

機械学習 (1)

概要

1. 機械学習の基礎

- 概論
 - 教師あり学習
 - 教師なし学習
 - 強化学習

補助資料：<http://small-island.work/trial/>

ユーザ名：trial

パスワード：trial

2. 教師あり学習

- 最近傍法・線形識別・決定木
- ランダムフォレスト
- サポートベクターマシン

3. 確率モデリングと教師なし学習

- クラスタリング
- ナイーブベイズモデル
- 混合ガウスモデル
- クロスバリデーション・モデル選択

4. データ構造と機械学習アルゴリズム実践

- テーブルデータ
- 行列データ
- スパースモデリング
- 時系列データ
- グラフデータ

この講義の目的とカバー範囲

目的

- 原理をわかりつつ実際に使うことを目的とする
 - 途中の厳密さや計算の詳細はある程度省略
- 言われた通りアルゴリズムやパラメータをいじるのではなく、いざとなれば自分で適切なアルゴリズムの選択・パラメータのチューニングができるレベルを目指す
- つまり、単に表面的な理解でライブラリを呼び出すということより踏み込んだ理解を目指す

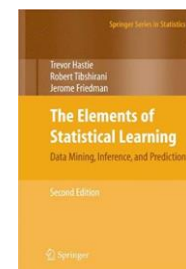
カバー範囲

- 実際どのように書くかなどはその都度触れるが、詳細は本などでのトレーニングが必要
- 理論的に高度すぎる話は扱わないがその中間を学ぶことで

より個別の詳細な教科書



幅広い教科書



実装の教科書



機械学習を学ぶ意義

- 多くの応用分野をもつ（医療・創薬・材料・Web・ロボット自動化）
→ キャリアアップ
- データ分析の人が何を考えているかがわかる
→ 外注丸投げの回避
- 趣味
→ 🍷 ('ω' 🍷)三 🍷 ('ω') 🍷 三(🍷 'ω') 🍷



関連分野

計算と人工知能

1950年：チューリングテスト
1956年：ダートマス会議

論理・探索

第1次AIブーム: 推論と探索

1956年－1974年

1972年：Prolog

第2次AIブーム:
知識

(エキスパートシステム)

1980年－1987年

機械学習

1967年：k-means法
(教師なし学習：
クラスタリング)

1989年：Q学習
(強化学習)

1992年：非線形SVM
(教師あり学習)

1990年～：統計的機械学習

- ・確率モデル
- ・汎化誤差理論
- ・ベイズモデル

ニューラルネットワーク

1958年：パーセプトロン

1969年：パーセプトロンの限界

1986年：誤差逆伝搬法

2006年：オートエンコーダー

第3次AIブーム:
深層学習
2006年－

関連分野

人工知能

論理・探索

第1次AIブーム: 推論と探索

1956年－1974年

1972年: Prolog

第2次AIブーム:
知識

(エキスパートシステム)

1980年－1987年

計算と人工知能

1950年: チューリングテスト

1956年: デルフォイン・トマス会議

機械学習

機械学習

1967年: k-means法
(教師なし学習:
クラスタリング)

1989年: Q学習
(強化学習)

1992年: 非線形SVM
(教師あり学習)

1990年～: 統計的機械学習

- ・ 確率モデル
- ・ 汎化誤差理論
- ・ ベイズモデル

ニューラルネットワーク

1958年: パーセプトロン

1969年: パーセプトロンの限界

1986年: 誤差逆伝搬法

深層学習

2006年: オートエンコーダー

第3次AIブーム:
深層学習
2006年－

関連分野

人工知能

論理・探索

第1次AIブーム: 推論と探索

1956年 - 1974年

1972年: Prolog

第2次AIブーム:
知識

(エキスパートシステム)

1980年 - 1987年

計算と人工知能

1950年: チューリングテスト

1956年: デルフォイダス会議

機械学習

機械学習

1967年: k-means法

(教師なし学習:
クラスタリング)

1989年: Q学習

(強化学習)

1992年: 非線形SVM

(教師あり学習)

1990年~: 統計的機械学習

- ・ 確率モデル
- ・ 汎化誤差理論
- ・ ベイズモデル

ニューラルネットワーク

1958年: パーセプトロン

1969年: パーセプトロンの限界

1986年: 誤差逆伝搬法

深層学習

2006年: オートエンコーダー

第3次AIブーム:

深層学習

2006年 -

本日の目標

- 教師あり学習・教師なし学習・強化学習の考え方を知る
- これらの実際の応用事例を知る
- モデルの考え方を学ぶ
- 実際の細かい手法等については次回以降に行う

本日、登場する単語

(数理) モデル

- ある現象を数式で表現したもの
- 本当に現実で意味のあるモデルと数理上都合の良いモデルのバランスをとっているものが多い

パラメータ

- モデルが持つ変数で、データによらない部分で何らかの値を持つ変数

学習

- 与えられたデータからその知識を後に再利用できるようにする
- 特に、モデルのパラメータを決定する学習をパラメータ学習とも呼ぶ

教師あり学習

- データとそれに対応したラベルの入出力関係を学習する

教師なし学習

- データ自体を学習し、そのデータに関する知見を得る

強化学習

- 行動によって得られる報酬を最大化するように学習を行う
学習：より良い行動を選べるよう改善すること

本日、登場する単語

(数理) モデル

- ある現象を数式で表現したもの
- 本当に現実で意味のあるモデルと数理上都合の良いモデルのバランスをとっているものが多い

パラメータ

- モデルが持つ変数で、データによらない部分で何らかの値を持つ変数

学習

- 与えられたデータからその知識を後に再利用できるようにする
- 特に、モデルのパラメータを決定する学習をパラメータ学習とも呼ぶ

教師あり学習

- データとそれに対応したラベルの入出力関係を学習する

教師なし学習

- データ自体を学習し、そのデータに関する知見を得る

強化学習

- 行動によって得られる報酬を最大化するように学習を行う
学習：より良い行動を選べるよう改善すること

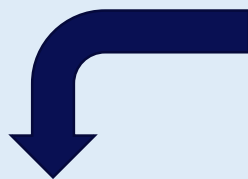
教師あり学習

学習



画像認識の例

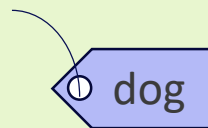
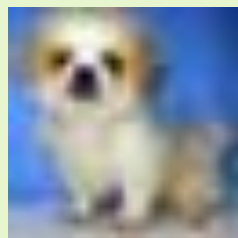
学習用
データセット



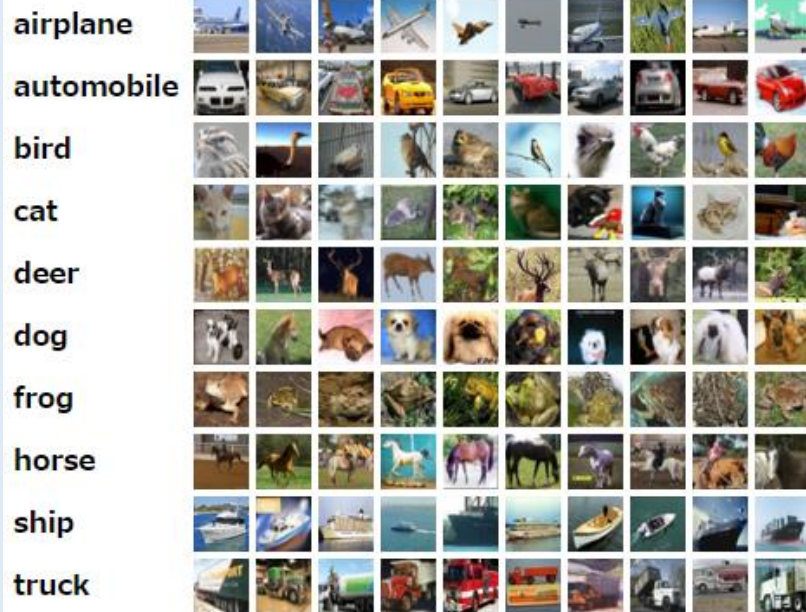
テスト・予測



テスト用
データ



Cifer10: 60000 32x32 color images in 10 classes



とりあえず動かしてみる



Home Installation Documentation Examples

Google Custom Search



Fork me on GitHub

Previous
sklearn.linear_...
Next
sklearn.linear_...
Up
API Reference

scikit-learn v0.20.3
Other versions

Please [cite us](#) if you use the software.

sklearn.linear_model.LogisticRegression
Examples using
sklearn.linear_model.LogisticRegression

sklearn.linear_model.LogisticRegression

```
class sklearn.linear_model.LogisticRegression (penalty='l2', dual=False, tol=0.0001, C=1.0, fit_intercept=True, intercept_scaling=1, class_weight=None, random_state=None, solver='warn', max_iter=100, multi_class='warn', verbose=0, warm_start=False, n_jobs=None) [source]
```

Logistic Regression (aka logit, MaxEnt) classifier.

In the multiclass case, the training algorithm uses the one-vs-rest (OvR) scheme if the 'multi_class' option is set to 'ovr', and uses the cross-entropy loss if the 'multi_class' option is set to 'multinomial'. (Currently the 'multinomial' option is supported only by the 'lbfgs', 'sag' and 'newton-cg' solvers.)

This class implements regularized logistic regression using the 'liblinear' library, 'newton-cg', 'sag' and 'lbfgs' solvers. It can handle both dense and sparse input. Use C-ordered arrays or CSR matrices containing 64-bit floats for optimal performance; any other input format will be converted (and copied).

The 'newton-cg', 'sag', and 'lbfgs' solvers support only L2 regularization with primal formulation. The 'liblinear' solver supports both L1 and L2 regularization, with a dual formulation only for the L2 penalty.

