

# Deep Learning 班活動報告書

立命館コンピュータクラブ 2017 年度 後期グループ活動  
Deep Learning 班

—Leader—

前川彰

—B1—

青木雅典 伊藤聡子 玄元奏 程瑞希  
藤原亜也佳 正岡玲於奈 吉田享平 稲垣和真

—B2—

井上諒也 大山航平 川上雄大 杉岡宏樹  
田中良樹 藤田朱門 湯浅健生 吉本裕貴

—B3—

川端諒 福井尚卿 奥村泰久

2018 年 2 月 5 日

## 概要

本プロジェクトは RCC2017 年度後期プロジェクト “Deep Learning 班” である。ここ数年の Deep Learning の活躍に対しては様々な立場があるが、社会的に大きな影響を与えていることに関してはどの立場からも異論はないだろうと考えられる。そのため本プロジェクトでは「Deep Learning を実装できる人材を増やす」ことを目的に、入門に最適と高く評価されている書籍 “ゼロから作る Deep Learning” を週一回の頻度で輪講を行った。参加人数は当初の想定を大きく上回るメンバー数となったため火曜水曜金曜の三つにチームを分割した。プロジェクト活動の具体的内容としてチームごとの活動内容を紹介した。また一部のメンバではあるが書籍の内容を基礎に Deep Learning の数学的妥当性を検討する論文 “Mathematics of Deep Learning” を熟読してもらい、解説を試みた。

なお本報告書には Deep Learning そのものに関する解説はほとんど載せていない。Deep Learning を学ぶための教材や機会は既に各所で得られるためである。その代わりに、参考文献として、メンバが価値があると思った論文や資料をあげている。時流に乗ることを恐れない諸氏に当たられては是非ともこの分野に果敢に挑戦していただきたい。

