

数理データサイエンスコース プログラミング体験講義

岡山大学 オープンキャンパス2023

指定された席に着席後

各自、下記を実施して下さい

- ① パソコン上のwebブラウザに、本日使うファイルの一覧が表示されている→これら全てをダウンロード
- ② (まだ作っていない人は) googleアカウントを作成する
- ③ webブラウザで"google colab"で検索し、 google colaboratoryのページに行く

ここをクリック



- ④ (時間が余った人は)
本スライドのp5~p10の
「google colab実行手順」を実際に実行してみてください

補足：自宅で今回の演習内容を実施するには

- 本日使用するファイルは
<http://www.mtds.okayama-u.ac.jp/OC/>
上に置いてあります
- 自宅（あるいは自分の）パソコンのwebブラウザで
上記URLにアクセスし、ファイルをダウンロードすれば、
今回の演習内容を家でも実施できます。

Google Colaboratoryとは

- 通称Google colab.
- googleのアカウントを持ち, かつインターネットさえ繋がれば, webブラウザ上でプログラミング (例: python) が行える

google colab実行手順： googleアカウントでログイン

- google colabトップページ（下記画面）の画面右上から、googleアカウントにログインする（すでにログインしている場合は不要）



google colab実行手順：

「.ipynb」形式のファイルのアップロード

- ログインできたら下記の画面になる
→ 「アップロード」をクリック



The screenshot shows the Google Colaboratory interface. The top navigation bar includes 'Colaboratory へようこそ' and '共有' (Share) settings. The main content area displays a modal window for uploading files. The modal has a header with tabs: '例' (Example), '最近' (Recent), 'Google ドライブ' (Google Drive), 'GitHub', and 'アップロード' (Upload). The 'アップロード' tab is highlighted with a blue circle. Below the tabs, there is a table with columns for 'タイトル' (Title), '最初に開いた日時' (Date first opened), '最終閲覧' (Last viewed), and a trash icon. The table contains two entries, both titled 'Colaboratory へようこそ', with '最初に開いた日時' and '最終閲覧' both set to '0分前' (0 minutes ago) and '6分前' (6 minutes ago) respectively. At the bottom of the modal, there are links for 'ノートブックを新規作成' (Create new notebook) and 'キャンセル' (Cancel). The background interface shows a sidebar with a table of contents and a main workspace area with a code editor and a terminal.

タイトル	最初に開いた日時	最終閲覧	
Colaboratory へようこそ	0分前	0分前	🗑️
Colaboratory へようこそ	6分前	6分前	🗑️

google colab実行手順：

「.ipynb」形式のファイルのアップロード

- アップロードをクリックすると下記の画面になる
→本日使用する「.ipynb」形式ファイルを，画面中央に向かってドラッグ&ドロップ
(本日は最初に"Python入門.ipynb"を使うので，このファイルを下記画面中央にドラッグ&ドロップして下さい)



google colab実行手順：

「.ipynb」形式のファイルの確認

- Python入門.ipynbファイルが表示された。
- 各セル（灰色の箱）に，コードが打ち込んである。

Python入門.ipynb ☆

コメント 共有

接続 | 編集

+ コード + テキスト

Python入門

四則演算：+, -, *, / , ()をつかってみよう。

例：12+5, 4-3, 2*4, 3/5, (3+2)*3

```
[ ] 2*3
```

```
[ ] 3/4
```

```
[ ] 3-5
```

```
[ ] 4*(5-3)
```

```
[ ] 5**2
```

```
[ ] 4+5*2
```

各自で入力してみよう

```
[ ]
```


google colab実行手順： コードを実際に実行してみる

- セルにマウスカーソルを合わせると、実行ボタンが表示される→これをクリックすると、コードが実行される（出力結果が下に表示される）

実行ボタン



The screenshot shows a Google Colab notebook interface. At the top, there are buttons for '+ コード' and '+ テキスト'. Below that, a search bar contains 'Python入門'. A dropdown menu is open, showing a folder icon and the text '四則演算：+, -, *, /, ()をつかってみよう。'. Below this, an example is given: '例：12+5, 4-3, 2*4, 3/5, (3+2)*3'. The main area shows a code cell with a play button icon (the execution button) circled in blue. The code in the cell is '2*3'. Below the code cell, there are three output cells, each starting with '[' and containing the result of the code above: '3/4', '3-5', and '4*(5-3)'. The play button is highlighted with a blue circle and a line pointing to the text '実行ボタン' on the left.

google colab実行手順：

外部ファイルをcolab上に追加

①画面左のフォルダのアイコンをクリックし，下記の画面にする（左側に「ファイル」に関するタブが出る）



②この余白部分に，colab上に追加したいファイルをドラッグ&ドロップ

"Python入門.ipynb"を開いている時は **sazae.csv**をcolab上に追加しておいて下さい

本日の流れ

- ミニ講義：機械学習とは
- 演習①：Python入門
 - 使用ファイル：Python入門.ipynb, sazae.csv
- 演習②：アヤメデータの教師あり分類学習
 - 使用ファイル：Ayame.ipynb, iris0.csv
- 演習③：手書き文字データの教師あり分類学習
 - 使用ファイル：NumberTest.ipynb