参照プログラム
lecture_logreg.ipynb

教師あり学習の基礎：回帰

回帰: ある数値を何らかの数式で予測する問題

たかし君はリンゴとミカンが4個ずつ買いました。金額はあわせて800円でした。また、リンゴ6個とミカン2個買、1000円でした。リンゴを3個とミカン2個を買うといくらでしょう？

学習

パラメータ

リンゴ： a 円

みかん： b 円

$$\begin{cases} 4a + 4b = 800 \\ 6a + 2b = 1000 \end{cases}$$

$$a = 150, b = 50$$

モデル

リンゴを x 個、みかんを y 個買ったとき、
 $ax + by$

リンゴを3個、みかんを2個買うと、
 $3 \times 150 + 2 \times 50 = 550$



予測



教師あり学習の基礎：回帰

たかし君はリンゴとミカンを4個ずつ買いました。金額はあわせて800円だった気がすると言っています。花子さんはリンゴ6個とミカン2個買い、1000円ぐらいだそうです。ひろしくんはリンゴ3個とミカン2個買い、500円ぐらいだそうです。

学習

リンゴ： a 円

みかん： b 円

$$\begin{cases} 4a + 4b \sim 800 \\ 6a + 2b \sim 1000 \\ 3a + 2b \sim 500 \end{cases}$$

[http://www.wolframalpha.com/input/?i=\(4x%2B4y+%E2%88%92800\)%5E2%2B\(6x%2B2y+%E2%88%921000\)%5E2%2B\(3x%2B2y+%E2%88%92500\)%5E2](http://www.wolframalpha.com/input/?i=(4x%2B4y+%E2%88%92800)%5E2%2B(6x%2B2y+%E2%88%921000)%5E2%2B(3x%2B2y+%E2%88%92500)%5E2)

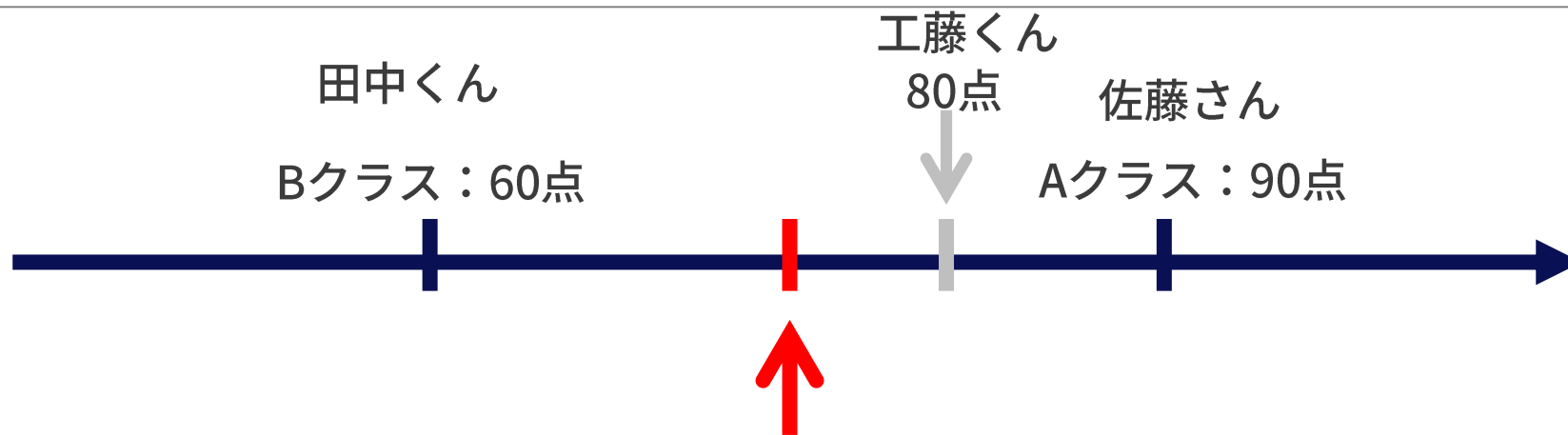
$$\min \begin{aligned} & (4a + 4b - 800)^2 \\ & + (6a + 2b - 1000)^2 \\ & + (3a + 2b - 500)^2 \end{aligned}$$

誤差関数
→ 2乗誤差

教師あり学習の基礎：分類

分類: データをクラスに分類する問題

テストを受けてもらいクラス分けを行いました。点数の高い人はAクラス、点数の低い人はBクラスになります。友達に点数を聞いたところ佐藤さんは90点でAクラス、田中くんは60点でBクラスでした。工藤くんは80点でしたどちらのクラスと思いますか？



ボーダーラインを求める問題と同じ

決定境界

教師あり学習の基礎：分類

文系100点満点と理系100点満点のテストを受けて、クラス分けを行いました。理系コースなので理系科目重視の重み付き平均点で決まります。点数の高い人はAクラス、点数の低い人はBクラスになります。

$$\begin{array}{l} \text{理系の点数：} x \text{ 点} \\ \text{文系の点数：} y \text{ 点} \end{array} \left\{ \begin{array}{l} ax + by \geq c : A \\ ax + by < c : B \end{array} \right.$$

決定境界 (c 点) を求める問題
と同じ

$$\left\{ \begin{array}{l} ax + by - c \geq 0 : A \\ ax + by - c < 0 : B \end{array} \right.$$



$$\left\{ \begin{array}{l} ax + by - c = 1 : A \\ ax + by - c = -1 : B \end{array} \right.$$

連立方程式問題と同じ
→回帰問題と同じ

教師あり学習の基礎：分類

文系100点満点と理系100点満点のテストを受けて、クラス分けを行いました。理系コースなので理系重視の重み付き平均点で決まります。点数の高い人はAクラス、点数の低い人はBクラスになります。



理系の点数： x 点

文系の点数： y 点

$$\begin{cases} ax + by - c = 1 & : A \\ ax + by - c = -1 & : B \end{cases}$$

友達がたくさんいたので以下のような式が出せました

$$\begin{cases} a \times 36 + b \times 53 - c \sim -1 \\ a \times 69 + b \times 80 - c \sim 1 \\ a \times 62 + b \times 84 - c \sim 1 \end{cases}$$

これらの式から理系・文系の重み a, b を計算する

$$\begin{aligned} & (a \times 36 + b \times 53 - c + 1)^2 \\ \min & + (a \times 69 + b \times 80 - c - 1)^2 \\ & + (a \times 62 + b \times 84 - c - 1)^2 \end{aligned}$$

[http://www.wolframalpha.com/input/?i=\(\(36a%2B53b%E2%88%92c%2B1\)%5E2%2B\(69a%2B80b%E2%88%92c%E2%88%921\)%5E2%2B\(62a%2B84b%E2%88%92c%E2%88%921\)%5E2+\)](http://www.wolframalpha.com/input/?i=((36a%2B53b%E2%88%92c%2B1)%5E2%2B(69a%2B80b%E2%88%92c%E2%88%921)%5E2%2B(62a%2B84b%E2%88%92c%E2%88%921)%5E2+))

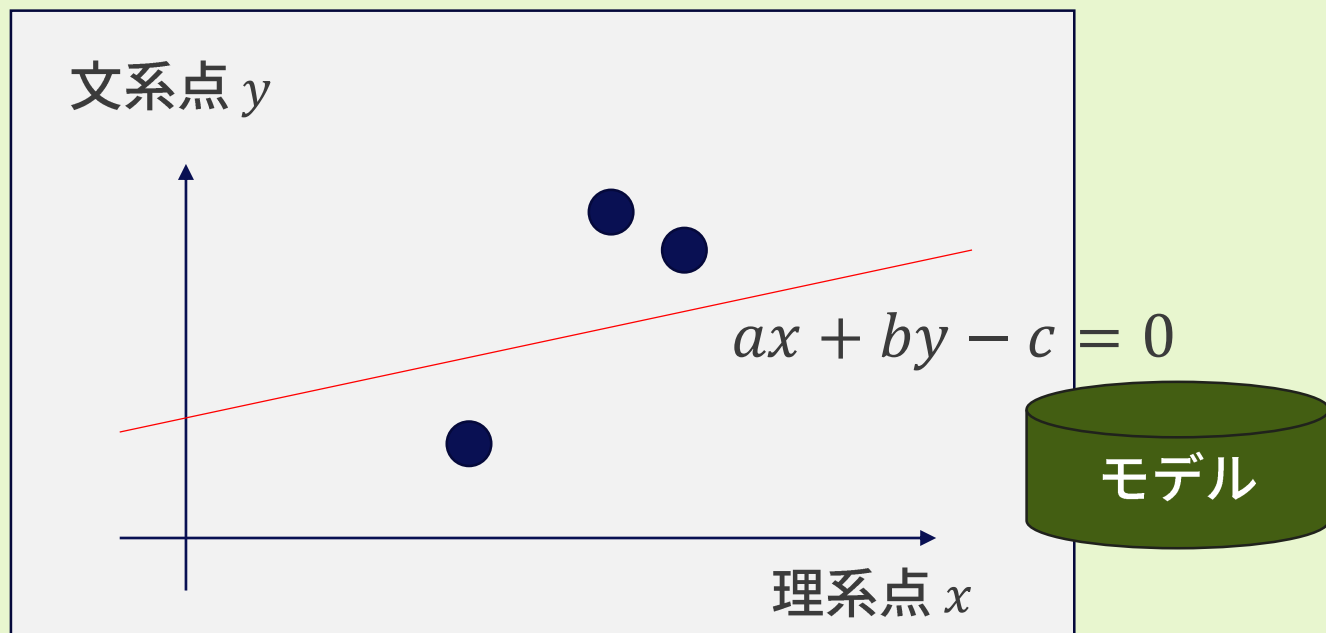
教師あり学習の基礎：分類

文系100点満点と理系100点満点のテストを受けて、クラス分けを行いました。理系コースなので理系重視の重み付き平均点で決まります。点数の高い人はAクラス、点数の低い人はBクラスになります。