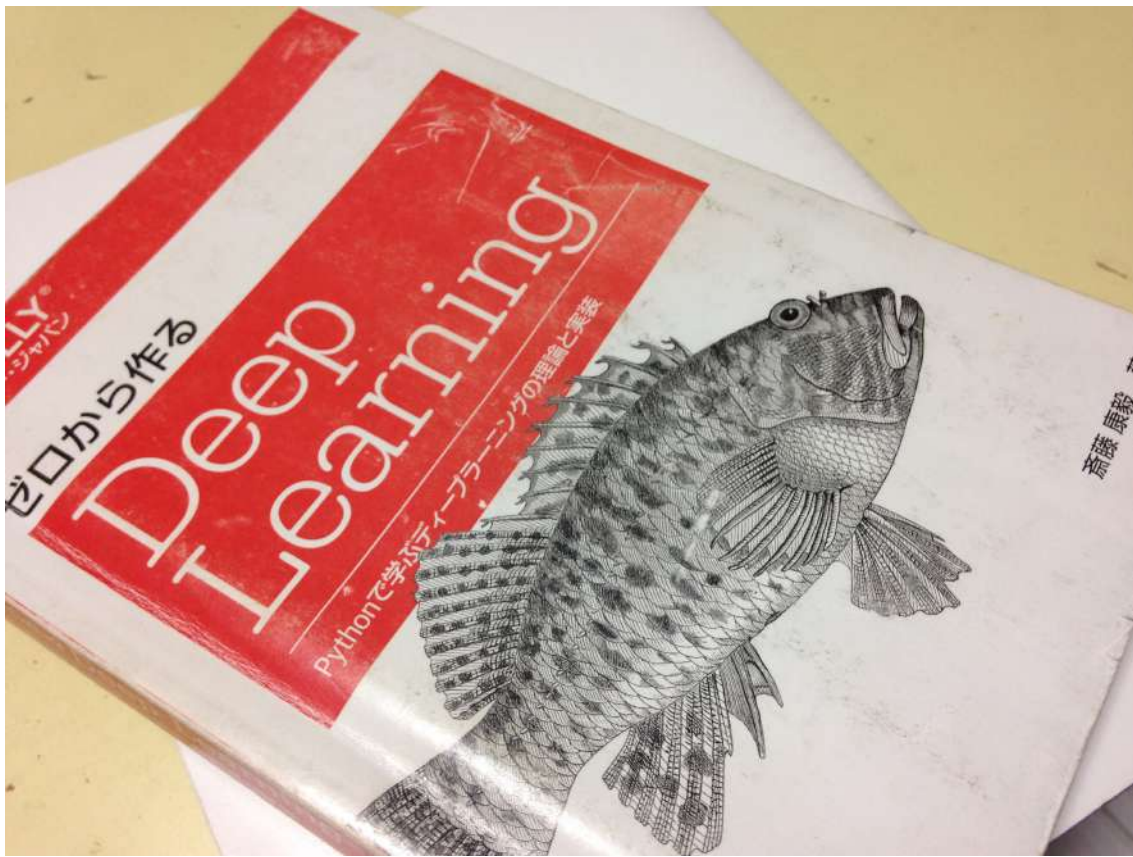


図1 ゼロから作る Deep Learning

## 目次

## 1 活動概要

### 1.1 活動目的

文責：大山 航平

近年は三度目の人工知能 (AI) ブームが到来していると言われており、様々な分野で AI が実用化され出している。このブームの発端には Deep Learning という機械学習手法の確立が大きく関わっており、コンピュータ囲碁プログラム「AlphaGo」を初めとする話題の AI の多くは DeepLearnig によって実現されている。Deep Learning がこれからの AI 時代に欠かせない技術となることは、もはや間違いない。本プロジェクトは「Deep Learning を実装できる人材を増やす」ことを目的として発足し、活動を行なった。

### 1.2 用いた本の紹介

文責：奥村 泰久

輪講には斎藤康毅著の「ゼロから作る Deep Learning Python で学ぶディープラーニングの理論と実装」という本を用いた。この本のコンセプトはディープラーニングの理論について深く理解することである。そのために理論をただ勉強するだけでなく、実際にディープラーニングを作成することで理解を深めようとしている。ディープラーニングを作成するのはライブラリを使えば簡単にできるが、この本ではその使用は最小限にしている。これは、ディープラーニングをゼロから作ることで理論を深く理解することを目的としているためである。

またこの本の画像を図??で示している。Deep Learning を勉強する上での良書であるため、学びたいと思っている方でまだ手に取られていない方には是非おすすめする。

## 2 活動内容

2017 年十月の終わりから年末にかけて火曜水曜金曜の各班に別れて輪講を行った。メンバは各班ごとに異なったため、その活動は各回ごとに特色が出ている。そのため、以下活動を各曜日ごとに活動内容を小節に分けて示す。また、その際の活動風景画像を以下に二枚紹介する。

