

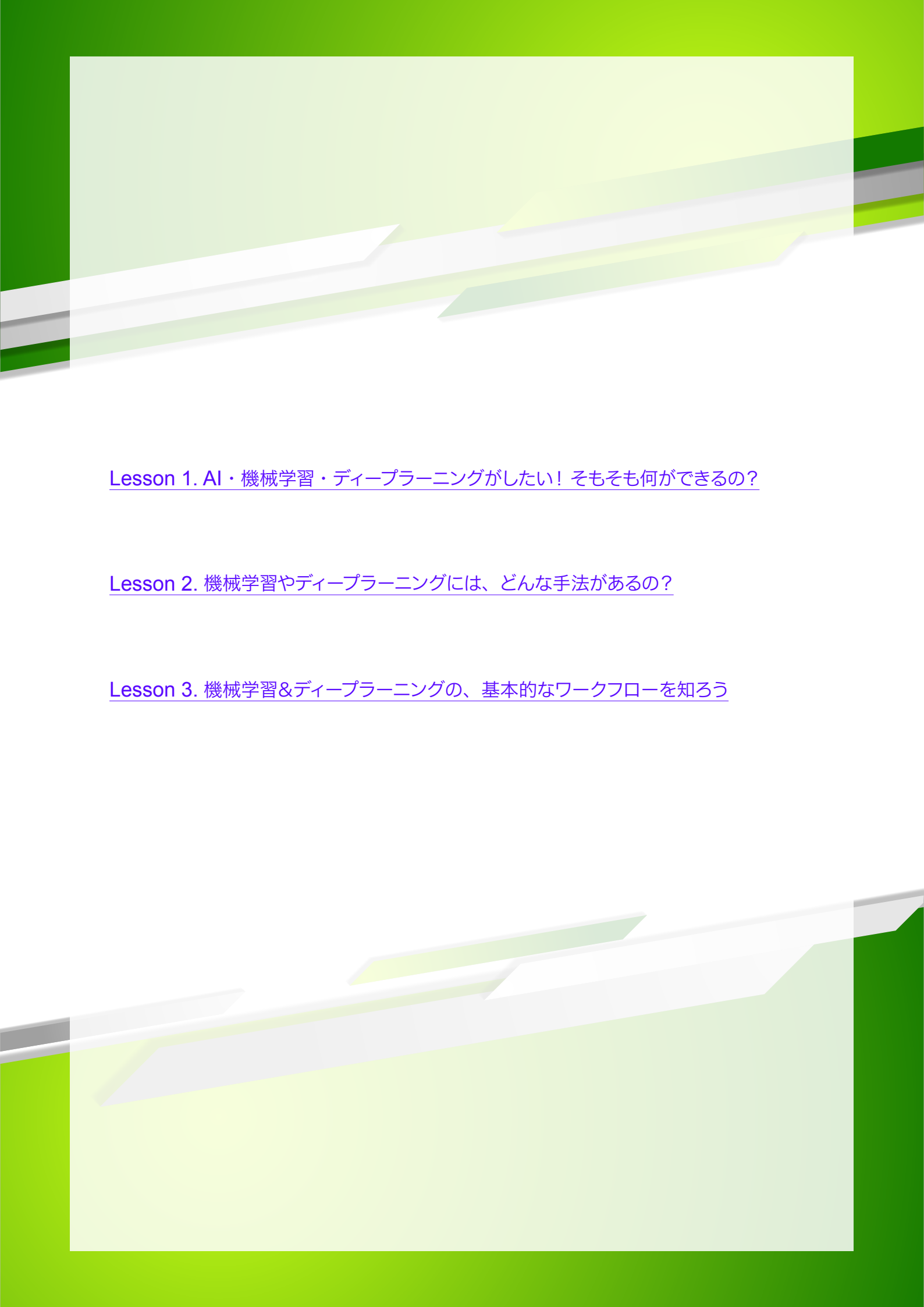


a t m a r k I T

# 普通のエンジニアでも分かる ディープラーニング概説

機械学習&ディープラーニング入門（概要編）

一色政彦, デジタルアドバンテージ



[Lesson 1. AI・機械学習・ディープラーニングがしたい! そもそも何ができるの?](#)

[Lesson 2. 機械学習やディープラーニングには、どんな手法があるの?](#)

[Lesson 3. 機械学習&ディープラーニングの、基本的なワークフローを知ろう](#)

# Lesson 1 AI・機械学習・ディープラーニングがしたい！ そもそも何ができるの？

機械学習専門家の藍博士と素人のマナブが会話形式で、AI・機械学習・ディープラーニングの基礎の基礎を分かりやすく紹介するシリーズがスタート。まずはAIとは何か、機械学習との違い、ディープラーニングで実現できることを知ろう。

(2018年04月16日)

ご注意:本記事は、@ IT / Deep Insider 編集部（デジタルアドバンテージ社）が「deepinsider.jp」というサイトから、内容を改変することなく、そのまま「@ IT」へと転載したものです。このため用字用語の統一ルールなどは@ IT のそれとは一致しません。あらかじめご了承ください。（インタビュー取材協力：DATUM STUDIO 安部 晃生）

## 登場人物紹介



### 深井藍（ふかい・あい）博士

最新の人工知能技術を応用して、次世代の人型ロボット（アンドロイド）を開発するのが仕事。試行錯誤の末にやっと開発できたのがマナブ（01号）である。

責任感が強く頑固で読書家だが、ドラマ好きで、超天然な一面もあるアラサー リケ女。

ちなみに藍が使っているタブレットには、マナブの学習状況をチェックできる機能だけでなく、万が一の安全対策としてマナブの暴走を制御するための「秘密機能」が搭載されているという。



### マナブ（01号）

現実社会の学習を進めるため、藍博士と24時間生活を共にしている次世代アンドロイド、0歳。

見るもの聞くものすべてに興味津々。藍が好きなテレビドラマとお笑い番組からも学習しているため、うわすべりな知識から勘違いな行動を取ったり、大阪風のボケをかまししたりすることもある。

この物語の主人公。エンジニアスキルはあるけど機械学習やディープラーニングについてまだ何も知らない。

## 2018年、春のある日

現実社会の学習をもっと深めようと、毎日、マナブをいろんなところに連れて行く藍。今日は、世界中の食べ物についてもっと学習させるため、横浜ワールドポーターズにやってきて、お店巡りをしていたときのこと。

## マナブ、人工智能に興味を持ち始める



ジャーン！これ知ってる？ハワイのドーナツ、マラサダだよ。このお店のは、ふわふわモチモチで美味しいのよ。



うわぁ～食べたい。見た目は、和風のアンドーナツだね。ミスドのドーナツにも似ているし、一緒に並べると見分けられないかも。藍は見分けられるの？



図1 マラサダ



（ふむふむ、ドーナツに対する学習は進んでいないと……）形が似ていると見分けるのは大変だね。でも、私が仕事でやっているAI（人工知能）はそういう見た目判定は得意で、場合によっては人間よりも優れてるんだよ。



そうなんだ。そういえば、昨日見たテレビで、将来、AIで仕事が無くなるって言ってたなあ。自分が大人になったときに、そんなんなったらかなわんわー。だから逆に、こっちからAIを作る側に回らんのだ！



（ヤバい！ヤバい！ヤバい！これはまさかシンギュラリティ\*1の始まり!!?）

### \*1【解説しよう】シンギュラリティ

シンギュラリティ（Singularity\*2、技術的特異点）とは、人工知能が、さらに優れた人工知能を再帰的に創造していくことで、人間を完全に超える圧倒的に高度な知性が生み出されるとする仮説のこと。ここでは詳しくは紹介しないが、この仮説に対しては、各方面からさまざまな賛否両論の主張がある。シンギュラリティを人類の危機と考える人もあり、例えば映画『マトリックス』や『ターミネーター』のスカイネットも、シンギュラリティにより生まれた危機的な人類を想像して描かれたフィクションだと言えるだろう。より詳しくは[キーワード解説](#)を参照してほしい。

**\*2** ディープラーニングや機械学習を実践していくには、APIリファレンスなどが英語になるので、「英語の文章を読むことは避けられない」と考えた方がよい。そこで本シリーズでは、英語のドキュメントを読むためのヒントになるよう、大切なキーワードについては、英単語をカッコ書きで記載する。

—— パニックになった藍は、気づいたらマナブの分のマラサダまで食べてしまっていた ——



あ！僕のマラサダまで……！！（泣）



（しまった……。突然だったのでビックリしちゃったけど、考えすぎかな……）ごめん、ごめん。マラサダはもう 1 個買うね。



もう、藍先生ったら食いしん坊なんだから！ まあ許すけど。お詫びに、AI の作り方を教えてよ。いつも仕事でやってんだよね？



（マナブに AI・機械学習・ディープラーニングを教えるの、研究材料としてむしろ面白いかも）はいはい、分かりました。でも、やる気があるのはいいことだけど、AI で何でも実現できるというわけでもないのよ。マナブは AI で何ができているの？



テレビで言ってたのは、タクシー／トラックの運転手、スーパーのレジ係、レストランの案内係、オフィスの警備員、銀行員の仕事なんかができるってことだったかな。こういう人たちの作業は、AI で全部実現できちゃうってことなのかな。



うーん。それらの作業は、AI とさまざまな技術が組み合わさってできているから、一概に「AI ができる」とは言いにくいよね。まずは「AI とは何か」についてきちんとした理解が必要だね。



はい、先生！（キリッ）

## AI とは？



**AI（Artificial Intelligence、人工知能）** とは、人間が行う「知的活動」をコンピュータープログラムとして実現することだよ。



知的活動？



知的活動とは、頭（厳密には脳）で考えて実行する活動全般のことだね。例えばさっきの「ドーナツを見分ける」とか、「絵を描く」「言葉を認識する」「ゲームをする」などなど、あらゆる人間の行動がこれに当てはまるんだよ。

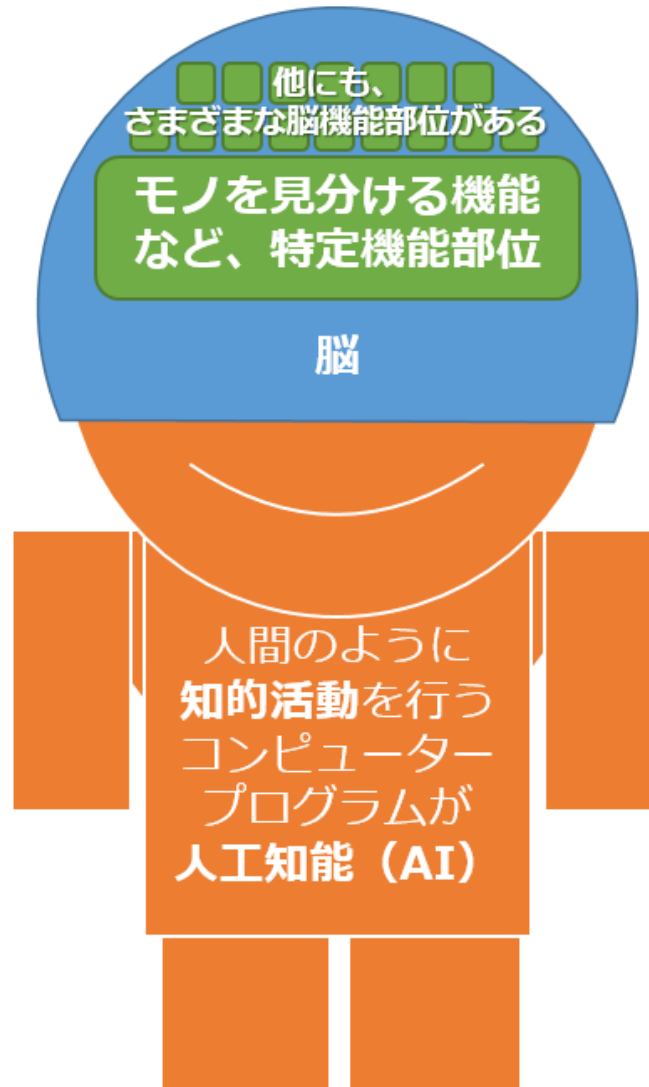


図2 人間の知的活動



ゲームといえば、囲碁で AI が人間のトップ棋士に 5 番勝負で勝利したという話をテレビで見たよ。



2016 年の **AlphaGo** だね \*3。

\*3 ちなみに、囲碁で AI がハンデ無しでプロ棋士に初めて 1 勝したのは、これより前の 2008 年のこと。





ところで「囲碁」ってどういうゲーム？どこが知的活動なの？



囲碁は、黒石と白石に分かれて盤面の陣地を取り合うゲームだけど、「どこに石を打てばよいのか」という課題に対する解答を考えながら勝負するの。最終的に陣地が広い方が勝ち。つまり囲碁における AI とは、「最大の陣地を得るためにどこに石を打つべきか」という課題に対して解答する**知的活動**をコンピュータプログラムとして実現すること、ということになるね。