予測



$$\begin{cases} a \times 36 + b \times 53 - c \sim -1 \\ a \times 69 + b \times 80 - c \sim 1 \\ a \times 62 + b \times 84 - c \sim 1 \end{cases}$$



教師あり学習の超基本部分

教師付き学習では“教師”が必要

- 買い物の例では合計いくらであったかの調査が必要である
- クラス分けの例では知り合いの点数を調査する必要がある

モデルをつかうと新たなデータに対して予測できる

- 買い物の例では単価×個数の和で合計金額を予測できる
- クラス分けの例ではボーダーライン以上・以下でクラスを予測できる

教師付き学習では教師を用いて変数の値（パラメータ）を推定していた

- 買い物の例ではリンゴとみかんの値段がパラメータ
- クラス分けの例ではボーダーラインがパラメータ

教師あり学習：線形でない場合

今までの例は
(線形の) ルールが
存在した

$ax + by$ 線形モデル

パラメータ
リンゴ： a 円
みかん： b 円

$$\begin{cases} 4a + 4b \sim 800 \\ 6a + 2b \sim 1000 \\ 3a + 2b \sim 500 \end{cases}$$

$$\begin{aligned} & (4a + 4b - 800)^2 \\ \min & + (6a + 2b - 1000)^2 \\ & + (3a + 2b - 500)^2 \end{aligned}$$

非線形な場合

- 現実のルールが線形でない
- 複雑な関数でとにかく当てたい

$f(a, b, x, y)$ 非線形モデル

$$\begin{cases} f(a, b, 4, 4) \sim 800 \\ f(a, b, 6, 2) \sim 1000 \\ f(a, b, 3, 2) \sim 500 \end{cases}$$

$$\begin{aligned} & (f(a, b, 4, 4) - 800)^2 \\ \min & + (f(a, b, 6, 2) - 1000)^2 \\ & + (f(a, b, 3, 2) - 500)^2 \end{aligned}$$

線形 vs 非線形

問題：次のうち金額が線形に変化するものはどれ（0以下は考えない、小数点以下もあるものとする）

1. 1000円買ったなら100円引き
2. 1割引
3. 1000円以上は定額1000円
4. X グラムあたり $1000 \log X$ 円
($X > 1$)

ヒント：線形の定義

$$f(X + Y) = f(X) + f(Y)$$

f ：購入したものの量→価格

注意事項：線形でないものはすべて非線形

4. に関してあらかじめ、 $\log X = X'$, $\log Y = Y'$ と購入した量を変換しておくと、以下のように線形性が成り立つ

$$f(X' + Y') = f(X') + f(Y')$$

例： $f(x) = 2x^3 + 3x^2 + 4x + 5$

は x のみに関する式だと思えば非線形だが、

ベクトル $\phi(x) = (1, x, x^2, x^3)^T$ とまとめると線形

$$g(\phi(x)) = \mathbf{w}^T \phi(x)$$

$$\phi(x) = \begin{bmatrix} 1 \\ x \\ x^2 \\ x^3 \end{bmatrix} \quad \mathbf{w} = \begin{bmatrix} 2 \\ 3 \\ 4 \\ 5 \end{bmatrix}$$

つまり、 $g(\phi(x) + \phi(y)) = g(\phi(x)) + g(\phi(y))$ は成り立つ

画像認識問題の例

$f(\text{鳥の画像}) \sim \text{bird} \rightarrow 1 \rightarrow (0,0,1)$

$f(\text{猫の画像}) \sim \text{cat} \rightarrow 2 \rightarrow (0,1,0)$

$f(\text{犬の画像}) \sim \text{dog} \rightarrow 3 \rightarrow (1,0,0)$

分類するクラスが複数ある場合は、ベクトルの内、クラスに対応した場所だけ1で残りがゼロのOne-hotベクトルに変換することが多い

特徴量抽出: 画像→数値 (ベクトル・データ構造) への変換

- 最近では画像そのまま = 各画素ごとに (R, G, B)
- Bag of features + 局所特徴量 (SURF, SIFT, KAZE, AKAZE)

深層学習: この関数を複雑にしたもの

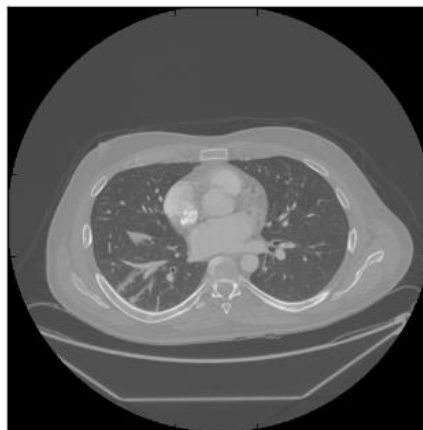
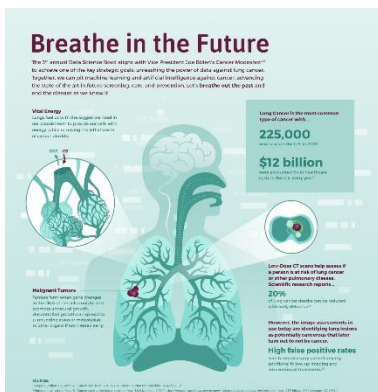
医療画像処理の例

Kaggle：機械学習のコンペサイト

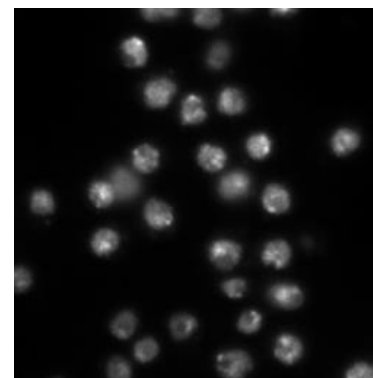
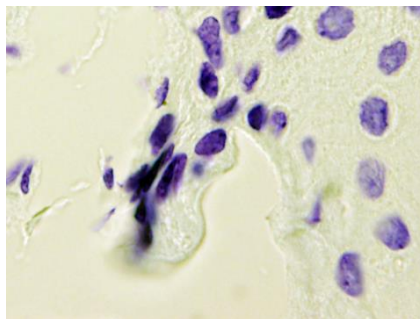
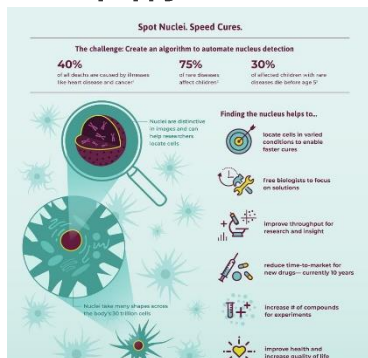
<https://www.kaggle.com/>

Data science Bowl

2017：CT画像からの肺がん判定問題

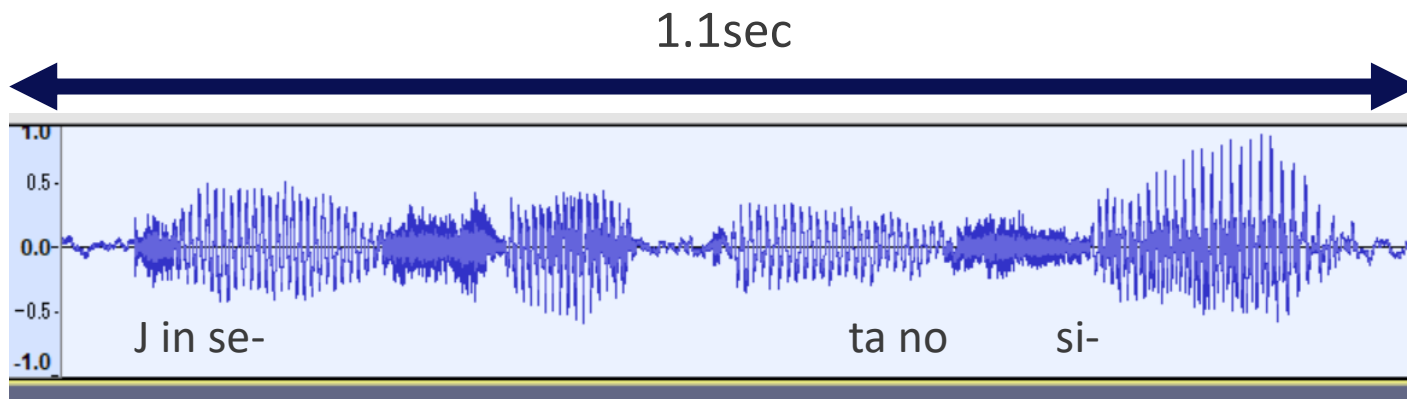


2018：画像ベースの細胞核検出問題



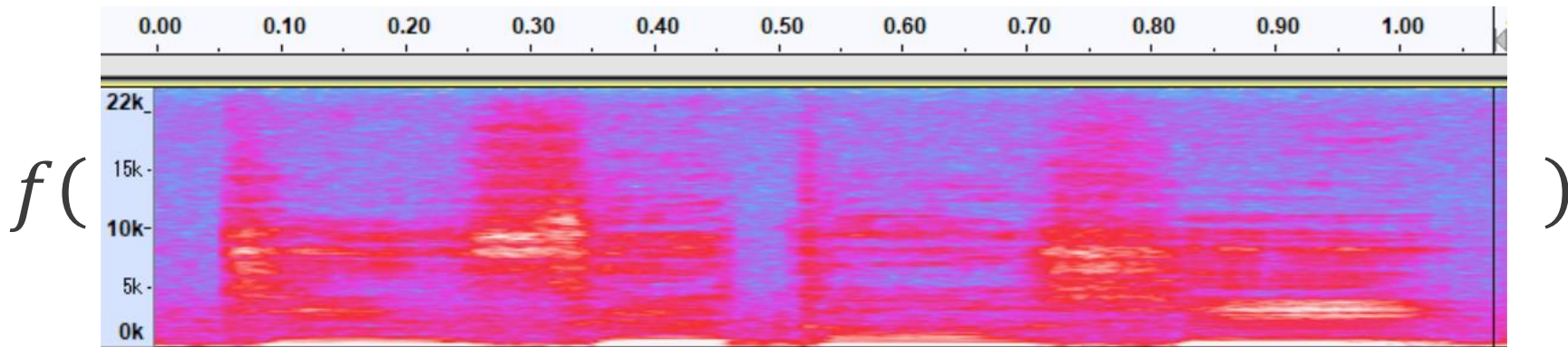
音声認識の例

音声



周波数解析

(STFT: 短時間フーリエ変換)



