロ多様体の情報幾何学」[2,4]に、そのやさしい説明を期待したい。

### 3.4 名人芸

上記の感想にあるように、講義を聴いてもらっても、学生には理解できた気分にはなってもらえていない。個々の講義、例えば、自己組織化モデルについて、黒板だけの講義を聞いて、「入力信号をモデルに次々と与え、ある学習法則に基づき、モデルのパラメータを変化させていく。その結果、入力の確率分布にしたがった素子の割り当てがおこなわれる」という認識にたどり着くのは難しい。講義にはどれくらいの意味があるのだろうか。もう最初から手を動かしてもらったほうがいいのではないかとさえ思うこともある。

一方、世の中には古今亭志ん朝師のような名人がいる。長岡浩司先生が数学通信に寄稿された一文を紹介しよう [23]。『M1の最初の学期に、甘利先生の大学院の授業を受講した。先生の講義と言えば、聴いている最中は実に気持ちよく、本質を理解できたような気分になるのだが、あとでその内容を自分で再構成しようとしても手も足も出ない、という名人芸で有名である』。ある程度の短時間で「わかった」気にさせるのは「学び」にとっても重要である。この甘利先生の芸の本質はどこにあるのだろうか。圧倒的な理解の深さに裏打ちされていることは間違いないが、それだけではなさそうだ。

## 4. 考察

Tensorflow や Open CV を使い、研究者ではなくても、現実問題の解決が比較的容易にできるよ

うになった。ニューラルネット、コンピュータビジョン、音声認識の研究者は非常に危機感を持っている。もはや高価で特殊な道具を必要としなくなった。個人研究者が、計算機パワー・大規模データを使わないでもできる意味のある研究は何だろうか [3]。それは各個人によって違うだろう。私の考えを以下に紹介する。

#### 4.1 25年前の論文からの示唆

深層学習の主要な認識エンジンは多層の神経回路モデルであるので、手法自体は、本質的には、約25年前の第2次ニューロブーム時と、変わっていない [1, 22, 27]。当時 S. Geman らは神経回路モデルによる学習法を総括する論文で “learning complex tasks is essentially impossible without the *a priori* introduction of carefully designed biases into the machine’s architecture,” および “the identification and exploitation of the ‘right’ biases are the more fundamental and difficult research issues in neural modeling” ([13] の p.3) と結論している。ここまで強い主張ではないが、バックプロパゲーションを考案した D. E. Rumelhart も神経回路モデルを現実の問題に応用する際の注意として “Carefully design appropriate input data” ・ “Build known symmetries (often through weight linking) into your network where ever possible” ([26] の pp.195-196) などを挙げている。現在の技術進歩はコンピュータハードウェアの進歩、悪く言えば力任せで得られた結果であることは否定できない。「学習アルゴリズム」よりは「表現」が挑戦すべき課題である点は変わっていないように思う。「表現」とは知識の表現方法、データ構造のことである。

#### 4.2 ものごとの関係性をどう表現するか

1973年に出版された Duda と Hart によるパターン認識の教科書「Pattern Classification and Scene Analysis」[10]を見ると、全482ページのうち、261ページ以降が第2部「Scene Analysis」として構成されている。ところが、2000年に出版された第2版では、タイトルが「Pattern Classifica-

tion」になり、第2部は完全に削除されている。機械学習 (Machine Learning) に関する専門書は最近たくさん出版されているが、物事の関係性を生成規則などで記述する手法 (英語では grammatical, syntactical, structural, relational, discriptive などという言葉が使われる) は、自然言語処理を除き、教科書からは姿を消している。つまり、このような、いわゆる AI アプローチ [29] は、一部では本質的に失敗だったと思われる。これは証明されたわけではない (証明される類のものでもないが)。

1990年代後半、S. Geman は構成性の数学的理論を作った [14]。これは形式言語における従来の生成規則 (production rule) を、構成規則 (composition rule) という観点から見直し、数学的に定式化したものである。数学的にしっかり整備されているが、この応用例は Huang [17] の他には見あたらない。理由はいくつか考えられる。第一に生成規則を、それぞれの応用に合わせて設計者が手作りする必要がある。これはかなり面倒臭い。より単純化したモデルを Geman ら自身が提案している [18]。ここでも、対象を再帰的に表現するという知識表現が、確率分布を使い実現されている。概念や規則を自動化する方法も考えられてきており [25]、教科書的に姿を消した文法的方法も、確率分布を用いた手法で盛り返しつつある [6, 30]。画像理解の確固たる性能評価の基準はまだないが、単なる認識率から変わりつつある [11]。このあたりが今後の鍵になりそうだ [5, 8, 12]。