うーん。何となく分かったけど、それっぽっちなの？ テレビで観てきた限りでは、「AI」といえば「まるで人間のように自分で考えて行動するもの」だとばかり思った。ドラえもんみたいにね。そうじゃないの？



広い意味ではそれも AI で、狭い意味と区別するために**強い AI (Strong AI、もしくは汎用的な AI、AGI: Artificial General Intelligence)**と呼ばれているの。強い AI は、まさに人間が行う知的活動を完全に模倣できるような AI。だから「ドラえもんを作る」っていうのは「強い AI を作る」ってことになるわね。ドラえもんができれば素敵だけど、残念ながら現在の技術は、まだそれを実現できるレベルまでには進んでいないのよ。(……と言いつつ実はマナブが強い AI なんだけどね)



じゃあ、藍先生が教えてくれるのは、狭い意味の AI になるの？



そうね。ちなみに狭い意味の AI は弱い AI (Weak AI、もしくは Narrow AI) と呼ばれていて、AlphaGo が囲碁に特化しているように、特定の処理のみを実現する AI のことよ。



ふーん。でも AlphaGo では「ディープラーニング」というのが話題になってるから、これが AI の中身なんだよね？

## AI / 機械学習 / ディープラーニングはここが違う



確かに AlphaGo もディープラーニングの技術が利用されているのよね。しかも「人間のトップ棋士を AI が打ち破った」というニュースがあまりにセンセーショナルだったので、一般の人には「AlphaGo = AI = ディープラーニング」というふうに固定的に認識されている部分があると思う。



ディープラーニング以外にも「AI」と呼べるものがあるということ？



そう。まず AI は、さっき説明したように「人間が行う知的活動をコンピュータプログラムとして実現すること」なんだけど、「どのくらい知的であれば“AI”と呼べるか」については厳密な定義があるわけではないの。



ほんなら、「AI」と呼べるポイントは何なん？



例えば AlphaGo が登場する前から、囲碁のゲームはあったよね。そういったゲームは、例えば「こういう配置であれば、こういう手を打つ」のように人が作ったルールのロジック（論理）に基づいて動作しているのだけど、そういうロジックによるプログラムも、マスメディアや一般の人などから「AI」と呼ばれる時代があったの。こういうのは最近では「AI」と呼ばれなくなっているのよね。



AI は、その時代時代の技術によって解釈が違うということなの？



そう。ある AI 技術が浸透して、人々にとって当たり前の存在になると、自然と「AI」と呼ばれなくなり注目されなくなることを時代は繰り返してきているわね。今は、ディープラーニングや機械学習を使ったものが、一般社会の人に「AI」と呼ばれて注目される時代だと思う。もちろん今でも、商品の営業戦略上、人が作ったロジックで動作が自動化されているものを「AI」と呼称しているケースもあるから、「本当に、私たちが想定している“機械学習を使った AI”なのか」は怪しいものも多いのよね。



そっか。それなら今、最新の AI 技術をを知りたいければ、「機械学習とディープラーニングを学ばばよい」ということだね。ところで、「機械学習」ってのは急に出来たけど何なの？



**機械学習（Machine Learning、ML）** は、さっきの AI の説明に似ているけど、人間が行う「学習」をコンピュータプログラムによって実現することだよ。



学習？



学習は、人間が人間らしく知的活動を行うための一つの要素だと言えるの。人間は、体験や知識から学ぶことで、新しい行動ができるようになるよね。例えば自転車に乗るのに何度もチャレンジしたら乗れるようになったり、英単語を何度も覚えることで英語の試験で良い点が取れるようになったり。この学ぶ活動を、AI 技術では**学習（learning、ラーニング）**と呼んでいるの。



あっ、ディープラーニングにも「ラーニング（学習）」って付いてる！





そうだね。ディープラーニング（Deep Learning、深層学習）は、機械学習の手法（アプローチ、技法）の一つというわけ。

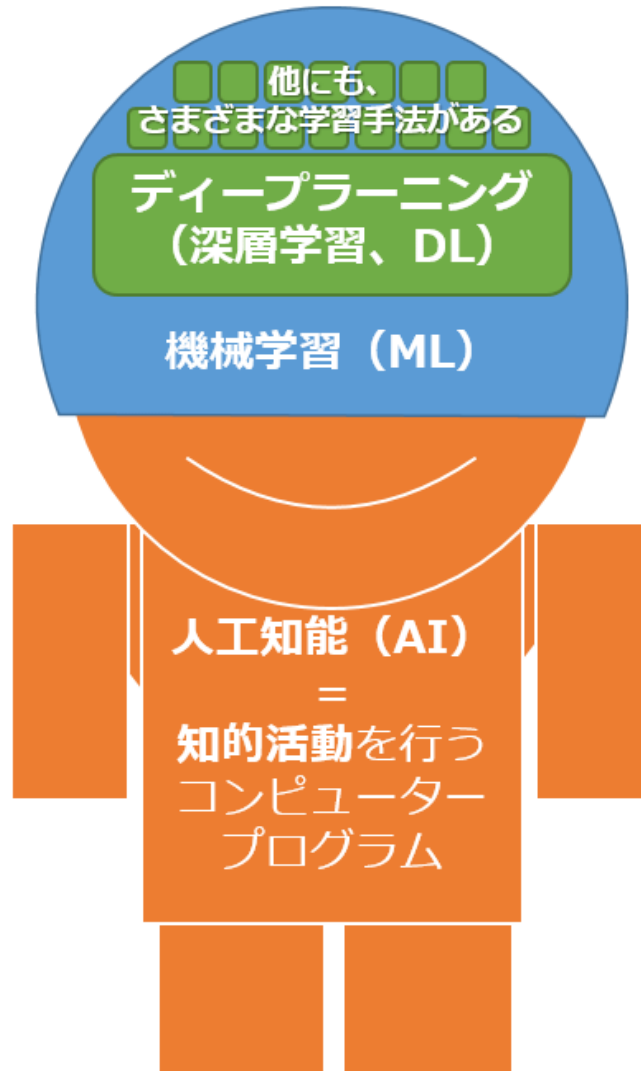


図3 AI・機械学習・ディープラーニングの包括関係



AlphaGo も学習してるの？



AlphaGo はたくさんの囲碁の対局結果から学習して、「この盤面ではこの手が最適」という判断できるようになったのね。このときに採用した機械学習の手法が「ディープラーニング」ということなの。



「機械学習 ⊃ ディープラーニング」という包括関係にあるんだね。そのディープラーニングでは具体的に何ができるの？

## これまでにディープラーニングで実現できたこと



前に話題になった例では、顔写真の画像をアップロードすると、その人の見た目の年齢を返すという Web サービスがあったよ。他には、画像の内容を読み取って「若い女性がほほえんでいる」みたいな文章を返すサービスもある。こういう「画像の内容を文章で返すプログラム」は画像キャプションと呼ばれているんだけど、ちょっと試してみようか？ 何か写真をこのサイトにアップロードしてみてごらん。結果として内容説明の文章（キャプション）が表示されるから。



OK。じゃあ、いつも観てるテレビを撮影した一番好きな画像をアップロードしてみよっと。（ポンッ）

—— 画面上のキャプション表示：「裸で激しく抱き合う 2 人 [アダルト判定：80%]」



んっ……!!? コラー！ いつも何を見てるんじゃ～。お仕置きだっぺ！  
<サンバイザー暗黒モード \*2、始動>

### \*2【解説しよう】サンバイザー秘密機能

マナブのサンバイザーには、安全対策として真っ黒にして何も見えないようにすることで、マナブの行動をやりわりと制限する機能が搭載されている。万が一、マナブが悪さをすると、藍はタブレットを操作して、サンバイザー暗黒モードのお仕置きを与えるのだ。



うわー、何も見えへん！ なんでなんや、いつも藍と観てるお相撲さんの写真やのに～～。



?!（紛らわしい画像を入れるな！）勘違いしたの。ごめん、ごめん。  
<サンバイザー暗黒モード、解除>



一体なんなの。ひどない、これ！



ふー（汗）。気を取り直して説明していくよ。確かに 2 人が裸で相撲をとっている写真だから、表示された文章は大体合っているよね。だけど今回使ったディープラーニングのプログラムは相撲取りの学習をしたことがなかったから、こういう結果になったんだと思う。相撲取りをたくさん学習させればもっと正確な文章を返してくれるようになると思うよ。



（なんか、ごまかされてるよーな）



これはあくまで一例で、他にもいろいろできるけどね。ここ最近、ディープラーニングを使うことで結果の**精度（accuracy）**が大きく向上したものには、次のようなことがあるよ。ただしこれらは代表的なものだけだし、実際には組み合わせて使われることも多いから、こんなふうにきれいに分類できるわけでもないけどね。

- **画像認識**：画像や映像から情報を抽出する。最近、特に精度が上がってきている。代表例：上記の「年齢当てサービス」
- **画像生成**：絵画の生成、画像や映像の自動加工など。代表例：線画自動着色サービスの「PaintsChainer」
- **音声認識**：言われたことを認識する。最近、特に精度が上がってきている。代表例：「Google Home」の音声認識機能
- **音声生成**：音声合成や作曲など。代表例：「Amazon Echo」での「Kindle 本の読み上げ」
- **自然言語処理（認識）**：言葉を理解して情報を抽出する。例：「Google 翻訳」
- **自然言語処理（生成）**：会話を生成するなど。いわゆるチャットボットなど。例：Twitter 上で提供されている女子高生 AI 「りんな」



チャットボットって昔からあるような気がするけど？



確かに、以前から「人工無能」と呼ばれる、決まった言葉に対して決まった言葉を返すようなチャットボットはあったわね。でもディープラーニングが活用され始めて、本当に人と会話しているような、より自然な会話を実現できるようになってきているの。



ディープラーニングで学習させれば、こういうことが何でもできる AI ができるんだね！



それは違うよ。ディープラーニングで学習させて、「**学習済みモデル（Learned model、学習モデル）**」というのを作って、それを使って各機能を実現するんだけど、このときできる学習モデルは、あくまで「特定の処理領域に特化したモデル」なの。つまり、目的別に学習させて作り込む必要があるのよ。ドラえもんみたいに汎用的に何でもできるわけではないこと（AGI ではないこと）に注意してね。



それなら、さっきのお相撲さんの写真はそれ専用で学習しないといけないということ？



基本的に、ある特定の領域向けに作った学習済みモデルは、他の目的には転用できないの。でも実は、ある特定領域で作った学習済みモデルに追加学習させることでカスタマイズし、別の領域に適応させる技術も存在するのよ。これは**転移学習 (Transfer Learning)** と呼ばれていて、さっきの Web サービスには**転移学習によって既存の学習済みモデルをカスタマイズするサービス**も提供されているから、相撲取りを認識できるようにするのはそれほど難しくないと思うわ。しかも本来、画像認識の目的で学習させるには数十万枚単位で膨大な数の画像が必要になったりするけど、転移学習であれば数十枚で学習が完了する場合も多いから、使えるなら使わない手はないわね。



それなら今の自分でも、ディープラーニングで独自の学習済みモデルを試しに作れそうやね。あとでやってみようっと。

## これからディープラーニングで実現できそうなこと



ディープラーニングは進化中だから、これからももっと実現できることが増えていくと思う。



へー、例えば例えば？



2020 年の東京オリンピックのころには、英語やフランス語、中国語と日本語をリアルタイムに通訳する翻訳 AI なんて出てくる可能性が高いんじゃないかな……。あと、声のサンプルを採って、同じ声でしゃべらせる、なんてことも可能になってきているから、例えば米国大統領のトランプさんの声で文章を読み上げさせるとかもできるかな。