~じんせいいたのしい

画像とほぼ同様の方法が適用可能

まとめ：教師あり学習

教師付き学習では“教師”が必要

- 回帰と分類があり、両者の問題は多くの場合でほぼ同様の方法で解くことができる
- モデルのパラメータを決定することで、学習できる
 - モデルは線形の場合と非線形の場合がある
- 画像や音声の場合でも、ほぼ同じ手続きで学習できる

ブラックボックス的に使うことも可能だが、実際の用途では手法ごとに様々な特性があり、それらを理解して使い分ける必要がある

→第2回講義にて解説

本日、登場する単語

(数理) モデル

- ある現象を数式で表現したもの
- 本当に現実で意味のあるモデルと数理上都合の良いモデルのバランスをとっているものが多い

パラメータ

- モデルが持つ変数で、データによらない部分で何らかの値を持つ変数

学習

- 与えられたデータからその知識を後に再利用できるようにする
- 特に、モデルのパラメータを決定する学習をパラメータ学習とも呼ぶ

教師あり学習

- データとそれに対応したラベルの入出力関係を学習する

教師なし学習

- データ自体を学習し、そのデータに関する知見を得る

強化学習

- 行動によって得られる報酬を最大化するように学習を行う
学習：より良い行動を選べるよう改善すること

教師なし学習

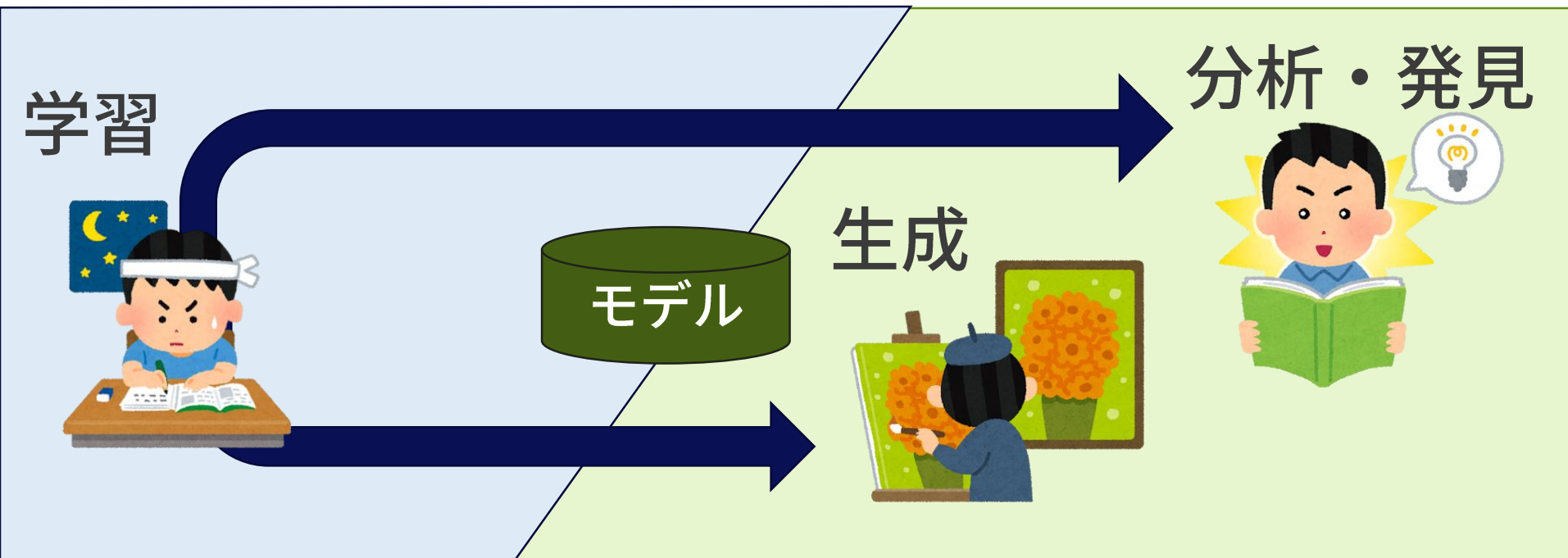
入力データを再現するようなモデルを学習する

[分析・発見]

扱いやすいデータに変換をし、分析に利用する

[生成]

似たデータを生成する





教師なし学習: 次元削減

列数の多いデータ：
同時にプロットできない

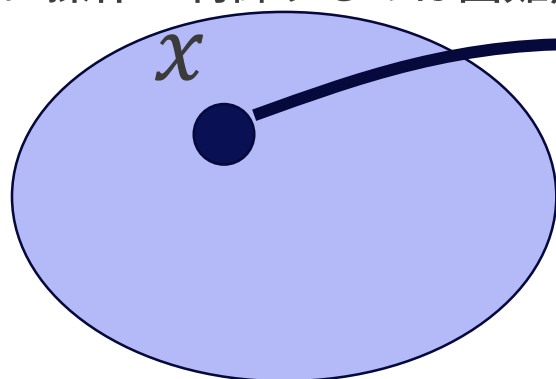
http://small-island.work/four_d_viewer/index.html

	体重	身長	白血球数	好中球数
Aさん	160	50	6000	3400
Bさん	170	55	7000	4200
Cさん	150	45	7500	4200

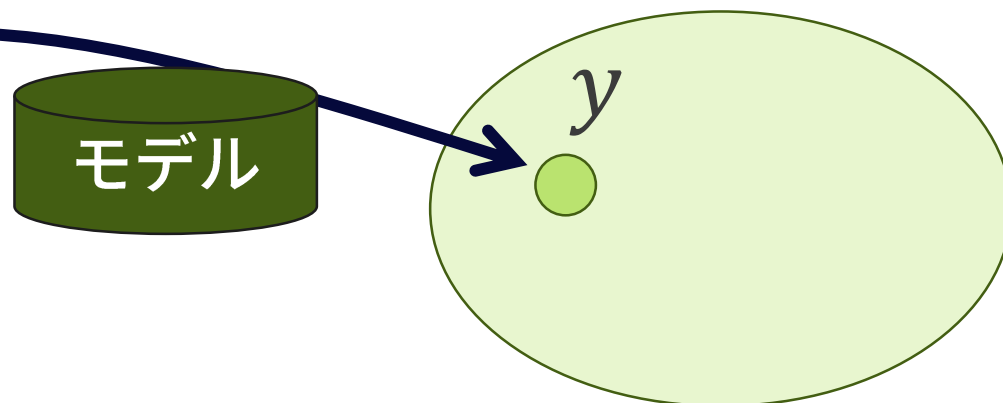
列数の少ないデータ：
同時にプロットできる

	BMI	好中球 (%)
Aさん	19.5	56
Bさん	19	60
Cさん	20	56

高次元の空間
(同時に操作・制御するのは困難)



低次元の空間 (例えば2次元)
(同時に操作・制御するのは容易)



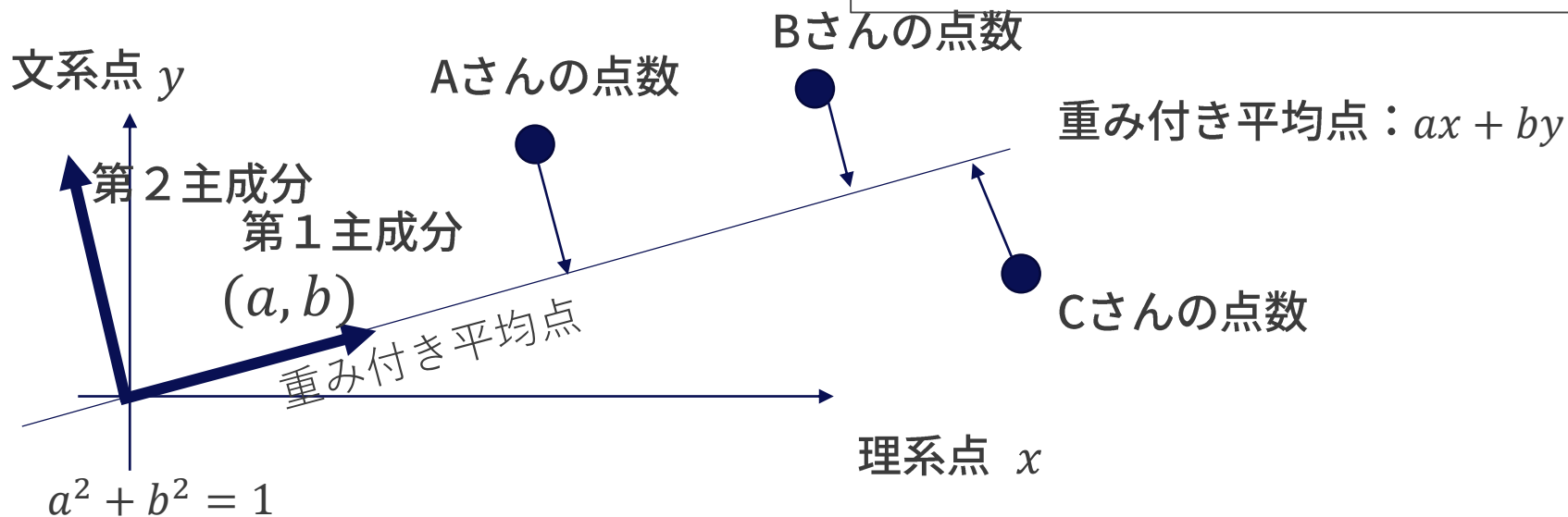
主成分分析 (1/2)

