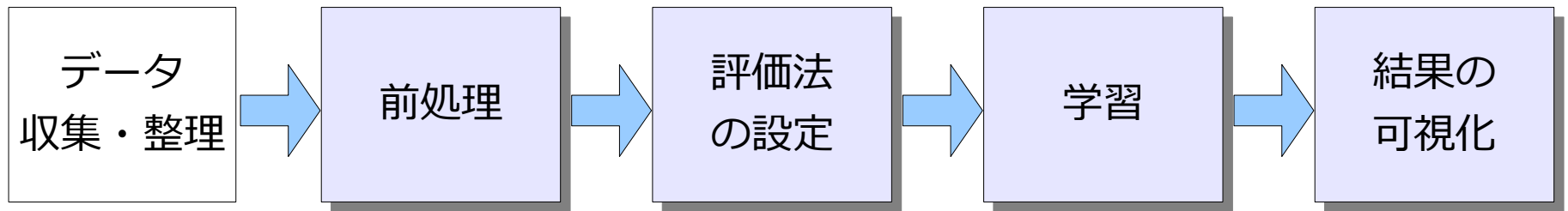



## 2. 機械学習の基本的な手順



 : ツールによる支援が可能

## 2.1 Weka を用いた機械学習

- Weka とは
  - Waikato Environment for Knowledge Analysis
  - 機械学習のアルゴリズムを実装した Java ライブラリ
  - データファイルを直接操作できる GUI を持つ
  - 説明は開発者版 3.9.3 に基づく
  - ライセンスは GNU GPL
    - プログラムの実行・改変・再配布が自由
    - ただし二次的著作物に対しても GNU GPL が適用される

# 勉強のためのデータセット

表 2.2 Weka 付属のデータ (一部)

データ名	内容	特徴	正解情報
breast-cancer	乳癌の再発	カテゴリ	クラス (2 値)
contact-lenses	コンタクトレンズの推薦	カテゴリ	クラス (3 値)
cpu	CPU の性能評価	数値	数値
credit-g	融資の審査	混合	クラス (2 値)
diabetes	糖尿病の検査	数値	クラス (2 値)
iris	アヤメの分類	数値	クラス (3 値)
ReutersCorn	記事分類	文字列	クラス (2 値)
supermarket	スーパーの購買記録	カテゴリ	なし
weather.nominal	ゴルフをする条件	カテゴリ	クラス (2 値)
weather.numeric	ゴルフをする条件	混合	クラス (2 値)

# 起動

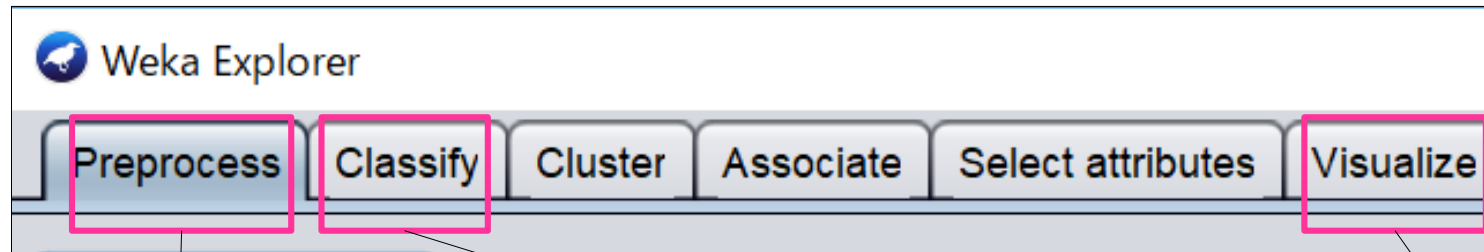
- アプリケーションの選択



- **Explorer** : データの読み込みから、特徴選択・学習・評価を試行錯誤的に行うのに適した操作を提供
- **Workbench** : すべてのアプリケーションをまとめた GUI (カスタマイズ可能)

- **Experimenter** : ハイパーパラメータ等を変えて性能を比較実験
- **KnowledgeFlow** : 実験プロセスを GUI で組み立て
- **SimpleCLI** : コマンドラインインタフェース

# Explorer での操作



- 前処理

- データの読み込み
- 標準化
- 特徴選択
- 特徴の分析

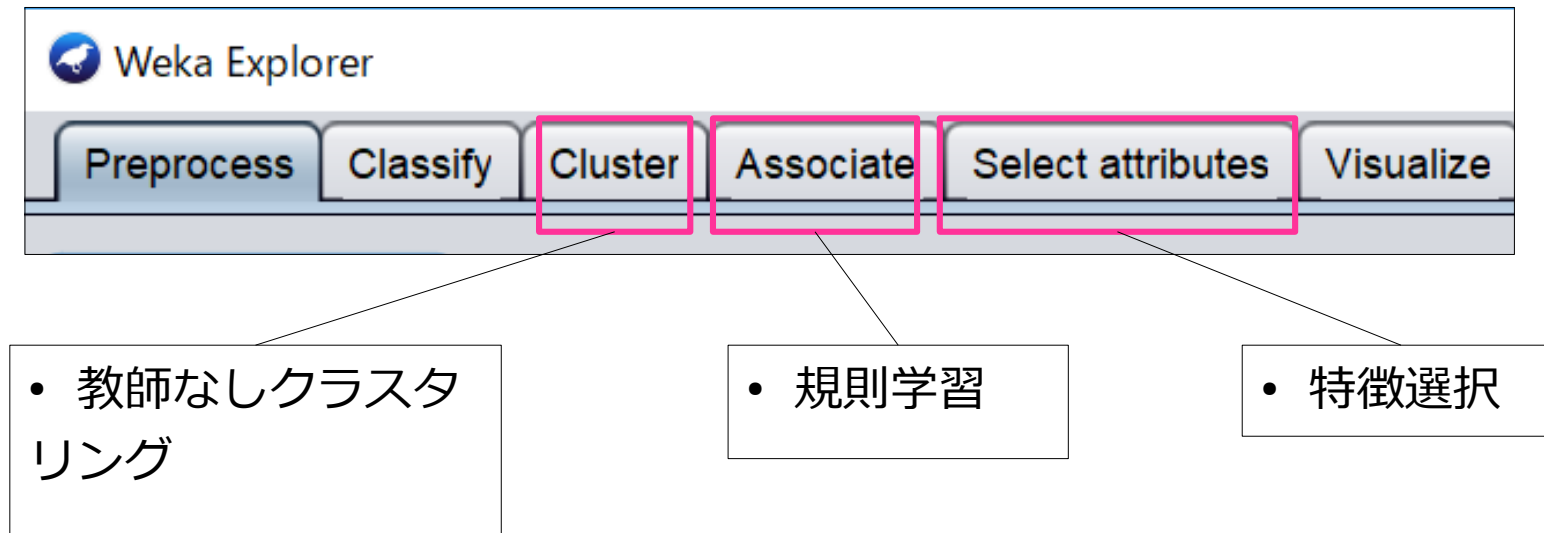
- 識別

- 100 以上の識別アルゴリズムの実装
- 学習の設定
- ハイパーパラメータの設定
- 学習結果の評価

- 可視化

- データの 2 次元プロット

# Explorer での操作



# 前処理 (Preprocess)

- 読み込み可能なデータ形式
  - ARFF (Attribute Relationship File Format) 形式
  - ヘッダ部とデータ部で構成
    - ヘッダ部
      - @relation : データ集合の名前 (ファイル名と同じでよい)
      - @attribute : 特徴の各次元の名前とデータの型を宣言
    - データ部
      - @data 以降に 1 行 1 件のデータを CSV 形式で記述
      - 各特徴・クラスラベルはカンマ区切り

