

演習で学ぶ

MATLAB による ディープラーニング

吉富 康成 編著

浅田 太郎

田伏 正佳 共著

横山 友也

コロナ社

【本著ご利用にあたって】

MATLAB および Simulink は、MathWorks 社の登録商標です。そのほかの商標のリストについては、www.mathworks.com/trademarks を参照ください。

また、カバーならびに表紙に記載されている MathWorks Publisher ロゴは、本著の内容に MATLAB® の内容が含まれることを示しており、MathWorks 社の許可を得て使用しております。

本ロゴは、MathWorks 社が本著のテキストや演習の正確性を保証するものではありません。

一方で、本著は MathWorks 社による MATLAB® およびその関連製品の特定の使用法を承認・推奨するものではありません。

なお、本著では®や TM は省略しています。

まえがき

本書は大学生，大学院生または企業の AI 実務担当（予定）者が学ぶディープラーニングの実践的入門書である。大学教養科目としての数学を基礎知識としてもち，プログラミングの経験があると，理解しやすい。しかしながら，これらの素養がなくても，その不足分を補うに足る勉強意欲があれば，実践力が育まれるように工夫して書いた。

大学・大学院での演習，または企業の研修での教科書・参考書として利用いただくことを想定しているが，独習用にも活用できるように配慮した。

ニューラルネットワークの研究は，神経生理学からスタートした。本書の執筆時点は，その第3発展期にあたる。本書で取り扱うディープラーニングは，多層のニューラルネットワークによる機械学習手法であり，ニューラルネットワークの研究の歴史の中で，特筆すべき成果である。

ニューラルネットワークは人工知能の一形態と考えられており，近年，ディープラーニングが AI の代名詞となってきた。本書は，初心者が実践力を身に着けることを優先して書いたので，ディープラーニングに至るまでの先人たちの貢献については，紙面の多くを割くことはできなかった。

筆者らは，研究における軌跡を記すというより，むしろ AI の活用を促進したいという思いから本書を企画した。MATLAB は，科学技術計算用のツールとして利用されており，ディープラーニングも対象としている。本書の執筆時点では，ディープラーニングを行うとき，プログラミング言語として Python を使われる方が多いと思われる。それでは何故，ディープラーニング入門のツールとして，MATLAB を取り上げたかということ，たんに使いやすかったからである。バージョンアップも毎年行われており，ヒントや例題も豊富で，サポート体制も完備されているため，ブラックボックス的色彩が拭えないディープラーニングの世界に，初心者が足を踏み入れるのに都合がよい。

また，MATLAB から Python のプログラムを呼び出したり，Python のプログラムから MATLAB を呼び出したりすることも可能とされているので，Python でのプログラムと経験の蓄積を MATLAB 使用時にも有効活用できる。

ディープラーニングに関する書籍はすでに数多く出版されているが、初心者がその分野に足を踏み入れるには敷居が高いきらいがある。そこで、ツールを演習で使用しながらディープラーニングを体得する進め方が、初心者の勉学意欲を喚起すると考えて本書を企画した。

まず1章で、ディープラーニングに至るニューラルネットワークの源流を概観し、2章で、具体例を踏まえて、ディープラーニングの基本的ネットワークの構造を把握してほしい。3章では、ディープラーニングの世界を俯瞰するのに役立つMATLABでのチュートリアル、例題をQ&A付で紹介している。ここまですが、演習をこなすための準備である。そして、4章の演習を通じて、ディープラーニングの世界に自ら足を踏み入れてほしい。

4章の演習問題で提供している解答例のプログラムは、Windows 11上のMATLAB R2022aの環境で動作確認をしている。4章の演習問題4.1.3, 4.1.4については、MATLAB R2023aで解答例を作成した。本書の解答例のプログラム、本書に関する画像データなどはコロナ社のホームページ上にてダウンロードできる（目次の末尾をご確認ください）。