#### 4.3 記憶のデータ構造

脳の記憶のデータ構造とアルゴリズム (書き込み、読み出し、整理の仕方) には不思議が多い。記憶のデータ構造には階層的な構造がうまく収まっている (各カテゴリー、カテゴリー間の関係、それを時間空間に位置づけたエピソード)。また、このようなデータ構造をどう改変しているのだろうか。

「脳はどのような風に動いているのか。その気分を味わうことができるのが神経回路網モデルの醍醐味」と先に書いたが、数理モデルがないため、その気分が味わえない現象は結構ある。この「もやもや」をもとに、問題設定を作っていく定式化の

作業（これは応用数学！）こそがもっとも意味のある研究であろう。

謝辞 本研究の一部は文部科学省科学研究費補助金（基盤 C） 課題番号 26330280 による。

#### 参考文献

- [1] 甘利俊一：神経回路モデルとコネクショニズム，東京大学出版会（1989）。
- [2] 甘利俊一：ニューロ多様体の情報幾何学，数理科学，No. 340, pp. 61–65（1991）。
- [3] 甘利俊一：脳・心・人工知能：数理で脳を解き明かす，講談社（2016）。
- [4] Amari, S., Kurata, K. and Nagaoka, H.: Information geometry of Boltzmann machines, *IEEE Transactions on Neural Networks*, Vol. 3, pp. 260–271（1992）。
- [5] Anishchenko, A.: Two Alternative Models of Neuronal Connectivity: Committed Complex Cells and Small World Networks, PhD Thesis, Brown University（2006）。
- [6] Chua, J. and Felzenszwalb, P. F.: Scene Grammars, Factor Graphs, and Belief Propagation, <http://arxiv.org/abs/1606.01307>（2016）。
- [7] 伊達 章：gnuplot を用いた「自己組織化マップ」学習ダイナミクスの理解，電子情報通信学会技術研究報告，NC2005-15（2005）。
- [8] 伊達 章，窪田 光，山田雄輔：構成性システムとその手書き文字認識への応用について，電子情報通信学会技術研究報告，PRMU2016-94（2016）。
- [9] 伊達 章，倉田耕治：ボルツマンマシンと自己組織化，*Clinical Neuroscience*, Vol. 34, pp. 885–888（2016）。
- [10] Duda, R. O. and Hart, P. E.: *Pattern Classification and Scene Analysis*, John Wiley & Sons, New York（1973）。
- [11] Geman, D., Geman, S., Hallonquist, N. and Younes, L.: Visual Turing test for computer vision systems, *Proceedings of the National Academy of Science U.S.A.*, Vol. 112, pp. 3618–3623（2015）。
- [12] Geman, S.: Invariance and selectivity in the ventral visual pathway, *Journal of Physiology-Paris*, Vol. 100, pp. 212–224（2006）。
- [13] Geman, S., Bienenstock, E. and Doursat, R.: Neural networks and the bias/variance dilemma, *Neural Computation*, Vol. 4, pp. 1–58（1992）。
- [14] Geman, S., Potter, D. F. and Chi, Z.: Composition systems, Technical report, Division of Applied Mathematics, Brown University, Providence, RI 02912 USA（1998）。
- [15] Hopfield, J. J.: Neural networks and physical systems with emergent collective computational abilities, *Proceedings of the National Academy of Science U.S.A.*, Vol. 79, pp. 2554–2558（1982）。
- [16] Hopfield, J. J., Feinstein, D. I. and Palmer, R. G.: ‘Unlearning’ has a stabilizing effect in collective memories, *Nature*, Vol. 304, pp. 158–159（1983）。
- [17] Huang, S.-H.: Compositional Approach to Recognition Using Multi-Scale Computations, PhD Thesis, Brown University（2001）。
- [18] Jin, Y. and Geman, S.: Context and hierarchy in a probabilistic image model, *Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '06)*, pp. 2145–2152（2006）。
- [19] Künsch, H., Geman, S. and Kehagias, A.: Hidden Markov random fields, *Annals of Applied Probability*, Vol. 5, pp. 557–602（1995）。
- [20] 倉田耕治：神経回路モデルとしてのボルツマン・マシン，数理科学，No. 289, pp. 23–28（1987）。
- [21] 倉田耕治：ボルツマン・マシンを応用したトポグラフィック・マッピングの形成モデル，電子情報通信学会技術研究報告，MBE87-154, pp. 393–398（1988）。
- [22] 倉田耕治，麻生英樹：神経回路網の理論的研究における最近の動向，電子情報通信学会論文誌，Vol. J73-D-II, pp. 1103–1110（1990）。
- [23] 長岡浩司：甘利俊一先生の2012年度文化功労者表彰に寄せて，数学通信，pp. 10–14（2013）。
- [24] 長岡浩司，小嶋徹也：統計的モデルとしてのボルツマンマシン，計算機統計学，pp. 61–81（1995）。
- [25] Ommer, B. and Buhmann, J.: Learning the compositional nature of visual object categories for recognition, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-32, pp. 2501–516（2010）。
- [26] Rumelhart, D. E.: Theory to practice: A case study-recognizing cursive handwriting, *Proceedings of the 3rd NEC Research Symposium*, SIAM, Philadelphia, PA, pp. 177–196（1993）。
- [27] Rumelhart, D. E., Hinton, G. E. and Williams, R. J.: Learning representations by back-propagating errors, *Nature*, Vol. 323, pp. 533–536（1986）。
- [28] Sejnowski, T. J., Kienker, P. K. and Hinton, G. E.: Learning symmetry groups with hidden units: Beyond the perceptron, *Physica*, Vol. 22D, pp. 260–275（1986）。
- [29] 白井良明：人工知能とはなにか，岩波書店，東京（1985）。
- [30] Si, H. and Zhu, S.-C.: Learning AND-OR templates for object recognition and detection, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-35, pp. 2189–2205（2013）。