# 前処理 (Preprocess)

- アヤメの分類データ (iris)



setosa



versicolor



virginica

```
% 1. Title: Iris Plants Database
@RELATION iris

@ATTRIBUTE sepallength    REAL
@ATTRIBUTE sepalwidth     REAL
@ATTRIBUTE petallength    REAL
@ATTRIBUTE petalwidth     REAL
@ATTRIBUTE class {Iris-setosa,Iris-versicolor,Iris-virginica}

@DATA
5.1, 3.5, 1.4, 0.2, Iris-setosa
4.9, 3.0, 1.4, 0.2, Iris-setosa
...
7.0, 3.2, 4.7, 1.4, Iris-versicolor
6.4, 3.2, 4.5, 1.5, Iris-versicolor
...
6.3, 3.3, 6.0, 2.5, Iris-virginica
5.8, 2.7, 5.1, 1.9, Iris-virginica
...
```

データセット名

特徴名と型

萼・花びらの  
長さ・幅

アヤメの  
種類

これ以降、1行に1事例  
(ExcelのCSV形式と同じ)



# 前処理 (Preprocess)

- 特徴抽出後のデータを読み込む
- いくつかの特徴の操作（フィルタの適用）が可能

The screenshot shows the Weka Explorer application window. The 'Preprocess' tab is selected. The 'Open file...' button is highlighted with a pink box and labeled '読み込み' (Load). The 'Filter' section has a 'Choose' button highlighted with a pink box and labeled 'フィルタ' (Filter). The 'Current relation' section shows 'Relation: iris' and 'Instances: 150'. The 'Attributes' list shows 'sepalength' selected. The 'Selected attribute' section shows statistics for 'sepalength'. The 'Class' dropdown is set to 'class (Nom)'. A histogram is displayed at the bottom right.

読み込み

フィルタ

データの全体像

分析対象の特徴（属性）の選択

データの表示

選択された特徴の分析

Statistic	Value
Minimum	4.3
Maximum	7.9
Mean	5.843
StdDev	0.828

Class: class (Nom) Visualize All

16 30 34 28 25 10 7

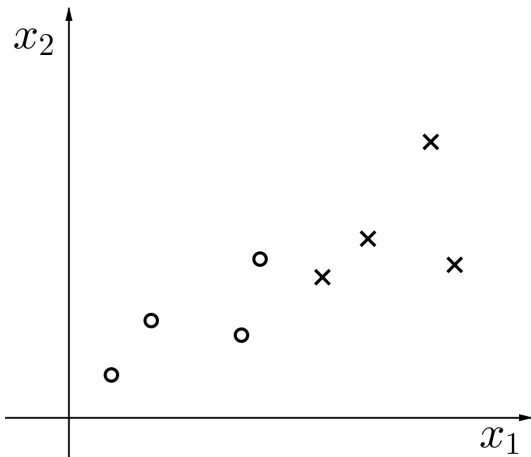
4.3 6.1 7.9

## 2.1.2 前処理

- 分析
  - 主成分分析（次元削減）
    - データの散らばりをできるだけ保存する低次元空間へ写像
    - データの可視化に有効
- データの標準化
  - すべての次元を平均 0、分散 1 にそろえる
  - 各次元に対して平均値を引き、標準偏差で割る

$$x'_i = \frac{x_i - m_i}{\sigma_i} \quad m_i, \sigma_i : \text{軸 } i \text{ の平均、標準偏差}$$

# 主成分分析の考え方



共分散行列 $\Sigma$ の計算

$\bar{x}_1, \bar{x}_2$  : 平均値、 $N$  : データ数

対角成分は分散、  
非対角成分は相関を表す

$$\Sigma = \frac{1}{N} \begin{pmatrix} \sum (x_1 - \bar{x}_1)^2 & \sum (x_1 - \bar{x}_1)(x_2 - \bar{x}_2) \\ \sum (x_1 - \bar{x}_1)(x_2 - \bar{x}_2) & \sum (x_2 - \bar{x}_2)^2 \end{pmatrix}$$

$\Sigma$ は

半正定値(→固有値がすべて0以上の実数)

対称行列(→固有ベクトルが実数かつ直交)

であるので、以下のように分解できる

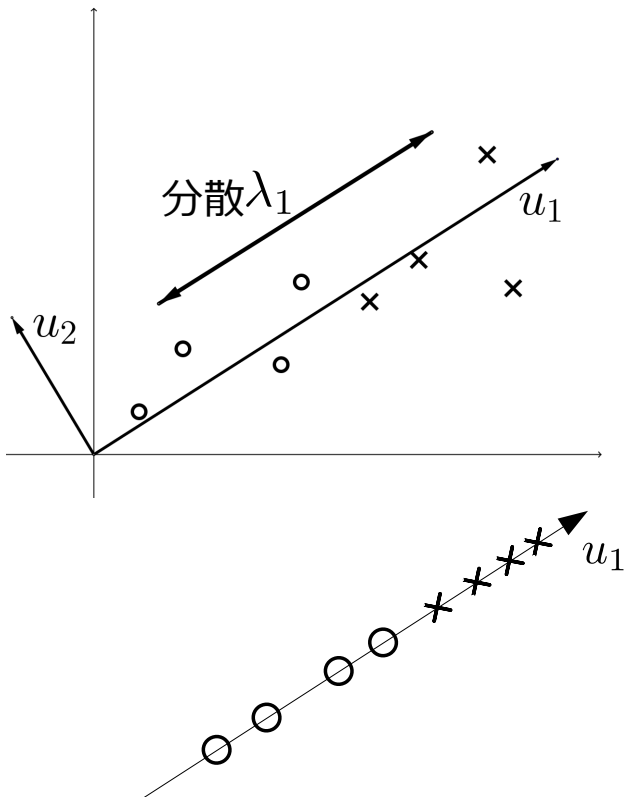
$$\Sigma' = U^T \Sigma U = \begin{pmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{pmatrix}$$

$\lambda$ は固有値の大きい順、 $U$ は対応する  
固有ベクトル $U_1, U_2$ を並べたもの

$\lambda_1$ に対応する固有ベクトル $U_1$ で  
2次元データを1次元に射影

$$u_1 = U_1^T x$$

$$\text{寄与率} = \frac{\lambda_1}{\lambda_1 + \lambda_2}$$



# 前処理 (Preprocess)

- 標準化

Weka Explorer

Preprocess | Classify | Cluster | Associate | Select attributes | Visualize | Collective | Forecast

Open file... | Open URL... | Open DB... | Generate... | Undo | Edit... | Save...