4章の演習問題を解くためにMATLAB本体以外にToolboxを用いる必要が生じる場合は、そのToolbox名が、実行できることを必ず確認するようにと、指示している参考例に記されている。ただし、演習問題4.1.3, 4.1.4については、参考例はない。しかし、これらの問題を解くためには、Computer Vision Toolboxを用いる必要がある。

本書の執筆にあたり多くの方々にお世話になった。本書はMathWorks社のブックプログラムによるご協力の基に作成したものである。MathWorks社の皆様にこの場を借りて感謝申し上げる。

本書を読まれて、ディープラーニングを「よし！使ってみよう！使いたい！」という気持ちになっていただければ、無上の喜びある。

2024年1月

著者を代表して 吉富康成

目 次

1 章 ニューラルネットワークの基礎

1.1 形式ニューロンとシナプスの可塑性	2
1.2 パーセプトロン	6
1.3 誤差逆伝搬学習法	11
1.4 ネオコグニトロン	17
引用・参考文献	18

2 章 ディープラーニングの基礎

2.1 AlexNet	22
2.1.1 ネットワークの構造と学習済みの事例	22
2.1.2 主な構成要素	23
2.1.3 転移学習の利用方法	28
2.2 GoogLeNet	29
2.2.1 ネットワークの構造と学習済みの事例	30
2.2.2 主な構成要素	31
2.2.3 転移学習の利用方法	33
2.3 LSTM	35
2.3.1 ネットワークの構造	35
2.3.2 主な構成要素	35
2.3.3 適用方法	40
引用・参考文献	40

3 章 MATLAB によるディープラーニングの予備知識

Question 一覧	44
3.1 チュートリアル Q&A	47
3.2 例題 Q & A	50
3.2.1 共通	50
3.2.2 例 1：深層学習を使用した Web カメラ イメージの分類	55
3.2.3 例 2：新しいイメージを分類するための深層学習ネットワークの学習	56
3.2.4 例 3：イメージ分類用の残差ネットワークの学習	60
3.2.5 例 4：深層学習を使用した時系列予測	61

4 章 MATLAB によるディープラーニング演習

4.1 画像認識	65
4.2 LSTM	85
4.3 モデルの調整	92
演習問題のヒント	96
演習問題の解答例	103
演習問題の解答例の解説	196
索引	208
英訳索引	211

※ 本著の解答例のプログラムや、本書に関する画像データなどは下記 URL および二次元コードより入手できる。また、本著に記載した URL および二次元コードはすべて 2024 年 1 月現在のものである。

<https://www.coronasha.co.jp/np/isbn/9784339029420/>

解凍用パスワードは **P95** をご確認ください。



1章 ニューラルネットワークの基礎

ニューラルネットワーク (neural network) における神経細胞 (ニューロン, neuron) の機能のモデル化には

- ① ミクロな立場 神経細胞内の電気パルス (electrical pulse) の発生, その伝搬機構などの性質をモデル化
- ② マクロな立場 神経細胞のミクロな立場にとらわれずに, ニューラルネットワークを構成する情報処理の要素として神経細胞をモデル化

がある。

本書においては, 実践的入門書を志向しているので, 後者の視座で執筆した。本章では, 2~4章でのディープニューラルネットワーク (deep neural network) の理解に必要なニューラルネットワークの基礎について述べる。ディープラーニングは人工知能 (AI: artificial intelligence) の一形態であり, 機械学習 (machine learning) にも属すが, 本書の性質上, AI および機械学習を俯瞰する構成はとっていない。

ゴール

形式ニューロン (formal neuron), シナプスの可塑性 (synaptic plasticity), パーセプトロン (perceptron), 誤差逆伝搬 (back propagation) 学習法, ネオコグニトロン (neocognitron) とディープニューラルネットワークとの関連性を理解する。

1.1 形式ニューロンとシナプスの可塑性

神経細胞の観察を基にして、ニューラルネットワークの基礎となる神経細胞の機能のモデル化が行われた。神経細胞の構造を図 1.1 に示す。神経細胞は、細胞体 (soma) から突き出した樹状突起 (dendrite) および他の神経細胞へ伸びた軸索 (axon) からなる。軸索と他の神経細胞との結合部分をシナプス (synapse) という。