機械学習とは

注目される人工知能（AI）技術

**データサイエンス：
国のAI戦略で「全大学生が習得」**

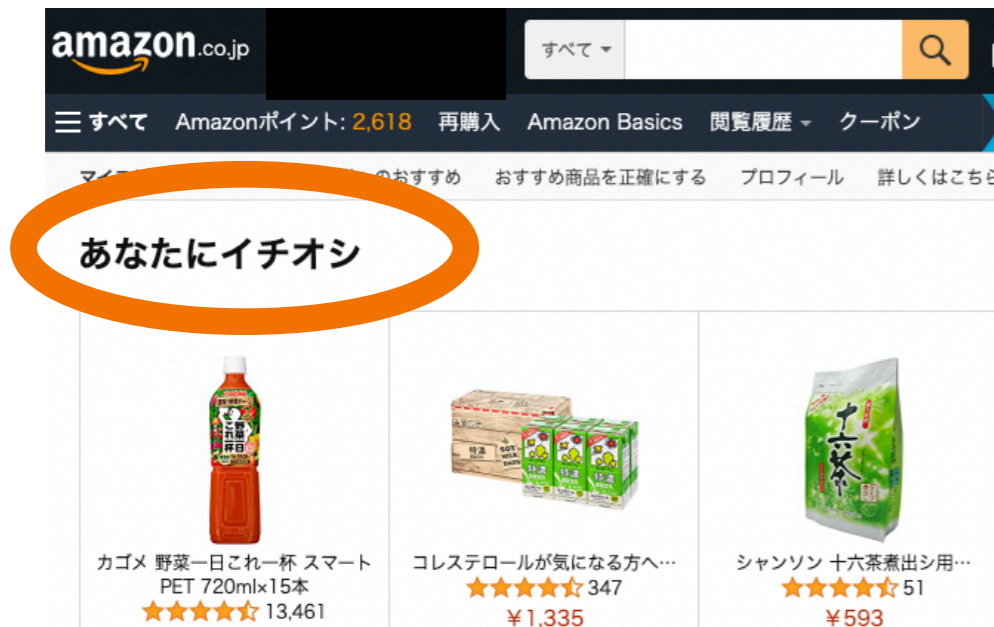


**デジタル人材* 「別枠採用」が3割
主要企業の来春新卒**

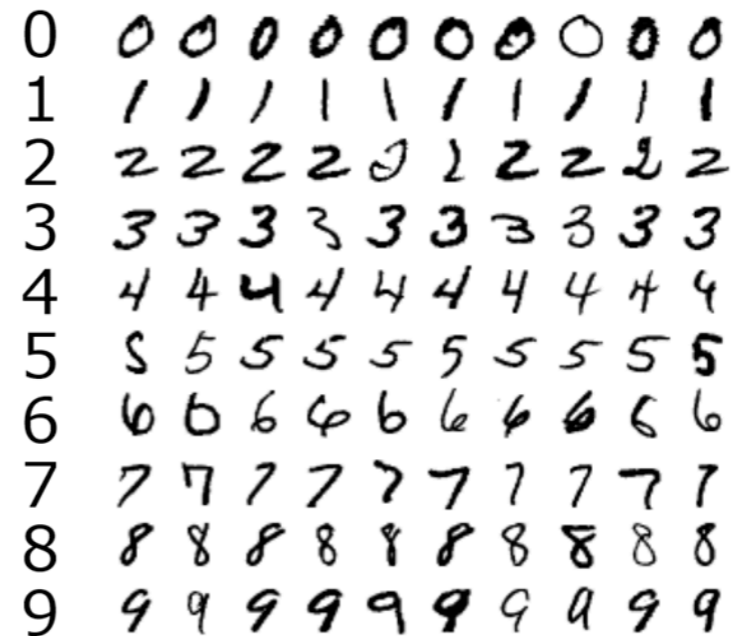
*（記事では）データ分析や人工知能（AI）
などの専門人材を指す

"AI" (機械学習) 技術の例

推薦システム



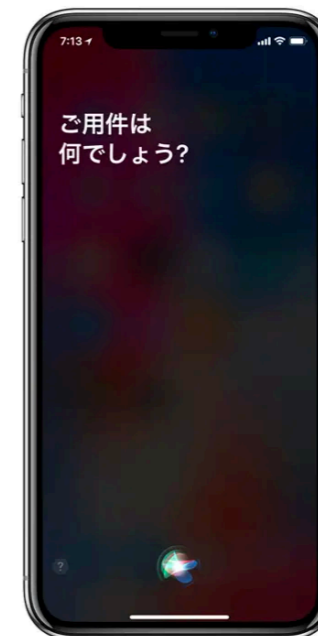
