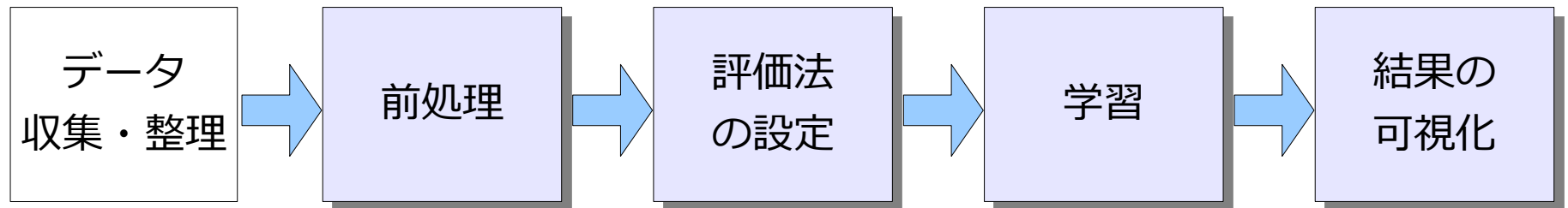



2. 機械学習の基本的な手順



 : 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
 - 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
 - この解説では開発版である ver. 3.9.3 を使用

Weka に関する資料

- 開発者による機械学習一般の解説書
 - Ian H. Witten et.al.: Data Mining: Practical Machine Learning Tools and Techniques, Fourth Edition, Morgan Kaufmann, 2016.
- web 教材
 - Waikato 大学 Mooc: Data Mining with Weka
 - <http://www.cs.waikato.ac.nz/ml/weka/mooc/dataminingwithweka/>
 - ビデオやスライドを公開

