

# 1章

## はじめに

我々人間は、日々五感を通して得られる情報から、対象を分類する能力や、事象の背後にある規則性を獲得してゆきます。これらと同じ能力をコンピュータに与えることを目指したものが、機械学習です。本章では、近年機械学習が注目されている背景を紹介し、本書で扱う機械学習の概要について説明します。

### 1.1

### ビッグデータの時代

近年、ビッグデータ (big data) というキーワードのもとに、様々な技術開発やビジネス展開がなされています。2012年3月に米国が巨費を投じると発表した「ビッグデータ研究開発イニシアティブ」<sup>1</sup>は、ビッグデータの収集・管理・分析のための中核技術の発展、科学技術分野での発見の加速、国家安全保障の強化、ビッグデータ技術分野の人材育成を目指すもので、ビジネスを超えた幅広い分野に、ビッグデータの影響が広がっています。

ビッグデータの特徴は、大量・多様・スピードだといわれています。情報機器・センサーネットワークの発達で、全世界のデータ量は、1日あたりエクサバイト<sup>2</sup>単位で大量に増加しているとみられています<sup>3</sup>。そして、その内容は人々がブログやSNS (social networking service) に投稿する文章・画像・動画、スマートフォンなどに搭載されているセンサーから収集される情報、オンラインショップやコンビニエンスストアの販売記録、IC乗車券の乗降記録など、多種多様なものです。また、それらのビッグデータを有効に活用するためには、必要なときに、必要な結果が得られるというスピードが要求されます。大量・多様なデータをスピードをもって処理することで、購買記録からお勧め商品を提示するようなおなじみの機能に加えて、不審な行動パター

1 [http://www.whitehouse.gov/sites/default/files/microsites/ostp/big\\_data\\_press\\_release.pdf](http://www.whitehouse.gov/sites/default/files/microsites/ostp/big_data_press_release.pdf)

2 1エクサバイトは1テラバイトの百万倍、1テラバイトは標準的なデスクトップPC 1台のハードディスクの記憶容量です。

3 <http://www-03.ibm.com/press/us/en/pressrelease/35633.wss>



図 1.1 ビッグデータの活用例

ンの検出や、インフルエンザの流行の予想など、これまでになかったサービスや機能を実現することもできるのです（図 1.1）。

ビッグデータは、インターネットの発展や様々なセンサ・通信機能をもった携帯端末の普及の結果としてもたらされましたが、その大量・多様なデータの効率のよい利用に関しては、これからますます技術開発が進められてゆく分野です。ビッグデータの応用に関して最も注目されている分野として、**データマイニング** (data mining) があります。データマイニングは、それ自体は単なる記録の蓄積に過ぎないデータの山から、貴重な知見を発見（マイニング＝発掘）しようとする試みです。

データマイニングの有用性を示す古典的な例として、スーパーマーケットの売り上げデータの分析から、「木曜日の夕方、スーパーマーケットで紙おむつを買う人はビールも買うことが多い」（だから、紙おむつの近くにビールを置けば、売り上げが増える可能性がある）ことがわかったというものがあります（図 1.2）。この話は、1992 年 12 月 23 日の「ウォールストリートジャーナル」に掲載された“Supercomputer Manage Holiday Stock”という記事が発端らしいのですが、その後、不正確な伝聞が広がって、一種の伝説のようなものになってしまいました<sup>1</sup>。後付けの理屈として、「紙おむつは



図 1.2 データマイニングの事例「紙おむつとビール」

<sup>1</sup> ITmedia 情報マネジメント用語辞典：おむつとビール

<http://www.itmedia.co.jp/im/articles/0504/18/news086.html>

かさばるので父親が買い出しに行くことが多く、そのついでに週末に飲むビールを「買うことが多い」という説明がなされています。普通の発想では無関係に思える木曜日・紙おむつ・ビールの3項目に対して、売り上げデータを通じてその関連性が抽出されたということで、データマイニングについての話の導入として、この「紙おむつとビール伝説」が取り上げられることが多いようです。

データマイニング以外にも、データを利用して識別・予測の精度を高めることで、省力化やサービスの向上を図るものは数多くあります。迷惑メールのフィルタや、特定の製品に関する SNS での評判情報の自動収集などは、言語データを対象とした機械学習の成果です。また、数値データを対象としたものには、過去のデータから今後の数値を予測する手法を用いた新製品需要の予測や、外れ値を検出することによる異常検出があり、いずれもビッグデータの応用例として期待されているものです。

その名のとおりビッグデータは大量であり、その処理にはスピードが要求されますので、人手による情報抽出は見込めません。また、ビッグデータの情報源は多様なので、必然的に矛盾・曖昧性・誤差を含みます。これらのデータを処理する規則を人手で見つけ出すのは非常に難しいことです。そこで、機械が知識や規則を発見し、それを使って推論・識別・予測をおこなうという、いわゆる**機械学習** (machine learning) の技術が必要になります。