文系100点満点と理系100点満点のテストを受けて、クラス分けを行いました。理系コースなので理系科目重視の重み付き平均点で決まります。点数の高い人はAクラス、点数の低い人はBクラスになります。

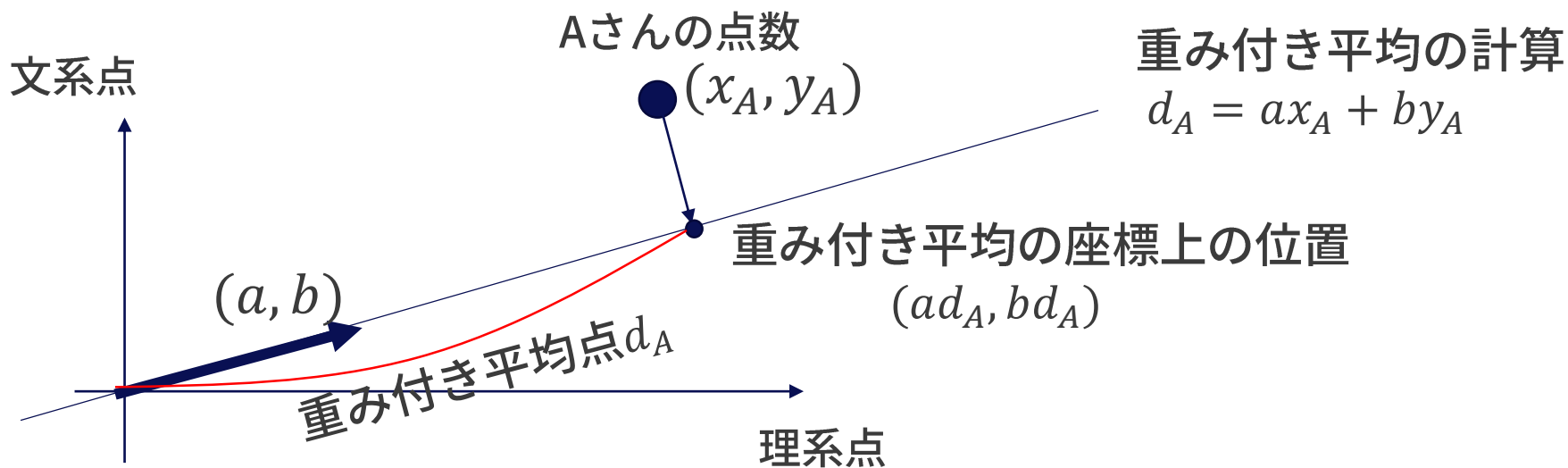
できる限り差をつけるための重み付き平均点は？

解き方

原点を平均点に持ってきて、
第一主成分の分散を最大にする
(第二主成分の分散を最小にする)

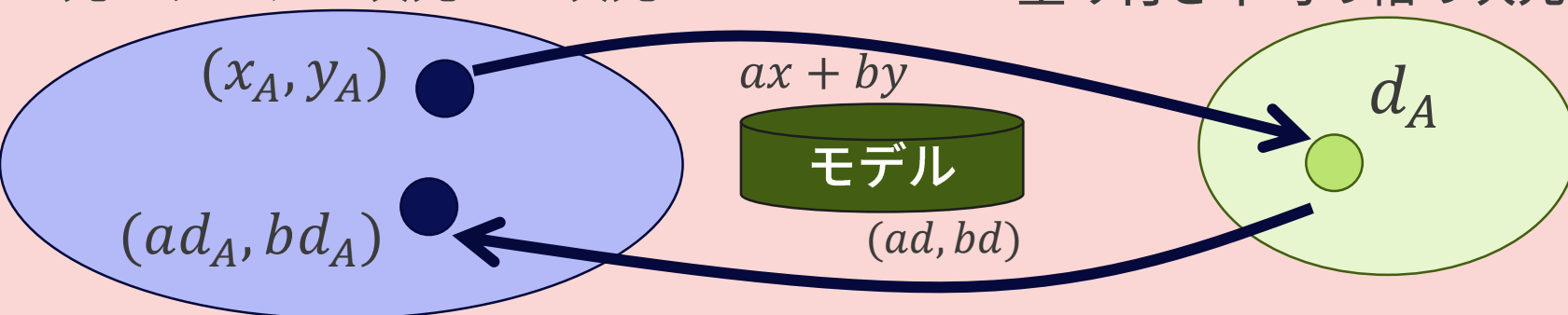


主成分分析 (2/2)