手書き文字認識



カメラの顔認識

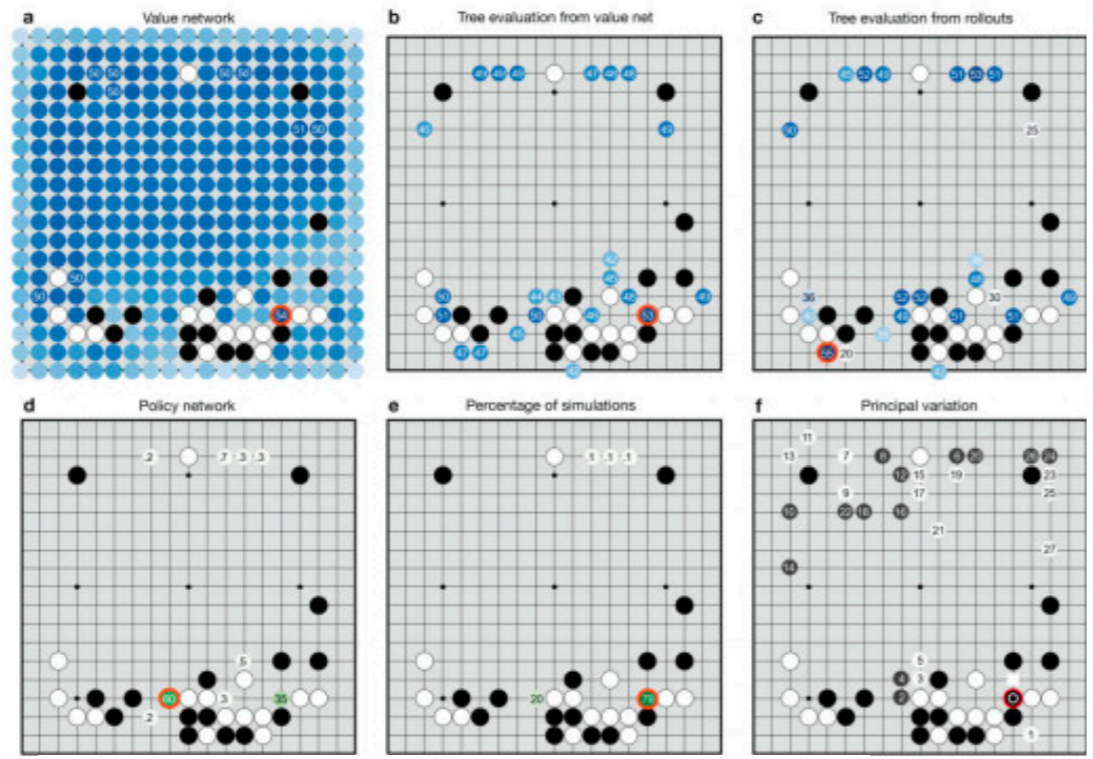


音声認識 (Siri)



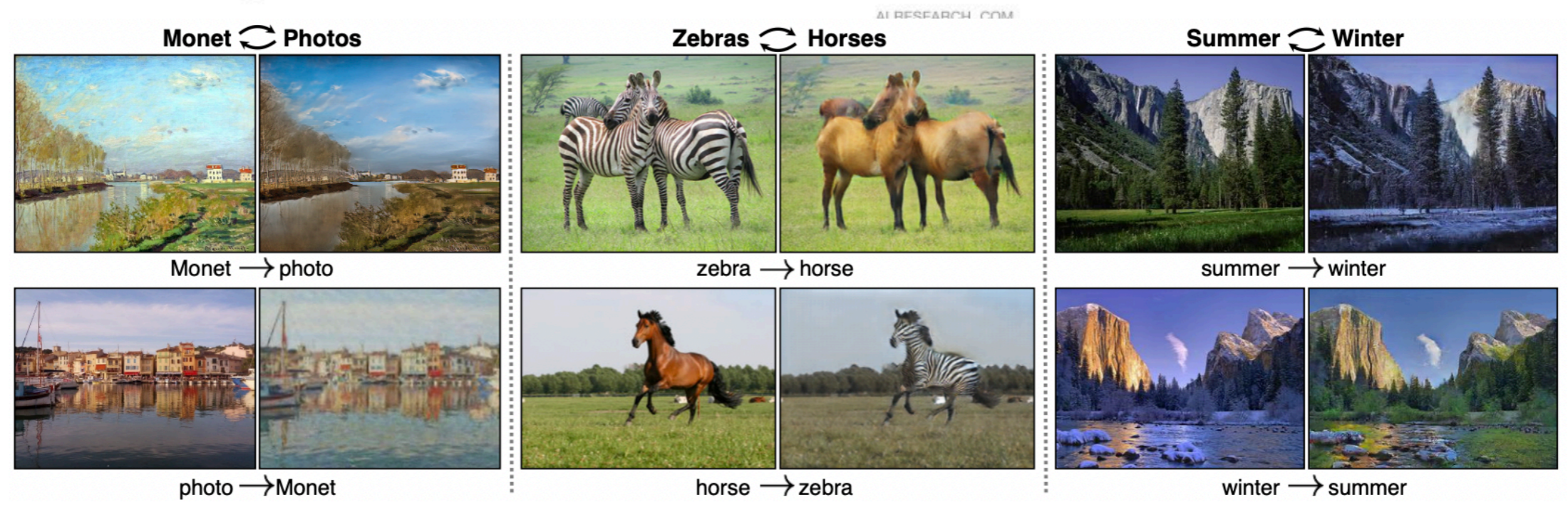
"AI" (機械学習) 技術の例

→プロ棋士を破った囲碁のAI (Alpha Go)



好みのイラストをAIで生成

↓ 画像の特徴を変換



機械学習とは

- 機械学習：多くの人工知能（AI）の基盤となっている技術.
- →この講義では，機械学習のカテゴリの1つ「教師あり学習」に注目し，その一部を実際に体験してみる.