適用

選択

Filter

Choose Standardize Apply Stop

Current relation

Relation: iris-weka.filters.unsupervised.attribu... Attributes: 5  
Instances: 150 Sum of weights: 150

Attributes

All | None | Invert | Pattern

No.	Name
1	<input checked="" type="checkbox"/> sepallength
2	<input type="checkbox"/> sepalwidth
3	<input type="checkbox"/> petallength
4	<input type="checkbox"/> petalwidth
5	<input type="checkbox"/> class

Remove

Status

OK

Selected attribute

Name: sepallength Type: Numeric  
Missing: 0 (0%) Distinct: 35 Unique: 9 (6%)

Statistic	Value
Minimum	-1.864
Maximum	2.484
Mean	-0
StdDev	1

平均 0  
標準偏差 1

Class: class (Nom) Visualize All

16 30 34 28 25 10 7

-1.86 0.31 2.48

Log x 0

# 前処理 (Preprocess)

- 主成分分析
  - iris データ (4 次元特徴) を 2 次元に

Weka Explorer

Preprocess | Classify | Cluster | Associate | Select attributes | Visualize | Collective | Forecast

Open file... | Open URL... | Open DB... | Generate... | Undo | Edit... | Save...

Filter: Choose **PrincipalComponents -R 0.95 -A 5 -M -1** Apply Stop

**Current relation**  
Relation: iris-weka.filters.unsupervised.attribu... Attributes: 3  
Instances: 150 Sum of weights: 150

**Attributes**  
All None Invert Pattern

No.	Name
1	<input checked="" type="checkbox"/> -0.581petallength-0.566petalwidth-0.522sepallength+0.2...
2	<input type="checkbox"/> -0.926sepalwidth-0.372sepallength-0.065petalwidth-0.02...
3	<input type="checkbox"/> class

Remove

**Selected attribute**  
Name: -0.581petallength-0.566petalwidth-0.522sepalle... Type: ...  
Missing: 0 (0%) Distinct: 147 Unique: ...