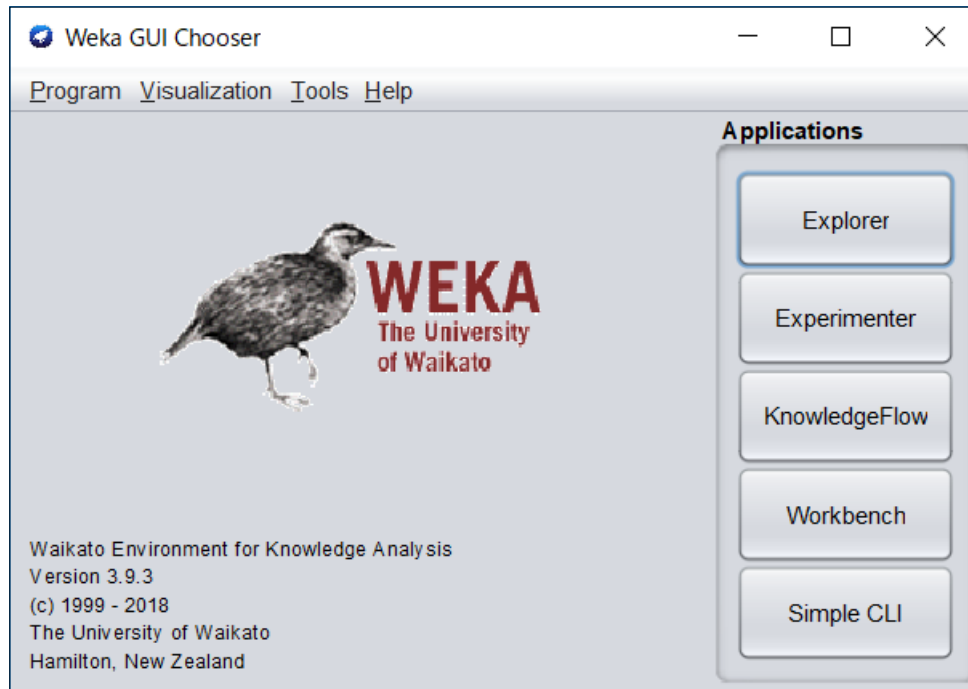
勉強のためのデータセット

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

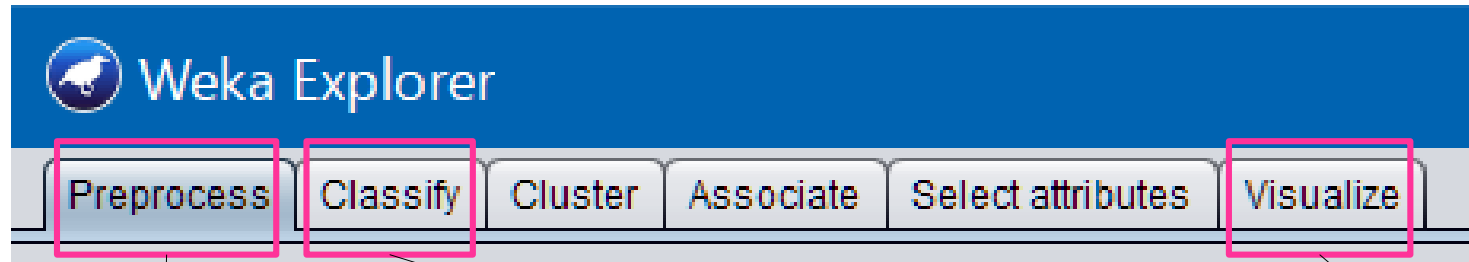
- アプリケーションの選択



- **Explorer** アプリケーション
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

- **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
- **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)
- **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

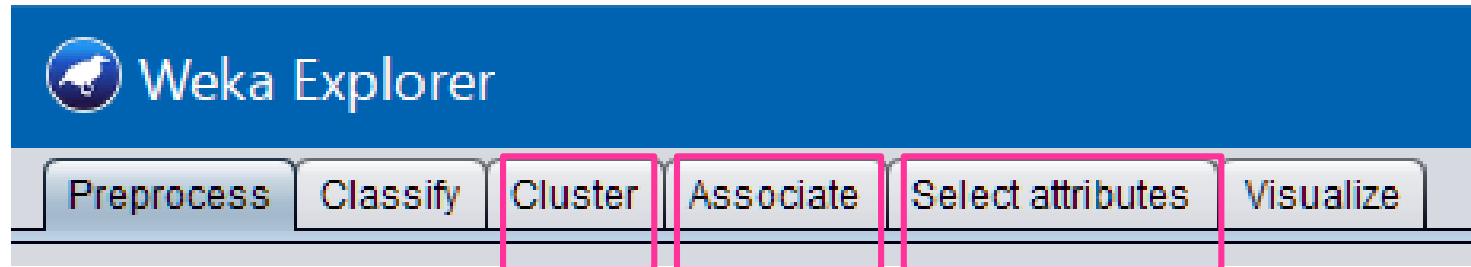
- 識別

- 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



• 教師なしクラスタリング

• 規則学習

• 特徴選択

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 'Edit...' button is highlighted with a pink box and labeled 'データの表示' (Data display). 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: ex7-1' and 'Instances: 15'. The 'Attributes' section has a table with columns 'No.' and 'Name'. The 'Selected attribute' section shows 'Name: f1' and 'Type: Numeric'. The 'Class' is set to 'vowel (Nom)'. A horizontal bar chart is displayed at the bottom right, showing the distribution of the 'vowel' class across the range of 'f1' values.

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

No.	Name
1	f1
2	f2
3	vowel

Statistic	Value
Minimum	210
Maximum	800
Mean	426.667
StdDev	201.092

Class: vowel (Nom)

Visualize All

11

4

210 505 800

Status: OK

Log

x 0

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 件のデータを記述
 - 各特徴・クラスラベルはカンマ区切り