質疑・応答

久野 暴力的なアプローチというのは？

伊知地 暴力的というのは、大量のデータを、コンピュータのパワーだけで解決しよう。こういう解釈でいいか。

伊達 はいそうです。

美馬 「わかった」というのは？「数式のフォローができた」、「その世界が見えた」、「わかった」にもいろんなレベルがある。プログラミング課題を通して分かった、といのはフォローできただけ。というレベルとっていいか。

伊達 はいそうです。

飯尾 モデルの数式もわかった、プログラムもわかった。教科書に掲載されているのと同じような図が描けた、それで本当にニューラルネットのモデルをちゃんと理解したと言えるのだろうか。

伊達 「ちゃんと」と言われるとそれは怪しい。レポートの考察に、こちらが講義中に説明していない事柄、実験して気づいた事柄が書いてあることが多いので、それで理解度は把握できる。

※ ニューラルネット（特に、連想記憶モデルやボルツマンマシン、自己組織化マップ）は並列の相互作用で学習が進むので、コンピュータシミュレーションしてはじめて分かる事が多い。「emergent collective computational properties」(J.J.Hopfield の論文タイトルより)と呼ばれているものが、シミュレーションなしでは理解しにくい。もちろんモデルを発明した人は、これ（原理を記述した少数の数式）でこうなるはずだとシミュレーションなしで理解していたわけで、数式だけで並列相互作用の結果を予想できる（超）人がある。一方、確率的生成モデルをつかったパターン認識（事後確率最大化など）は、数式の理解だけで、（凡人でも）、その動作は理解できる。

伊知地 ニューラルネットは並列に動いている。それはきちんと数学的に書けているのか。（ボルツマンマシンが？）うまく動かないという話、数学的に表現できていないのではないか。

並列処理に関しては、新しい数学を作らないといけない。

伊達 逐次処理のコンピュータシミュレーションではあるが、並列の相互作用は記述できている。※ 同期して更新するか非同期にするか状態更新の仕方など、細かなバリエーションがあるが、そこが問題ではない。

伊知地 学生はわかったつもりになっているが、本当には理解していないのではないか。プログラムを書いて少し雰囲気は分かっただけではいまいか。

伊達 うーん。※ まずは、雰囲気を味わえるというのが重要。

竹迫 理論と実際のプログラムにはギャップがある。機械学習を勉強した際、数式から、数式の行間を読み取り、コンピュータで処理できるように、Python のコードを書き起こすことが難しかった。学部生は、そもそも数式の意味を理解するのが難しいので、機械学習を学部生に教えるのは結構難しいのではないか。

伊達 はい。※ Python で for 文を使わず一挙に処理するコードを書こうとするのは、C 言語にはない、別の意味の難しさもあるように思う。

伊知地 応用数学というなら、数学の拡張をしないといけない。既存数学で書いてはいけないのでは？

伊達 そうは思っていない。※ もやもやしている問題があって、それをきちんと数学的に定式化したい。そういう定式化する努力をしないといけない。ここで言った「応用数学」というのはそういうこと。問題の本質は高校数学レベルで記述できると思っている。現存するニューラルネットの基本的なモデルは1970年代までに出揃っていて、80年代にボルツマンマシンが提案されて以降30年間、それらに匹敵するような数理モデルがモデルが出現していない。流行には外れるけど、基本モデルを作る「暴力的でないアプローチ」をもっと進めよう、というのが言いたいこと。