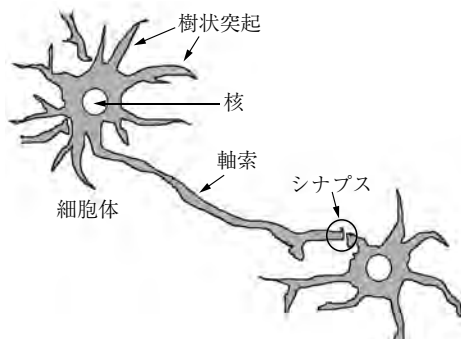


図 1.1 神経細胞

細胞体は細胞膜 (cell membrane) で囲まれており、中央には核 (nucleus) がある。シナプスの部分で、樹状突起または細胞体で他の神経細胞からの入力を受け、細胞体で信号処理し、軸索を通して他の神経細胞へ出力を出している。

シナプスにおいて、出力を出す側の神経細胞であるシナプス前神経細胞 (presynaptic neuron) の軸索の末端は、入力を受ける側の神経細胞であるシナプス後神経細胞 (postsynaptic neuron) の樹状突起または細胞体に、非常に狭い間隔で向かい合っている。

細胞体の働きは、機能的にはしきい値作用 (threshold effect) である^{1)†}。このしきい値作用は、細胞膜のイオン透過性の変化から生じている。他の神経

[†] 肩付き番号は章末の引用・参考文献を示す。

細胞からの入力を受けて細胞膜のイオン透過性が高まり，その結果として細胞膜の膜電位（membrane potential）がしきい値より高くなれば[†]出力を出す[‡]が，しきい値以下であれば出力を出さない。

このしきい値作用が，機能として，入力を出力に非線形変換する効果を有している。また，シナプスには細胞膜の膜電位を高める興奮性シナプス（excitatory synapse）と低くする抑制性シナプス（inhibitory synapse）がある。さらに，複数のシナプスからの入力が加算的に膜電位へ影響を与えている。

神経細胞の機能において，ニューラルネットワークの数理モデルを作る際の要点は

- ① 入力を出力に変換する際のしきい値作用
- ② 入りに正，負の種類があること
- ③ 入力の加算性
- ④ シナプスの可塑性

である²⁾。シナプスの可塑性については本節の後半で記述する。

マッカロ（W.S. McCulloch）とピッツ（W. Pitts）は，これらの機能のうち①，②，③を，シナプス後神経細胞が n 個のシナプス前神経細胞 $k(k=1,2,\dots,n)$ から入力を受け，出力を出すモデルとして，式(1.1)～(1.3)で表される形式ニューロン（**図 1.2**）で表現した³⁾。

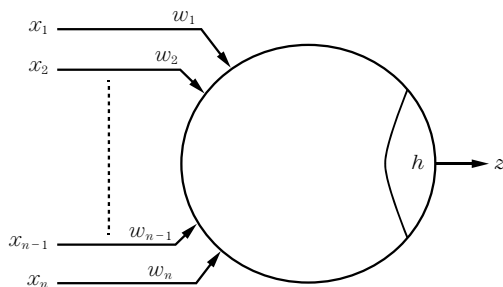


図 1.2 形式ニューロン

[†] この状態を，神経細胞が興奮した（excite）または発火した（fire）という。

$$y = \sum_{k=1}^n w_k x_k \quad (1.1)$$

$$f(u) = \begin{cases} 1, & u > 0 \text{ のとき} \\ 0, & u \leq 0 \text{ のとき} \end{cases} \quad (1.2)$$

$$z = f\left(\sum_{k=1}^n w_k x_k - h\right) \quad (1.3)$$

ここで、 $x_k \in \{0, 1\} (k=1, 2, \dots, n)$ は、シナプス後神経細胞が受け取るシナプス前神経細胞 k の出力、 y はシナプス後神経細胞の膜電位の変化量、 h は興奮のしきい値、 w_k はシナプス後神経細胞とシナプス前神経細胞 k とのシナプス結合荷重 (synaptic weight)、 z はシナプス後神経細胞の出力である。