Statistic	Value
Minimum	-3.298
Maximum	2.765
Mean	0
StdDev	1.706

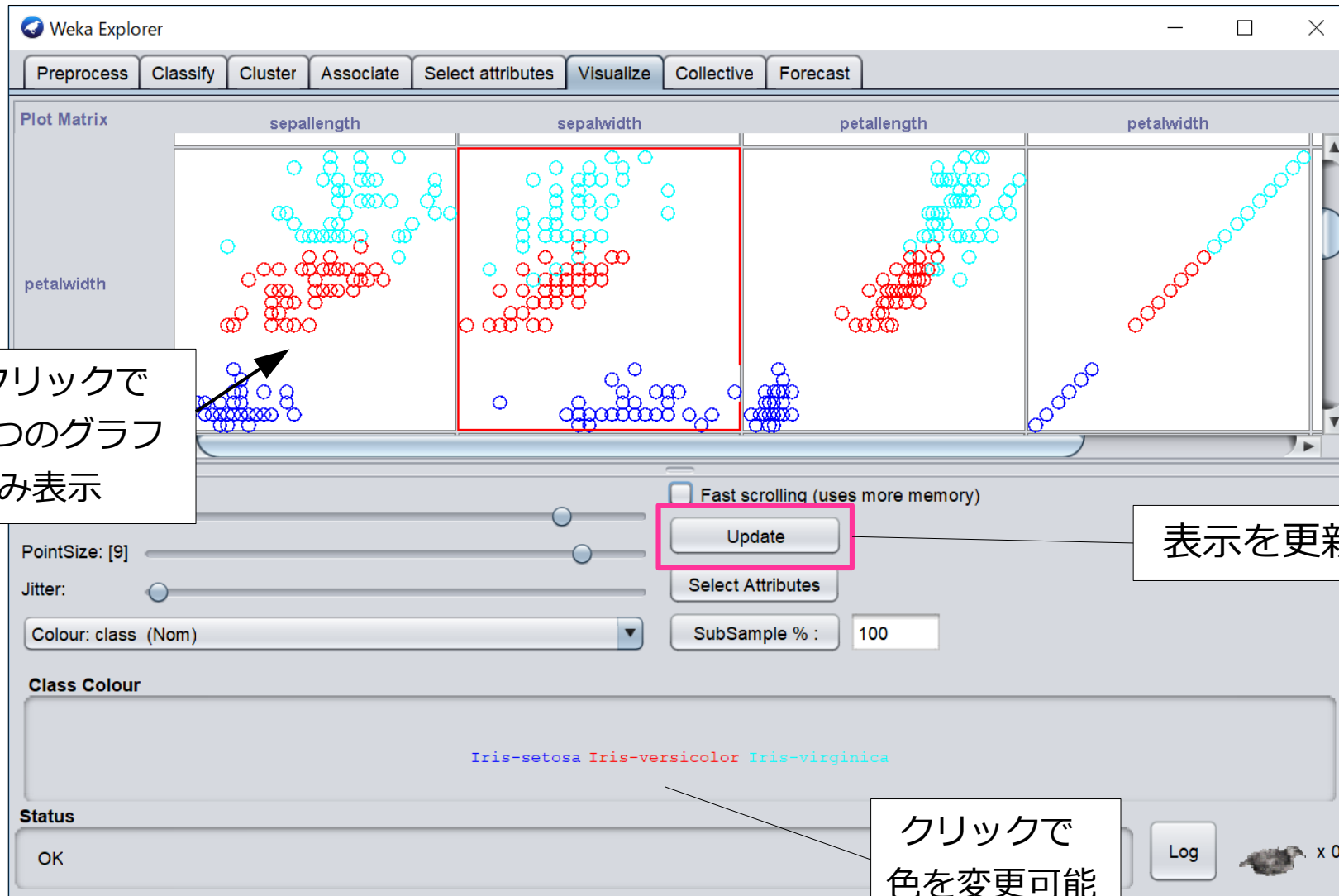
Class: class (Nom) Visualize All

10 51 36 3 50

-3.3 -0.27 2.76

**Status**  
OK Log x 0

# データのプロット (Visualize)



# データのプロット (Visualize)

- 1つのグラフのみ表示

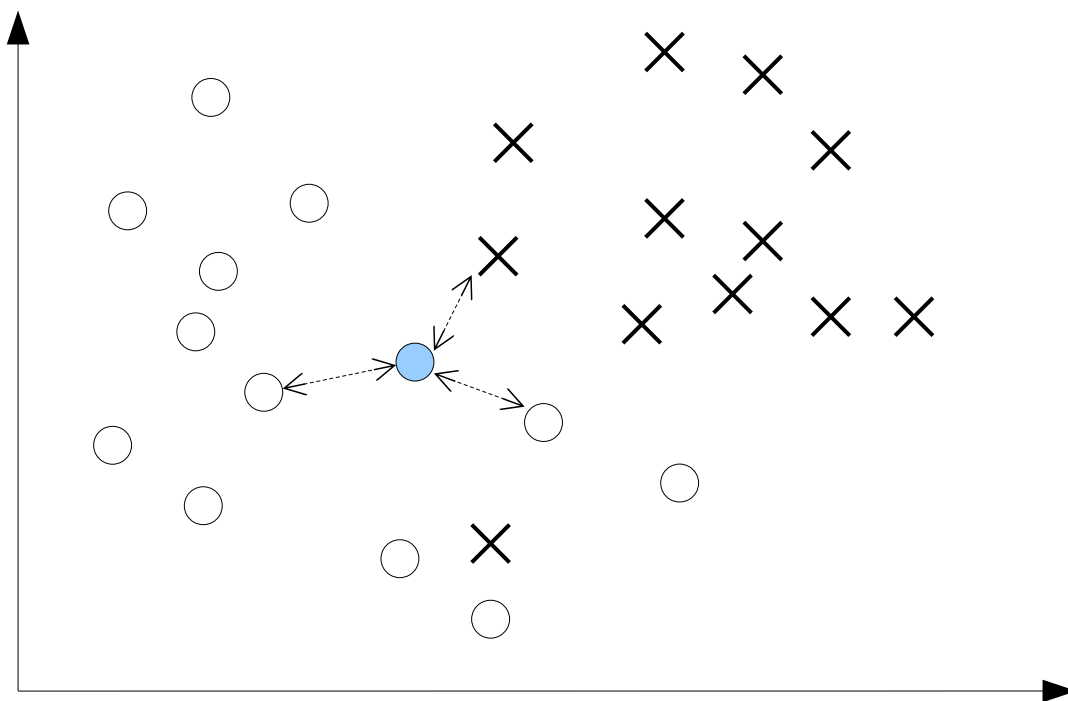


x 軸、y 軸、色の  
基準が選べる

クリックで  
色を変更可能

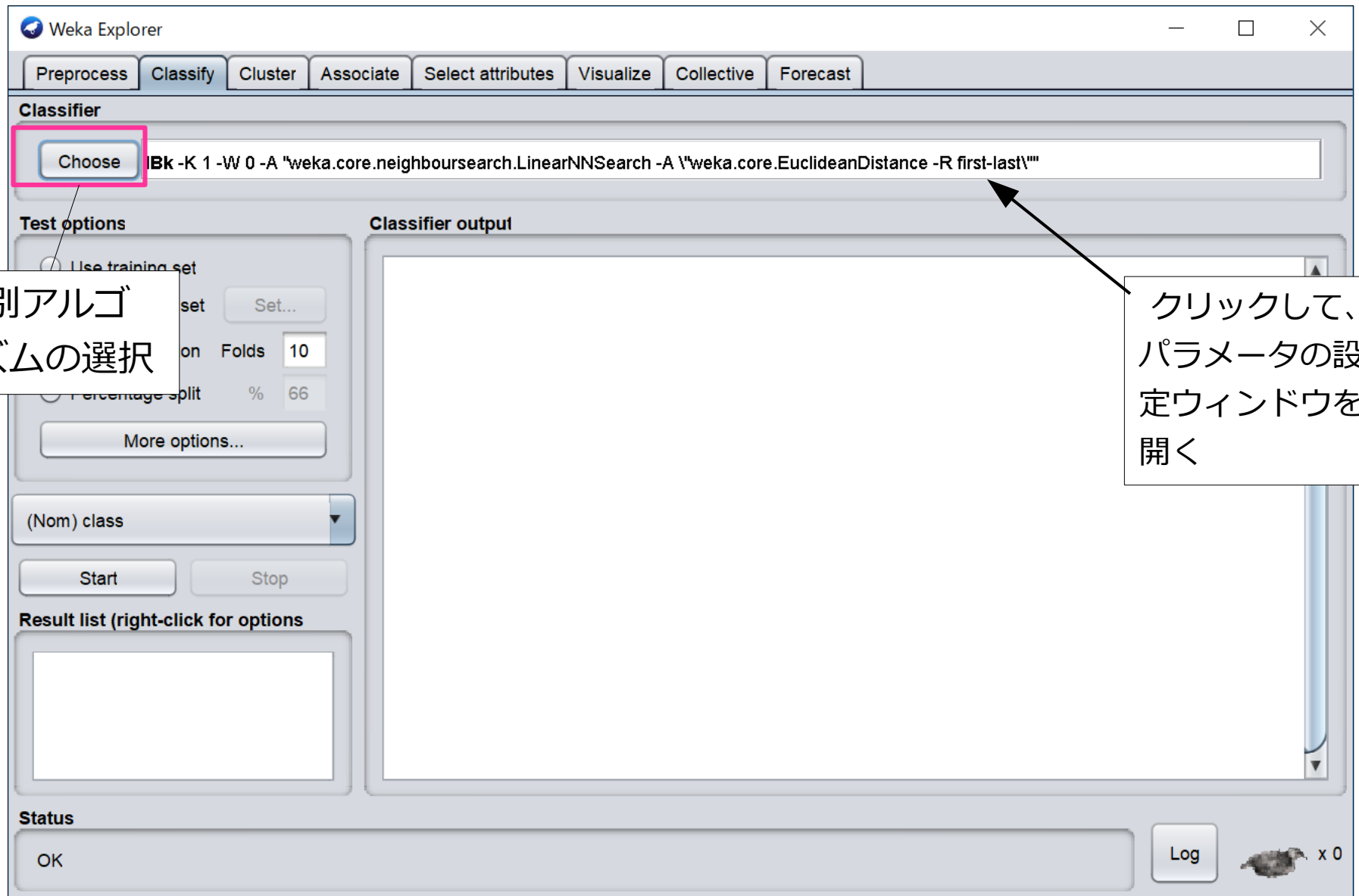
## 2.1.4 学習 k-NN 法

- NN (Nearest Neighbor: 最近傍) 法
  - 識別したいデータと最も近い事例を求め、その事例の属するクラスを識別結果とする (1-NN 法)
  - k 番目までの近い事例を求め、多数決を採るのが k-NN 法