## 1.2

## 機械学習とは何か

「人手で扱えない」ものはすべて「機械学習」を使うべき、というわけではありません。大量のデータを特定の順番に並べ替えることは、人手では気の遠くなる作業量でも、ソーティング（並べ替え）プログラムを使えば可能です。同じく大量のデータを扱う場合でも、特定の条件を満たすデータの検索・抽出・平均値の計算などの問題は、その解法であるアルゴリズムがすでにわかっています。

これに対して、機械学習で扱うのは、解法が明確にはわかっていない問題です。ここでは、機械学習で対象とする問題を**タスク** (task) とよびます。

たとえば、人間は文字や音声の認識能力を学習によって身につけますが、どうやって認識をおこなっているかを説明することはできません。その人間の認識能力を何らかの手法で**モデル** (model) 化して、コンピュータでその能力を再現しようとする技術が、機械学習の一種である**パターン認識** (pattern recognition) とよばれる問題です。

ここでのアイデアは、明示的にその手順は記述できないけれども、データ（この場合は入力とその答え）は大量に用意することができるので、そのデータを使って人

間の知的活動（場合によってはそれを超えるもの）のモデルを作成しようというものです。これ以降、機械学習のために用いるデータのことを**学習データ** (training data) とよびます。

ここまでをまとめると、機械学習の基本的な定義は、

アルゴリズムとして明示的に解法が与えられないタスクに対して、そのタスクを遂行するためのモデルを、学習データから構築すること

となります。

また、文献[1]では、コンピュータプログラムが学習するとは

- あるタスクにおいて
- ある学習データによって
- モデルの性能測定基準の値が向上すること

であるとされています。単にモデルを構築するだけではなく、その性能測定基準の値が向上するというのを、機械学習の定義として定めています。

単に知的なモデルを作って「それで終わり」というわけではなく、学習を続けて性能が向上しつづけることが、定義の中に含まれています。たとえば、迷惑メールフィルタを考えます。製品によって相対的に性能の良し悪しはありますが、誰にとっても最高の性能であるような迷惑メールフィルタはありません。各人の利用環境によって迷惑メールの基準が異なるので、利用者が「これは迷惑メール」、「これは迷惑メールではない」と操作した履歴に基づいて再学習をおこない、その利用者の操作の結果が適切に分類に反映されれば、このプログラムは学習能力が高いといえるでしょう。

これらの定義に沿って考えてゆくと、機械学習のタスクとしては、推論・認識・予測・適応などを伴う、人間の知的活動の様々な分野がその対象となりそうです。これらは人間の高度な知的活動ですので、実現するのは非常に難しそうですが、これらの機能が規則や関数などで表現できると仮定すると、それらを性能測定基準に沿って最適化してゆく作業を機械学習とみなすことで、適切な理論化がおこなえそうです。また、学習データも一見多様にみえますが、購入した商品名や天候のような「ラベルの集合」、あるいは金額やセンサーからの入力のような「数値の集合」、さらにはそれが入り交じったもの、というように整理して考えることができます（図 1.3）。

機械学習の役割をこのように位置付けると、図 1.3 中の「機械学習」としてまとめられた中身は、タスクの多様性によらず、目的とする規則・関数などのモデルを得るために、どのような学習データに対して、どのようなアルゴリズムを適用すればよいか、ということを決める学習問題と、その学習の結果得られたモデルを、新たに得ら