$f(x) = ax + b$   $x+y$

①  $f(x+y) = a(x+y) + b = ax + ay + b$

$f(x) + f(y) = (ax + b) + (ay + b) = ax + ay + 2b$

$f(x+y) \neq f(x) + f(y)$

...非線形

$h_2(x) = cx^2$

$h_2(h_2(x)) = h_2(cx^2)$

$= c(cx^2)^2$

$= c^3 x^4$

$x = \begin{pmatrix} x_1 & x_2 \\ x_3 & x_4 \end{pmatrix}$

$f(x) = ax$  ← 線形

$f(A+B) = a(A+B) = aA + aB = f(A) + f(B)$

$f(kA) = a(kA) = k(aA) = k f(A)$

$5V = aI$

図2 活動風景 線型性の説明を試みている図

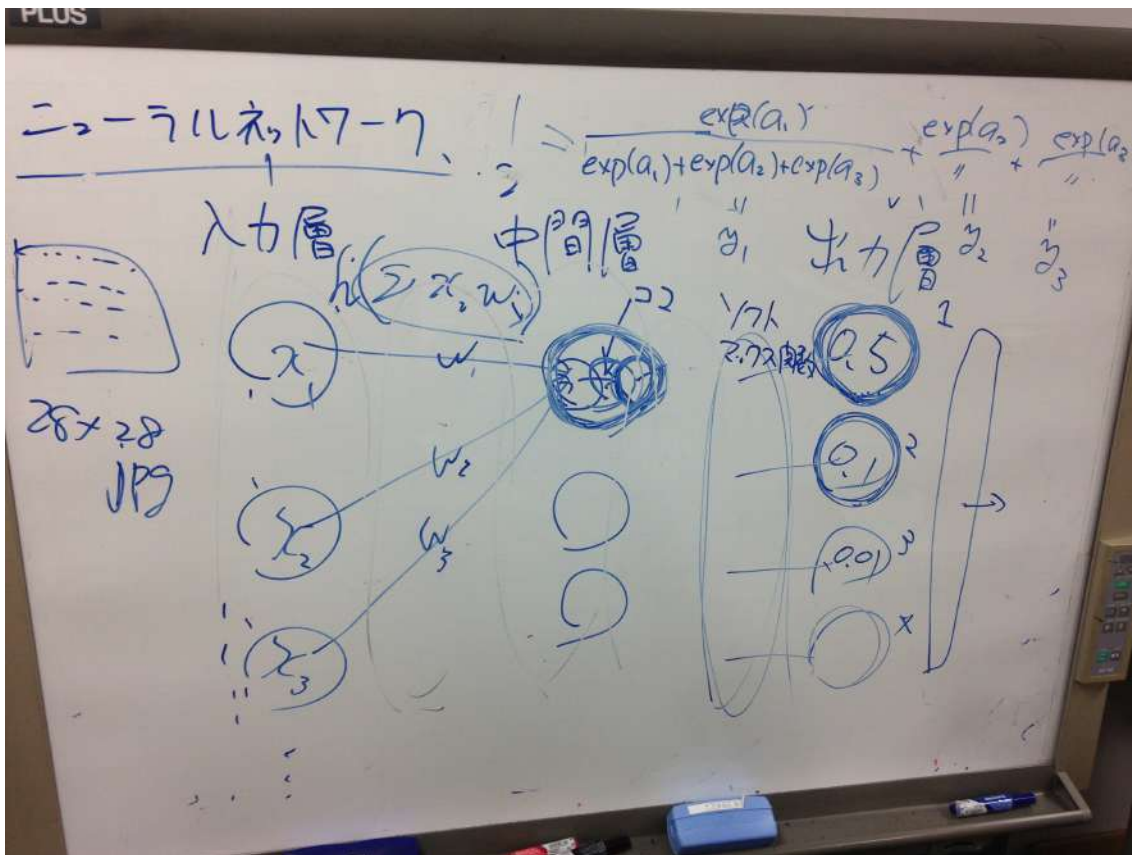


図3 活動風景 ニューラルネットワークの説明を試みている図

## 2.1 火曜日班活動

文責：稲垣 和真

火曜日班は全部で六回の活動を行なった。以下にその活動内容を各回ごとに簡単に示す。

第一回は「ゼロから作る Deep Learning が python を用いての記載をしていたので、python の環境構築から始めた。使うライブラリは主に Numpy と matplotlib だった。次に、活性化関数の扱い方や線形性についての学習、Numpy を用いた内積計算の手法やニューラルネットワークの実装方法などの基本的な部分を学習した。

第二回はパーセプトロンについて学んだ。この部分はいわゆる true か false を返す部分で、次の層に伝えるか否かを判定するのに使われる。パーセプトロンは深層学習において基本単位であり、かつ極めて重要な部分であるため、じっくり時間をかけて理解に努めた。

第三回は新たな活性化関数であるソフトマックス関数や恒等関数について学習した。また、新たに損失関数であるエントロピー（誤り情報）や二乗和誤差について学習した。また、なぜ損失関数を使うかなどの本質的

な疑問の解決なども行なった。

第四回は、勾配法や二層ニューラルネットの実装、誤差逆伝播などを学習した。また、誤差逆伝播を用いる理由や計算グラフの利用法なども学んだ。

第五回は深層学習において重要な重みの初期値について学習した。適切な初期値の設定は、深層学習の重大なミスである過学習の防止に繋がるので、重点的に学習した。  
第五回は内容がかなり複雑であったため、重点的に理解に努めた。

第六回は過学習が起こる原因や防止法、CNN(convolutional neural network) について学習した。

## 2.2 水曜日班活動

文責：杉岡 宏樹

水曜日班では合わせて五回の活動を行った。事前に割り振られた箇所の理解を深め、当日にはその内容を内容を発表する方針で Deep Learning の理解を深めあった。以下に活動内容を各回ごとに簡単に示す。