# 識別器の学習 (Classify)

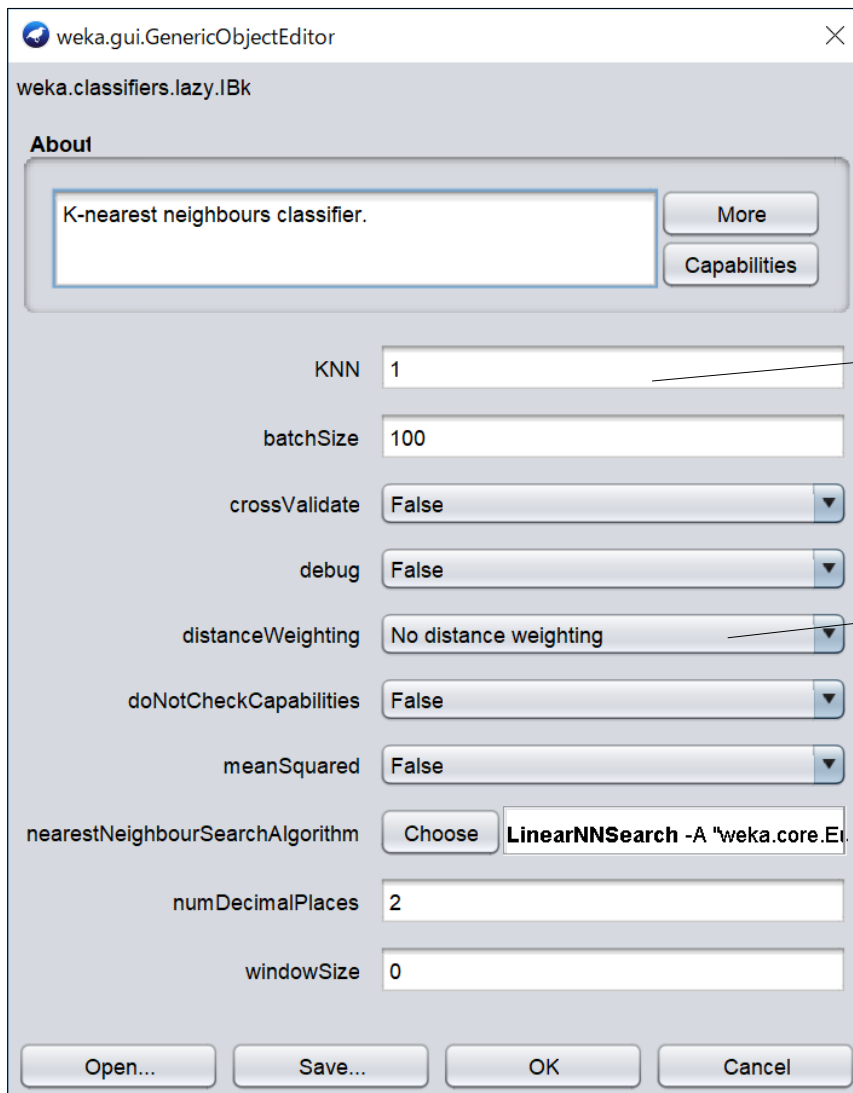


識別アルゴリズムの選択

クリックして、パラメータの設定ウィンドウを開く

# 識別器の学習 (Classify)

- IBk (k-NN 法) のパラメータ



The screenshot shows the 'weka.gui.GenericObjectEditor' window for the 'weka.classifiers.lazy.IBk' classifier. The 'About' section describes it as a 'K-nearest neighbours classifier.' with 'More' and 'Capabilities' buttons. The main parameter list includes:

- KNN: 1
- batchSize: 100
- crossValidate: False
- debug: False
- distanceWeighting: No distance weighting
- doNotCheckCapabilities: False
- meanSquared: False
- nearestNeighbourSearchAlgorithm: Choose LinearNNSearch -A "weka.core.Et...
- numDecimalPlaces: 2
- windowSize: 0

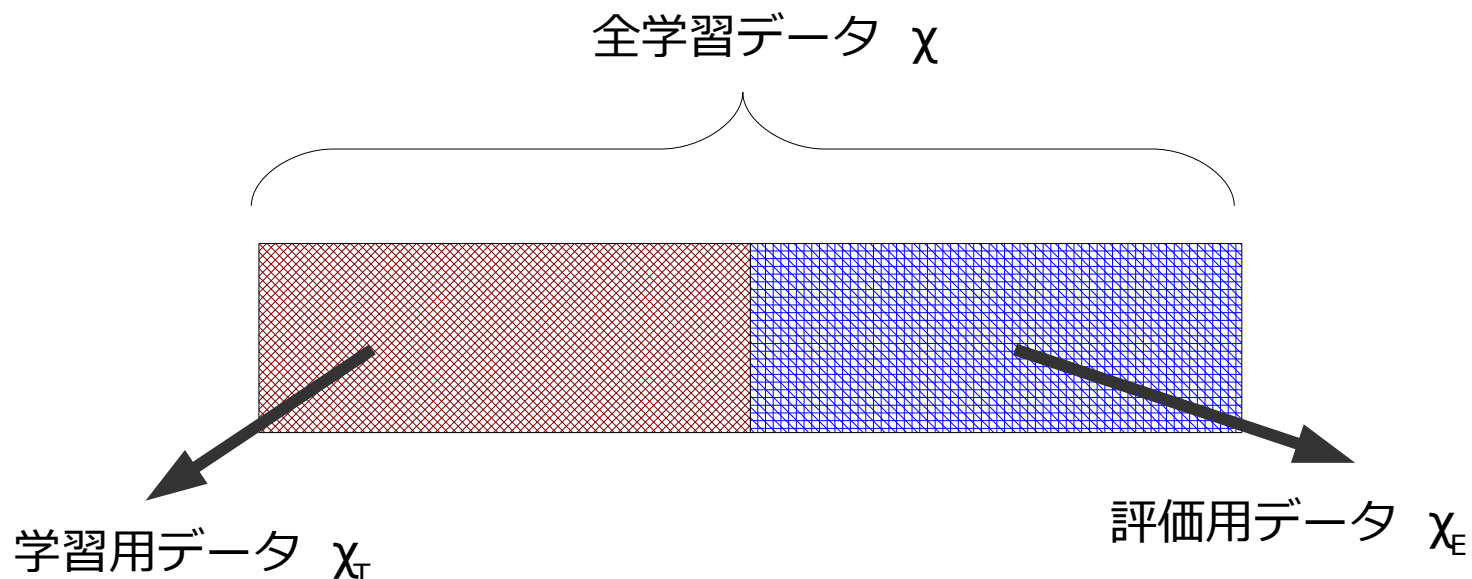
Buttons at the bottom: Open..., Save..., OK, Cancel.