それって……偽の声で悪いことされたら嫌やん？



確かに倫理的な問題や、犯罪に悪用される可能性はあるのよね。とはいっても科学技術の進歩は、いつも悪用や誤用の危険性と隣合わせだから、それも同時に考えながら対策を考えていく必要はあると思う。



ニュースで大騒ぎになっている自動運転は？



複合技術の領域になるので、すぐに実現できるとは思えないけど、いつか実現しても全然おかしくはないわね。実際に、アメリカでは公道で実証実験が行われたりしているわ。



将来性あるなあ〜。ディープラーニングを中心に機械学習を学ぶのがますます楽しくなってきた！あと、ディープラーニング以外の機械学習にはどんなのがあるの？



挙げるとキリがないけど、今、思い付く**機械学習の手法**（**approaches**、その中身となる計算方法は「**アルゴリズム**：**algorithm**」とも呼ばれる）をいくつか挙げると、

- **決定木 (Decision Tree)**：ツリーを自動的に作ってカテゴリを分類する手法
- **最近傍法 (Nearest Neighbor algorithm)**：一番近い隣を使う手法
- **単純ベイズ分類器 (Naive Bayes classifier)**：ベイズの定理を使う手法。スパムフィルターの技術として使われている
- **サポートベクターマシン (Support Vector Machine : SVM)**：点と点のマージンが最大になるように線引きする手法
- **ニューラルネットワーク (Neural Network : NN)**：人間の神経回路を真似して、入力層→1つ以上の隠れ層→出力層という多層ネットワークを構成する手法

などがあるよ。説明はまだ分からなくても大丈夫。これらはあくまで代表的なもので、これらからさらに細分化した手法も存在するのよね。例えば決定木の一つには「ランダムフォレスト」という手法などもあるよ。

ちなみに、今後、解説するディープラーニングの代表的な手法についてはニューラルネットワークの一種になっているの



図 4 機械学習の手法



へえ。何だかよく分からないけど、いっぱいあるんだね。



本来であれば、こういった手法が生まれた流れに沿って、ステップ・バイ・ステップで機械学習を体系的に学んでいくと、課題に対して最適な手法を選択する知識も身についていくと思うのだけれども、それ全部をゼロから説明するにはかなり時間がかかるから、今日はとりあえず主な名前を挙げるだけで説明は省略させてもらうね。ここからは、マナブが一番興味がありそうなディープラーニングを中心に、できるだけ簡単に理解できるように説明していくね。



## 【コラム】ただしディープラーニングも万能ではない

注目の AI 技術となったディープラーニングだが、ディープラーニングに欠点がないわけではない。初回から深い話は荷が重いので、今回は参考程度に紹介しておこう。以下が主な欠点である。

### 学習済みモデルの中身（ネットワーク）は解釈できない

多くの機械学習の手法では中身は「計算式のアルゴリズム」になるので解釈可能である。「賃貸物件の家賃」を求めるアルゴリズムを例にとると、「家賃 = 駅からの徒歩距離  $\times a$  + 部屋数  $\times b$  + 築年数  $\times c$  + ……」のような計算式が提示されれば、その意味は理解できるだろう。

しかし、ディープラーニングなどのニューラルネットワークの手法は、学習によって最終的に出来上がったネットワークを見ても人間には解釈できない（図 5）。これは、脳神経のつながりのネットワークを見ても、意味が分からないのと同じだ。このため、ディープラーニングの学習済みモデルが表現する内容を、顧客ユーザーらに説明するのも不可能で、**ブラックボックス**として示すしかない。

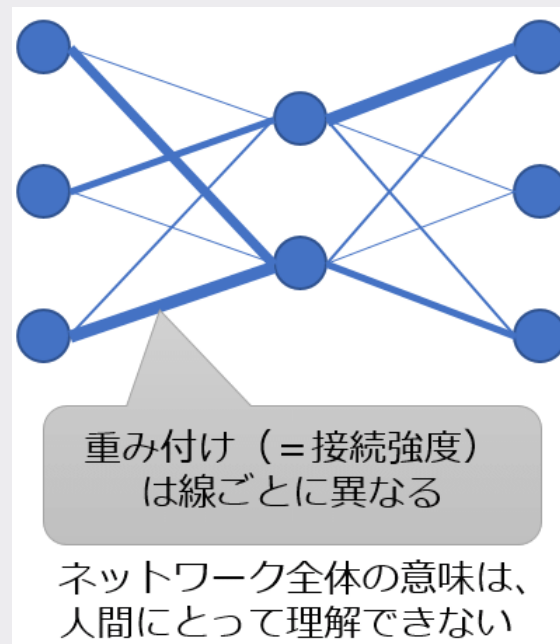


図 5 ネットワークは解釈できない

### 大量のデータが必要

画像認識の説明でも簡単に示したが、ネットワークが構築されるまでには非常に膨大なデータが必要となる。ただしこれについては、データを拡張する方法（第 3 回で説明）など、さまざまな改善手段も提案されている。

### ネットワーク作成には時間がかかる

ネットワーク作成時の計算には、データ量やネットワークのサイズによって、何時間も、ひどいときには何週間もかかったりする。しかも、第 3 回で説明することになるが、学習して最適解を出せるようになるまでには、そのネットワーク作成を何度となく繰り返して結果を評価する試行錯誤が必要になる。

## まとめ



AI については大体分かってきた！



では今日はここまで。お疲れさま。



おおきに、藍先生！勘違いしているところもいっぱいあったけど、やっぱおもしろいでこれからが楽しみや！



あらあら、得意の大阪弁が出たわね（テレビの漫才見せるのもいい加減にしないとなあ……）。じゃあ明日の朝食はパンケーキを食べに行こう。次は機械学習がどんなものなのか、もっと具体的に教えてあげるね。

### 【まとめよう】AI / 機械学習 / ディープラーニングの概要

- **AI（人工知能）**：人間が行う「知的活動」をコンピュータープログラムとして実現すること
- **シンギュラリティ**：AI 自らが AI を生み出せるようになると、人間を完全に超える圧倒的に高度な知性が生み出されるとする仮説
- **強い AI**：人間が行う知的活動を完全に模倣できる AI で、汎用的な AI（AGI）とも表現される
- **弱い AI**：特定の処理のみを実現する AI で、現在はこれを実現されている
- **今の時代に注目されている AI の技術領域**：ディープラーニングもしくは機械学習を使った AI
- **機械学習**：人間が行う「学習」をコンピュータープログラムによって実現すること
- **ディープラーニング（深層学習）**：機械学習の手法の一つで、また機械学習は AI の一種。内容の詳細は次回説明
- **ディープラーニングで実現できていること**：画像の認識／生成、音声の認識／生成、自然言語処理（認識／生成）など
- **ディープラーニングで将来的に実現される可能性があること**：リアルタイム通訳、自動運転など
- **ディープラーニング以外の機械学習の手法**：決定木、最近傍法、単純ベイズ分類器、サポートベクターマシン、ニューラルネットワークなど

## Lesson 2 機械学習やディープラーニングにはどんな手法があるの？

藍博士とマナブの会話から機械学習とディープラーニングの基礎の基礎を学ぼう。機械学習を始めるための最低限の基礎用語から、ディープラーニングの代表的な学習方法と代表的なアルゴリズムまでをできるだけシンプルに紹介する。

(2018 年 04 月 17 日)

ご注意:本記事は、@ IT / Deep Insider 編集部（デジタルアドバンテージ社）が「deepinsider.jp」というサイトから、内容を改変することなく、そのまま「@ IT」へと転載したものです。このため用語用語の統一ルールなどは@ IT のそれとは一致しません。あらかじめご了承ください。（インタビュー取材協力：DATUM STUDIO 安部 晃生）

### 登場人物紹介



#### 深井藍（ふかい・あい）博士

最新の人工知能技術を応用して、次世代の人型ロボット（アンドロイド）を開発するのが仕事。試行錯誤の末にやっと開発できたのがマナブ（01 号）である。

責任感が強く頑固で読書家だが、ドラマ好きで、超天然な一面もあるアラサー リケ女。

ちなみに藍が使っているタブレットには、マナブの学習状況をチェックできる機能だけでなく、万が一の安全対策としてマナブの暴走を制御するための「秘密機能」が搭載されているという。



#### マナブ（01 号）

現実社会の学習を進めるため、藍博士と 24 時間生活を共にしている次世代アンドロイド、0 歳。

見るもの聞くものすべてに興味津々。藍が好きなテレビドラマとお笑い番組からも学習しているため、うわすべりな知識から勘違いな行動を取ったり、大阪風のボケをかまししたりすることもある。

この物語の主人公。エンジニアスキルはあるけど機械学習やディープラーニングについてまだ何も知らない。

### ディープラーニングを教え始めて 2 日目

毎日、マナブを連れだし、現実社会を深く学ばせる藍。今日は、さらに食べ物について学習させようと、お台場のハワイアン・パンケーキ屋さんにやってきた。

## マナブ、ディープラーニングを基礎から学び始める



うわ、めっちゃボリュームある〜！ 美味しそう。早く食べたい。藍先生、美味しそうなお店を知ってるなあ。甘い物好きだね。



職場の友達がスイーツ好きで、その影響かな。うふふ。しかも前にテレビでやってた深夜ドラマ『さぼりーまん甘太郎』全話もブルーレイで買って持ってるしね。



何それ www



(やばい、口が滑った……)

—— マナブのサンバイザーには、動画検索して映像を見るための機能が搭載されている ——



動画検索して見てみよっと。ん—————!!? 何これぶっ飛びすぎやろ www



(ヤバい! ヤバい! ヤバい! 趣味がばれてしまう!)

—— パニックになった藍は無意識のうちにマナブのサンバイザーの秘密ボタンを連打してしまった ——

<サンバイザー暗黒モード \*1、始動>

### \*1【解説しよう】 サンバイザー秘密機能

マナブのサンバイザーには、安全対策として真っ黒にして何も見えないようにすることで、マナブの行動をやりわりと制限する機能が搭載されている。万が一、マナブが悪さをすると、藍はタブレットを操作して、サンバイザー暗黒モードのお仕置きを与えるのだ。



うわー。なんでや、なんでや、何も悪いことしてないっつーのに〜〜。しかもしょっぱなからやん。



図1 パンケーキ



(しばらくこのままにして忘れさせようかな……と思ったけど良心が痛む) ごめん、ごめん。解除するね。  
<サンバイザー 暗黒モード、解除>



かんたろ……



あ! そうそう、今日は機械学習の話をする約束だったね。



そやで! (目がキラキラ)



(しめしめ、忘れたな。機械学習に興味津々で助かったわ)

## 機械学習における手法／アルゴリズム／学習済みのモデルはここが違う



昨日、

**機械学習の手法** (その中身となる計算方法が**アルゴリズム**) の一つに**ディープラーニング**がある。

……

ディープラーニングで学習して実現できるようになった各機能——その中身である「**学習済みモデル**」……

と、さらっと説明したよね? 「手法」「アルゴリズム」「モデル」という単語は全部似ている気がして何が違うのかがよく分からなかったよ。



確かにそこは、今後の説明のためにも、いったん整理しておいた方がいいね。ここからは、徐々に学術的な話が入って少し難しくなるから、覚悟して頑張って付いてきてね。



- **機械学習の手法／技法 (approaches)**：アルゴリズムを活用してモデルを作成する手順・方法。例えば「ニューラルネットワーク」
- **アルゴリズム (algorithm)**：学習前の計算式／計算方法。例えば「線形回帰」\*2であれば  $y = ax + b$  のようなパラメーター（ $a$  や  $b$ ）が決まっていない抽象的な式パラメーターを決める手続きのこと
- **モデル (model)**：学習後の計算式／計算方法。例えば上記の「線形回帰」\*2であれば  $y = 2x - 1$  のようなパラメーターが決まった具体的な式のこと

## \*2【解説しよう】線形回帰／非線形回帰

線形回帰（Linear regression）のイメージを説明すると、「データの1つ1つの値を点としてプロットしたグラフ上で、それに近似（フィット）する直線を引くこと」と同じと言える（図2の左）（詳しくは後述）。ちなみに、非線形回帰（Nonlinear regression）のイメージは「曲線を引くこと」と同じと言える（図2の右）。