機械学習とは（教師あり学習の場合）

教師あり学習とは...予測や判別等のために使う関数を、得られたデータから"学習"すること。

- 一般の数学でいう"関数"：入力 x に対し，何らかの加工して， y を出力（例： $f(x) = 2x + 3 = y$ ）

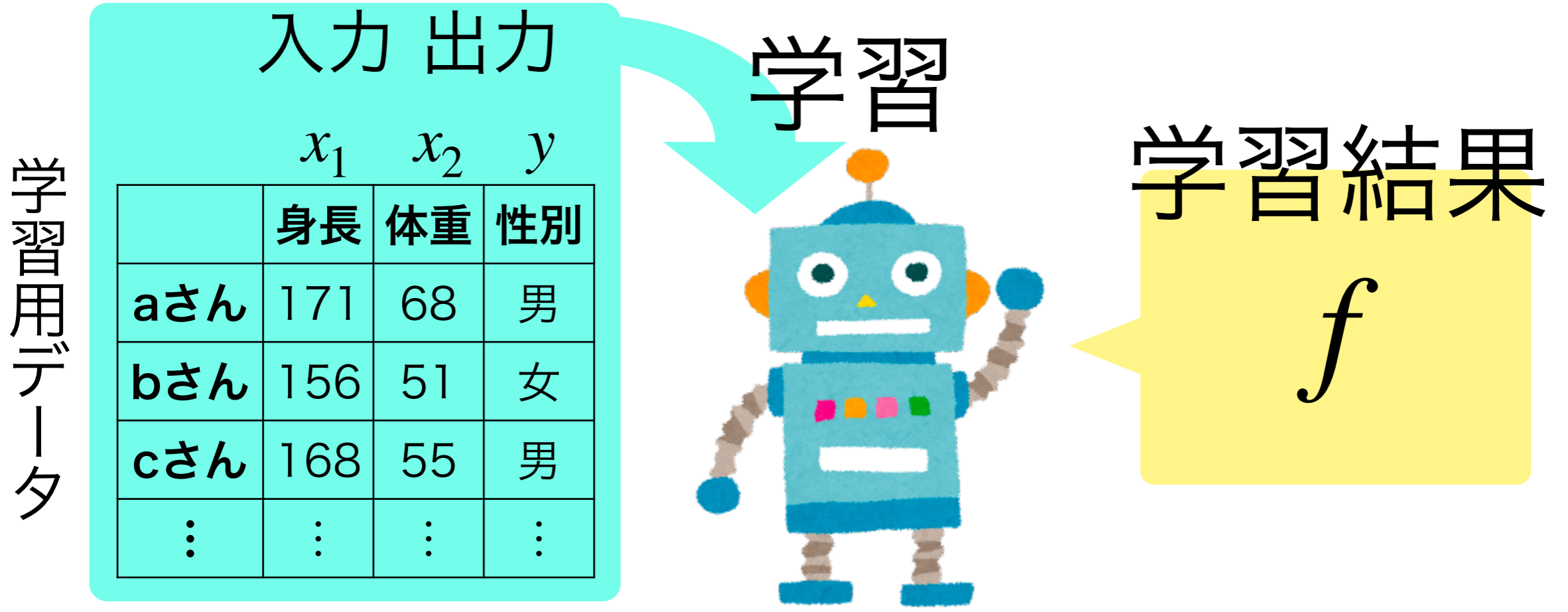


- 機械学習（教師あり学習）で扱う"関数"：入力から出力を予測。関数 f の中身は未知。



教師あり分類学習の例：身長・体重から性別予測

- 目的：身長・体重から性別を予測する関数 f を学習。



- 学習結果を用いると...