初回はパーセプトロンについて学習を行った。AND, OR, NAND, XOR とパーセプトロンの類似性を学んだ。

第二回はニューラルネットワークについて学習を行った。ニューラルネットワークとは何か、それに用いる活性化関数とは何かについて有名なものを例に挙げ議論を行った。

第三回は損失関数や勾配について学習を行った。学習を行う際には評価基準について考慮しなければならない。認識精度と損失関数という二つの尺度を比較して、議論を行った。

第四回は Deep Learning の学習方法や計算グラフについて学習を行った。

最終回は計算グラフの続き、活性化関数や学習に関するテクニックについて学んだ。学習に関するテクニックは SGD を始めとするパラメタの更新方法や適切な初期値の設定方法に関して学ぶことができた。

## 2.3 金曜日班活動

文責：藤原 亜也佳

金曜班は全部で三回の活動を行った。以下にその活動内容を各回ごとに簡単に示す。

第一回目はパーセプトロンについて学習した。パーセプトロンとは入力と出力をもつアルゴリズムであり、バイアスや重みを調節することで AND,OR,NAND の論理回路を実装することができる。さらに二層のパーセプトロンを用いることで XOR ゲートの作成も可能で、このように層を重ねたものを多層パーセプトロンと

いう。多層パーセプトロンを利用することでコンピュータが行う任意の処理を表現することができる。

第二回目はニューラルネットワークについて学習した。ニューラルネットワークには入力層、隠れ層、出力層があり各層には活性化関数がある。活性化関数は入力信号の総和を出力信号に変換する関数であり、以下の五種類が有名である。いずれも非線形関数であり、層を深める上でその非線形性は重要となる。

- シグモイド関数：0 を境にして出力が 0 から 1 に連続的に切り替わる
- ステップ関数：0 を境にして出力が 0 から 1 に非連続的に切り替わる

これらは二つの活性化関数は入力が大きければ出力は 0 に近く、入力が小さければ出力は 1 に近くなる

- ReLU 関数：入力が 0 以下なら 0 を、0 を超えていればその値をそのまま出力する
- 恒等関数：入力をそのまま出力する。回帰問題で用いられる。
- ソフトマックス関数：出力の総和が 1 になる。分類問題に用いられる。

第三回は勾配法、誤差逆伝播法について学習した。勾配とは各点において関数の値を最も減らす方向であり、勾配の方向に何度も変化させることで関数の値を減らす手法を勾配法という。誤差逆伝播法を理解するために計算グラフを用いた。実際に計算グラフを用いて解いてみることで、時間はかかったものの、問題を単純化でき効率よく計算ができることが確認できた。

### 3 発展的活動

文責：玄元奏 川端諒

発展的課題として「Mathematics of Deep Learning」の論文の要旨と目新しい用語を以下にまとめた。この論文は以下の要旨にあるように Deep Learning の数学的妥当性を複数の視点から検討したものである。やや難解であるが以下に要旨の訳と用語の解説を示していく。

#### 3.1 Abstract(要旨)

近年、表現学習や分類のための深層設計の導入によって認識システムの劇的な発展が起きている。しかしながら、これらの躍進の裏側にある数学的な根拠は未だ理解しにくいままである。このチュートリアルは大域最適性や幾何的安定性や表現の不変性などの深層学習における、いくつかの要素に対する数学的な正しさを提供する目的で最近の業績を見直すものである。

#### 3.2 Global optimality(大域最適性)

最適化とは損失関数の値をできるだけ小さくするパラメータを見つけることである。データの範囲が広い場合、最小値はある範囲を定めてその範囲内で求めるが、定めた範囲外も含めるとさらに最小値が見つかることもある。前者（図の f, h 点）を局所的最小値、後者（図の b, d 点）を大域的最低値という。

#### 3.3 Geometric stability(幾何的安定性)

この論文を読む限り安定性の定義がわからなかった。論文を読むかぎり、安定性とは「幾何的に微小な変化があったとしても同様のものとしてみなす性質をもつこと」と捉えることができる。そこで安定性の定義はこ