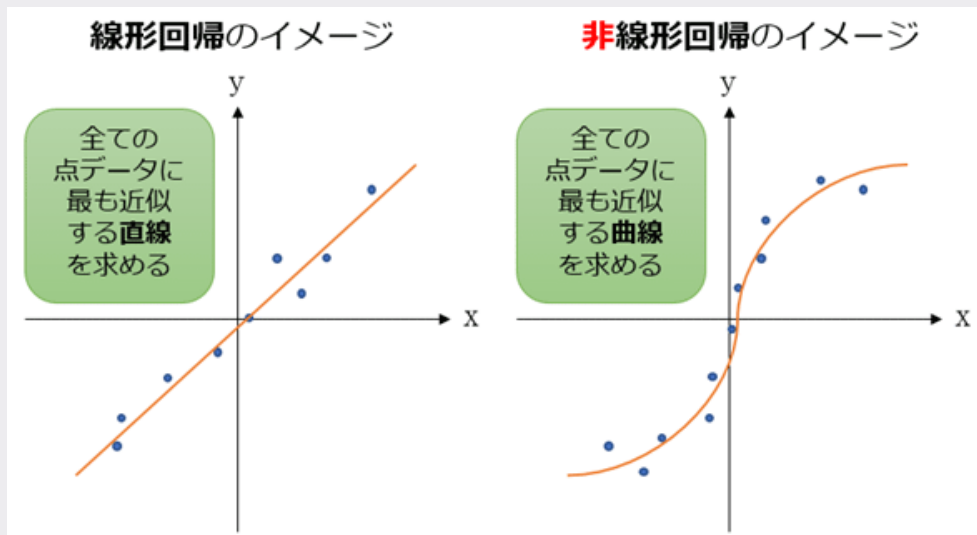


図2 線形回帰と非線形回帰



「モデル」を説明的に丁寧に言うと「学習が済んだモデル」という理解でいいのかな？



まーそういうことだね。本来のモデルの意味は「入力を与えると、何らかの処理後の出力を返すもの」のことなんだけど、機械学習においては、アルゴリズムを活用して学習した結果として「出来上がったもの」がその「モデル」ということになる。なので例えば「サポートベクターマシン（SVM）」のアルゴリズムで学習したものは「SVM モデル」と呼ばれたりするね。



それでニューラルネットワークやディープラーニングでは「学習済みモデル」（もしくは「学習モデル」）と呼ぶんだよね？





そう。それ以外にも、その中身であるネットワークに着目して「**ネットワークモデル (Network model)**」とも呼ばれることがあるね。ただし現実には、もっとルーズに「アルゴリズム」や「手法」のことも「モデル」と表現されることもあるから、そのあたりは文脈によって判断するしかないよ。

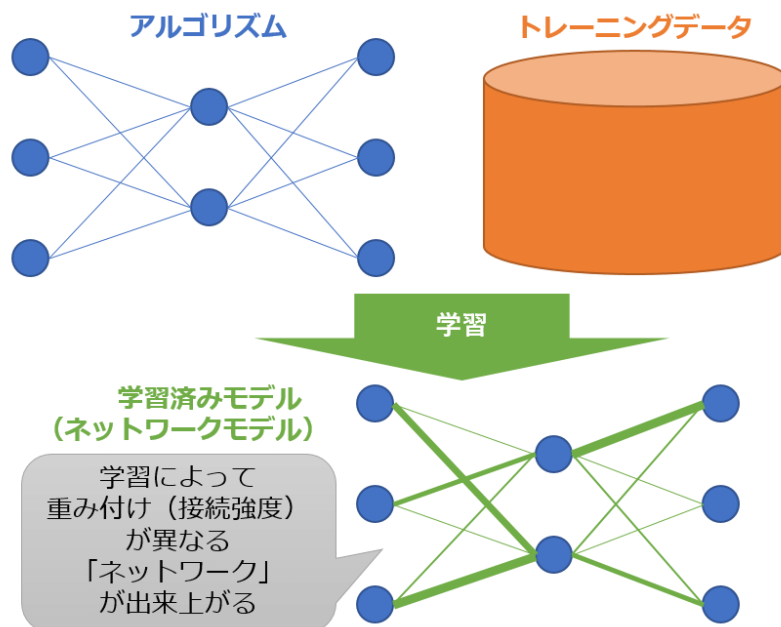


図3 アルゴリズムとモデル (ニューラルネットワーク手法による機械学習の場合)



ところでスルーしてたけど、例に挙げた「線形回帰」の  $y = ax + b$  が、なんで学習したら  $y = 2x - 1$  になるの？



図4を見てみて。まず学習用の**トレーニングデータ (Training data、訓練データ)**をXY座標のグラフに点としてプロットしたとするわね。そのすべての点データに最も近似 (フィット) する  $y = ax + b$  の線を引こうとすると、**a** や **b** のパラメーターがおのずと決まってくるでしょう？ そうやって決まったのが  $y = 2x - 1$  で、これがモデルになるの。ちなみに機械学習の課題はおおむね、この例のように「トレーニングデータに近似する線を引いてください」という問題に置き換えることができるの。

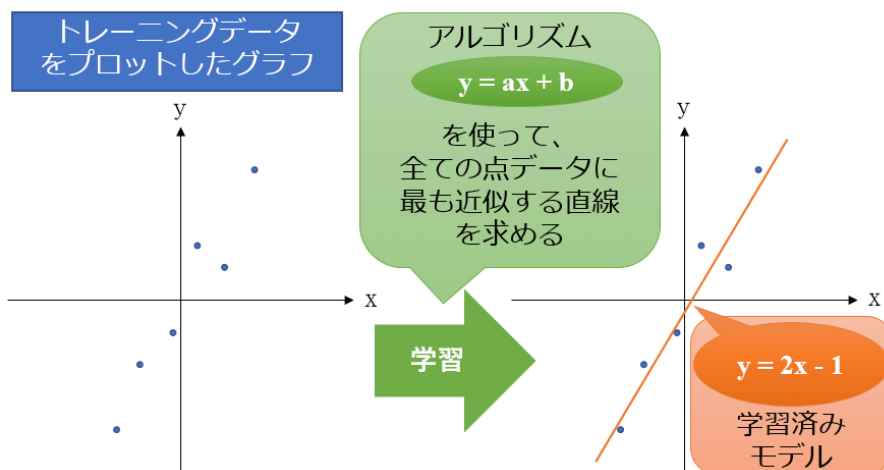


図4 機械学習の課題≒「トレーニングデータに近似する線を引いてください」という問題



ふ〜ん、分かったような、分からないような……



では身近なもので例えてみようか。例えばジャンケンって、人によって癖があるよね。ある友達の癖を学習して、勝ちパターンの**戦略モデル**を構築することは、機械学習に近いのよ。具体的には、その友達が普段どういう手を出しているかを観察して**データ**を蓄積していけば、

「2 回連続勝利する戦略 = 1 回目は **a** の手 + 2 回目は **b** の手」

という**計算式アルゴリズム**における勝ちパターンの**戦略モデル**が見えてくるよね。例えば友達が「1 回目はチョキ、2 回目はグーを出すことが多い」という「データ」があれば、それに勝てるような **a** と **b** を求めていくと、

「2 回連続勝利する戦略 = 1 回目は **グー** + 2 回目は **パー**」

という勝ちパターンの**戦略モデル**が構築されるでしょう。こんな感じで**モデル**を構築していくのが機械学習における**学習**なんだよ。



確かにそういう戦略モデルを自分も立ててる！これは学習していつに使うということなんだね。

## 機械学習の学習と推論



学習したモデルは、実践で使うことになるよね。これに関しては以下のような用語があるので覚えておいてね。

- **学習 (learning)** : モデルを「作る」こと
- **推論 (inference)** : モデルを「使う」こと



OK。ジャンケンの過去データから勝ちパターンを**学習**して、実践では勝ちパターンによってジャンケンの次の手を**推論**する、というふうに言えばいいのかな？



そんな感じ。推論はそれほど難しくないのよ、次回説明するね。今日は学習について、もう少し掘り下げていこう。

### 3つの代表的な学習方法



学習にもやり方がいくつかあって、代表的なものが次の3つになるよ。ちなみにこれ以外にも**半教師あり学習 (Semi-Supervised Learning)** というものもあるけど、これはまだ覚えなくていいと思う。

- **教師あり学習 (Supervised Learning)** : 正解が決まっているトレーニングデータを使って学習し、過去の正解にできるだけ近似 (回帰/分類) する**入出力パターンモデル**を構築すること
- **教師なし学習 (Unsupervised Learning)** : 正解が決まっていないトレーニングデータを使って学習し、クラスタリングや次元削減によって**本質的なデータ構造のモデル**を構築すること
- **強化学習 (Reinforcement Learning)** : プログラムの行動に対するフィードバック (報酬・罰) をトレーニングデータとして使って学習し、次に最も取るべき**行動方針のモデル**を構築すること



……、あーあー……何言ってるのか全く分からないよ。まずは、**最初の教師あり学習**を身近な例で教えてよ。

#### 教師あり学習

例えば動物の写真を見て「犬か」「猫か」を判断する問題があるとするよね。子供の場合、両親から「これはワンワンですよ」「これはニャンニャンですよ」などと教えられながら、犬と猫の区別を学んでいく。これと同じように教師あり学習では、動物の写真ごとに「これは犬です」「これは猫です」などの正解**ラベル (labels = 教師データ:labeled training data)** と一致するかをフィードバックされながら、犬と猫の区別を学んでいくということよ (図5)。

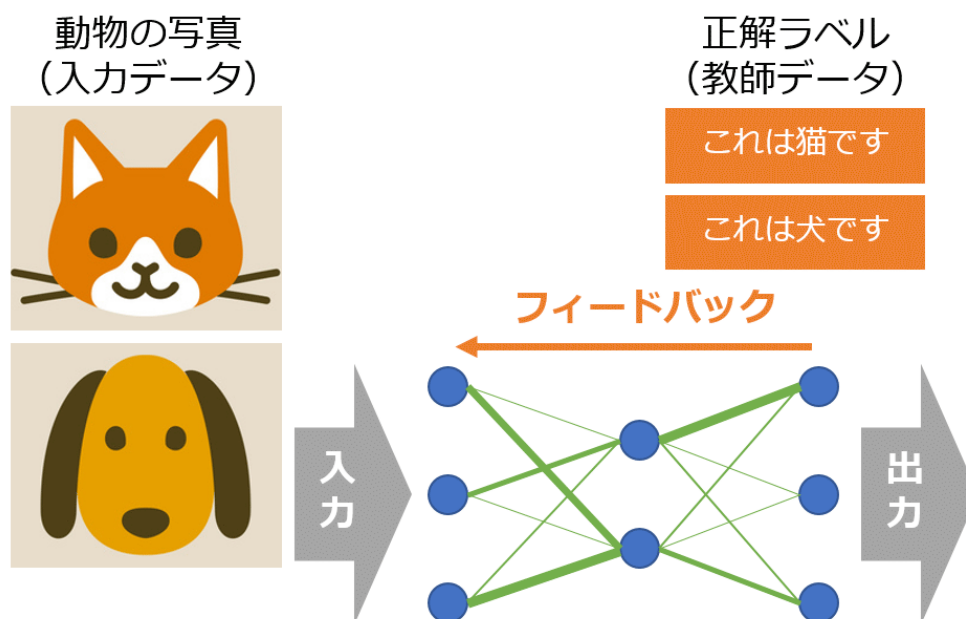


図5 教師あり学習のイメージ



名前のとおり、教師がいるときの学習ということだね。



先ほど「フィードバック」と言ったのは、学術的には「ネットワークに誤差信号を戻す」という意味で、**バックプロパゲーション（Backpropagation、誤差逆伝播）**と呼ばれているよ。



バックプロパゲーション、長いけど覚えた！あと、回帰とか分類とか言ってたけど、これは何なの？



回帰は「連続的なデータ」の問題を、分類は「離散的なデータ」の問題を解決するために使うよ。

## 回帰



連続的なデータ？



例えばマナブは毎日体重計に乗っているよね。1日1日ちょっとずつ値が変わりながら体重の線が推移していくでしょ。こういう点々にフィットする線を引いていける（＝学習できる）のが連続的なデータ（図6）。ちなみにこの場合、正解ラベル（教師データ）は、次の日の体重になる。他には、毎日の温度とか、株価の推移とか、そういうものが例として挙げられるよ。