検証用データ

入力情報のみ

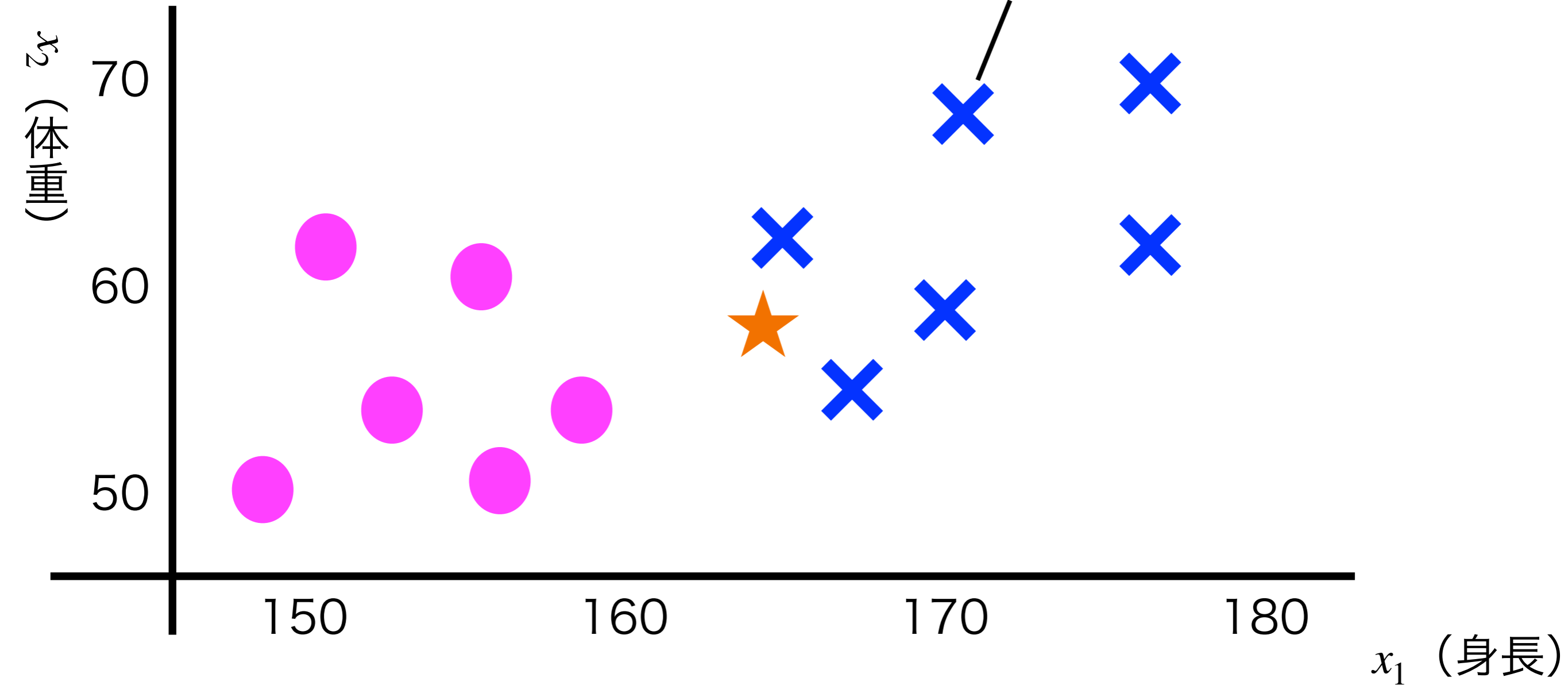
$$f(165, 60) = \text{男}$$

身長・体重を入力すると性別を予測

「関数 f を学習する」とは？

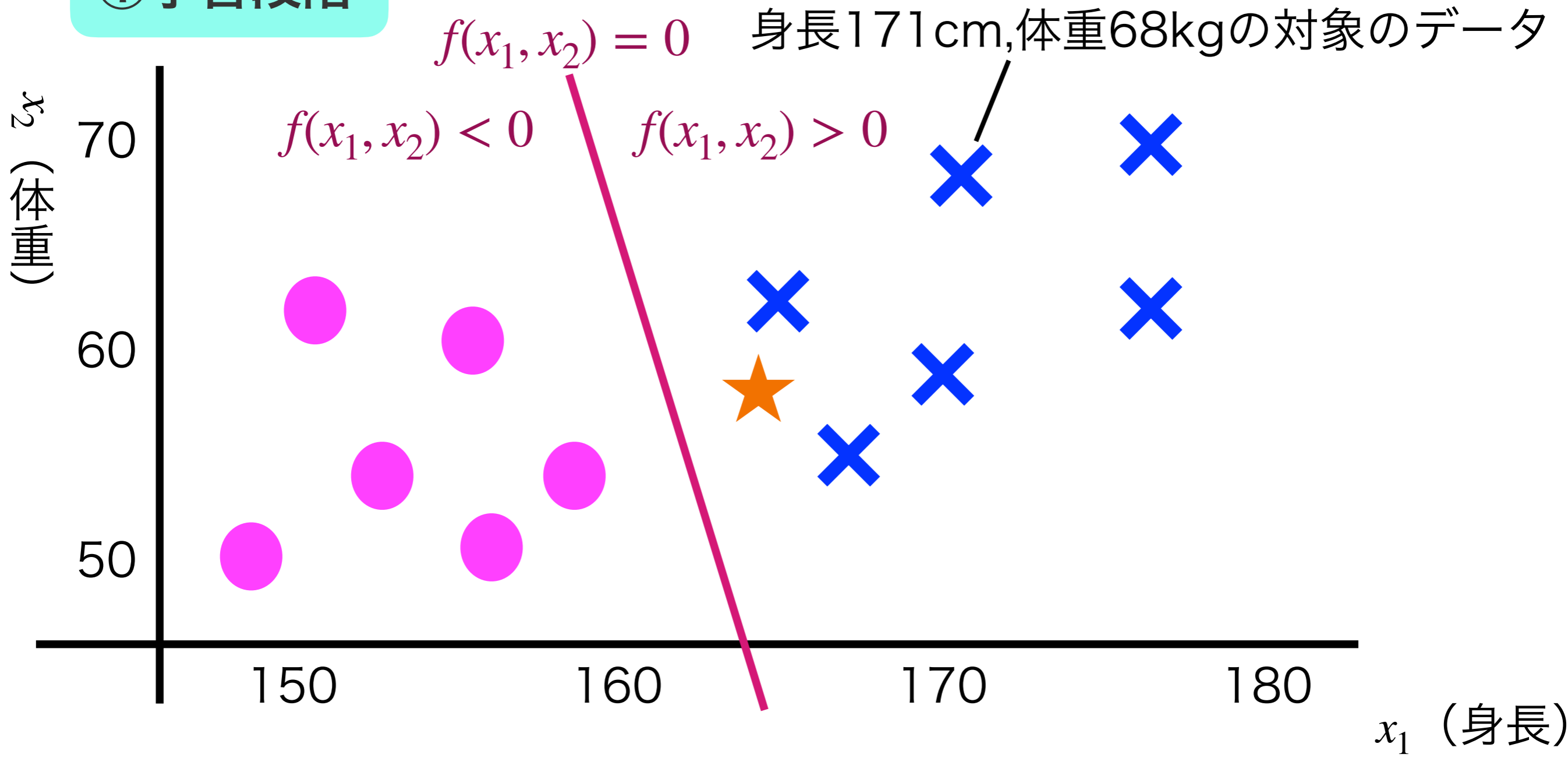
①学習段階

身長171cm,体重68kgの対象のデータ



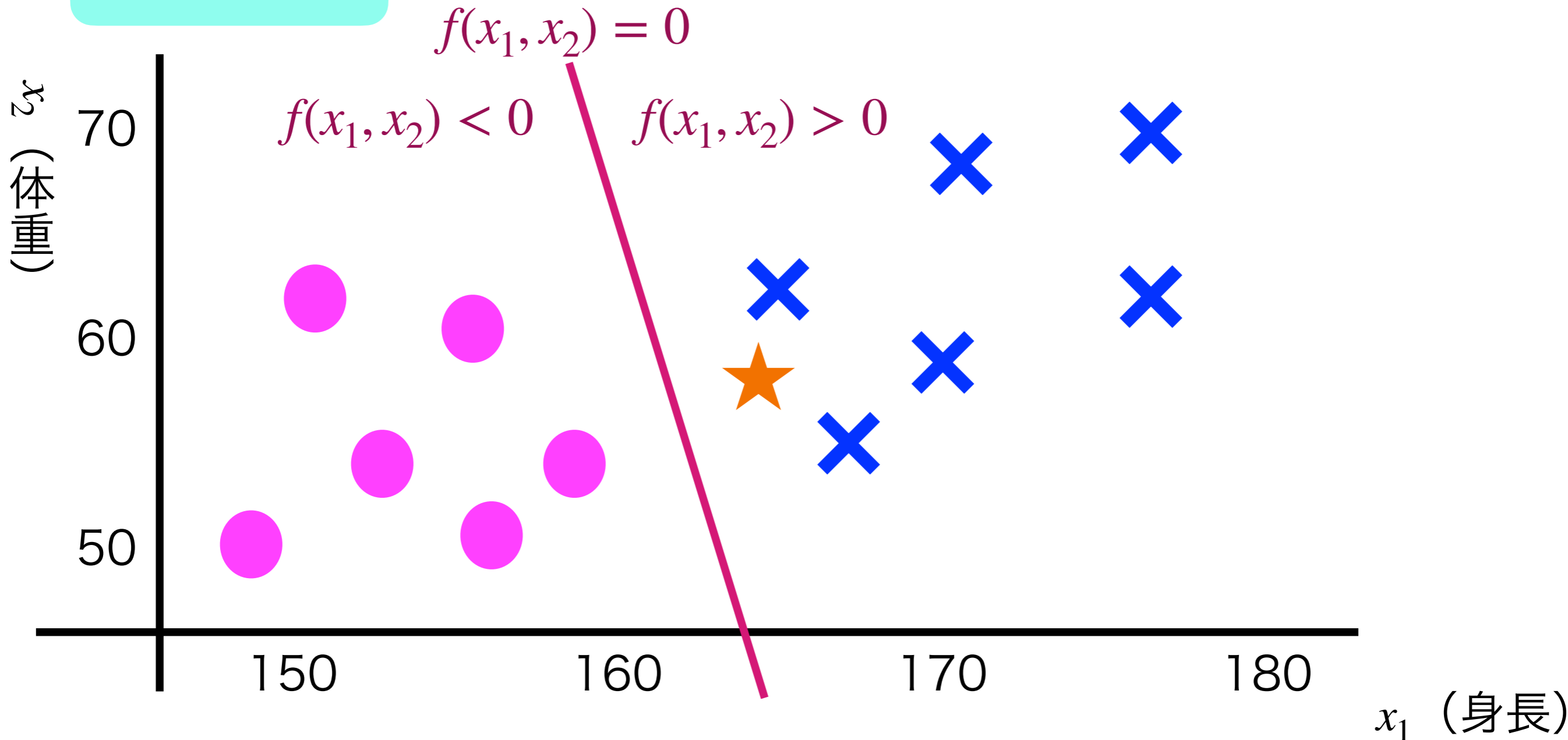
「関数 f を学習する」とは？

①学習段階



「関数 f を学習する」とは？

②検証段階



zさん（性別の情報なし）は身長164，体重58.

$f(164, 58) = 0.01 > 0 \rightarrow$ zさんは男性と予測

「関数 f を学習する」とは？