このモデルでは、(1) 着目したシナプス後神経細胞とシナプス前神経細胞 k の位置関係や各神経細胞の構造、(2) シナプスの構造や機能などの影響因子で決まるシナプス後神経細胞とシナプス前神経細胞 k の間の信号の伝わり方を、シナプス結合荷重 w_k で総括的に表現している²⁾。

このモデルが、ディープニューラルネットワークを含めたニューラルネットワークの基礎になっている。形式ニューロンでは、シナプス後神経細胞の膜電位の変化を、シナプス前神経細胞の出力 (x_1, x_2, \dots, x_n) の荷重和という形で単純化しており、この荷重和がしきい値を超えると、シナプス後神経細胞が興奮して、出力 1 を出す。

入出力関数[†] (output function) f として、式(1.2)および図 1.3(a)で表されるステップ関数 (step function) 以外にも、式(1.4)および図 1.3(b)で表されるシグモイド関数 (sigmoid function) が用いられるようになってきた。このシグモイド関数は、全領域で微分可能であり、常に有限な出力を出すのが特徴である。

$$f(u) = \frac{1}{1 + e^{-\varepsilon u}}, \quad \varepsilon > 0 \quad (1.4)$$

上記のモデル化に至る研究の軌跡を詳しく知りたい読者は、文献 2)~7) を読まれることをお勧めする。文献 2) で概観を掴むことができる。文献 3) が

[†] 活性化関数 (activation function) ともいう。

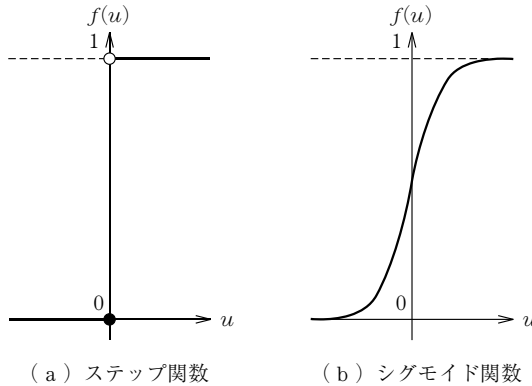


図 1.3 入出力関数の例

このモデルの提案であり、文献4)~7)には、基礎となる生物学、生理学の関係する知見が記されている。

ここで、シナプスの可塑性について説明する。脳で行われている記憶をつかさどる素過程として、ヘブ (D.O. Hebb) がヘブの学習則 (シナプス強化則) を提案した⁸⁾。形式ニューロンでいえば、式(1.1)におけるシナプス結合荷重 w_k が時間的に変化することで記憶が蓄えられるという仮説である。

さらに具体的に説明すると、この仮説は、「神経細胞 A の出力が神経細胞 B に入力されたときに神経細胞 B が興奮すると、神経細胞 A から神経細胞 B へのシナプス結合荷重が大きくなる」というものである⁹⁾。このシナプス結合荷重が入力との関係で変化することをシナプスの可塑性と呼ぶ。

この可塑性における塑性とは、「物体 (例えば粘土) に力を加えて変形させた後に、力を加えるのをやめても、元の形状に戻らない性質」をいう。シナプス結合荷重は、シナプスの機能をモデル化したものであるので、シナプスの生理学的性質が変化することがシナプスの可塑性の本質である。

ディープニューラルネットワークを含めたニューラルネットワークの数理モデルでの学習則は「ヘブの学習則」を基にしている。また、シナプス結合荷重は「シナプス」を省略して、単に「結合荷重」と表記している場合もある。

MATLAB の中では、「シナプス結合荷重」を「Weight」、興奮の「しきい値」

索引

【あ】

新たな各クラス 34
新たなクラス 29, 34

【い】

イオン透過性 2
位置選択性 18
イメージラベラー 69
陰 12

【え】

エポック 60, 83

【お】

オブジェクト 48
重み付きフィルター 24

【か】

回帰層 62
階層型ニューラルネット
ワーク 6, 7, 27
外側膝状体 17
核 2
学 習 8
学習機能 8, 25, 26
学習された畳み込み
フィルター 34
学習済みネットワーク 28
学習則 10
学習対象 8, 23, 25, 26, 31
学習データ 27
学習データの過適合 62
学習に用いたデータセットに
含まれないクラス 28
学習のしくみ 10
学習の収束 61