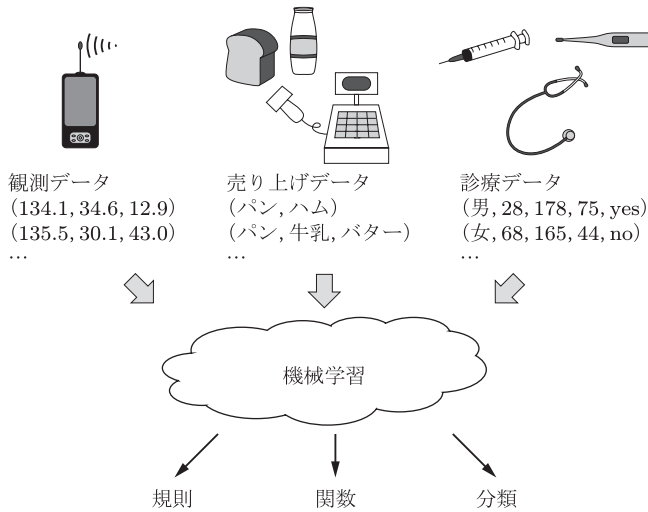


図 1.3 機械学習の位置付け

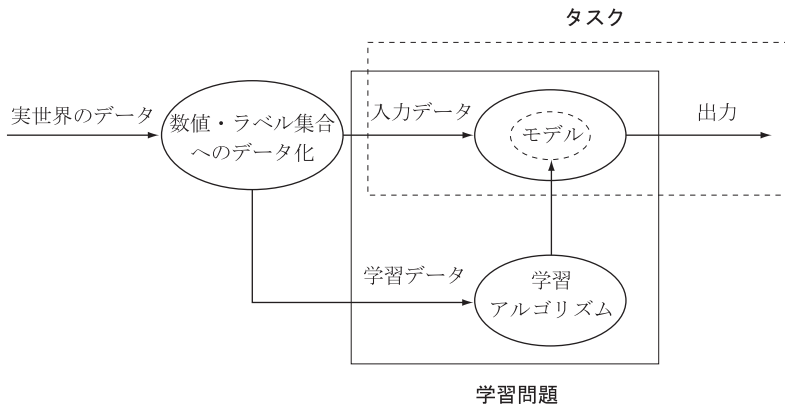


図 1.4 機械学習の要素

れる入力に対して適用する実運用段階に分割して考えることができます (図 1.4)。

本書の対象は、主として図 1.4 の学習問題と定義された部分ですが、いかに実運用の際により性能を出すか、すなわち、学習段階ではみたことのない入力に対して、いかによい結果を出力するか (この能力は、「学習データからいかに一般化されたモデルが獲得されているか」、ということになるので、**汎化 (generalization) 能力**といいます) ということを常に考えることになります。

## 1.3

## 機械学習の分類

ここでは、前節で説明した学習データと学習結果を基準に、機械学習の分類を試みます。機械学習には様々なアルゴリズムがあり、その分類に関しても様々な視点があります。機械学習の入門的な文献では、モデルの種類に基づく分類がおこなわれていることが多いのですが、そもそもそのモデルがどのようなものかというイメージをもっていない初学者には、なかなか納得しにくい分類にみえてしまいます。そこで本書では、入力である学習データの種類と出力である学習結果の種類の組み合わせで機械学習のタスクの分類をおこない、それぞれに分類されたタスクを解決する手法として、モデルを紹介します。

まず、学習データにおいて、正解（各データに対してこういう結果を出して欲しいという情報）が付いているか、いないかで大きく分類します（図 1.5）。学習データに正解が付いている場合の学習を**教師あり学習** (supervised learning)、正解が付いていない場合の学習を**教師なし学習** (unsupervised learning) とよびます。また、少し曖昧な定義ですが、それらのいずれにも当てはまらない手法を**中間的手法**とよぶことにします。

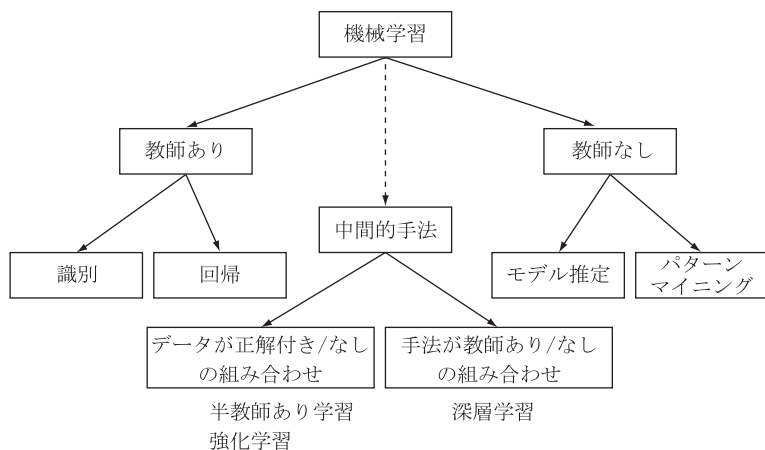


図 1.5 機械学習の分類

教師あり／なしの学習については、それぞれの出力の内容に基づいてさらに分類をおこないます。教師あり学習では、入力をあらかじめ定められたカテゴリーに分類した結果を出力するものを識別とし、入力から予測される数値を出力するものを回帰と

します。一方、教師なし学習では観点を変えて、入力となるデータ集合全体を説明する情報を出力するものをモデル推定とし、入力となるデータ集合の一部から得られる特徴的な情報を出力するものをパターンマイニングとします。

中間の手法に関しては、何が「中間」であるのかに着目します。学習データが中間である場合（すなわち、正解付きの学習データと正解なしの学習データが混在している場合）と、学習手法が中間である場合（すなわち、教師あり／なしの両方の手法を組み合わせる場合）とで分類します。

以下では、それぞれの分類について、その問題設定を説明します。

### 1.3.1 教師あり学習

教師あり学習では、正解の付いた学習データを用います。このデータを訓練例とよぶこともあります。学習データは、入力データに対応するベクトル  $\mathbf{x}_i$  と、正解情報  $y_i$  のペアからなります。

$$\{(\mathbf{x}_i, y_i)\}, \quad i = 1, \dots, N$$

ここで、 $N$  は学習データの総数、添字  $i$  は学習データ中の  $i$  番目の事例であることを示します。

当面、入力ベクトル  $\mathbf{x}_i$  は次元数  $d$  の固定長ベクトルであると考えておきます<sup>1</sup>。

$$\mathbf{x}_i = (x_{i1}, \dots, x_{id})^T$$

図 1.3 の上部に示したような、(134.1, 34.6, 12.9) や、(女, 68, 165, 44, no) などが、入力ベクトル  $\mathbf{x}_i$  の例です。