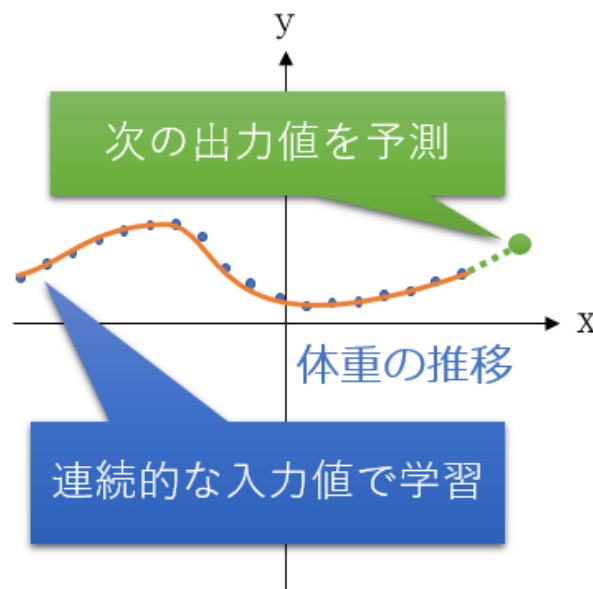


図6 連続的なデータによる回帰の例



なるほど。確かに毎日、体重が減っていったら、今日の体重から連続した翌日の体重はこれくらい減っていると予測できるね。じゃあ離散的なデータは？

## 分類



例えばさっきの犬・猫の分類判断がまさにそれ。動物の顔の特徴（耳・目・鼻・口など）を何らかの方法で数値化してグラフ上に点としてプロットすると、これは体重のように綺麗に連続するデータとはならないで、ばらばらの点になってしまうよね。写真に対する正解ラベル（「これは犬です」「これは猫です」）は分かっているので、「犬の点の集まり」と「猫の点の集まり」があれば、その集まりの間に線を引いていける（＝学習できる）よね。そうやって線で分けられるのが離散的なデータ（図7）。他には、手書きで書いた数字画像を0～9の数値に分類するとか、そういうものが例として挙げられるよ。

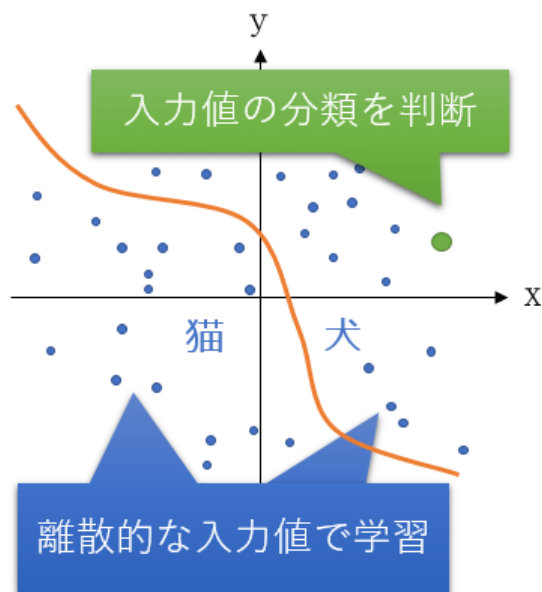


図7 離散的なデータによる分類の例



新しいデータを与えたときに、その点が「犬側に入るか」「猫側に入るか」が予想（判断）できるということか。何となく分かった。



「次の値を予測する、もしくは分類を予想（判断）する」というのがポイントで、これが教師あり学習の目的となっているわけ。回帰と分類の用途を学術的な言葉でまとめると、次のようになるよ。

- **回帰 (regression)**：連続する入力値に対する**次の値を予測**し、その結果を出力したい場合に使う
- **分類 (classification)**：離散的な入力値を、事前に定義された複数のクラスに**分類**し、その結果を出力したい場合に使う



教師なし学習も具体的な例をお願い。

## 教師なし学習

これは先ほどの教師あり学習の分類に似ているけど、違うから注意して聞いてね。例えば点が散らばっていて、「これを2つのグループに分けてください」と言ったときに、人間だったら「この辺とこの辺はグループになっている」と判断して枠線を引ける（＝学習できる）よね（図8）。機械学習でも、同じように枠線を引いてデータを複数のグループに分割していくのよ。このときの分割には、教師あり学習の場合と違って、**正解があるわけではない**というのがポイントね。分割に対する正解ラベルはない（つまり教師データがない）ので、「教師なし学習」と言われているの。

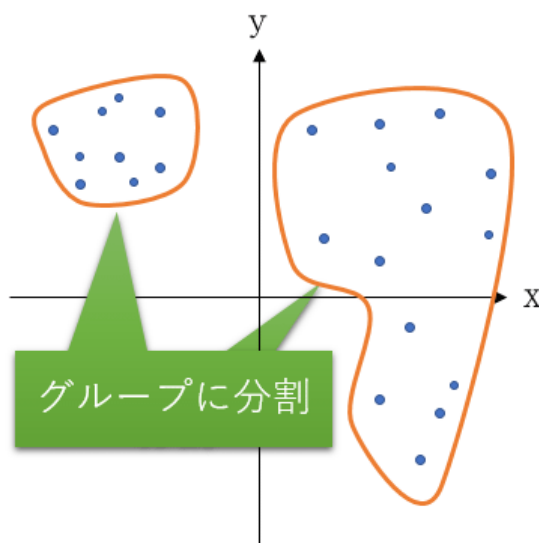


図8 離散的なデータによるクラスタリングの例（教師なし学習）

## クラスタリング



「複数のグループに分割する」って言われても、うまくイメージできないなあ……。



例を挙げるなら、商品の購買者データ（年齢、性別、居住地など）を基に顧客を無理やりいくつかのグループに分けるとか、が考えられるね。このように複数のグループ（＝クラスター）に分割することは「クラスタリング」と呼ばれているの。



何のためにクラスタリングするの？



例えばグループ内のある人が買った商品は、同じグループ内の別の人も購入する可能性が高いなどの傾向が分かれば、その商品を同じグループ内の別の人々にもお勧めするなどの活用方法が考えられるよね。回帰や分類と同じように、クラスタリングの用途を学術的な言葉でまとめておくと、次のようになるよ。



- ・ **クラスタリング (clustering)** : 入力値を、事前に定義されていないグループに分割したい場合に使う



じゃあ、「教師なし学習＝クラスタリングの問題を解決する方法」と覚えとけばいいの？

## 次元削減



いや、それだけでなく、例えばデータの圧縮や可視化を行うための「**次元削減 (Dimensionality Reduction)**」などにも、教師なし学習はよく使われているのよね。次元削減については、後で出てくる「オートエンコーダー」のところで詳しく説明するね。クラスタリングや次元削減に共通するのは、「データの背後に存在する本質的な構造を抽出している」ということ。これがポイントで、教師なし学習の目的となっているわけ。



ふうん。それじゃあ最後の**強化学習**も具体的お願い。

## 強化学習



例えば「運転する」「ゲームで対戦する」というような、ある目的に対して**望ましい行動**をさせたいプログラムを作るときに使う学習方法だね。要するに、行動方針を学習するってこと。どうやって学習するのかというと、ダイナミックに変化する環境下において、プログラムが何かしらの行動を起こしたときに、それが「良かったか」「悪かったか」というフィードバックを与えられることで学習を実現しているの。



あれ？ フィードバックと言え、教師あり学習だったよね？



教師あり学習のフィードバックは正解ラベル（教師データ）だったけど、強化学習のフィードバックは**報酬 (rewards)** もしくは**罰 (penalty)** という点が全く違うわね。これによって、正解の出力をそのまま学習するのではなく、長期的に見て価値（＝報酬の合計）が最大化する行動を「望ましい行動」として学習していくことになるの。

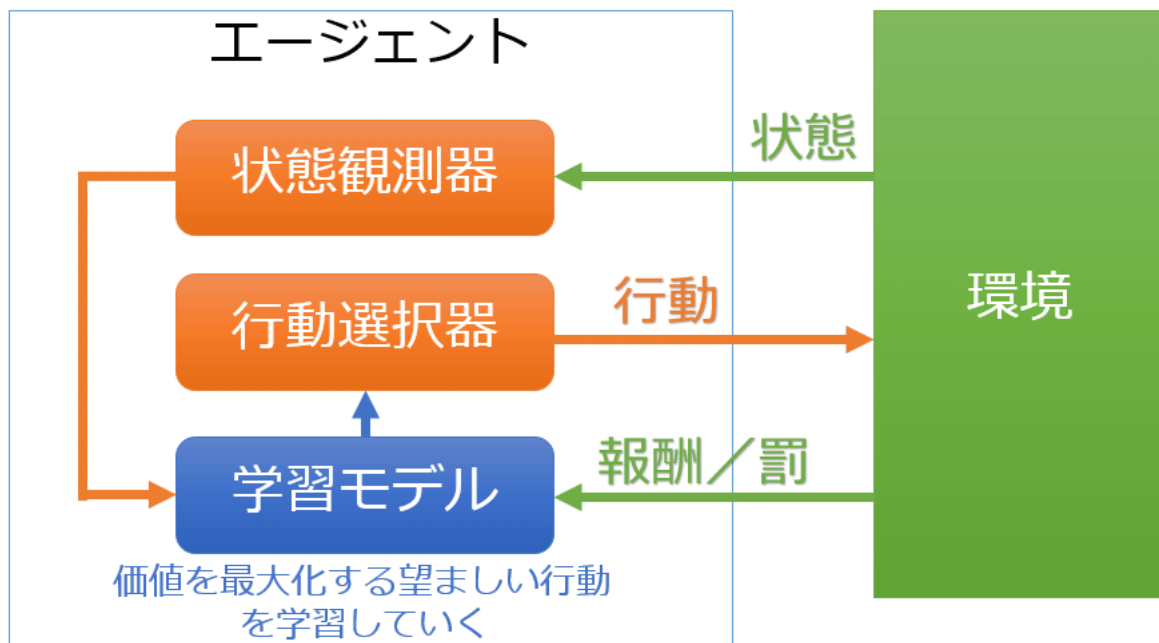


図9 強化学習のイメージ



やっぱよく分からないなあ。身近な例で言ってよ。



例えば犬のしつけを考えてみて。「お手」と言ったら、「飼主の手の上に犬自身の手を乗せる」のが、本来の望ましい行動になるよね。しつけでお手をできるようにさせたいなら、たまたまでも「お手」と言ったときに「手を乗せる」ことがあったら犬を褒めてあげる。これが報酬。ちなみに、それ以外の行動を取ったときは怒るなら、それは罰になるね。この報酬（や罰）を重ねることで、「お手」という言葉に対して「手を乗せればよいのだ」という行動パターンを学び、報酬が最大になるのが「望ましい行動なのだ」と学習していけるというわけ。

#### “望ましい行動”



ところでさっき、教師あり学習／教師なし学習が解決できる問題として「回帰」「分類」／「クラスタリング」「次元削減」という用語が出てきたけど、強化学習が解決できる問題は何て言うの？



強化学習については、ひと言で表現する用語はないと思う。強いて言葉にするなら「望ましい行動」の問題になるけど、伝わりにくいから「強化学習」としか言い様がないわね。

## ディープラーニングと学習方法

---



テレビでは、「今のディープラーニングによる AI は、自ら学習できるからスゴイ」と言ってた。これは何学習のことになるの？



話題になった囲碁の AlphaGo あたりを指して言ってるんだと思うけど、それは強化学習のことだろうね。囲碁で最終的に勝利するための行動を、自己対戦を繰り返すことなどで学習しているので、そういう言い方になるんだと思う。



それなら、ディープラーニングは強化学習を使うことが一番多いの？



いえいえ、教師あり学習の方が多いわね。例えば画像の犬／猫の判定だったり、音声の日本語／英語の判定だったり。他には自然言語だったら、直前の入力単語に対して、次に来るべき単語は何かとか。そういった分類問題や回帰問題の解決のために、ディープラーニングはよく使われているという印象を私は持っているわ。