k

距離による重み付けの有無

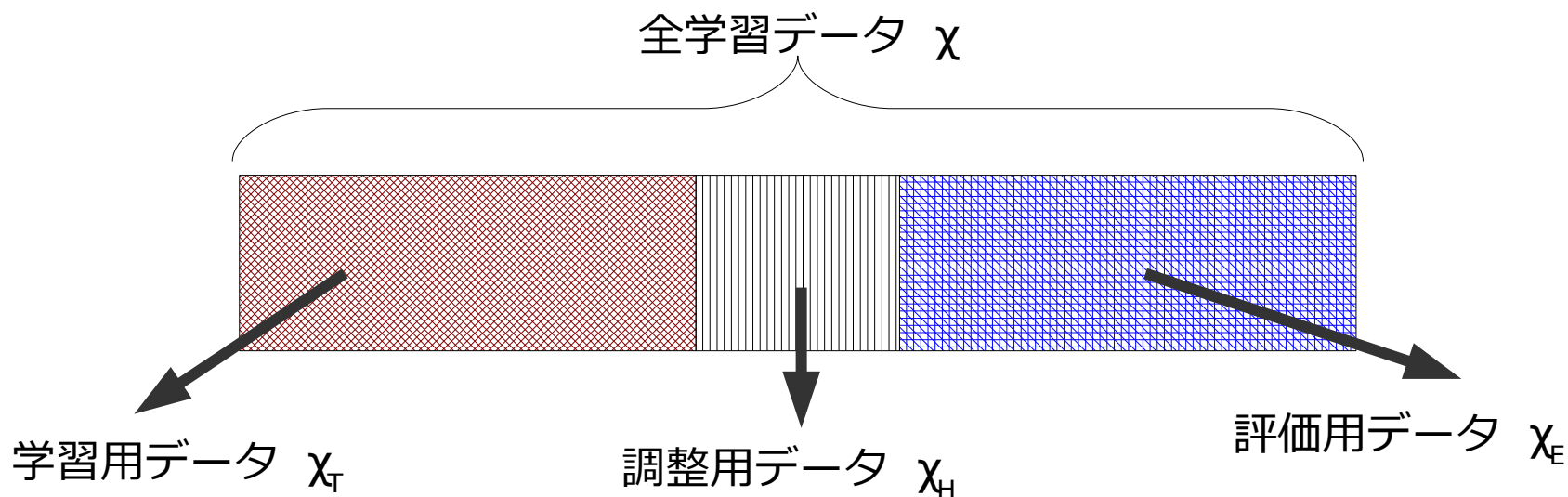
## 2.1.3 評価基準の設定

- 分割学習法（１）
  - 全学習データ  $\chi$  を学習用データ集合  $\chi_T$  と評価用データ集合  $\chi_E$  に分割する
  - $\chi_T$  を用いて識別機を設計し、 $\chi_E$  を用いて誤識別率を推定する



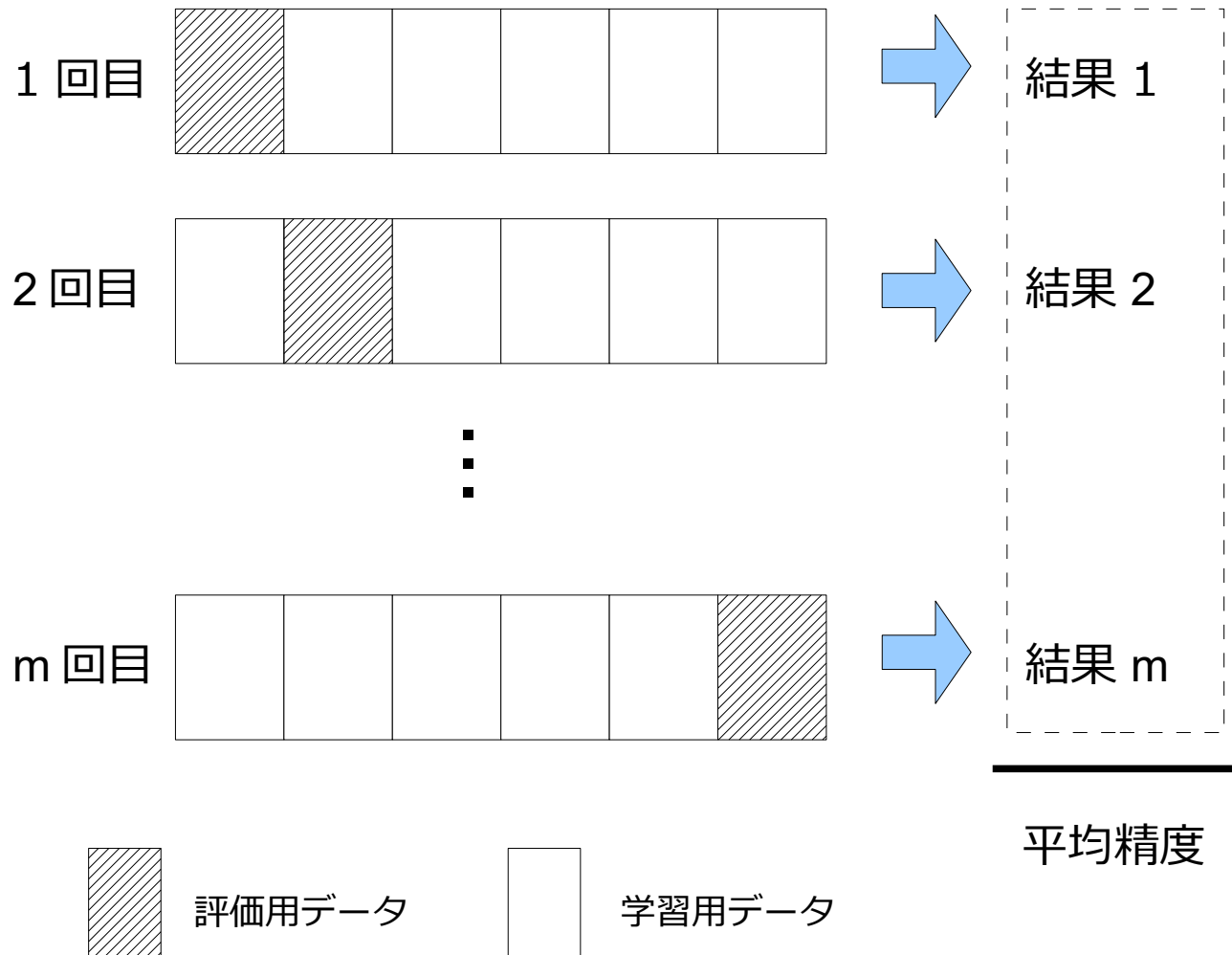
## 2.1.3 評価基準の設定

- 分割学習法（2）
  - 全学習データ  $\chi$  を学習用データ集合  $\chi_T$ 、調整用データ集合  $\chi_H$ 、評価用データ集合  $\chi_E$  に分割する
  - $\chi_T$  を用いて識別機を設計、 $\chi_H$  を用いてハイパーパラメータを調整、 $\chi_E$  を用いて誤識別率を推定する



## 2.1.3 評価基準の設定

- 交差確認法



# 識別器の学習 (Classify)

- 評価法の設定

学習データを使  
って評価

分割学習法

交差確認法

**Test options**

☐ Use training set

☐ Supplied test set Set...

☒ Cross-validation Folds

☐ Percentage split %

More options...

データ分割数

# 識別器の学習 (Classify)

- 学習結果の見方

```
=== Summary ===
```

Correctly Classified Instances	143
Incorrectly Classified Instances	7
Kappa statistic	0.93
Mean absolute error	0.0399
Root mean squared error	0.1747
Relative absolute error	8.9763 %
Root relative squared error	37.0695 %
Total Number of Instances	150

...

```
=== Confusion Matrix ===
```

a	b	c	<-- classified as
50	0	0	a = Iris-setosa
0	47	3	b = Iris-versicolor
0	4	46	c = Iris-virginica

正解率

95.3333 %  
4.6667 %

## 混同行列

縦方向が正解、横方向が予測  
対角成分が正解数

# 結果の可視化

- 混同行列

	予測+	予測-
正解+	true positive (TP)	false negative (FN)
正解-	false positive (FP)	true negative (TN)