入力ベクトルの各要素  $x_{i1}, \dots, x_{id}$  を、**特徴** (feature) あるいは**属性** (attribute) とよびます<sup>2</sup>。特徴は、数値あるいはラベルのいずれかの値をとります。数値は長さや温度などの連続値をとる場合もありますし、商品の購入回数や単語の出現回数などの離散値をとることもあります。ラベルは文字列データで、たとえば性別を表す「男・女」や天候を表す「晴・曇・雨」などを値とします。

教師あり学習は、この学習データから、入力  $\mathbf{x}$  を正解  $y$  に写像する関数  $c$  を学習することを目的とします。

1 ほかの教科書などの記法に従って、入力は列ベクトルとします。ただし、本文中や数式中で要素を示すときは、スペースを節約するため、転置を表す記号  $T$  を右肩に付けて行ベクトルとして表記します。また、図中などでとくに差し支えない場合は、転置の記号を省略します。

2 これ以降での学習手法の説明においては、(深層学習の説明を除いては) 問題設定に適した特徴がすでに選ばれているものとします。観測対象から、問題設定に適した特徴を選んでデータ化する作業は、特徴抽出とよばれます。

$$c: \mathbf{x} \rightarrow y$$

この関数  $c(\mathbf{x})$  は理想的な写像を表すので、とても複雑な形をしているかもしれませんが、それを限られた学習データから求めるのは難しいので、通常は関数の形を扱いやすいものに仮定して、その関数のパラメータを推定するという問題に置き換えます。この推定する関数を  $\hat{c}(\mathbf{x})$  と記述します。関数  $\hat{c}(\mathbf{x})$  の実際の形は、入力ベクトル  $\mathbf{x}$  と正解  $y$  の種類によって異なります。

また、正解情報（あるいは関数  $\hat{c}(\mathbf{x})$  の出力） $y$  も、数値（スカラー・ベクトルいずれの場合もあります）あるいはラベルのいずれかになります。正解  $y$  がラベルの場合を**識別** (classification) 問題、数値の場合を**回帰** (regression) 問題とよびます。識別問題の正解をクラス (class)、回帰問題の正解をターゲット (target) とよぶこととします。

具体的な教師あり学習問題の説明に入る前に、性能測定基準について少し説明します。学習結果である関数  $\hat{c}(\mathbf{x})$  は、学習データに含まれていない未知のデータ  $\mathbf{x}$  に対して、なるべく正しい答えを出力するように一般化されなければなりません。学習データに対する正解率ではなく、未知のデータに対する正解率が重要なのです。学習データに対しては、その正解をすべて表形式で記録しておけば、間違いなく正解を出力することができます。しかし、未知データに対して正解を出力するには、「学習データの背後にある法則のようなもの」を獲得する必要があるのです。機械学習は、人間が解き方のわからない問題に対して適用するものであることを、前節で説明しました。「学習データの背後にある法則のようなもの」をいかにして獲得するかということが、教師あり学習のテーマになります。

## 識別

識別は、入力をあらかじめ定められたクラスに分類する問題です。典型的な識別問題には、音声や文字の認識、レビュー文章のPN判定 (positive (ほめている) か negative (けなしている) か)、疾病の有無の判定などがあります。

ここで、識別問題として最も単純な、2値分類問題を考えてみましょう。2値分類問題とは、たとえば、ある病気かそうでないか、迷惑メールかそうでないかなど、入力を2クラスに分類する問題です。さらに、入力を数値のみを要素とするベクトルと仮定します。入力ベクトルが2次元の場合、学習データは図 1.6 (a) に示すように、平面上の点集合になります。クラスの値に対応させて、それぞれ丸とバツで表しました。

最も単純に考えると、識別問題はこの平面上で二つのクラスを分ける境界線を決めるという問題になります。未知のデータが入力されたとき、この境界線のどちらにあるかを調べるだけで、属するクラスを解答できるので、このことによって、識別能力