もしかして強化学習ってあまりうまくいってないの？



確かに強化学習は、教師あり学習などよりも、なかなか成果を出しにくい難しい技術という側面はあると思うわ。だけど大きな成功事例もあって、例えば 2017 年 11 月に発表された「AlphaGo Zero」は、教師となるプロ棋士との対局データがない状態から、強化学習のみで学習を行った結果、以前の AlphaGo に全勝できるほど強くなったんだよ。他にはグーグル傘下の DeepMind 社の成果で、「Atari 2600」という昔のゲーム機のゲームプレイを強化学習したら人間を上回るスコアを記録できた、という論文が科学雑誌『nature』で 2015 年 2 月に掲載されているわね。



やっぱ強化学習ってめっちゃスゴイやん！



強化学習は自動運転でも重要な学習方法として研究されているので、今後さらに発展していく分野なのは間違いないわ。

## ディープラーニングの代表的なアルゴリズム



学習方法が主に 3 つあるんは分かったけど、どういうふうに学習するの？



いい質問ね。ディープラーニングで有名なアルゴリズムには、次の 3 つがあるよ。

- **オートエンコーダー**：主に教師なし学習に用いられる
- **CNN**：主に教師あり学習に用いられる
- **RNN**：主に教師あり学習に用いられる



うわぁ、まーたまた意味不明な名前ばかり……。で、これは何？



各アルゴリズムの内容を理解するのは、まだ難しいと思うけど、それぞれどんなことが得意なのか、ちょっとだけ紹介しておくね。詳しくは使う場面が来たらまた説明するよ。

### オートエンコーダー



まずはオートエンコーダーからお願いします。



**オートエンコーダー (Autoencoder)** は、入力層 (input layer) のデータと同じデータを出力層 (output layer) の教師データとして使う珍しいタイプのアルゴリズムよ。入力層と出力層の間にある隠れ層 (hidden layer) は、入出力層よりもノード (nodes：ネットワーク内での線と線の結び目) を減らして表現する。このため、データは入力層から隠れ層に流れるときに圧縮 (エンコード) され、隠れ層から出力層に流れるときに復元 (デコード) されるの (図 10)。

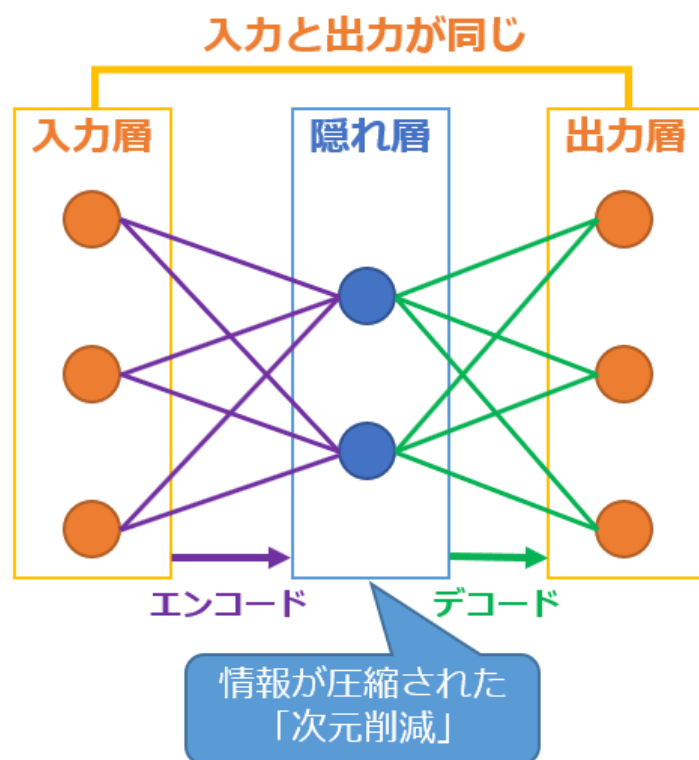


図 10 オートエンコーダーのイメージ



こうすると隠れ層の情報量は、本来の情報よりも小さくなるよね。これを**次元削減**って呼ぶの。さっきもこの単語は出てきたのを覚えてる？



覚えてる。次元削減は、データの圧縮やデータの可視化に使うという話だったよね。



具体的には、画像のノイズ除去とかで使われているわ。画像の純粋な**特徴 (features)**、つまり画像の中に描かれているエッジや角、線といった本質的な特徴のみを抽出して、そこから復元するわけ。これによって、元の画像にある揺らぎなどが抑えられて純粋な特徴だけが表現された画像になるの。



んーと、ノイズ除去って AI というより画像の加工だね。AI による問題解決には使われていないの？



今のところ、AI として使える範囲はそんなに広くないみたいね。CNN や RNN の方がよく使われていると思う。ちなみにオートエンコーダーの功績については、『**人工知能は人間を超えるか**』という本で一般人向けに分かりやすく説明されているので、気になる場合は一読してみるのをお勧めするわ。

## CNN（畳み込みニューラルネットワーク）



じゃあ次、CNN って何？



**CNN（Convolutional Neural Network：畳み込みニューラルネットワーク）**は、主に画像データから、「畳み込み」と呼ばれる処理によって局所的な特徴を抽出し、それを「プーリング」と呼ばれる処理によってぼかす（＝縮小する）ことで、位置ズレなどの揺らぎに対するロバスト性（頑強性）が高い抽象的な画像表現が得られるアルゴリズムよ。入力層と出力層の間にある隠れ層として、畳み込み層（convolutional layer）とプーリング層（pooling layer）を何度か挟んで、最後に全結合層（fully connected layer）でまとめあげることで、画像の認識といった最終的な判定が行えるの（図 11）。

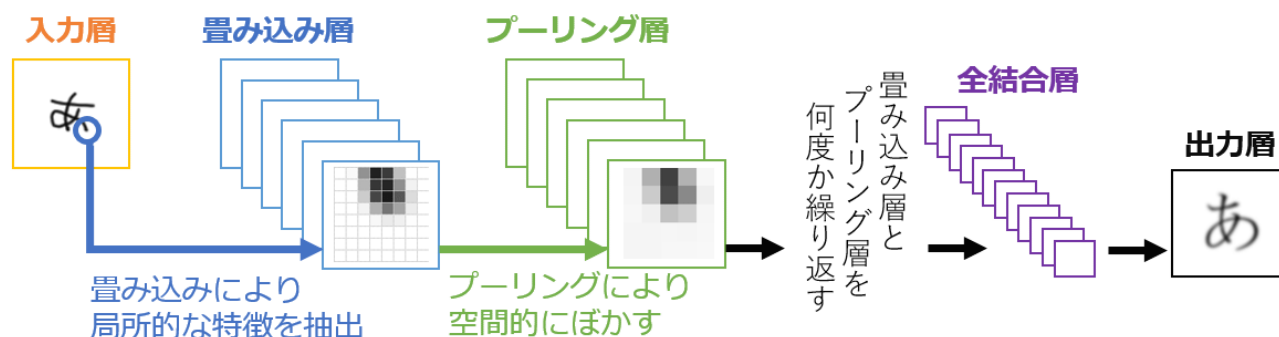


図 11 CNN のイメージ



図 11 の例だと、画像から「あ」という文字を認識しているんだよね？



そうよ。画像認識の例。CNN は、空間的構造の表現を取り扱えるので、主に画像認識や音声認識に使われているの。他には、時系列データをグラフとして書き出せば、CNN で扱えるわね。例えば証券会社のサービスで、現在の株価のグラフから「買いか、売りか」のヒントを顧客に提示するものがあるけど、こういったケースでも CNN が使えると思う。



うーん……、難しいなあ……。



そうね。CNN については、具体的な使い方を説明するときに、もっと詳しく説明するね。



## RNN（再帰型ニューラルネットワーク）



結構もう疲れてきたんだけど、最後に RNN も教えといて。



**RNN（Recurrent Neural Network：再帰型ニューラルネットワーク）**は、内部に再帰構造を持つアルゴリズム。時間方向にネットワークを展開して、幾重にも深くつなげていく際、この再帰構造により、現在の出力データを、次のネットワークの入力データとして使いながら学習できるの（図 12）。

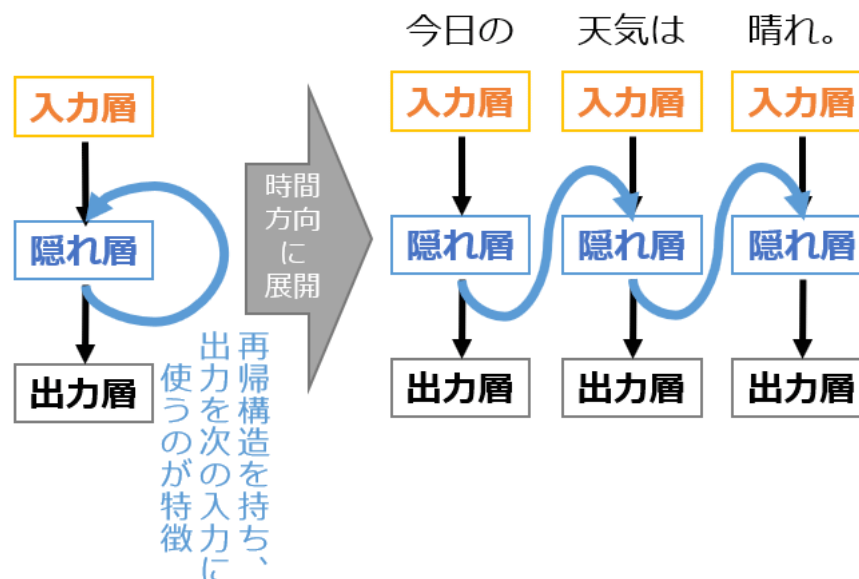


図 12 RNN のイメージ



えーと、図 12 の例は、文章を学習させているのかな？



そうよ。これは自然言語処理の例。RNN は、時間的構造の表現を取り扱えるので、主に自然言語や時系列データなど、連続性のあるデータの識別や生成に使われているの。例えば機械翻訳や、音声認識してテキスト化するサービス、文章の生成などで使えるわね。他には、画像の縦方向を時間軸として扱うことで、画像認識に応用したりするパターンも聞いたことがあるわ。



RNN を使った現実のサービスってあるの？



マイクロソフトが提供している Twitter 上で会話できる女子高生チャットボットの「りんな」というサービスは、一部に RNN を使っているらしいよ。他には証券会社のドキュメントを自動校正するツールなどで、RNN が使われているって聞いたことがあるわ。

## LSTM



ふーん。RNN については以上かな？



RNN には拡張バージョンが何種類かあって、その中でも **LSTM (Long Short-Term Memory)** が有名だから、そういうのがあるということは覚えておいてほしいな。LSTM は、短期記憶を長期間、活用できるようにしたネットワークで、長期的な依存関係を学習できるのが特徴。多くの場合で、標準バージョンより良い結果を出しているの。



はあー……、RNN もよく分からないのに、その拡張バージョンかあ……。



……。RNN についても後日詳しく説明するね。

## まとめ



うーん。もう頭もお腹もいっぱい！



そうね。今日はかなりいっぱいしゃべっちゃったね。基礎が分からなくなったら、今回の話を振り返ってみると理解が少しずつ深まると思うよ。



はい、先生！でも自分でも何かやってみたい。早くやり方を教えて。



やる気いっぱいだね。でも一歩ずつやっていこうね。チョコレートは好き？



大好き！



では明日は有給休暇を取ってあるので、チョコレート工場に一緒に行こうよ。そこで、どうやってディープラーニング（機械学習）をすればよいか、手順を教えてあげるよ。



わーい！

### 【まとめよう】機械学習／ディープラーニングを実践するための基礎知識

- **機械学習における手法／アルゴリズム／モデル**：モデルを作成する手順・方法／学習前の抽象的な計算方法／学習後の具体的な計算方法
- **機械学習の学習／推論**：モデルを「作る」こと／モデルを「使う」こと
- **機械学習の代表的な学習方法**：教師あり学習、教師なし学習、強化学習
- **教師あり学習**：正解ラベル（教師データ）がある学習。回帰問題や分類問題を解決するのに使う
- **教師なし学習**：正解がない学習。クラスタリング問題を解決したり、次元削減をしたりするのに使う
- **強化学習**：報酬によって、長期的に価値を最大化する「望ましい行動」をするように学習する
- **ディープラーニングの代表的なアルゴリズム**：オートエンコーダー、CNN、RNN
- **オートエンコーダー**：入力と出力を同じにするアルゴリズム。主に次元削減に使う
- **CNN**：畳み込みとプーリングの層を持つアルゴリズム。主に画像認識や音声認識に使う
- **RNN**：内部に再帰構造を持つアルゴリズム。LSTM という拡張バージョンが有名。主に自然言語や時系列データの識別・生成で使う