2.1.1 データ収集・整理

- Weka のデータ形式 ARFF フォーマット

```
% 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)

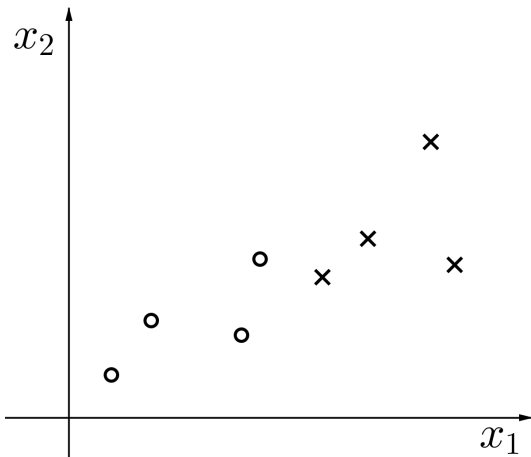
2.1.2 前処理

- 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

前処理 (Preprocess)

- フィルタの適用
 - 有用なフィルタのほとんどは
weka → filters → unsupervised → attribute
の下にある
- Standardize : 標準化 (平均 0, 分散 1)
 - 各次元に対して平均値を引き、標準偏差で割る
- Normalize : 値を [0,1] に変換
- PrincipalComponents : 主成分分析

主成分分析の考え方



共分散行列 Σ 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

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

Σ 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

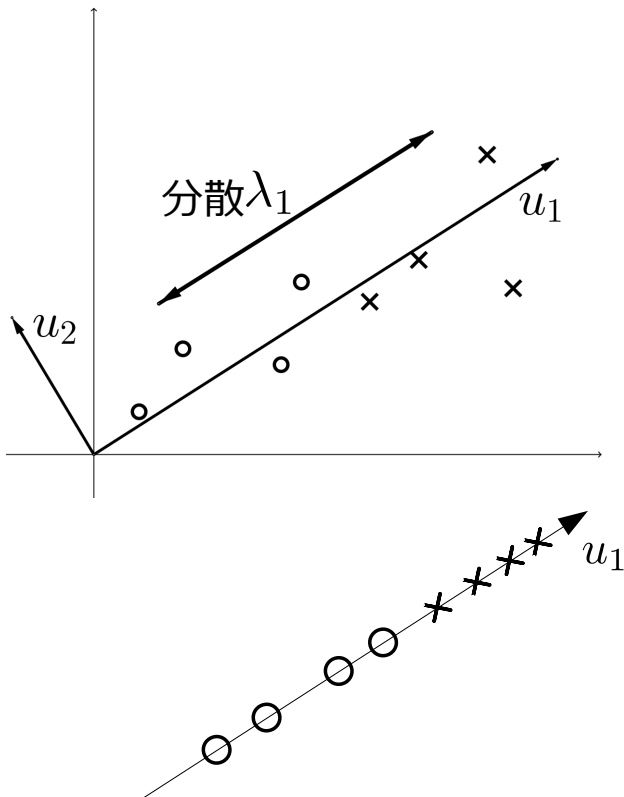
$$\Sigma' = U^T \Sigma U = \begin{pmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{pmatrix}$$

λ 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

λ_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

$$u_1 = U_1^T x$$

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



前处理 (Preprocess)

- 標準化

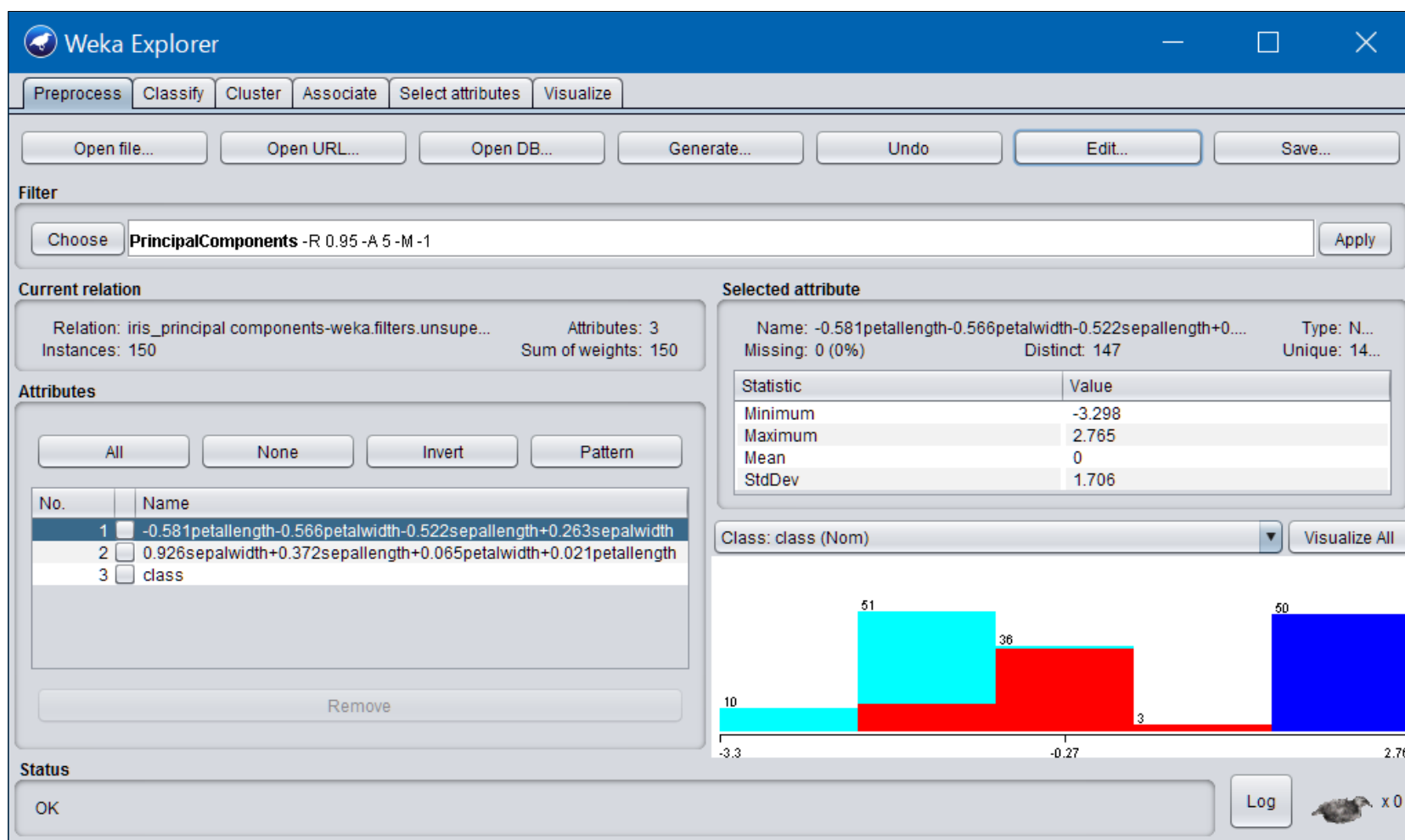