のようなものであるとみなして、以下説明をする。例えば画像データを扱う際、画像を画像として扱うのではなく

$$X : (0, 1) \times (0, 1) \rightarrow \mathbf{R}^2$$

なる関数  $X$  として扱う。この関数を、作用素やノルムや確率測度と呼ばれる関数の性質を調べる数学的な道具を用いて解析的に分析する。ここで一つ例を用いて説明する。1辺の長さが  $\frac{1}{2}$  の正方形の画像をディープラーニングで識別することを考える。正方形を二変数関数で表そうとするならば次のようになる。

$$X(x, y) = \left(\frac{1}{2}x, \frac{1}{2}y\right), (x, y) \in [0, 1] \times [0, 1]$$

データセットにあるすべての1辺の長さが  $\frac{1}{2}$  である正方形のような形をした画像は上記の式と同様に二変数関数の式で表すことができる。そこでデータセットに入ってる画像を全て関数で表し、その集合を  $D := \{X_1, X_2, X_3, \dots\}$  とする。先ほど述べた作用素やノルムと呼ばれる道具を用いて  $D$  の各元にたいして解析していく。そうしてえられる結果の一つは論文によると次の通りである。

$$|f(\mathbf{L}_\tau(X)) - f(X)| \approx \|\nabla f\|$$

この式を日本語で説明するならば、「関数がほんの少し違ったところで同じ関数だと認識してくれる」ことを意味する。先ほどの理論だと、画像を関数という数学の言葉に変換していた。そこで、逆に関数を画像という語句に入れ替えて先ほどの式の意味をもう一度考え直してみると次のようになる。「画像がほんの少し違ったところで同じ画像だと認識してくれる」。「画像がほんの少し違った」、すなわち、「データの幾何的性質がほんの微小量変化」しても許されるということである。すなわち、数学を用いて幾何的性質は安定しているということを示した。

### 3.4 Invariance of the learned representations(表現の不変性)

深層学習の手法の一つには畳み込みニューラルネットワーク (以下 CNN) というものがある。CNN には畳み込みとプーリングという二つの層が含まれており、畳み込みは行列に対するオペレータのようなもので入力データにフィルタをかけて縦と横方向の空間を小さくする処理である。プーリングも入力データの空間を小さくするという点では同じであるが、こちらは範囲内の最大値 (または平均値や最小値) を取り出して入力データの特徴を抽出するという違いがある。プーリングでは局所領域からフィルタをかけて特徴を抽出するため、入力データに微小なズレが生じていてもその特徴を検知することができるので特徴を検知する対象がデータのどこにあっても検知することができる。これを移動不変性という。

## 4 活動目的に対する達成度および活動を通して得られたもの

文責：福井 尚卿

今回のプロジェクト活動では「ゼロから始める Deep Learning」を輪読するという形で Deep Learning に関して理解を深めることができた。どの班も第五章まで読み進めることができ、一つの班は本の最終目標としていた七章まで読了することができた。しかしながらディープラーニングの基本的な考え方を学ぶことができたものの、実装まで試みるのができた者は一部であり、当初の目的である「Deep Learning を実装できる人材を増やす」ことが十分に達成できたとは言い難い。とはいえ、今後班員が Deep Learning を学ぶに当たって不

自由しない基礎的な知識を身につけることができたように感じる。さらに一部の班員は英論文”Mathematics of Deep Learning”に取り組むといった発展的課題をこなすことができた。

当初は環境の実行環境のセットアップに手こずっていたメンバもいたが、活動内で最低限環境は作り上げることができた。そうしたセットアップといった基礎的な知識や、今まで漠然と耳にしていたであろう関連語句に対する知識も多く得られたため、今後自分たち自身でさらなる学習を進められることを期待している。

## 5 大人数プロジェクトに関する知見

文責: 前川 彰

本プロジェクトでは大人数での活動となってしまったが、活動する曜日を分けそれぞれの曜日にリーダーを定めることで一部の班が機能停止することなく活動を行うことができた。今後、今回のような大人数のプロジェクトが結成される可能性があるため、気をつけるべき点を五つだけあげておく。