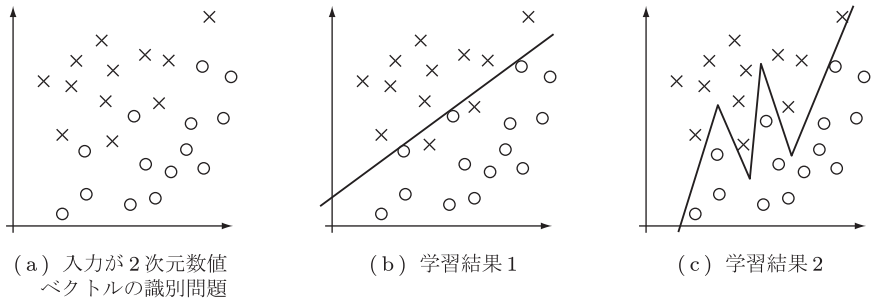


図 1.6 識別問題（入力が数値ベクトルの場合）

を身につけたとみなすことができます。

境界線として1本の直線を考えてみましょう。図 1.6 (b) に示すように、このデータでは切片や傾きをどのように調整しても、1本の直線で二つのクラスをきれいに分離することはできません。一方、図 1.6 (c) のように複雑に直線を組み合わせると、すべての学習データに対して同じクラスに属するデータが境界線の片側に位置するようになります。きれいに分離できたことになります。しかしここで、「識別とは、図 1.6 (c) のような、すべての学習データをきれいに分離する、複雑な境界線を探すことだ」という早とちりをしてはいけません。識別の目的は、学習データに対して 100% の識別率を達成することではなく、未知の入力をなるべく高い識別率で分類するような識別面を探すことでした。もう一度図 1.6 (a) に戻って、二つのクラスの塊をぼんやりとイメージしたとき、その塊を区切る線として図 1.6 (b), (c) のいずれがよいと思えるでしょうか。おそらく、大半の人が (b) を支持するでしょう。これが、我々人間が身につけている汎化能力で、そのようなことを学習結果に反映させるように、学習アルゴリズムを考える必要があります。

しかし、一般的な識別問題は、このような単純なものばかりではありません。識別結果が一つのクラスになるとは限らない場合があります。たとえば、受信した電子メールに対して、重要・緊急・予定・締切・…などのタグを付与する自動タグ付けを識別問題に当てはめると、一つの入力に対して、複数の出力の可能性がある問題設定になります。また、どのクラスにも当てはまらない入力が入ってくる可能性もあります。たとえば、スキャンした文書に対して文字認識をおこなっているときに、文字コードにない記号が含まれている場合があるかもしれません。

本書で扱う識別は、これらの複数出力の可能性や、識別不可能な入力の問題を除外して、すべての入力に対してあらかじめ決められたクラスのうちの一つを出力とする、というように単純化します。識別の代表的な手法には、決定木、ナイーブベイズ識別、ニューラルネットワーク、サポートベクトルマシンなどがあります。これらを3章か

ら7章で説明します。また、9章では複数の識別器を組み合わせる手法を、12章では系列データの識別手法を扱います。

## 回帰

回帰は入力から予測される妥当な出力値を求める問題です。典型的な回帰問題には、消費電力の予測、中古車の価格算出、生産量計画<sup>1</sup>などがあります。

回帰の単純な例として、入力を気温、出力をビールの売上高とした架空のデータ(図1.7(a))を考えてみます。

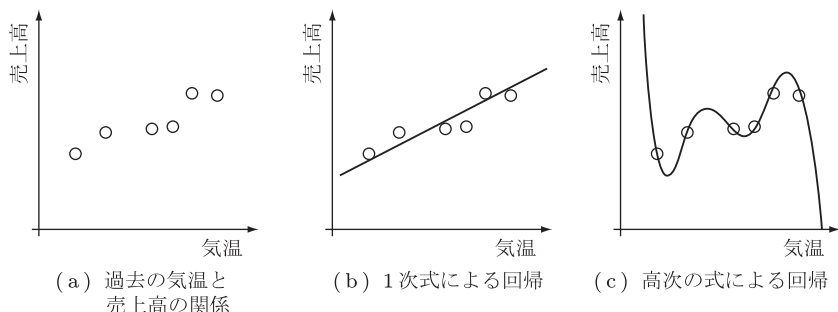


図 1.7 回帰問題の例

未知データに対して妥当な出力値を求めるために、入力データがある関数に基づいてターゲットを出力していると考え、その関数を求める問題が回帰問題です。ただし、関数の形として1次関数、2次関数、3次関数だけでなく、三角関数や指数関数などとの組み合わせまで考えてもよいとなると手がつけられなくなりますので、通常は関数の形を先に決めて、その係数を学習データから推定するという問題とみなします。

1次関数を仮定して、学習データとの誤差が最も少なくなるように係数を求めると、図1.7(b)に示すような直線が得られます。ほとんどの点がこの直線を外れているので、あまりよい近似とはいえないようにみえるかもしれません。一方、複雑な高次の関数<sup>2</sup>を前提とすれば、図1.7(c)に示すように、すべての学習データを通る関数を求めることができます。このどちらを採用するべきかについて、回帰問題でも識別問題と同様の立場をとります。すなわち、未知データに対する出力として、どちらが妥当

<sup>1</sup> ここでもビールの例がよく挙げられます。昨年同月の売り上げ、長期予報による平均気温、他社の新製品動向などを入力として、何ヶ月か先のビールの生産量を決めるという話です。

<sup>2</sup> 学習データが六つなので、5次式を仮定すると、全データを通る関数を求めることができます。一般に、 $M$ 次式は係数が $M+1$ 個なので、 $M+1$ 個の学習データを順にこの $M$ 次式に代入して得られる制約式を連立させて解くことで、すべての制約を満たす係数を求めることができます。

かということを考えるわけです。気温の少しの変化に対して売り上げが大きく変わるところがある図 1.7(c) の関数は、やはり不自然な回帰にみえます。この例のように、入力と出力を 2 次元で眺めることができれば、その妥当性ある程度直観的に議論できますが、通常の場合、入力は多次元ですので、直観に頼らずに学習結果の妥当性を吟味する方法を考えなければなりません。