- Q,境界線の役割を果たす関数 f はどうやって得られるの？
- A, f を下記のように定義すればよい.

$$f(x_1, x_2) = w_0 + w_1x_1 + w_2x_2 = w_0 + \mathbf{w}^T \mathbf{x}_i \quad \dots\dots\dots(\star)$$

- ただし, $w_0, \mathbf{w} = (w_1, w_2)^T$ は未知.
- 学習=関数 f の未知係数 w_0, \mathbf{w} の具体的な値を得ること

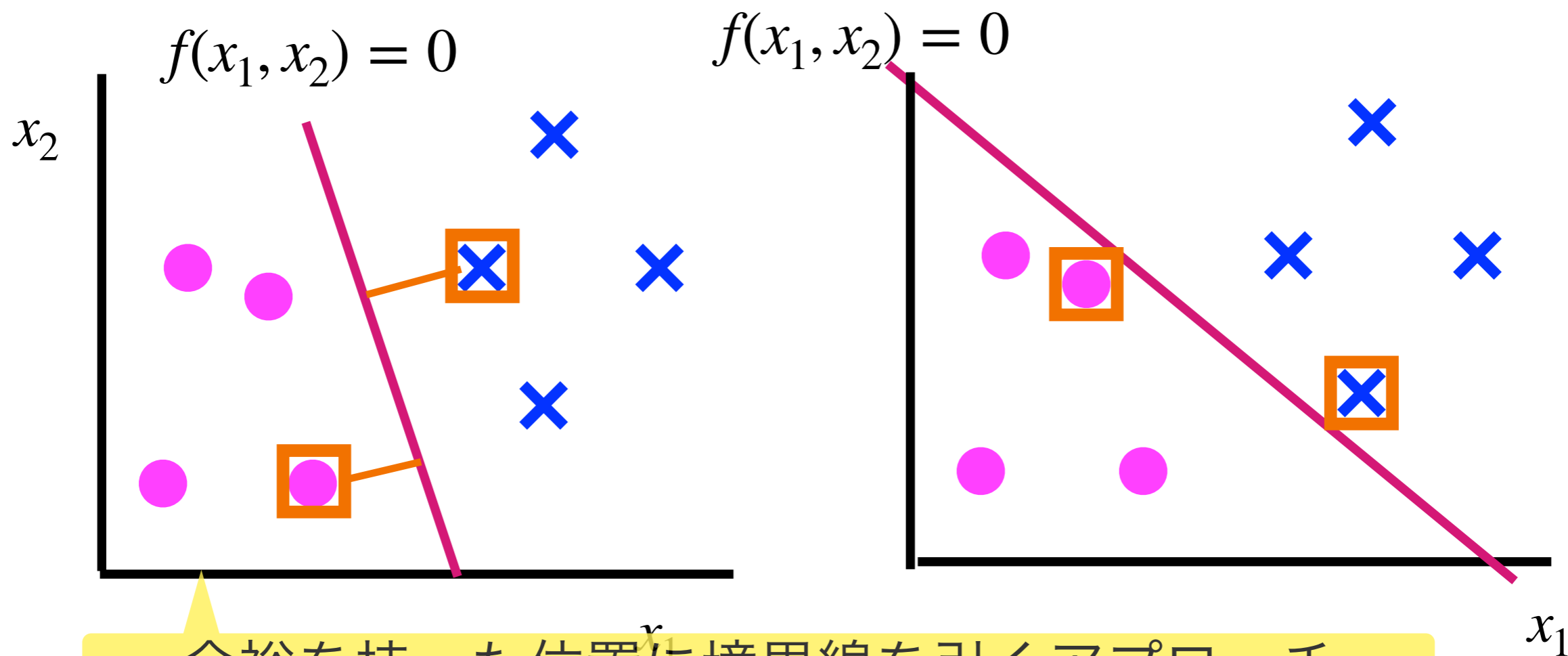
- この場合, 境界線は下記を満たす x_1, x_2 の集合.

$$f(x_1, x_2) = w_0 + w_1x_1 + w_2x_2 = 0$$

→直線の方程式!