学習の発散 61
学習の評価関数 12
学習率 10, 59
拡張イメージ データストア 57

確率的勾配学習法 27
隠れ層 6
仮想的なニューロン 12
画像認識 30, 65
仮想ユニット 36
活性化関数 4, 23
活性化層 22
カテゴリ 23, 25
カーネル 24
感覚ユニット 7
慣性項 17

【き】

記憶機構 35
機械学習 1
教師あり学習 9, 12
教師信号 10, 12
教師データ 27
教師なし学習 9

【く】

クラウドコンピューティング 54
クラス 10, 18, 28, 29
クラス確率 57
クラス数 29, 34
クラススコア 23
グループ化された 26
グループ化された2次元畳み
込み層 22, 26
グループ別 26
クロスチャネル正規化層 22

【け】

形式ニューロン 1, 3, 4, 5
結合荷重 4, 5, 9, 23

【こ】

興奮性シナプス 3
興奮のしきい値 4
興奮のしきい値は可変 8
極小値 12
コグニトロン 18
誤差関数 12
誤差逆伝搬学習法 1, 7, 11,
13, 16, 21, 28, 33, 38
誤差逆伝搬学習法での計算
方法 14

【さ】

最急降下法 12
最小値 13
細胞体 2
細胞膜 2

【し】

視覚情報処理 18
視覚野 17
しきい値 3, 5, 6, 9, 23
しきい値作用 2
軸 索 2
シグモイド関数 4
次元圧縮 25
シーケンス 61
事前学習 28
事前トレーニング済み 23
時相内挿のアルゴリズム 71
実験マネージャー 55, 92, 93
実行ファイル 64

シナプス	2
シナプス強化則	5
シナプス結合荷重	4, 5, 8, 9
シナプス結合荷重は可変	8
シナプス後神経細胞	2
シナプス前神経細胞	2
シナプスの可塑性	1, 3, 5
シナプスの生理学的性質	5
樹状突起	2
出力ゲート	38
出力層	6
出力層文脈ユニット	37
出力ユニット	36
受容野	17
順伝搬型ネットワーク	38
人工知能	1
人工ニューラルネットワーク	8
【す】	
スクリプト	65
スクリプト形式のプログラム	64
スコア	25
ステップ関数	4
ストライド	25
【せ】	
正規化層	22
セッション	70
線形分離可能	11
線形分離不可能	11
全結合層	22, 27, 29, 33, 34, 58
全結合層の各ユニットの要素と直前の層の全ユニットの要素との結合を表す値	33
【そ】	
塑性	5
ソフトマックス層	22, 27
損失値	58
【た】	
ダウンサンプリング	27
畳み込み	23, 24

畳み込みカーネル	24
畳み込み処理	24
畳み込み層	18, 22, 23
畳み込みニューラルネット	
ワーク	17
畳み込みフィルター	24, 27
単純型細胞	17
単純パーセプトロン	7
単純パーセプトロンにおける学習	9
【ち】	
チャンネル	24
チャンネル単位の局所応答	
正規化層	22
中間層	6
長期記憶	35
超複雑型細胞	17
超平面	11
【つ】	
追加学習	84
通時的誤差逆伝搬	38
【て】	
ディープニューラルネット	
ワーク	1, 43, 64
ディープネットワークデザイナ	
イナ	31, 67, 75, 77, 79, 81, 83, 84, 85, 87, 88, 90, 93
ディープラーニング	21, 28, 43
データ拡張	27, 28, 60
データストア	49
データセットに含まれないクラス	34
データの標準化	61
転移学習	21, 28, 29, 33, 34, 75, 77, 84, 92, 93
【と】	
特徴抽出器	28, 34
特徴抽出	23, 25, 26, 28, 34
特徴抽出細胞S細胞	18
ドロップアウト層	22, 23