元のデータの次元：2次元

重み付き平均の軸の次元：1次元

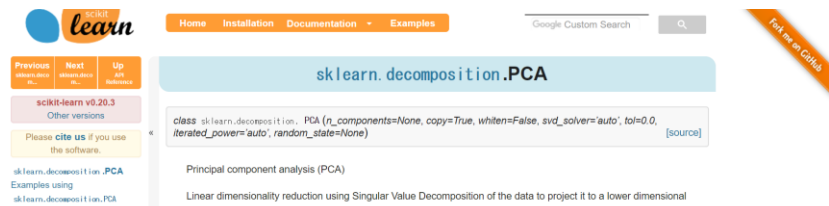


$$\min_{a, b} \|(x_A, y_A) - (ad_A, bd_A)\|^2$$

実際には特異値分解を使って計算する
→ 第4回講義

人数（データ数）が多い場合・次元数が大きい場合も同様の方法で計算可能

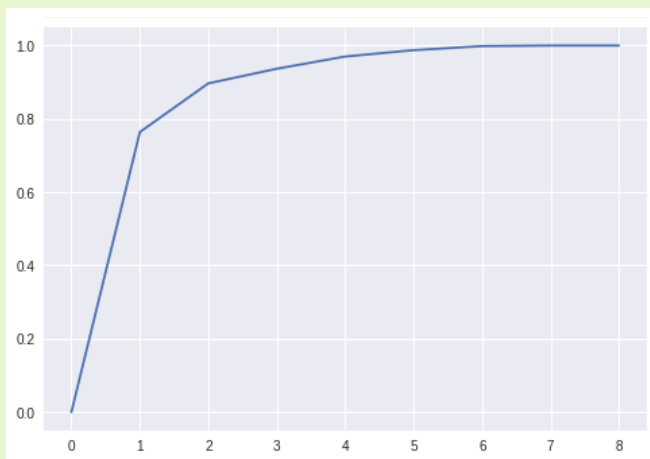
例：主成分分析



参照プログラム
lecture_pca.ipynb

元のデータの
どれくらいを説明できるか

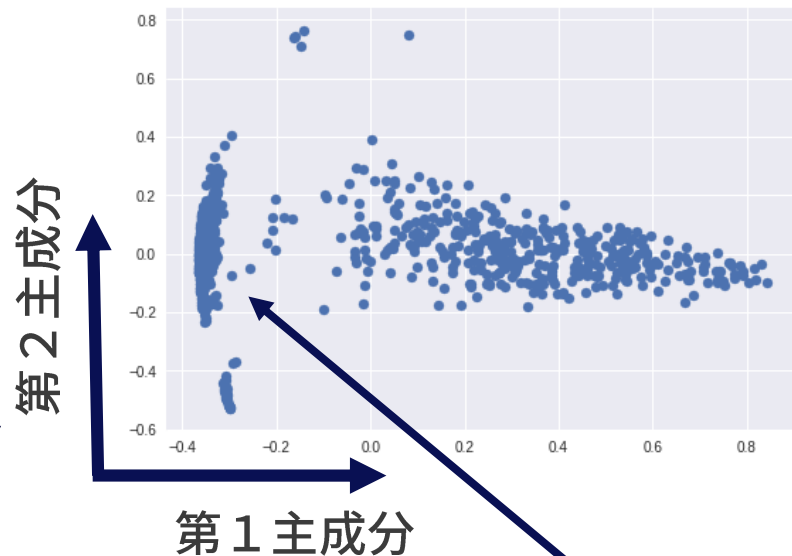
累積貢献度



削減後の次元数

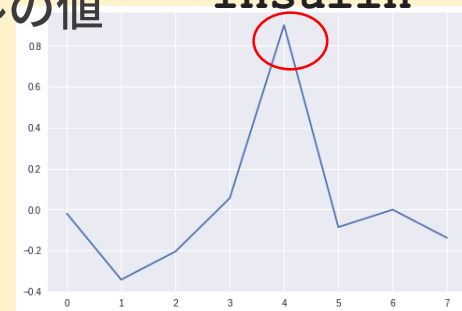
→ 2次元でもかなり元のデータの情報を持っていることがわかる

糖尿病データセットの説明変数のPCA



主成分ベクトルの値

Insulin



元データの属性

Insulinの値がゼロ（欠損）になっていることによる分散が第1主成分と想像できる

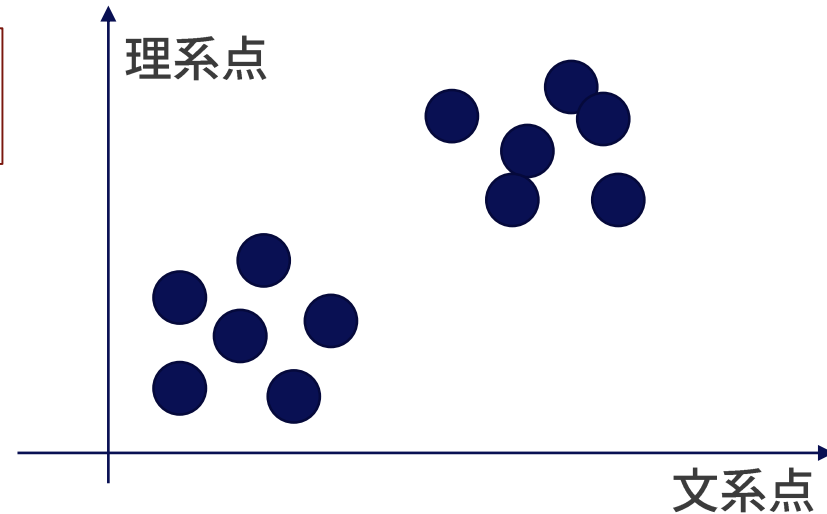


教師なし学習: クラスタリング

クラスタリング：教師なしでデータをいくつかのグループ（クラスター）に分離する手法

例：理系点・文系点ともに高いグループと、ともに低いグループに分かれる

2次元であれば目で見て判断可能



高次元でデータにも適用できる方法論が必要
(説明は2次元で)

	体重	身長	白血球数	好中球数
Aさん	160	50	6000	3400
Bさん	170	55	7000	4200
Cさん	150	45	7500	4200

k-means 法

1. 各データに対してランダムにクラスターを割り振る
2. 各クラスターの中心を計算する
3. データ点に最も近いクラスター中心を見つけて、そのクラスターに割り当てる
4. クラスターの割り当てが変化しなくなるまで2と3を再計算

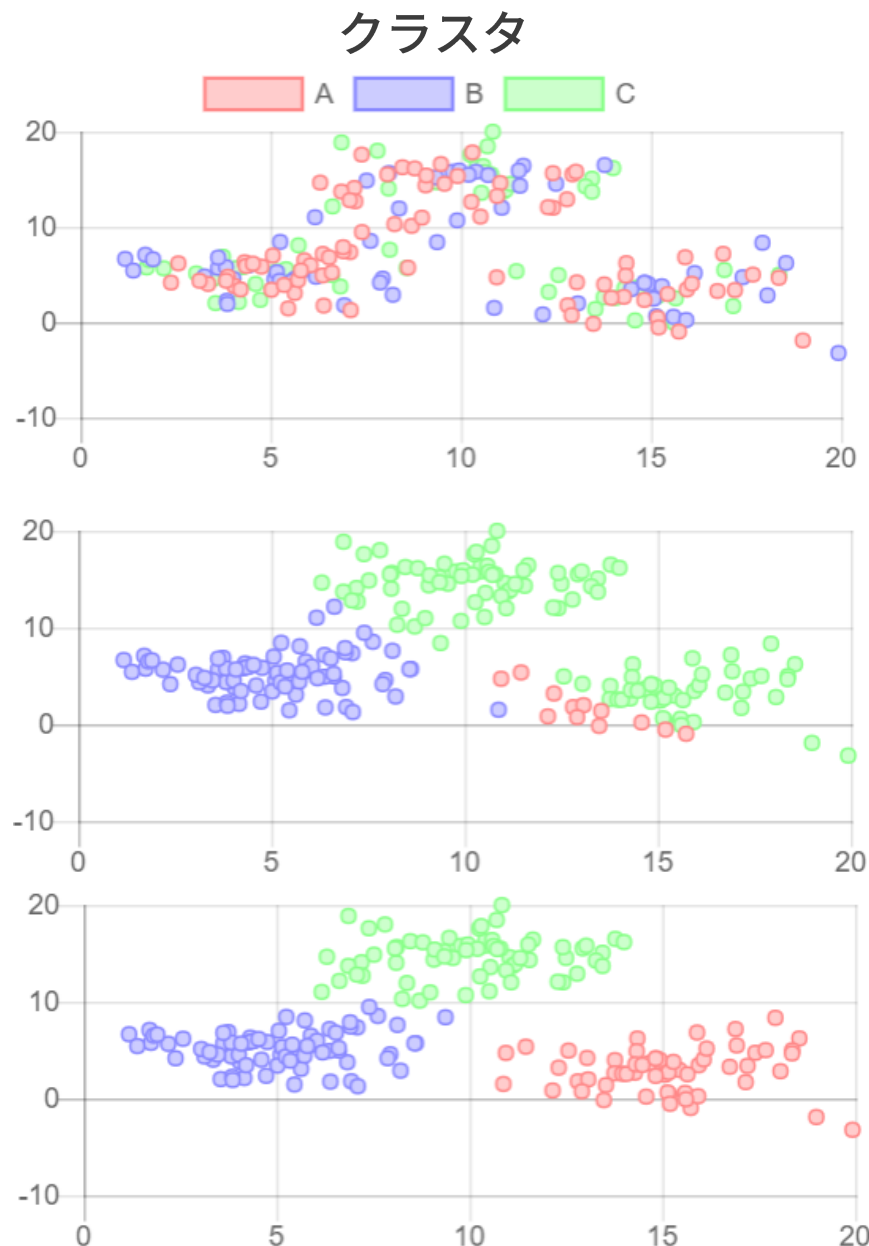
デモ：<http://small-island.work/clustering/>

疑問点

- この繰り返しは必ず止まるのか？
- クラスターの数kをどうやって決めるのか？
- 各クラスターのサイズや形を変えたい場合はどうするのか？



第3回講義にて解説



教師なし学習（再掲）

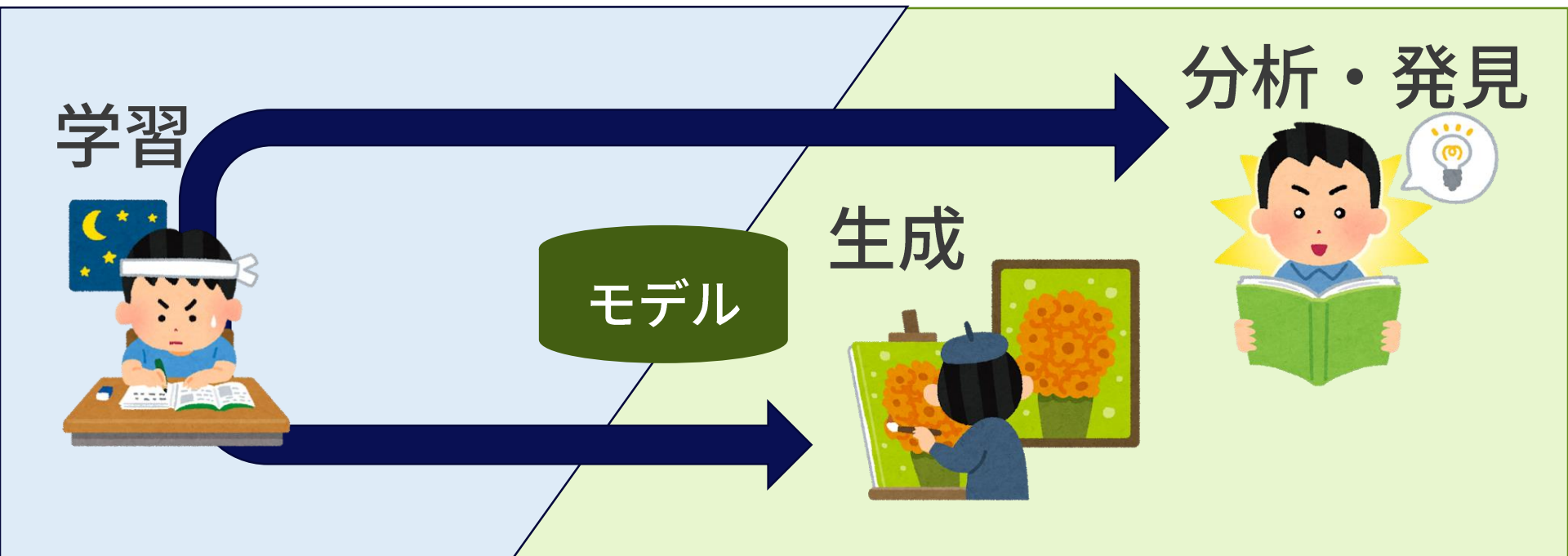
入力データを再現するようなモデルを学習する

[分析・発見]

扱いやすいデータに変換をし、分析に利用する

[生成]

似たデータを生成する



教師なし学習：生成の基本的な考え方

「いびつな6面サイコロ」を100回振った出目を記録して、そのサイコロの出目の偏りを再現したい。100回振った結果は以下のような回数であった

出目 X	1	2	3	4	5	6
回数	10	20	30	20	10	10



それぞれの目が出る確率を計算

出目	1	2	3	4	5	6
確率 $P(X)$	0.1	0.2	0.3	0.2	0.1	0.1

パラメータ

モデル

一様な1000面サイコロ（コンピュータで簡単に作れる）の出目からモデルを使って仮想上のサイコロの目に変換する

仮想上の出目	1	2	3	4	5	6
累積確率	0.1	0.3	0.6	0.8	0.9	1.0
実際の出目	~100	100~ 300	300~ 600	600~ 800	800~ 900	900~ 1000