回帰の代表的な手法には、線形回帰、回帰木、モデル木などがあります。これらを 8 章で説明します。

### 1.3.2 教師なし学習

教師なし学習では、学習に用いられるデータに正解情報が付いていません。

$$\{\mathbf{x}_i\}, \quad i = 1, \dots, N$$

入力ベクトル  $\mathbf{x}_i$  の次元数に関しては、教師あり学習の場合と同様に、 $d$  次元の固定長ベクトルで、各要素は数値あるいはラベルのいずれかの値をとると考えておきます。

$$\mathbf{x}_i = (x_{i1}, \dots, x_{id})^T$$

教師なし学習は、入力データに潜む規則性を学習することを目的とします。ここで着目すべき規則性としては、2 通り考えられます。一つめは、入力データ全体を支配する規則性で、これを学習によって推定する問題が**モデル推定** (model estimation) です。もう一つは、入力データの部分集合内あるいは部分集合間に成り立つ規則性で、通常は多くの要素の中に埋もれてみえにくくなっています。これを発見する問題が**パターンマイニング** (pattern mining) です。

#### モデル推定

モデル推定は、入力データの中から何らかの共通点をもつデータをまとめることで、入力データを生じさせたクラスが存在や、そのパラメータを推定するものです。図 1.8 に、モデル推定の考え方を示します。

観測されたデータは、もともと何らかのクラスに属していたものが、揺らぎを伴って生成されたものと考えます。その逆のプロセスをたどることができれば、データを生成したもととなったと推定されるクラスを見つけることができます。発見されたクラスの性質は、そこから生成されたと推定されるデータを分析することでわかります。もしかしたら、その発見されたクラスは、誰も考えつかなかった性質をもつものかもしれません。

このように、入力データ集合から適切なまとまりを作ることでクラスを推定する手

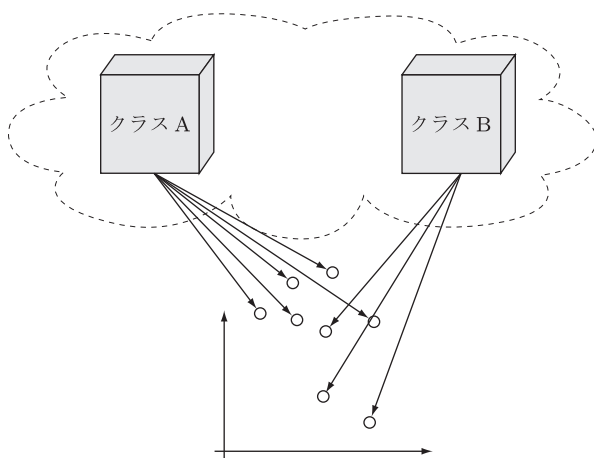


図 1.8 モデル推定の考え方

法を、**クラスタリング** (clustering) とよびます。顧客をクラスタリングした結果から特徴的な属性を見つけ出し、それぞれに適したマーケティングをおこなうような応用や、製品に対する口コミ文書をクラスタリングして、典型的な不満や要望を抽出する応用などが考えられています。

一方、もともとのクラスは何らかのデータを生成する関数をもっていると仮定して、その関数のパラメータを入力データから推定する手法を**密度推定** (density estimation) とよびます。密度推定はクラスタリングを発展させたものとみることができますが、その応用はクラスタリングだけでなく、不完全データを対象としたモデル推定<sup>1</sup>の問題や異常なデータの検出など、いろいろな場面で用いられています。

クラスタリングの代表的な手法には、階層的クラスタリングや k-means 法があり、密度推定の手法としては EM アルゴリズムがあります。これらを 10 章で説明します。

## パターンマイニング

**パターンマイニング**は、データ中に何度も出現するパターンや、そのパターンに基づいた規則を発見する手法です。前述した「紙おむつとビール」のようなバスケット分析が、代表的な応用例です。図 1.9 にパターンマイニングの考え方を示します。

パターンマイニングの敵は、膨大な計算量です。図 1.9 に示した例では、発見された規則の条件部も結論部も一つの要素しか含んでいないのですが、通常はどちらも要

<sup>1</sup> 教師あり学習の学習データ  $\{(\mathbf{x}_i, y_i)\}$  を「完全」なデータとすると、教師なし学習のデータ  $\{\mathbf{x}_i\}$  は  $y_i$  が欠損した「不完全」データとなり、密度推定はこの不完全データを対象とした識別問題とみなすことができます。

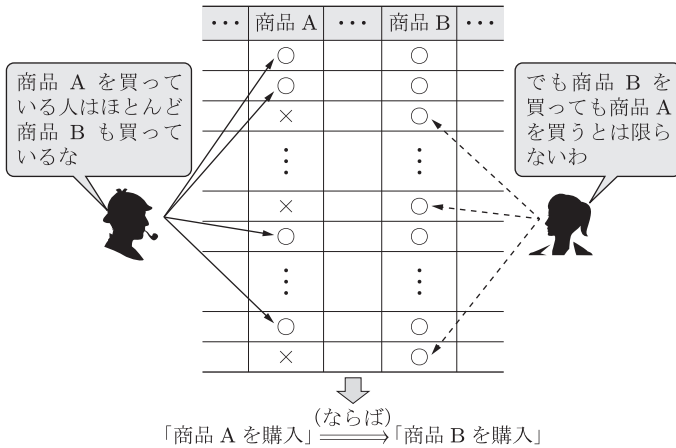


図 1.9 パターンマイニングの考え方

素の集合となります。それらのあらゆる組み合わせに対して、マイニングの対象となる大きなデータ集合から出現数を数えあげなければならないので、単純な方法では気の遠くなるような計算量になってしまいます。そこで、効率よく頻出パターンを見つけ出す手法が必要になります。