## Lesson 3 機械学習&ディープラーニングの基本的なワークフローを知ろう

機械学習／ディープラーニングの作業フローの基礎を学び、実践へ踏み出す準備をしよう。機械学習モデルは、どのようなステップで作成していくのか？ データ収集～学習～運用の一連の流れをできるだけシンプルに紹介する。

(2018 年 04 月 18 日)

ご注意:本記事は、@ IT / Deep Insider 編集部（デジタルアドバンテージ社）が「deepinsider.jp」というサイトから、内容を改変することなく、そのまま「@ IT」へと転載したものです。このため用字用語の統一ルールなどは@ IT のそれとは一致しません。あらかじめご了承ください。（インタビュー取材協力：DATUM STUDIO 安部 晃生）

### 登場人物紹介



#### 深井藍（ふかい・あい）博士

最新の人工知能技術を応用して、次世代の人型ロボット（アンドロイド）を開発するのが仕事。試行錯誤の末にやっと開発できたのがマナブ（01 号）である。

責任感が強く頑固で読書家だが、ドラマ好きで、超天然な一面もあるアラサー リケ女。

ちなみに藍が使っているタブレットには、マナブの学習状況をチェックできる機能だけでなく、万が一の安全対策としてマナブの暴走を制御するための「秘密機能」が搭載されているという。



#### マナブ（01 号）

現実社会の学習を進めるため、藍博士と 24 時間生活を共にしている次世代アンドロイド、0 歳。

見るもの聞くものすべてに興味津々。藍が好きなテレビドラマとお笑い番組からも学習しているため、うわすべりな知識から勘違いな行動を取ったり、大阪風のボケをかまししたりすることもある。

この物語の主人公。エンジニアスキルはあるけど機械学習やディープラーニングについてまだ何も知らない。

### ディープラーニングを教え始めて 3 日目

現実社会の学習を深めようと、毎日、マナブをいろんなところに連れて行く藍博士。ワークフロー（作業工程）の概念について理解させるため、有給休暇を利用して埼玉県坂戸市のチョコレート工場を見学を訪れたときのこと。

## マナブ、機械学習のワークフローを学び始める



図1 チョコレート工場



マナブがたまにおやつで食べている板チョコレートって、どうやって作られているか、知ってる？ 意外にたくさん手順（ステップ）を踏んで手間暇をかけてやっと完成しているのよね。



へえ。そんなに大変なの？



そうよ。実は、機械学習やディープラーニングで何らかの AI サービスを作るまでのワークフローも、チョコレートみたいに意外と手順がたくさんあって大変なの。だから今日は、チョコレートの製造工程を見ながら、機械学習のワークフローのイメージをつかんでもらうために、チョコレート工場の見学に来たんだよ。

## チョコレートの製造工程



楽しみ～。しかもさっき、受付でお土産のチョコをもらっちゃった。



良かったね。これからチョコレートが出来るまでの工程を説明してくれるそうだから、まずは聞いてみよう。大きく分けて次の工程があるんだって。

- 原料：チョコレートを作るための材料を準備する
- 製造：チョコレートを作る
- 出荷：チョコレートを小売店に卸す





分かる分かる。機械学習なら次のような感じだね。

- 材料：機械学習のモデルを作るためのデータを準備する
- 作成：機械学習のモデルを作る
- 納品：機械学習のモデルを AI サービスとしてユーザーに提供する



そのとおり！チョコレートの製造工程の流れをまとめたのが次の図。詳しい意味は「[チョコレートができるまで（ようこそ明治の工場見学へ）](#)」などが参考になるわね。



図2 チョコレート製造工程の流れ



うわぁー、手順が長いなぁ。想像以上に複雑な工程を経て出来てたんだね。特に (3) の前処理は手順数が多くて大変そうだ。



そうだね。同じような感じで、機械学習の作業工程の流れを図にすると次のようになるよ。



## 一般的な機械学習のワークフロー



図3 機械学習のワークフロー



うわぁ！機械学習もチョコレートに負けず劣らず手順が多いなあ。意味がよく分からないものがいくつかあるから、最初から1つずつ説明して。

### (1) データの準備



図4 データの準備



いいよ。まず初めに、「どんな AI サービスを作りたいのか」という目標が決まったら、そのためにどんなデータが必要かを検討するの。そして実際にデータを集める。例えば「犬と猫の画像判定を行う AI サービスを作りたい」としたら、どうすればいい？



犬と猫の画像のデータを集める必要があるよね。



そう。自分が大量にデータを保有していたり、自分で収集可能だったりするのなら、それを使う。もしそうでない場合は、誰かがどこかで提供してくれているデータセット（＝データ集。例えば [Kaggle 犬猫画像](#) / [MNIST 手書き文字画像](#) / [ImageNet 画像データベース](#) などがある）をダウンロードするか、インターネット上をクロール **\*1** しながら **Web スクレイピング (Web scraping)** **\*2** して画像をかき集める必要があるわね。その際、もちろん利用許諾などには注意する必要があるよ。

**\*1** インターネット上のファイル群を自動的に検索して、HTML ファイルや画像ファイルなどの情報データベースを作成していくこと。代表例としては、Google 検索用の情報データベースを作る「Googlebot」がある。

**\*2** HTML ソースから必要な情報だけを抽出すること。



スクレイピングは以前にもやったことがあるなあ。この手順は今の自分でもできそうだよ。



（な、何ドヤ顔なのよ。私の開発力が優秀ってことじゃないの!）

## (2) 手法の選択

### ② 手法の選択

機械学習の“手法／学習方法／アルゴリズム”を選ぶ

図 4 手法の選択



次に、どのような手法を採用するかを検討する。「ディープラーニングがいいか？ ディープラーニングなら、CNN か？ RNN か？」など、適切な機械学習の手法や、学習方法、アルゴリズムを選んだり、（まれだけど）適切なものがなければ独自に設計したりする。ちなみに、今回はディープラーニングを中心に話しているわけだけど、実際にはディープラーニングよりも、他の機械学習手法の方が精度が良かったり、効率的に作成できたりする場合があって、実際に現場ではディープラーニング以外を選択するケースも多いのよ。



ふ〜ん。この手順は、幅広い機械学習の知識と経験が必要そうだね。ボクにはまだ難しいかも。



そうね。まずはディープラーニングができるようになってから、少しずつ知識と経験を広げていくといいと思うわ。

### (3) 前処理

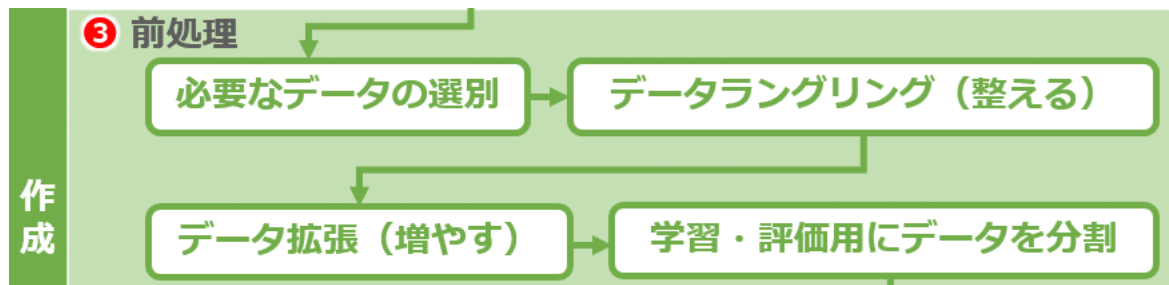


図5 前処理



で次に、収集しておいたデータから、必要なデータを選別する。ただし、選別したデータがそのままの形ですぐに使えるというのは、練習用データでもない限り、一般的な実用データではあまりないの。例えばデータの一部が欠損していたらそこにはランダム値を入れたりだとか、数値のフォーマットやテキスト項目がデータ入力者によってバラバラで統一されていなかったら名寄せ処理をしたりだとか、そのままでは機械学習で使用できないデータを何とかして使用可能な形に整える必要がある（＝**データクレンジング：Data cleansing**、もしくは**データクリーニング：Data cleaning**）。「暴れん坊の牛や馬を集めて飼いならす」みたいに力技が必要で大変だから、**データラングリング（Data wrangling）**って呼ばれたりもするのよね。



ネットで検索したら、“**wrangler**” は、馬などの家畜を操るために雇われたカウボーイのことだって書いてある。カウボーイかぁ…、大変そうだね。



ちなみに機械学習では、より良い結果が出るように、データの数値を前処理するためのテクニックがいろいろとあるの。最も代表的なものには次のものがあるよ。

- **正規化（Normalization）**：トレーニングデータの値が **0 ～ 1** などの指定範囲に収まるように、値を加工するテクニック
- **標準化（Standardization）**：トレーニングデータの平均が **0**、分散が **1** になるように、値を加工するテクニック



正規化…標準化…!? なんかにピンとこないなぁ…。



そうだね。そのあたりはそのうち具体的な例を見ながら学んでいくとしよう。



数値は前処理が必要として、画像データの場合は、前処理しなくていいのかな？



画像の場合も、前処理をすることで精度が良くなることが多いの。例えば人の顔を画像認識させたいのであれば、顔をあらかじめ切り抜いておいた方が学習しやすいよね。このような画像や映像の加工には、**OpenCV**というライブラリがよく使われているのよ（参考：「[OpenCV 入門](#)」）。



じゃ、文章データの場合は？



自然言語のテキストデータを、品詞情報に基づき、意味の最小単位である「形態素（けいたいそ）」に分解する技術を形態素解析（Morphological Analysis）と呼ぶんだけど、まずはデータの前処理として形態素を切り出しておくの。日本語の形態素解析では、**MeCab**というライブラリがよく使われているよ。



OpenCVとMeCabは、暇なときに試してみるよ。あと「データ拡張」というのは？



主に、不足する画像データを補うために使われるテクニックで、その内容は次のとおり。自前で用意するデータは不足しがちなので、最近では頻繁に利用されているのを見かけるわね。

- **データ拡張 (Data augmentation)**：トレーニングデータの画像に対して移動／回転／拡大／縮小／歪曲／ノイズ付加などの操作をすることで、データ数を何倍にも増やすテクニック



なーるほど。画像を加工して、たくさんの画像があるものとして学習させるわけだ。それにしても、データを学習する前には、いろんなことをやるんだね。



前処理で学習結果が変わってくるから、意外に重要なのよね。手順 **(5)** からここに戻ってきて、試行錯誤しながら機械学習モデルの再作成を何回も繰り返すこともあるので、意外にこの手順 **(3)** の作業時間が一番長かったりするのよ。



「学習・評価用にデータを分割」とあるけど、データを分けるの？



教師あり学習では、何度も登場しているトレーニングデータとは別にあと2つ、合わせて下記の3つのデータセットに分ける必要があるの。

- トレーニングデータ (training data、もしくは訓練データ)
- 精度検証データ (validation data、もしくは評価データ: evaluation data)
- テストデータ (test data)



教師あり学習以外は？



教師なし学習や強化学習は、正解ラベルと一致するかチェックしてパフォーマンス（性能）を評価する必要がないから、精度検証データは用意せずに、

- トレーニングデータ
- テストデータ

の2つのデータセットに分割すればいいよ。



精度検証データとテストデータは何が違うの？



精度検証データは、手順 (5) の「モデルの評価」で使うチューニング用データのことだよ。一方、テストデータは、同じく手順 (5) で完成候補のモデルに対して使う最終テスト用のデータね。学習やチューニングに使ったデータは良い結果が出るようにバイアスがかかっている可能性があるの、そういったバイアス問題などを回避するためにも、最終テストには全く未使用のデータを使った方がいいから、この2つは分けておくの。



確かに、学習に使っていないデータが全く残っていないと、本番環境の AI サービスをチェックする際にも困りそうだもんね。気を付けないと。で、全てのデータに対して、どれくらいの割合で分割すればいいの？