生成



→いびつなサイコロを振ったデータを生成できる

画像生成

学習する次元を高次元にし、確率分布がより複雑な場合を考えると、画像のような複雑なデータも生成できる

x :画像

$p(x|z)$

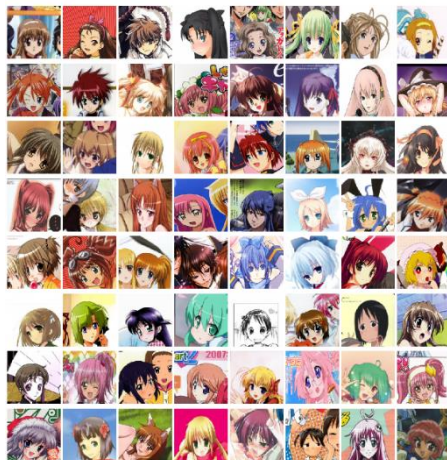
z :パラメータ

モデル

生成



学習データ



アニメイラストの顔データセット
<http://www.nurs.or.jp/~nagadomi/anime-face-character-dataset/>

生成したデータ

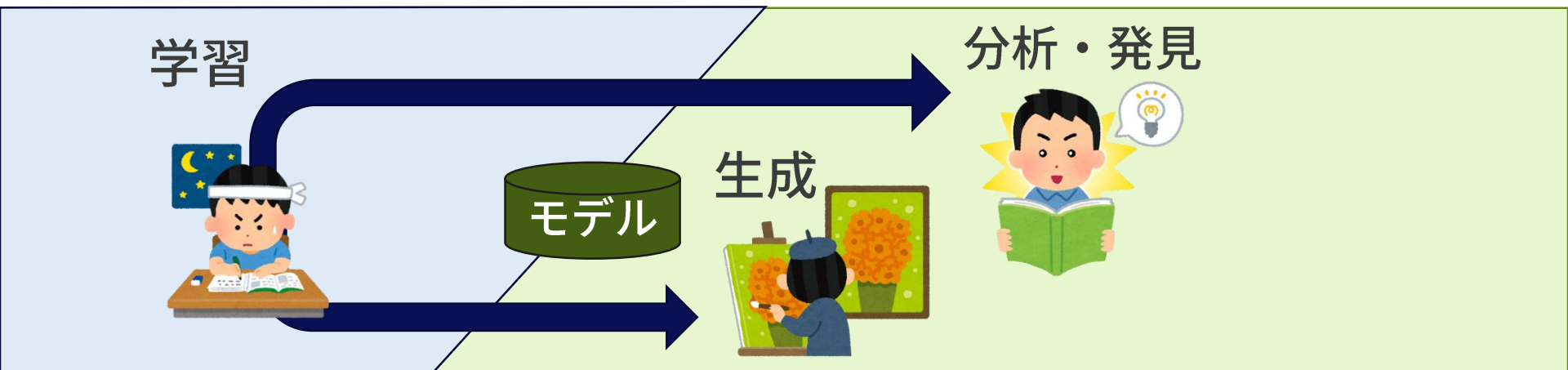


- <https://deepdreamgenerator.com>
- <https://affinelayer.com/pixsrv/>
- <https://deepart.io/>
- <https://paintschainer.preferred.tech/>

教師なし学習まとめ

教師なし学習は分析や生成のために使われており、様々な手法がある
いずれも、モデルの考え方が重要

次元削減：主成分分析 → データをより少ない次元で表現する
クラスタリング：k-means法 → データをクラスタに分離する
生成モデル → データが生成される確率分布を学習する



教師あり学習よりもデータやモデルに関する深い理解が必要なことが多い
→ 第3回講義にて解説（k-means法のモデルに関しても第3回講義）

本日、登場する単語

(数理) モデル

- ある現象を数式で表現したもの
- 本当に現実で意味のあるモデルと数理上都合の良いモデルのバランスをとっているものが多い

パラメータ

- モデルが持つ変数で、データによらない部分で何らかの値を持つ変数

学習

- 与えられたデータからその知識を後に再利用できるようにする
- 特に、モデルのパラメータを決定する学習をパラメータ学習とも呼ぶ

教師あり学習

- データとそれに対応したラベルの入出力関係を学習する

教師なし学習

- データ自体を学習し、そのデータに関する知見を得る

強化学習

- 行動によって得られる報酬を最大化するように学習を行う
学習：より良い行動を選べるよう改善すること

強化学習

あるエージェントが報酬を最大化するために取るべき行動を決定する問題



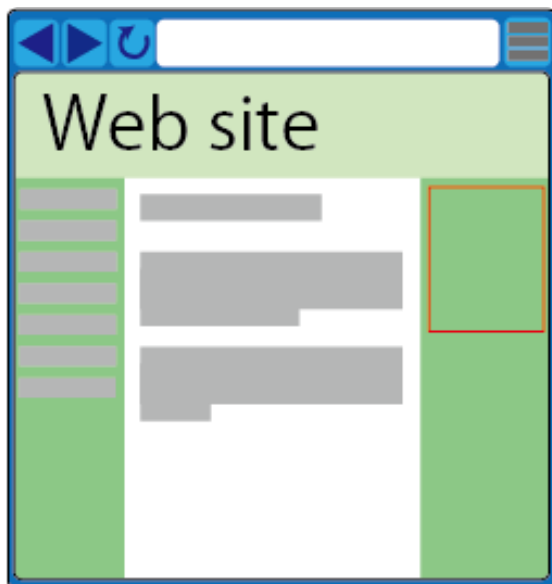
応用例：

- Web広告・デザイン：ユーザのクリック率を最大にするために最適な広告の配置を決定する問題
- 医療・治験：最適な治療法を選択する問題
- ゲーム：勝率を最大化するためにコンピュータの操作を決定する
- 経路制御：最も少ない移動で、目的地に到達する

多腕バンディット問題

問題：

あなたはWebページの運営者で、赤枠内に表示される広告のクリック率を上げたいです。以下の3つのデザインの内、どのデザインを選びますか？



なんらかのアルゴリズムで報酬（例えば広告クリック数の合計）を最大化する

多腕バンディット問題の戦略

今までの履歴から、行動を決定するなんらかのアルゴリズムを考える (**方策**)。

- ランダム
- greedy法

今までの履歴での各デザインのクリックの期待値の大きいものを選択

- ϵ -greedy法

確率 ϵ でgreedy法を選択し、残りの $1 - \epsilon$ でランダムに行動を選択

- ソフトマックス行動選択

行動 i を以下の確率で選択

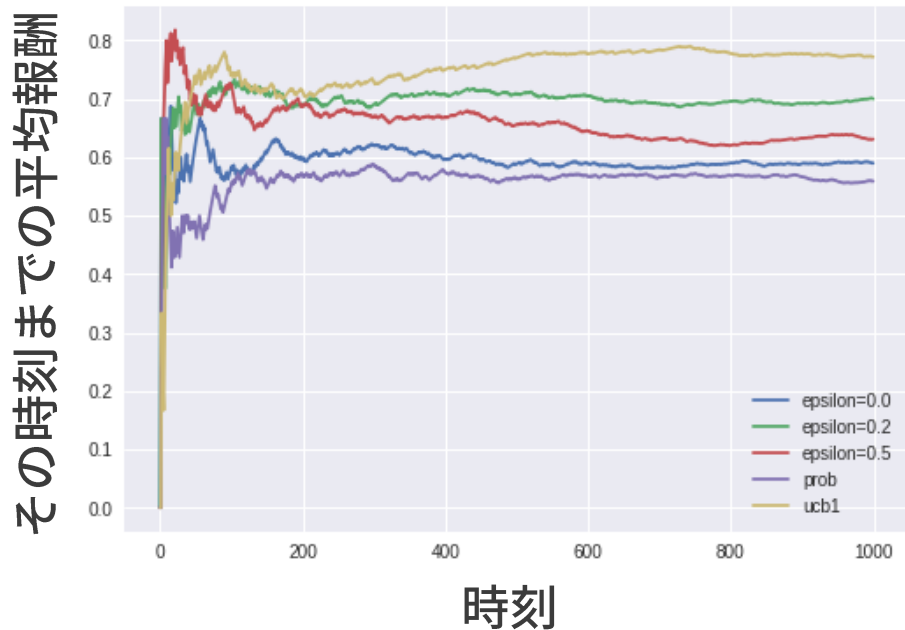
$$p_i = \frac{\exp(\text{expectation}_i)}{\sum_i \exp(\text{expectation}_i)}$$

- UCB1 (Upper confidence bound)