- 正解率  $Accuracy = \frac{TP + TN}{TP + FN + FP + TN}$

- 精度  $Precision = \frac{TP}{TP + FP}$

- 再現率  $Recall = \frac{TP}{TP + FN}$

- F 値  $F-measure = 2 \times \frac{Precision \times Recall}{Precision + Recall}$

正解の割合  
クラスの出現率に  
偏りがある場合は不適

正例の判定が  
正しい割合

正しく判定された  
正例の割合

↑  
トレードオフ  
↓

精度と再現率の  
調和平均

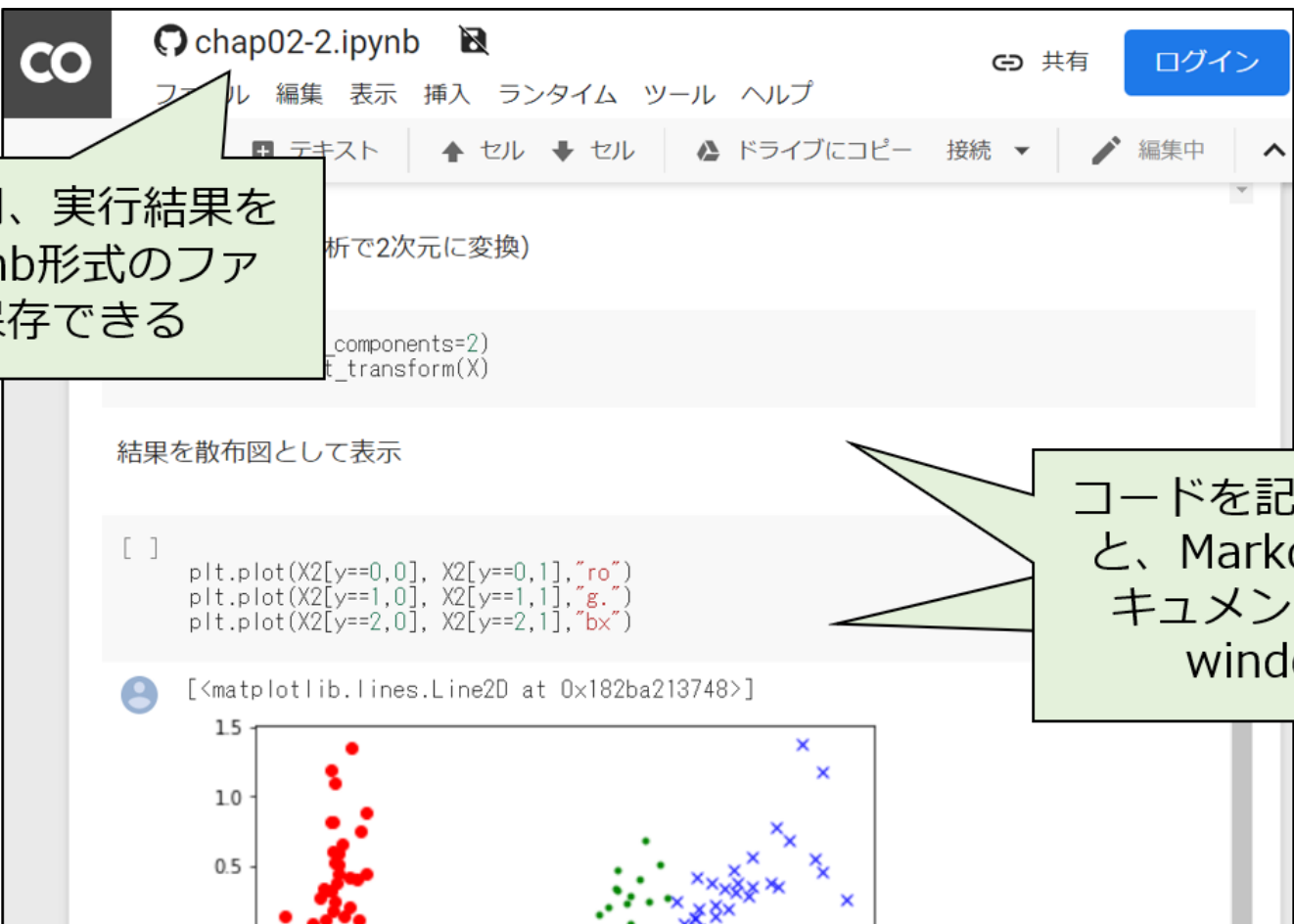


## 2.2 Python による機械学習

- プログラミング言語 Python
  - オブジェクト指向スクリプト言語
  - 特徴
    - 動的型付け・インデントによるブロック化・デフォルト引数を用いた関数呼び出し
- 機械学習に関連する多くのライブラリが充実
  - matplotlib : グラフ描画
  - pandas : データの読み込み・解析を支援
  - scikit-learn : 多くの機械学習アルゴリズム
  - tensorflow : 深層学習

## 2.2 Python による機械学習

- Jupyter notebook
  - ブラウザで実行できる Python 開発環境



The screenshot displays a Jupyter Notebook window titled 'chap02-2.ipynb'. The interface includes a top bar with a '共有' (Share) button and a 'ログイン' (Login) button. Below the bar, there are tabs for 'ファイル' (File), '編集' (Edit), '表示' (View), '挿入' (Insert), 'ランタイム' (Runtime), 'ツール' (Tools), and 'ヘルプ' (Help). The main area shows a code cell with the following Python code:

```
[ ]  
plt.plot(X2[y==0,0], X2[y==0,1], "ro")  
plt.plot(X2[y==1,0], X2[y==1,1], "g.")  
plt.plot(X2[y==2,0], X2[y==2,1], "bx")
```