【に】	
入出力関数	4
入力ゲート	38
入力層	6
入力層系列ユニット	37
入力層文脈ユニット	37
入力の加算性	3
入力の記憶	39
ニューラルネットワーク	1
ニューロンの興奮のしきい値	8
認識	28
【ね】	
ネオコグニトロン	1, 7, 17, 18, 21
【は】	
排他的論理和	11
配列の大きさの宣言	48
パーセプトロン	1, 6
パターン認識	18
パディング	25, 63
パラメータチューニング	8
反応ユニット	7
【ひ】	
光受容細胞	18
ビデオラベラー	70
微分可能	4
【ふ】	
フィードフォワード型の階層的ニューラルネットワーク	12
フィルター	24
フィルターをもつ新しい畳み込み層	45
深さ連結層	33
複雑型細胞	17
プーリング	23, 25
プーリング層	18, 22
分類	28
分類出力層	29

【へ】		ライブスクリプト形式の プログラム	64	imageDataAugmenter	60, 71, 73
ヘブの学習則	5			ImageNet	65, 67, 75, 77
【ほ】		【り】		ImageNet データセット	23, 28, 30, 33, 56
忘却ゲート	38	リカレントニューラルネット ワーク	35	Inception モジュール	30, 32, 33
【ま】		【れ】		LeNet	21
膜電位	3	連合ユニット	7	LSTM	21, 35, 85
【み】		【ろ】		LSTM 層	35
ミニバッチ サイズ	61	ロジスティックシグモイダル	38	MATLAB Compiler	52
【め】		【わ】		MATLAB リリース ノート	55
メモリセル	38	ワークスペース	47	Parallel Computing Tool Box	53
【も】		ワークスペース変数	76, 79, 84, 93, 94, 99, 202	Places365	56
網 膜	17			Places365 データセット	30
モーメント法	17	【欧文】		ReLU 層	22, 23
【ゆ】		Adam 最適化	62	RNN	35
有向閉ループ	37	AI	1	Simulink	53
有向閉路	35	AlexNet	21, 22	SqueezeNet	67, 75
ユニット	23, 27, 29, 33	AlexNet での学習	27	S 細胞	18
ユニット要素	24	analyzeNetwork	57	tanh 関数	38
ユニット要素値	25	Bias	6, 23, 27, 29, 31, 33, 34	Weight	5, 23, 27, 29, 33, 34
【よ】		CNN	17, 18, 21, 22, 23, 24, 25, 26	【数字】	
抑制性シナプス	3	C 細胞	18	1×1 畳み込み層	59
予測ラベル	58	Deep Network Designer	52	2次元グローバル平均プー リング層	30
【ら】		Elman 型ネットワーク	36	2次元最大プーリング層	22
ライブスクリプト	65	GoogLeNet	21, 29, 65, 77	2次元情報	24
		GoogLeNet での学習	33	2次元畳み込み層	22, 27, 33
				2次元畳み込み層の畳み込み フィルター	33

英 訳 索 引

【A】				【E】				【L】	
activation function	4	excitatory synapse	3	lateral geniculate nucleus	17	layered neural network	6	learning	8
activation layer	22	【F】		learning rate	10	linearly separatable	11	long short-term memory	35
artificial intelligence	1	feature extractor	28	long-term memory	35				
associate unit	7	feed-forward neural network	38						
axon	2	filter	24						
【B】		formal neuron	1						
back propagation	1	fully-connected layer	22						
backpropagation through time	38	【G】							
【C】		gradient decent method	12						
cell membrane	2	group-wise 2-dimensional convolution layer	22						
channel	24	【H】							
class score	23	hidden layer	6						
cognitron	18	hypercomplex cell	17						
complex cell	17	hyperplane	11						
convolution filter	24	【I】							
convolution kernel	24	implicit	12						
convolution layer	18	inhibitory synapse	3						
convolutional neural network	17	input layer	6						
【D】		【K】							
data augmentation	27	kernel	24						
deep neural network	1	【N】							
dendrite	2	neocognitron	1						
dimension compression	25	neural network	1						
dropout layer	22	normalization layer	22						
				nucleus	2				
				【O】					
				output function	4				
				output layer	6				
				【P】					
				padding	25				
				perceptron	1, 6				
				photoreceptor cell	18				