サポートベクターマシン (SVM)

- 関数 f (境界線) の推定の方法は様々.
- Q, 新たなデータが得られた時, 分類の失敗が少なそうな境界線はどっち?



余裕を持った位置に境界線を引くアプローチ
→サポートベクターマシン (SVM)

補足：SVMにおける境界線の学習

- SVMでは下記の最適化問題を通じて w_0 , \mathbf{w} を求める。
 - 境界線から x の点までの最短距離（マージン）が最大になるように求める ($y_i \in \{-1, 1\}$)

$$\max_{\mathbf{w}, w_0} \min_i \frac{|\mathbf{w}^\top \mathbf{x}_i + w_0|}{\|\mathbf{w}\|}$$

$$\text{s.t. } y_i(\mathbf{w}^\top \mathbf{x}_i + w_0) \geq 0, (i = 1, \dots, n)$$

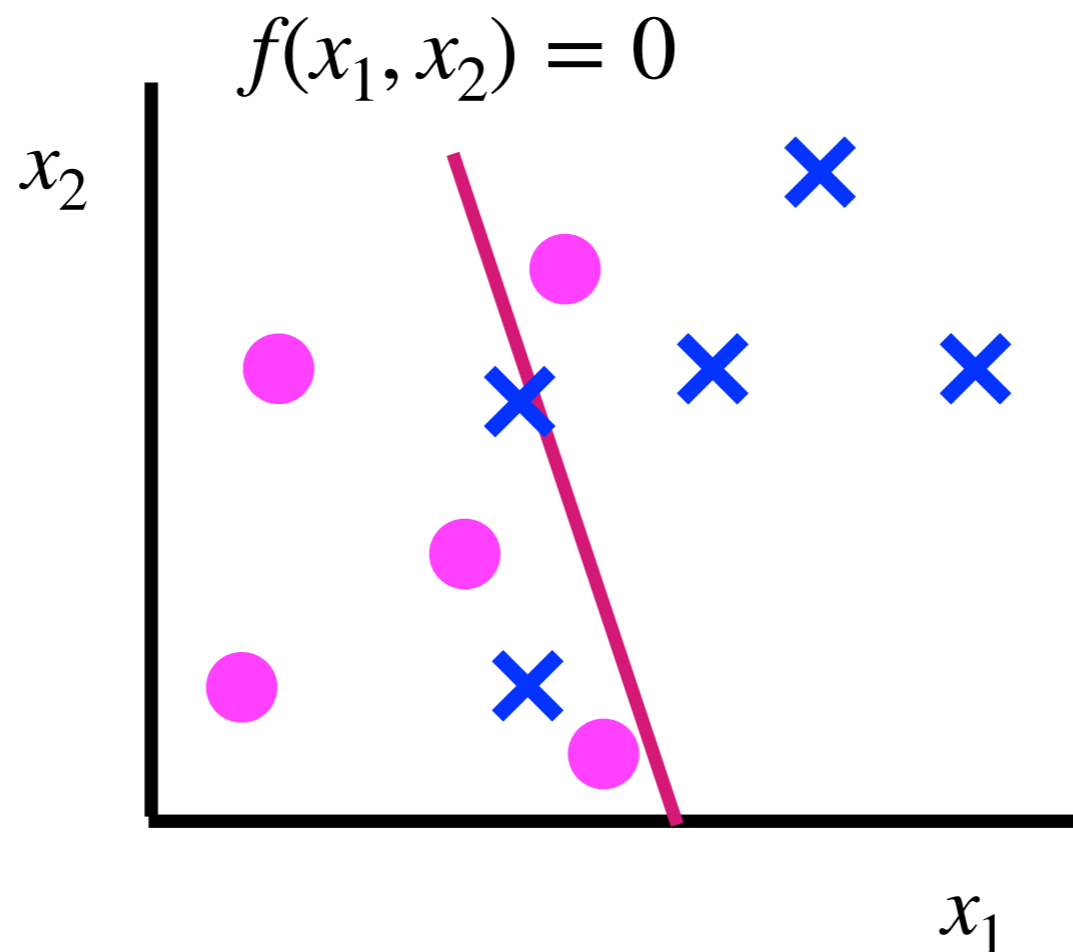
- 実際は上記を簡略化した下記の最小化問題をとく

$$\min_{\mathbf{w}, w_0} \frac{1}{2} \mathbf{w}^\top \mathbf{w}$$

$$\text{s.t. } y_i(\mathbf{w}^\top \mathbf{x}_i + w_0) \geq 1, (i = 1, \dots, n)$$

サポートベクターマシン (ソフトマージン)

- 下記のような場合：直線で分類しようとしても，多少の分類の失敗が生じてしまう...
- 分類の失敗を許容しつつ，直線で分類するSVM
→ソフトマージンSVM (今回使用!!!)



補足：ソフトマージンSVMにおける 境界線の学習

- SVMでは下記の最適化問題を通じて w_0 , \mathbf{w} を求める

$$\min_{\mathbf{w}, w_0, \xi} \frac{1}{2} \mathbf{w}^\top \mathbf{w} + C \sum_{i=1}^n \xi_i$$

$$\text{s.t. } y_i(\mathbf{w}^\top \mathbf{x}_i + w_0) + \xi_i \geq 1,$$

$$\xi_i \geq 0, \quad (i = 1, \dots, n)$$

- $\xi = (\xi_1, \dots, \xi_n)^\top$: 各対象の誤差
- $C \geq 0$: 誤差の重要度を示すハイパーパラメータ
 - 事前にユーザーが設定
 - C が大きいほど、誤差に対し寛容になる

これまでのまとめ

- 分類の教師ありの機械学習では，下記の2段階を行う
 - ① 学習（訓練）段階：データ \mathbf{X} , \mathbf{y} で，関数 f （境界線）を学習.
 - ② 検証段階：データ \mathbf{X} , \mathbf{y} で，学習した関数 f の性能検証
 - 例, 新たなデータ \mathbf{X} に対する予測, 正解率の計算
- ここでの学習とは，「直線の境界線を求める」こと
 - 理由：学習する関数 f が，直線の方程式を表すため
- 今回学習には（ソフトマージン）SVMを用いる.