パターンマイニングの代表的な手法としては、Apriori アルゴリズムやその高速化版である FP-Growth があります。これらを 11 章で説明します。

### 1.3.3 中間的手法

ここでは、これまでに説明した教師あり学習／教師なし学習に当てはまらない手法について説明します。その中でも、学習データが正解付き／正解なしの混在とみなせるものが半教師あり学習と強化学習です。一方、学習手法そのものが教師あり／教師なしの組み合わせとなっているものが深層学習です。

#### 半教師あり学習

これまでに述べてきた分類では、学習データすべてに対して正解が与えられているか、あるいはすべてに与えられていないかのいずれかでした。その中間的な設定として、学習データの一部にだけ正解が与えられている場合が考えられます。

学習データに正解を与えるのは人間なので、正解付きのデータを作成するにはコスト（費用・時間）がかかります。一方、正解なしのデータは、容易にかつ大量に入手可能という状況があります。たとえば、ある製品の評価をしているブログエントリーを識別するような問題では、正解付きデータを 1000 件作成するのはなかなか大変ですが、

ブログエントリーそのものは、Web クローラプログラムを使えば、自動的に何万件でも集まります。このような状況で、正解付きデータから識別器を作り、それを正解なしデータに適用して、得られた識別結果の中から確信度の高いデータを新たに正解付きデータとみなす、ということを繰り返す学習手法を、**半教師あり学習** (semi-supervised learning) といいます。半教師あり学習は、主として識別問題に対して用いられます。半教師あり学習の考え方を図 1.10 に示します。

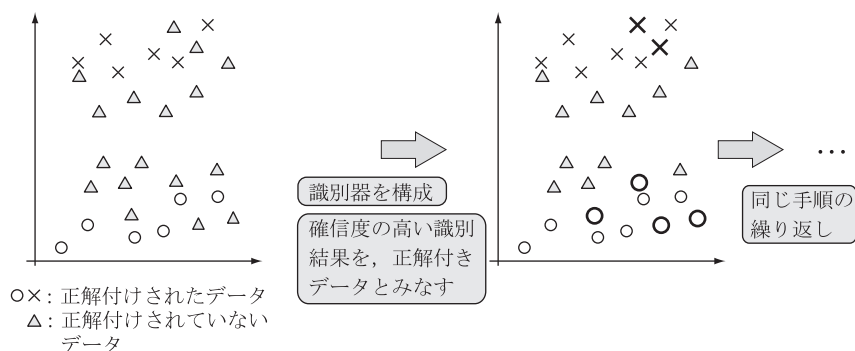


図 1.10 半教師あり学習の考え方

図 1.10 左のように、全データの中で正解の付加されたデータを丸・バツで表し、正解のないデータを三角形で表します。最初は、丸・バツが付いたデータだけで識別器を作り、たとえば、その中間あたりに境界直線を引いたものとしします。これに従って三角形のデータを分類しますが、境界線近辺のデータはあまり信用せず、境界線から大きく離れたものを確信度が高いとみなして正解を付与します。今度は、これらの新しく正解を付与されたデータも加えて、再度識別境界を計算します。これを、新しい正解付きデータが増えなくなるまで繰り返します。

この学習法は、識別すべきクラスがうまくまとまっているようなデータや、識別結果によって有効な特徴が増えてゆくような、やや特殊なデータ<sup>1</sup>に対して適用するときにうまくゆきます。この手法を 13 章で説明します。

## 強化学習

問題の性質によっては、間接的に正解が与えられる場合があります。図 1.11 のような迷路を抜けるロボットを学習させる場合を考えてみましょう。この場合、入力ロボットのもつセンサーからの情報で、これによって、ロボットはどの部屋にいるかが

<sup>1</sup> 文書分類の問題で、対象分野のキーワードが芋づる式にたどられるような状況です。

わかるものとします。出力はロボットの移動コマンド（この場合、上下左右いずれかへ進む）であるとして、もし、すべての部屋（すなわち、すべてのロボットの状態）において正解（各状態での最適な移動コマンド）が与えられれば、ロボットはスタートから回り道することなくゴールにたどり着けます。