スコアが最も高いものを選ぶ

$$\text{UCB1} = \underbrace{\text{expectation}_i}_{\text{今までの履歴の活用}} + \underbrace{\sqrt{\frac{2 \log t}{N_{i,t}}}}_{\text{探索：今まであまり選んでないものを選ぶ}}$$

t ：現在時刻
 $N_{i,t}$ ：時刻 t までに選んだ回数



ソフトマックス関数：期待値が負でも、この関数の出力は0以上で足して1になる

理論限界をもとにそれに近づけようとして導出されるスコア

今までの履歴の活用

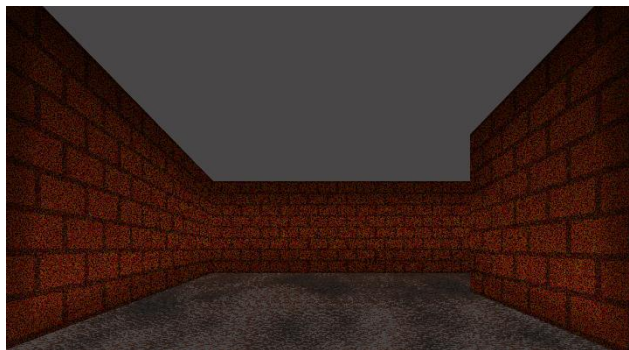
探索：今まであまり選んでないものを選ぶ

状態を考慮した強化学習

先ほどの例ではどのデザインをどういった順番で見せると言った事は関係ない。
状態を考えることで、順序や時間による変化を考えることができる。

問題：

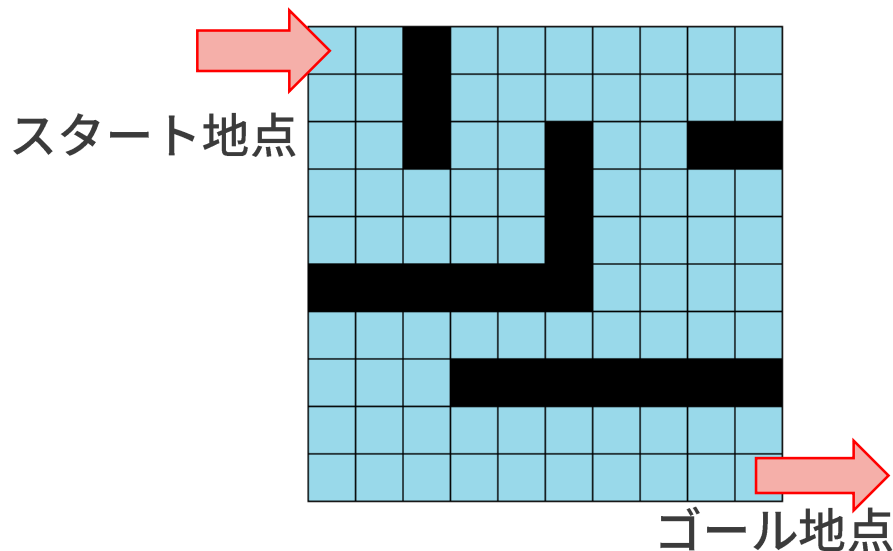
あなたは迷路を進んでいます。分かれ道でどちらに進むかを選択します。何回も繰り返すときできるだけ少ない移動回数でゴールするにはどうすればいいでしょう？



状態：迷路のどこにいるか ($N \times N$ の升目を考える)

行動：左右前後のどちらに移動するか

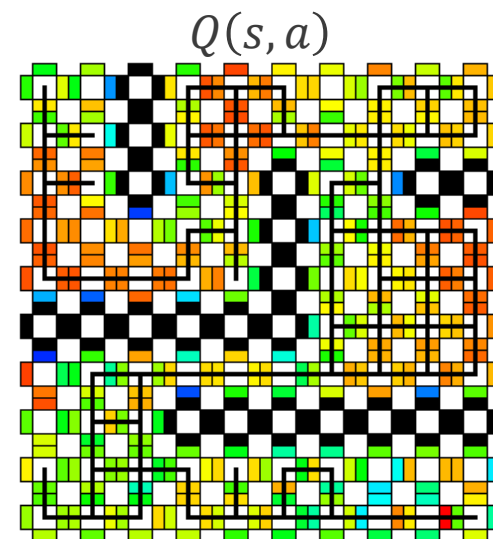
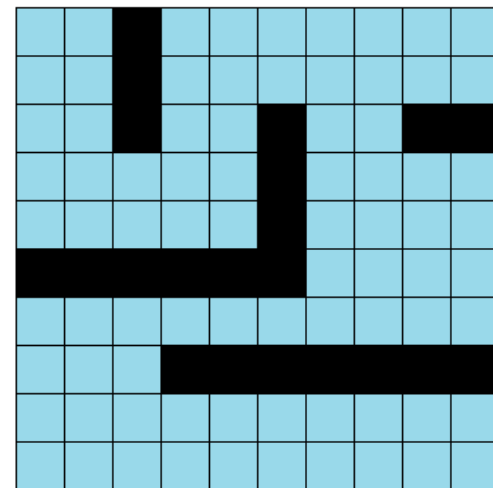
報酬：-(何回移動したか)



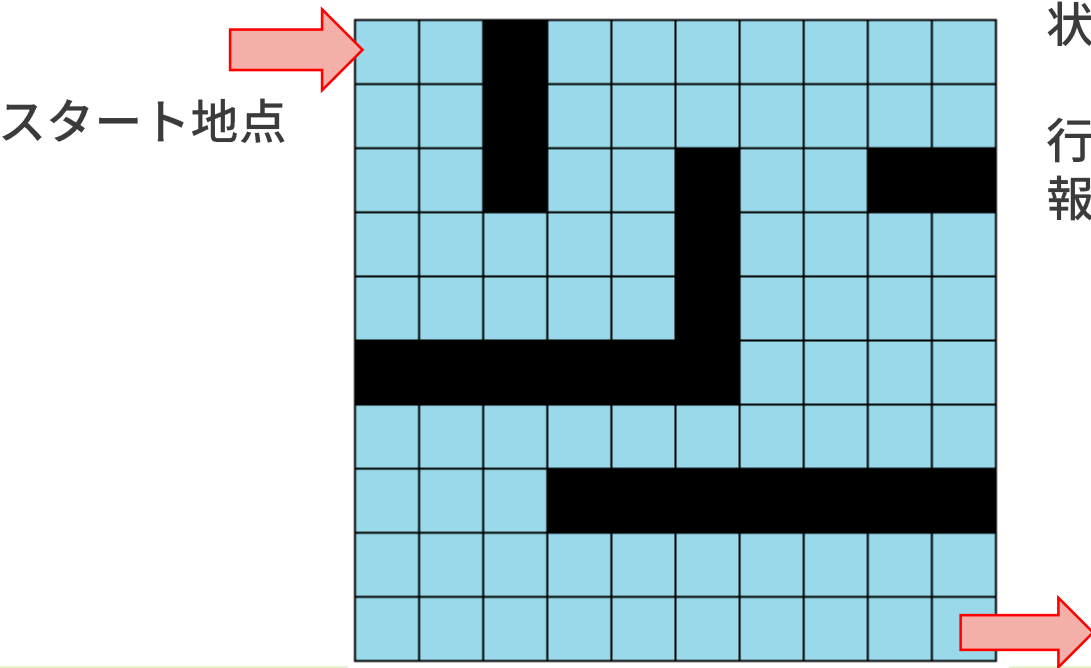
Q学習

$Q(s, a)$: 状態 s で行動 a をとる (残りの最適な行動を行った際の得られる報酬の期待値を) 価値を学習する

1. すべての $Q(s, a)$ をランダムに初期化
2. スタート地点に戻る
3. 現在の状態 s から各行動を行った場合の価値 $Q(s, a)$ に基づいて次の行動を決定する (Q学習の方策: 例えば、 ϵ -greedy 法)
4. 行動を行い、報酬を獲得する、次の状態を決定する
5. $Q(s, a)$ 値を更新
 - 報酬が得られたら Q を上昇させる
 - 将来的に報酬が得られそうであれば Q を上昇させる
6. ゴールに到達 or 一定時間経過したら 2. に戻る
7. そうでなければ、3. に戻る



迷路問題をQ学習で学習させる



状態：迷路のどこにいるか
($N \times N$ の升目を考える)
行動：左右前後のどちらに移動するか
報酬：-(何回移動したか)



強化学習の考え方

実際の問題から強化学習の枠組みに落とし込む必要がある

- 強化学習はMarkov Decision Process(MDP) という枠組みで取り扱われることが多い
- MDPではエージェントは環境から状態を受け取り行動を方策に従って決定し、それによって確率的に環境の状態が変化し、報酬を受け取るというプロセスを繰り返し行う



方策：行動を決定するなんらかのアルゴリズム

強化学習のまとめ

- バンディット問題は単純なため、実際の問題に応じて多くの変更を加えることができる。例えば実際のweb広告などではターゲティング広告や複数の広告を同時に載せている。
- 強化学習を実問題で使う際には、現実の複雑な現象を適切に強化学習の枠組みに落とし込む必要がある
 - 状態が何で、行動が何で、行動によって状態がどう変わるのか、報酬をどうするかといったことを決める

本日のまとめと次回予告

1. 機械学習の基礎
 - 概論
 - 教師あり学習
 - 教師なし学習
 - 強化学習
2. 教師あり学習
 - 最近傍法・線形識別・決定木
 - ランダムフォレスト
 - サポートベクターマシン
3. 確率モデリングと教師なし学習
 - クラスタリング
 - ナイーブベイズモデル
 - 混合ガウスモデル
 - クロスバリデーション・モデル選択
4. データ構造と機械学習アルゴリズム実践
 - テーブルデータ
 - 行列データ
 - スパースモデリング
 - 時系列データ
 - グラフデータ

ここまでのeラーニングでわからない人向けの外部のeラーニング授業動画・教材の紹介

- 【YouTubeチャンネル】 Able Programming
<https://www.youtube.com/channel/UCh5M2YUAPW7HnnpfTUv7XHmA>
「機械学習をはじめよう」
https://www.youtube.com/watch?v=gWL_E3zub9g
- Aidemy 機械学習概論 人工知能と機械学習とは？
<https://www.youtube.com/channel/UCqmWJJJolqAgjldLqK3zD1QQ>
- フォーサイトAI講座：機械学習入門 01 機械学習とは
<https://www.youtube.com/watch?v=C-zw1TZjPQ8>