- できる限り人数の単位を小さくし、各単位には信頼のおける人を配置する
- 最終目標とその過程での短期目標をはっきりさせ、達成不可能であることがわかれば柔軟に対応する
- メンバには明確なタスクを設定する
- “分からない”“ことに対して過剰に反応しない
- 少しの遅刻等にくじらを立てすぎない

大人数プロジェクトはその人数の多さゆえに目が行き届かなくなる。その際にリーダーが強権的にプロジェクトを当然立ち行かなくなることが想定できる。そのため単位ごとに小リーダー的な存在を配置し、目標が達成できるかななどを調整してもらえると良い。またある程度の決定権も渡しておく尚良い。

またプロジェクト内での不和を起こさず、なおかつ十分な議論できるようにするために、心がけるべき事がある。それはあるメンバが”わからない”と発言した際にその発言をバカにしたり軽くあしらわない事である。疑問を退けるのは議論をする上で無意味であり、健全な議論を行う上で最も恥ずべき行為である。これだけは気をつけてほしい。これは状況によるが、少しの遅刻も多めに見るほうが良い。これは単純に余裕を持って行動するほうが良き思考が働くと考えるからである。

## 6 おわりに

文責: 前川 彰

本プロジェクトは当初四、五人を想定していたが二十人を超えるメンバが参加してくれることとなった。リーダーの時間的身体的制約により全ての活動を把握できたわけではないが、メンバが主体となって活動を行うことができた。活動時の活発な議論はすでに学んだことのある者とない者の間で分け隔てなく行うことができた。どの参加者にとっても意義のある者になっていれば、幸いである。

来年度以降も「IoT で得られたビッグデータをディープラーニングでいい感じに解析して！」というあまりにも馬鹿げた言説が社会を席捲し続けるであろう。賢い諸氏はこうしたいわゆるバズワードに辟易としてしまい、距離を置いてしまうこともあると思う。しかしながらあえてそこにある技術の本当の意味や力を見抜き、パラダイムの変化を捉え、社会を先導するような存在に皆がなってくれることを願ってやまない。最後にリーダーが主観で好きな言葉を引いて締めくくる。

十分に発達した科学技術は、魔法と見分けがつかない.\*<sup>1</sup>

我々は、技術について短期的な影響を高く見積もりすぎ、長期的な影響を低く見積もりすぎる.\*<sup>2</sup>

## 参考文献

- [1] ゼロから作る Deep Learning Python で学ぶディープラーニングの理論と実装, O'REILLY, 齋藤康毅 著
- [2] Mathematics of Deep Learning, Rene Vidal, Joan Bruna, Raja Giryes, Stefano Soatto, <https://arxiv.org/abs/1712.04741>
- [3] 定番の Convolutional Neural Network をゼロから理解する, [https://deepage.net/deep\\_learning/2016/11/07/convolutional\\_neural\\_network.html](https://deepage.net/deep_learning/2016/11/07/convolutional_neural_network.html), 2016-11-7 記 2018-2-5 最終閲覧
- [4] TensorFlow 大量の画像から学習するには・・・～(ほぼ) 解決編～, @ikki8412, <https://t.co/eeGvI7XnzO>, 2015-12-21 最終更新, 2018-2-5 最終閲覧
- [5] [PReLU]Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification, Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun, <https://arxiv.org/abs/1502.01852>
- [6] Adam: A Method for Stochastic Optimization, Diederik P. Kingma, Jimmy Ba, <https://arxiv.org/abs/1412.6980>
- [7] Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift, Sergey Ioffe, Christian Szegedy, <https://arxiv.org/abs/1502.03167>
- [8] [LSTM]Long Short-Term Memory, Sepp Hochreiter, Jrgen Schmidhuber, <https://www.mitpressjournals.org/doi/abs/10.1162/neco.1997.9.8.1735>
- [9] He の初期値, Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification, Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun, <https://arxiv.org/abs/1502.01852>
- [10] [GAN]Generative Adversarial Networks, Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, Yoshua Bengio, <https://arxiv.org/abs/1406.2661>

---

\*<sup>1</sup> [Arthur Charles Clarke's third law]"Any sufficiently advanced technology is indistinguishable from magic."

\*<sup>2</sup> [Amara's law] "We tend to overestimate the effect of a technology in the short run and underestimate the effect in the long run."