pooling layer	18	soma	2	unit element	24
postsynaptic neuron	2	steepest descent method	12	unsupervised learning	9
presynaptic neuron	2	step function	4		
		stride	25	【v】	
【R】		supervised learning	9	visual cortex	17
receptive field	17	synapse	2		
recurrent neural network	35	synaptic plasticity	1	【w】	
response unit	7	synaptic weight	4	weighted filter	24
retina	17			【数字】	
		【T】		2-dimensional convolution	
【S】		threshold effect	2	layer	22
sensory unit	7	training data	27	2-dimensional max pooling	
sigmoid function	4	transfer learning	28	layer	22
simple cell	17				
simple perceptron	7	【U】			
softmax layer	22	unit	23		

— 編著者略歴 —

吉富 康成（よしとみ やすなり）

- 1982年 京都大学大学院工学研究科修士課程修了（数理工学専攻）
- 1982年 新日本製鐵株式会社勤務
- 1991年 工学博士（京都大学）
- 1995年 宮崎大学助教授
- 2001年 京都府立大学教授
- 2008年 京都府立大学大学院教授
- 2022年 京都府立大学名誉教授
現在に至る

— 著者略歴 —

浅田 太郎（あさだ たろう）

- 2010年 京都府立大学大学院人間環境科学研究科博士後期課程修了（環境情報学専攻）
博士（学術）
- 2010年 京都府立大学特任講師
- 2011年 京都府立大学特任准教授
- 2022年 京都府立大学大学院教授
現在に至る

田伏 正佳（たふせ まさよし）

- 1988年 神戸大学大学院自然科学研究科博士課程修了（物質科学専攻）
学術博士
- 1992年 宮崎大学助手
- 1999年 宮崎大学助教授
- 2003年 京都府立大学助教授
- 2007年 京都府立大学准教授
- 2008年 京都府立大学大学院准教授
- 2018年 京都府立大学大学院教授
現在に至る

横山 友也（よこやま ゆうや）

- 2015年 京都工芸繊維大学大学院工芸科学研究科博士後期課程修了（設計工学専攻）
博士（学術）
- 2016年 京都府立大学特任助教
- 2017年 京都工芸繊維大学非常勤講師
- 2023年 東京都立産業技術大学院大学助教
現在に至る

演習で学ぶ MATLAB によるディープラーニング

Deep Learning with MATLAB

© Yoshitomi, Asada, Tabuse, Yokoyama 2024

2024年3月1日 初版第1刷発行

★

検印省略

編著者 吉 富 康 成
著者 浅 田 太 郎
田 伏 正 佳
横 山 友 也
発行者 株式会社 コロナ社
代表者 牛来真也
印刷所 壮光舎印刷株式会社
製本所 株式会社 グリーン

112-0011 東京都文京区千石 4-46-10

発行所 株式会社 コロナ社

CORONA PUBLISHING CO., LTD.

Tokyo Japan

振替00140-8-14844・電話(03)3941-3131(代)

ホームページ <https://www.coronasha.co.jp>

ISBN 978-4-339-02942-0 C3055 Printed in Japan

(田中)



JCOPY < 出版者著作権管理機構 委託出版物 >

本書の無断複製は著作権法上での例外を除き禁じられています。複製される場合は、そのつど事前に、出版者著作権管理機構（電話 03-5244-5088, FAX 03-5244-5089, e-mail: info@jcopy.or.jp）の許諾を得てください。

本書のコピー、スキャン、デジタル化等の無断複製・転載は著作権法上での例外を除き禁じられています。購入者以外の第三者による本書の電子データ化及び電子書籍化は、いかなる場合も認めていません。落丁・乱丁はお取替えいたします。