Pythonの演習

本日の流れ（再掲）

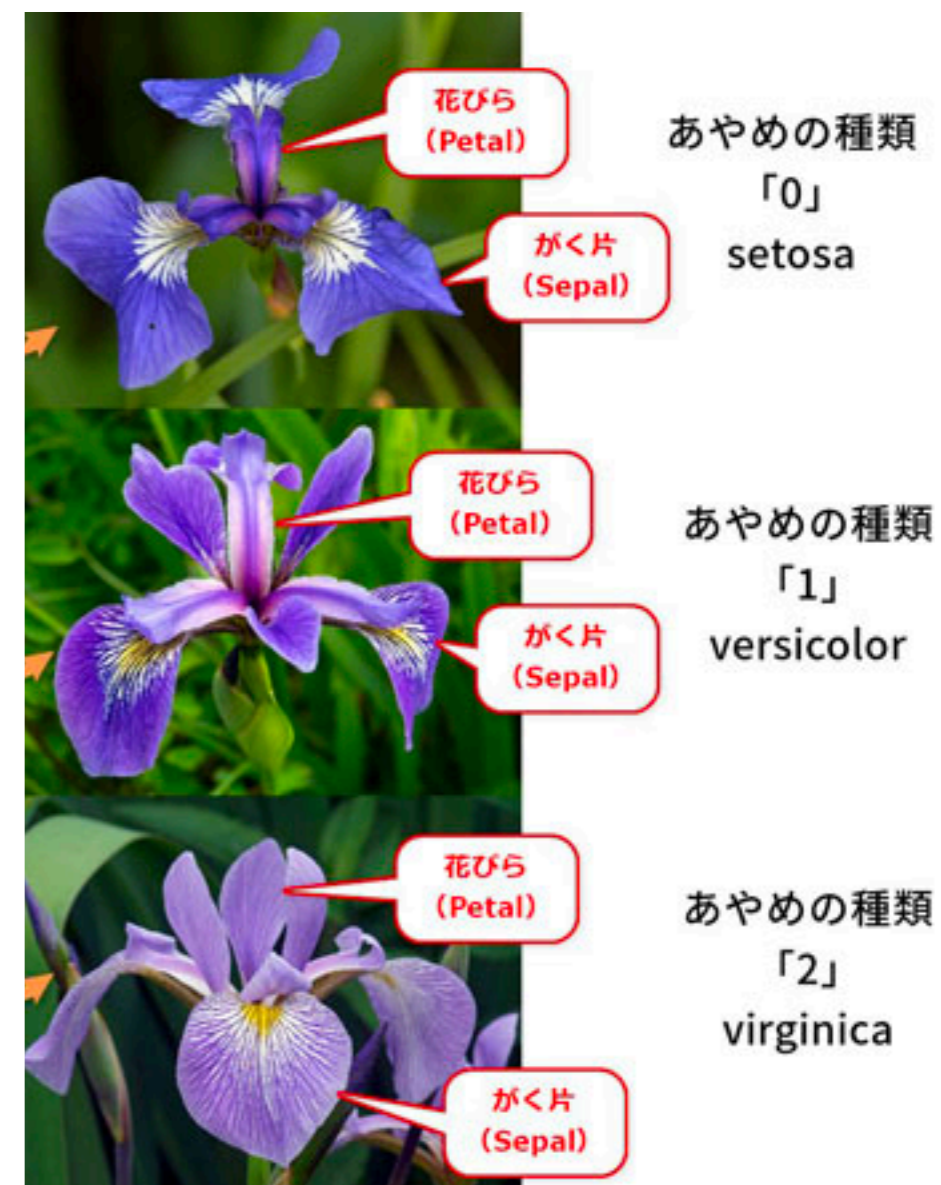
- ミニ講義：機械学習とは
- 演習①：Python入門
 - 使用ファイル：Python入門.ipynb, sazae.csv
- 演習②：アヤメデータの教師あり分類学習
 - 使用ファイル：Ayame.ipynb, iris0.csv
- 演習③：手書き文字データの教師あり分類学習
 - 使用ファイル：NumberTest.ipynb

Python (パイソン) について

- Pythonとは：
 - プログラムを作るための言語 (プログラミング言語) の1つ.
 - 機械学習やデータ分析を行うのに優れている.

使用データ1：アヤメ (iris) データ

- 花びら・がく片の大きさから，アヤメの種類を予測する
- 入力（特徴量） X (150x4 行列)：
 - がく片の長さ (cm)：Sepal Length
 - がく片の幅 (cm)：Sepal Width
 - 花びらの長さ (cm)：Petal Length
 - 花びらの幅 (cm)：Petal Width
- 出力（クラス） y (長さ150ベクトル)：
 - setosa (セトーサ)
 - versicolor (バーシーカラー)
 - virginica (バージニカ)



使用データ2：手書き文字データ

- 手書き文字画像から，その文字を予測する.
- 入力（特徴量） \mathbf{X} （1797x64 行列）：
 - 8x8ピクセルの濃さが変量となっている（8x8=64）
 - 濃さは16段階で表現（ここでは0が白で，16が黒とする）.
- 出力（クラス） \mathbf{y} （長さ1797ベクトル）：
 - 0~9の数字.