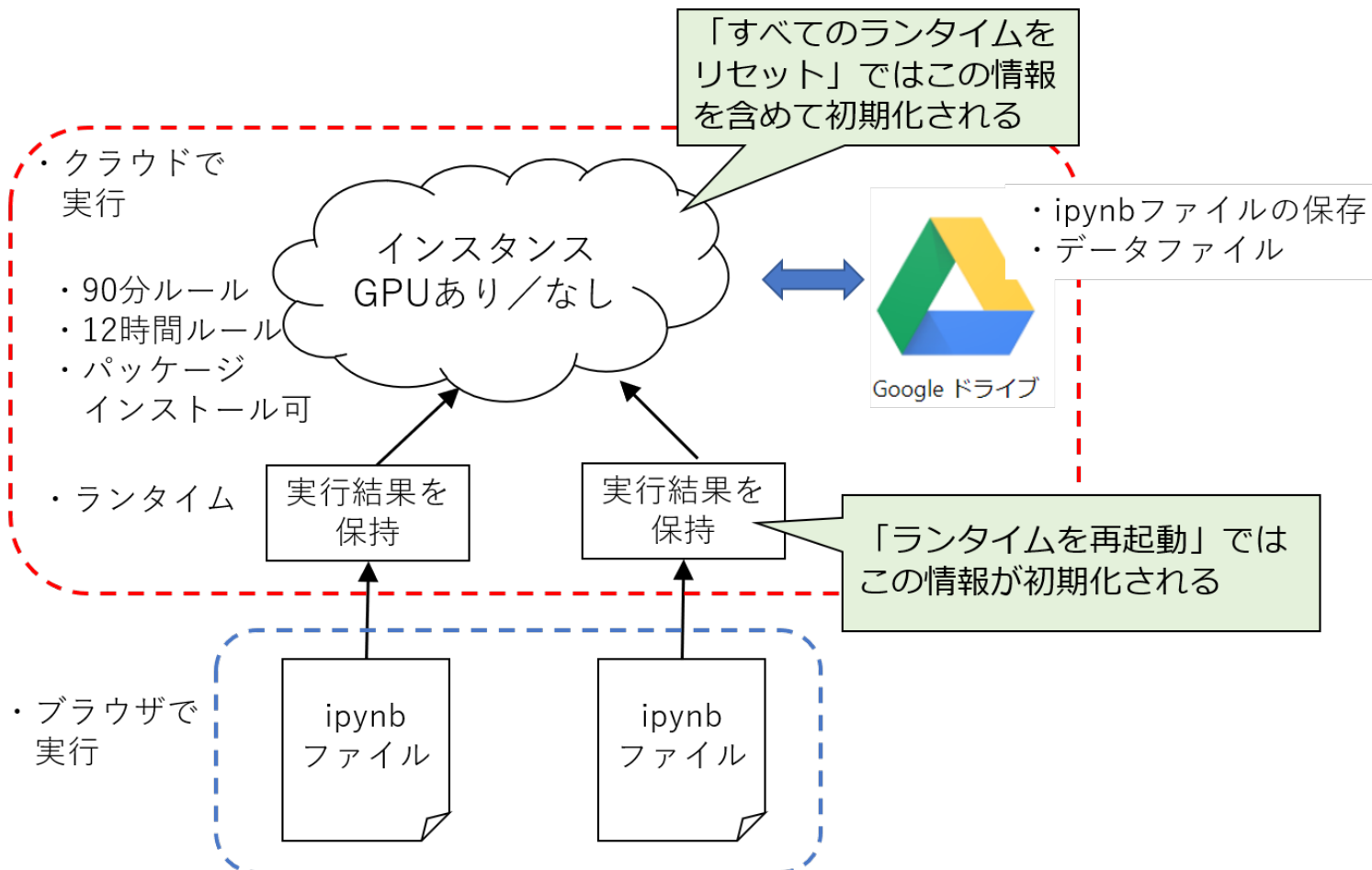
Below the code cell, the output is displayed as a scatter plot. The plot shows three distinct clusters of data points: red circles ('ro') on the left, green dots ('g.') in the middle, and blue crosses ('bx') on the right. The x-axis ranges from 0 to 1.5, and the y-axis ranges from 0 to 1.5.

Two callout boxes provide additional information:

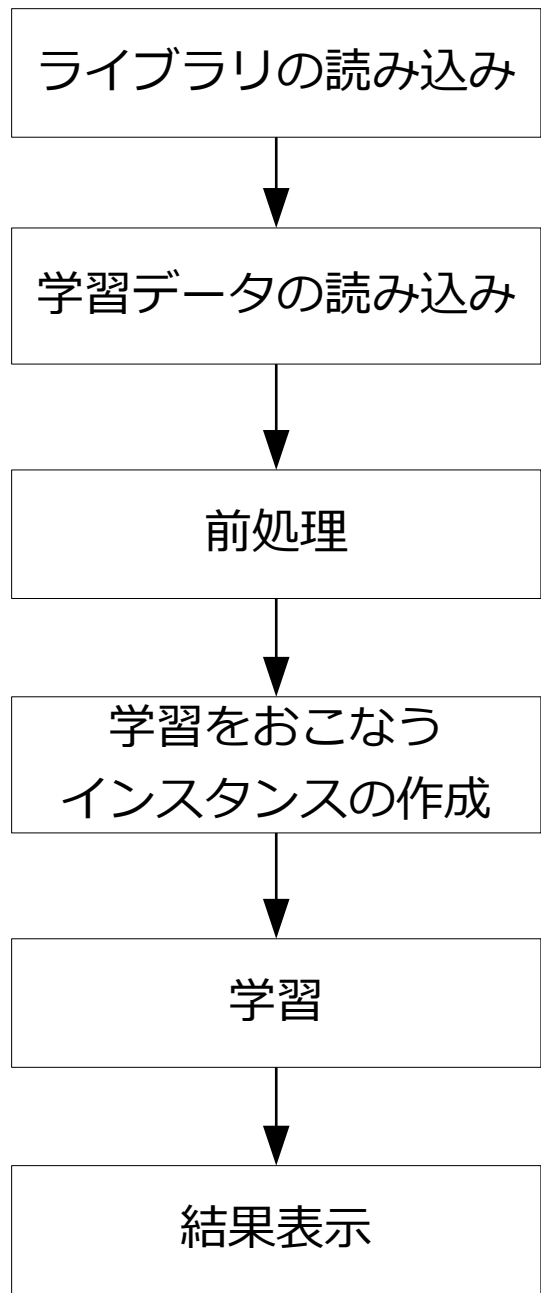
- Left Callout:** コード、説明、実行結果をまとめてipynb形式のファイルとして保存できる (Code, explanation, and execution results can be saved together as an ipynb format file).
- Right Callout:** コードを記述するwindowと、Markdown形式でドキュメントを記述するwindowがある (There are windows for describing code and windows for describing documents in Markdown format).

## 2.2 Python による機械学習

- Google Colaboratory
  - クラウドで実行される Jupyter notebook 環境
  - 機械学習関係のライブラリはインストール済み



## 2.2 Python による機械学習



- 組み込みデータは datasets パッケージを利用
- 外部データは pandas の read\_csv 等を利用

- 標準化 : scale
- 主成分分析 : PCA

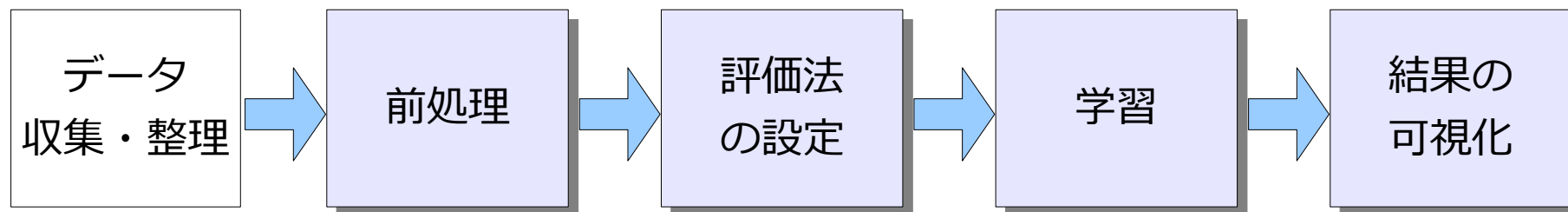
- 学習パラメータを与えてインスタンスを作成

- fit に学習データを与えて学習
- 分割学習法では predict で予測を得る
- 交差確認法では cross\_val\_score を実行

- 分割学習法では confusion\_matrix で混同行列を求める
- 交差確認法では、結果から平均・標準偏差などを求める

# まとめ

- 機械学習の基本的なプロセス



- データの前処理で有効な技法
  - 標準化、主成分分析
- 代表的なベースライン手法
  - k-NN 法（1-NN は最も近い事例を答とする）
- 結果の評価
  - 分割学習法、交差確認法
  - 正解率、精度、再現率、F 値