それはケース・バイ・ケースなので一概に言えないよね。強いて目安を挙げるとすると、3つのデータセットに分割する場合は、

- トレーニングデータ：70%
- 精度検証データ：20%
- テストデータ：10%

ぐらい。2つのデータセットに分割する場合は、

- トレーニングデータ：80%
- テストデータ：20%

ぐらいがよいと思うよ。でも、データ量がかなり多いとき、例えば数百万のデータがあるなら、テストデータに数十万も必要ないだろうから、通常は精度検証データやテストデータの割合をもっと減らして、その分、トレーニングデータの割合を増やした方がいいわね。

#### (4) モデルのトレーニング

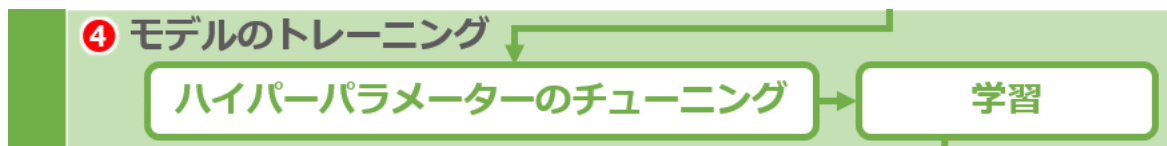


図6 モデルのトレーニング



次がよいよ学習だよ。あれ、「トレーニング」って書かれているけど、トレーニングと学習は意味が違うの？



微かな違いだけど、**トレーニングデータ**を使って、機械学習のモデルを**トレーニング**（訓練）していくことが「学習」、と理解しておくといいよ。ちなみに、ディープラーニングのニューラルネットワーク（＝**ディープ・ニューラル・ネットワーク：Deep Neural Network**と呼ばれる。以下、**DNN**）においては、次のような意味になるわね。

- **トレーニング (training)**：機械学習アルゴリズムを活用して、ネットワークの**重み (weight、重み付け)**を変えていくこと
- **学習 (learning)**：機械学習アルゴリズムを活用してトレーニングしながら、ネットワークモデルを構築していくこと





あと「ハイパーパラメーターのチューニング」って、カタカナが多くて読み間違えちゃいそうだけど、これは何？



DNN では、モデルの構成や設定はトレーニングプロセスによってある程度自動的に調整・最適化されるんだけど、一部には人間がチューニング（調整）しなければならない設定——例えばネットワークの層の数やユニット数など——があって、そういった設定を**ハイパーパラメーター（hyperparameter）**と呼んでいるの。ハイパーパラメーターが最適な値になるまで、次の手順（5）の「モデルの評価」からこの手順（4）「モデルのトレーニング」に舞い戻ってきて、「チューニング～トレーニング～評価」を何度も繰り返し実行していく必要があるの。でも最近は、最適なハイパーパラメーターを探すための機能をツールが提供してくれていたりもするので、比較的簡単にチューニングできるようになってきてはいるのよ。

あと、**過学習（Overfitting）**：学習で過去のデータにフィットし過ぎると、新しい未知データには逆にフィットしにくくなる現象）を抑制するために**正則化（Regularization）**という処理が行われることも多いから、この言葉も覚えておいてね。

## （5）モデルの評価



図7 モデルの評価



手順（5）の「推論」は前回の説明で出てきたね。学習済みモデル（**トレーニング済みモデル：Trained model**とも呼ばれる）を使用することだったよ。



そう。よく覚えてたね！ **推論（inference、もしくは推定）**とは、モデルにデータを入力して、そのモデルから結果を出力として受け取ること。つまりここでは、教師あり学習であれば、**精度検証データ**を使ってモデルの**精度（accuracy）**をチェックするということね。



チェックって、どうやってするの？



通常は出力と正解ラベルの値を比較して、モデルに適した統計的技法を使用することで「どれくらいの精度が出ているか」という数値的な性能指標（つまり成果）を見るわね。他には例えば画像生成などであれば、出力画像などを人が見て問題ないかを検証することもあると思う。



あとは、先ほどのチューニングに戻って、いろんなパターンでモデルを作ってから評価していけばいいんだよね？



そうそう。そうやって、学習する際のパラメーターをちょっとずつ変えながら、できた複数の学習済みモデルの中から、最も精度が高いモデルを選択すればいいの。



これで完成かな？



最後に、未知のテストデータに対する学習済みモデルのパフォーマンス（汎化性能:**Generalization performance**）を評価しておいた方がいいよ。すでに説明したとおり、精度検証データにはデータ慣れによるバイアス問題の可能性があるから、あらためて真新しい**テストデータ**を使って、運用環境にできるだけ近いコンテキストで最終チェックするの。



あ、そういう話もあったね。ちなみに、どうやっても高い精度が出なかった場合はどうすればいいの？



「**(2)** ～ **(5)** を繰り返す」と書いておいたように、「機械学習の手法の選択」にまで戻ってやり直すことも考えた方がいいね。



え～！？ **(2)** って振り出しみたいなものじゃないの！ そりゃ大変だなあ。

## (6) 納品・本番運用



図 8 納品・本番運用



最後に、選択した学習済みモデルを、運用環境で実行できる形でエクスポートして、AI サービスやアプリケーションに組み込めば完成。よくあるパターンは、機械学習エンジニア以外の開発者でも利用しやすいように、**Web API** として呼び出せるようにサーバー（主にクラウド）を用意して、その AI を使いたい開発者や会社に引き渡すことね。



IT のソフトウェア / Web 開発は得意なので、ここは自前の知識でできそう。でも、手順 (2) ~ (5) をマスターするまでには、何カ月ぐらいかかるんだろう……。

## 本シリーズで最初にフォーカスする領域



そうだね。筋が良ければ半年ぐらいで、何とか 1 人で機械学習できるようになる人はいるよ。だけど、機械学習が本当に適切にできるようになるまでには、たくさん覚えることがあるのは確か。だけど心配しないで。一步一步、「習うより慣れろ」で、ちょっとずつできることを増やしていったらいいから。任せて。



藍先生、よろしくお願いします！



……

(あら、急にしおらしくなっちゃって……) もちろん正攻法なら、データサイエンスの基礎から丁寧に学習すべきなんだけど、このシリーズでは、特に「(2) 手法の選択」と「(4) モデルのトレーニング」にフォーカスして具体的に説明していくよ。それ以外の (1) / (3) / (5) / (6) については概要紹介だけにして、具体的な説明には踏み込まないようにするね。



はい。(シリーズ??? よく分かんないけど、スルーしよっと)



明日から、まずは藍のやることをまねてみて。特に (4) の工程は、**ライブラリ**という便利なものがあるから意外と簡単なんだよ。だからソフトウェア開発スキルのあるマナブであれば、とりえずディープラーニングを体験することは今すぐにできるはずだよ。



よかった！ところでディープラーニングのライブラリにはどんなのがあるの？

## ディープラーニングの代表的なライブラリ



ディープラーニングが扱える代表的なライブラリには、

- **TensorFlow** : グーグルが開発し、世界的にも有名で人気があるライブラリ。ラッパーライブラリの Keras (「ケラス」と読む) と組み合わせて使われるパターンも多い。読み方は、英語風の「テンサーフロー」、もしくは数学のテンソルに基づく日本語風の「テンソルフロー」
- **Chainer** : 日本の Preferred Networks (プリファード・ネットワークス) が開発し、特に国内で人気が高いライブラリ。そのため日本語情報も多い。読み方は「チェイナー」

があるよ。どちらもできることに大きな違いはないから、あとは使ってみてどちらの使い心地が良いかで決めればいいと思う。好みのプログラミング言語を選ぶのと一緒にね。



どちらもマスターしたい!!!



このシリーズでは、まず TensorFlow を使ったパターン、次に Chainer を使ったパターンで、ディープラーニング実践の基礎を一通り学んでいくよ。だから各ライブラリが、具体的にどういうものかは、それぞれの回で紹介するね。



はい。必死でついていきます!

## 心構え：ソフトウェア開発や Web 制作との違い



今日は最後に、マナブのようなソフトウェア / Web 開発が得意な人が、機械学習に取り組む際の心構えについても説明しておくね。



えっ、何?



一般的にソフトウェアや Web の開発は、「具体的な完成目標」があるよね。最初の **0%** の状態から作り始めて、完成度の%が徐々に上がっていき、**100%** になったら、ソフトウェア / Web をリリースするイメージ。目標に向けて前進して積み上がっていくから、常に達成感を感じられる作業なのではないかと思う。基本的に自分が思ったものが思い通りに作れる。いわゆるクリエイティブ職ね。



機械学習は違うの？



機械学習は、ここまでに「試行錯誤」「繰り返す」という言葉が何度か出てきたように、「調整～作成～評価」を繰り返しながらベストな精度を探していく作業になるの。完成度というか精度が **100%** になることは基本的にないのよ。自分が思ったものが思い通りに作れるとは限らないの。しかも、より良い精度が出せるように、最新の手法やアルゴリズムも追いかけて試していく必要がある。だから**機械学習エンジニア**は、いわゆる研究職にかなり近いエンジニア職だと思うの（ちなみに**データサイエンティスト**は研究職にかなり近いデータアナリスト職のことね）。もちろんプログラミングスキルはムダにはならないんだけど。



大丈夫。試行錯誤しながら作り上げるのも大好きだから。



それは良かった。あと、ライブラリがプログラミングの難しい部分をかなりカバーしてくれているから、機械学習のプログラミング作業は、ソフトウェア／Web 開発のそれと比べると、圧倒的に短くて簡単になるわね。まとめると、「プログラミングする」というのはあくまで目的を実現するための補助的な作業となって、「手法を選んでモデルを学習させて精度を高める」という今日説明した作業が、機械学習エンジニアの日常的なメインの仕事になるよ。

## まとめ



了解！それにしてもチョコレート工場、そっちのけだったね。



（うーん、確かに……。没頭すると他のことが見えなくなっちゃうのよね、わたしは……）あはは……。とりあえずお土産にもらった新製品の板チョコを食べてみようよ。

——マナブが板チョコを食べて、白目を抜いて「うああああ」と奇声を上げる——



テレビドラマの甘太朗（かんた〇〇）の真似 www



……。 (前回の冒頭で「忘れた」と思っていたのに、まだ覚えていたのか……。)



あっ、前みたいに暗黒にするのはナシやで。





(ぎくうー) そ、そ、そんなことしないよ。えへへ、へ、へ……



で、次回のおやつは何？



それは秘密だよ。次回からは手作業しながら学ぶから、会話形式ではなくなるので。



え〜！（泣）



ここまでの話は、ディープラーニングをマスターする冒険に出発するための準備体操のようなもの。これから長い旅が始まるから頑張ってね。



はい、藍先生！（キリッ）



次は、GPU 付きの PC がある人は、次回「[機械学習&ディープラーニング環境構築入門](#)」に進んでね。それ以外の人は「[ゼロからのディープラーニング最速入門【TensorFlow 編】](#)」か「同（Chainer 編）」に進んでね。



## 【まとめよう】ディープラーニングの基本的なワークフロー

- **機械学習のワークフロー**：以下の **(1) (2) (3) (4) (5) (6)** の順に進め、**(2) ~ (5)**（特に **(4) ~ (5)**）を繰り返すことで、精度を高める
- **(1) データの準備**：使用するデータを決めてから収集する。公開データセットの利用や、Web スクレイピングによる収集などの方法もある
- **(2) 手法の選択**：最適な結果が出せる機械学習の手法を選択する。ディープラーニングであれば、学習方法やアルゴリズムも決定する
- **(3) 前処理**：データを選別。機械学習で使用可能な形に修正する（データラングリング）。変換テクニックを活用。データを分割する
- **データ変換テクニック**：正規化／標準化、画像・映像の加工（OpenCV など）、形態素解析（MeCab など）、データ拡張など
- **データセットの分割**：トレーニングデータとテストデータに分割。教師あり学習では、さらに精度検証データも分割して用意する
- **(4) モデルのトレーニング**：トレーニングデータで機械学習モデルをトレーニングする（＝学習）。ハイパーパラメーターを調整する
- **(5) モデルの評価**：評価データで推論して学習済みモデルの精度をチェック（教師あり学習）。最後にテストデータで汎化性能も評価する
- **(6) 納品・本番運用**：選択した学習済みモデルをエクスポートして、AI サービスやアプリケーションで使えるようにすれば完成
- **ディープラーニングが扱える代表的なライブラリ**：TensorFlow や Chainer が特に有名。本シリーズで使い方の基礎を解説していく