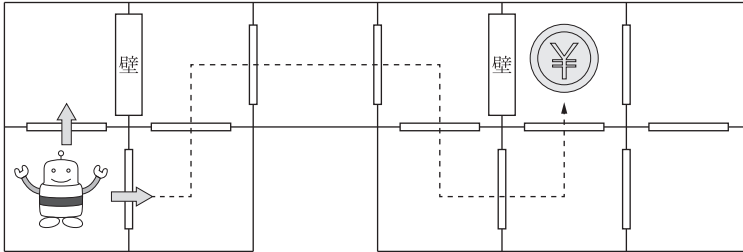


図 1.11 強化学習の問題例

このようなロボットに対して、ゴールに着いたときだけ報酬を与えるという方法で、ロボットに試行錯誤を繰り返させながら、最終的に各状態における最適な出力（この場合は移動コマンド）を獲得させる学習手法を、**強化学習** (reinforcement learning) とよびます。報酬を教師信号とみなすと、これは教師時々あり学習ということができます。すなわち、教師あり／なしの中間的な設定とみなすことができ、個々の決定に正解は与えられず、決定の連続に対して、後で形を変えた教師信号が与えられる、という難しい設定になります。強化学習に関しては 14 章で説明します。

## 深層学習

近年、機械学習の新しい手法として、**深層学習**（ディープラーニング (deep learning)）ともよばれます）が注目されています。そのブームはこれまで機械学習分野では前例のないもので、その研究成果が一般雑誌にも取り上げられたほどです。

深層学習の正確な定義は難しいのですが、最も狭い意味では、多層のニューラルネットワークを用いた学習（図 1.12）とみなせます。深層学習の入力は、特徴抽出をおこなわない生データに近いものが使われます。音声信号をデジタル化して一定時間幅の周波数分析をおこなったものや、画像のピクセル値などを入力とします。学習手順は、入力層に近いほうの階層から順に教師なし学習をおこなって、特徴抽出と同じはたらしをおこなうネットワークを構成します。これらの学習を積み上げてゆく段階を、**事前学習** (pre-training) とよびます。この事前学習によって得られたパラメータを初期値として、全体のネットワークを教師ありで学習することで、これまでの機械学習

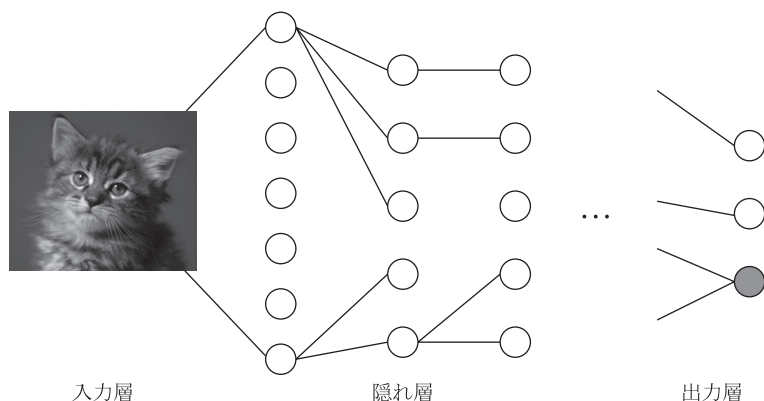


図 1.12 深層学習に用いる多層ニューラルネットワークの例

手法を大きく上回る識別結果を実現しています。また、事前学習の結果、教師なし学習で高次の概念（たとえば、猫の顔や人間の体など）が獲得できたという報告もあります。深層学習に関しては 15 章で説明します。

## 1.4

## まとめ

この章では、近年ビッグデータが注目されてきている背景を説明し、ビッグデータを扱う際の中心的な技術である機械学習についての概要を示しました。また、教師あり／なしという基準から機械学習における様々な問題を分類し、それぞれの目的を説明しました。

機械学習の分類に関しては、ここで説明した以外にも、様々な観点のものがあります。文献[2]では、モデルを構成する原理から、幾何的・確率的・論理的モデルという観点と、それとは直交して、モデルの振るまいから、グループ化・グレード化モデルという観点を挙げています。また、文献[3]、[4]では、採用する確率モデルの推定法の違いから、生成モデル[3]と識別モデル[4]という観点が立てられています。体系的に機械学習を学ぶには、このようなモデルに基づく分類からスタートするほうが好ましいと思われますが、本書では、まず全体像をできるだけ平易に俯瞰することを目指しましたので、問題設定に基づく分類をおこないました。

以後、本書ではここで示した分類に基づいて、順に機械学習技術を紹介してゆきます。識別・回帰など個別の問題に対して、それらを解決する機械学習アルゴリズムを示してゆきますが、それぞれはそこで取り上げられた問題専用のアルゴリズムではあ



りません。ラベル入力の識別問題を解決するアルゴリズムが、連続値入力の識別問題にも適用できたり、識別問題のアルゴリズムを回帰問題に適用することなども可能です。それらの方法を逐一解説しては非常に読みにくくなってしまうので、いくつかの重要な手法を除いては、その章で示した問題を解決する手順として、アルゴリズムを考えてゆきます。

それでは、機械学習を楽しむ旅に出かけましょう。

## 演習問題

- 1.1 識別・回帰・モデル推定・パターンマイニングのそれぞれについて、本書で挙げた以外の応用事例を調査せよ。