The screenshot shows the Weka Explorer interface with the 'Preprocess' tab selected. The 'Filter' dropdown is set to 'Standardize'. The 'Choose' button is highlighted with a pink box and labeled '選択' (Select). The 'Apply' button is also highlighted with a pink box and labeled '適用' (Apply). The 'Current relation' section shows 'Relation: ex7-1-weka.filters.unsuper...' and 'Instances: 15'. The 'Attributes' section shows a list of attributes: f1, f2, and class. The 'Selected attribute' section shows statistics for attribute f1: Name: f1, Missing: 0 (0%), Distinct: 11, Type: Numeric, Unique: 7 (47%). The statistics table is as follows:

Statistic	Value
Minimum	-1.077
Maximum	1.857
Mean	-0
StdDev	1

An arrow points from the 'Mean' value (-0) to a text box labeled '平均 0' (Average 0) and '標準偏差 1' (Standard deviation 1). The 'Class' dropdown is set to 'class (Nom)' and the 'Visualize All' button is visible. A horizontal bar chart is shown at the bottom, with values -1.08, 0.39, and 1.86 on the x-axis. The status bar at the bottom shows 'OK' and a 'Log' button.

前処理 (Preprocess)

- 主成分分析
 - iris データ (4 次元特徴) を 2 次元に



補足 – Select Attributes での主成分分析

Weka Explorer

Preprocess | Classify | Cluster | Associate | **Select attributes** | Visualize

Attribute Evaluator
Choose **PrincipalComponents** -R 0.95 -A 5

Search Method
Choose **Ranker** -T -1.7976931348623157E308 -N -1

Attribute Selection Mode
☒ Use full training set
☐ Cross-validation Folds: 10 Seed: 1

(Nom) class

Start Stop

Result list (right-click for options)
11:40:05 - Ranker + PrincipalCompon...

Attribute selection output

eigenvalue	proportion	cumulative	
2.91082	0.7277	0.7277	-0.581petallength-0.566petalwidth
0.92122	0.23031	0.95801	0.926sepalwidth+0.372sepalwidth

Eigenvectors

V1	V2	
-0.5224	0.3723	sepalwidth
0.2634	0.9256	sepalwidth
-0.5813	0.0211	petallength
-0.5656	0.0654	petalwidth

Ranked attributes:

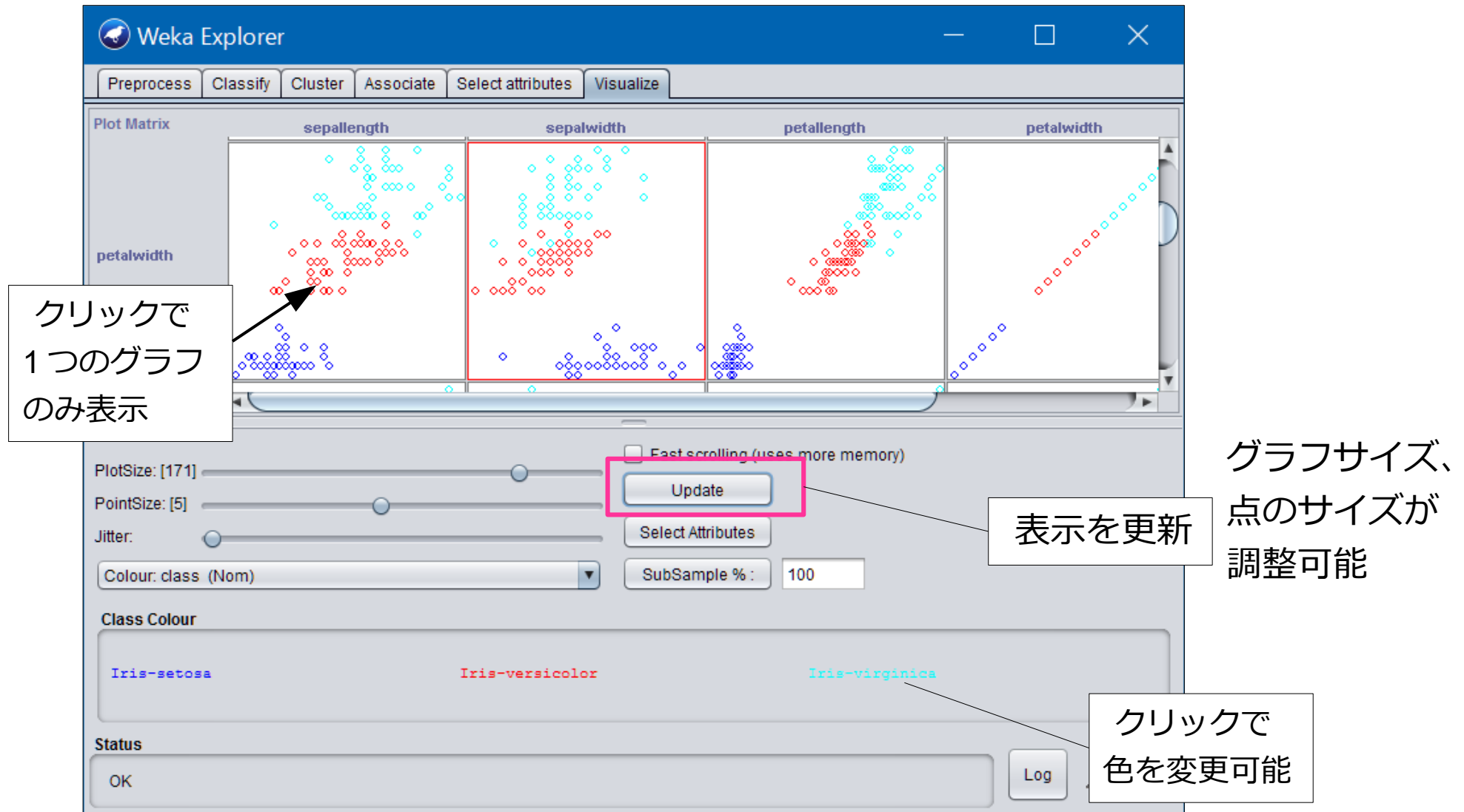
Rank	Attribute
1	-0.581petallength-0.566petalwidth-0.522sepalwidth+0.263sepalwidth
2	0.926sepalwidth+0.372sepalwidth+0.065petalwidth+0.021petallength

Status
OK Log x 0

Annotations:

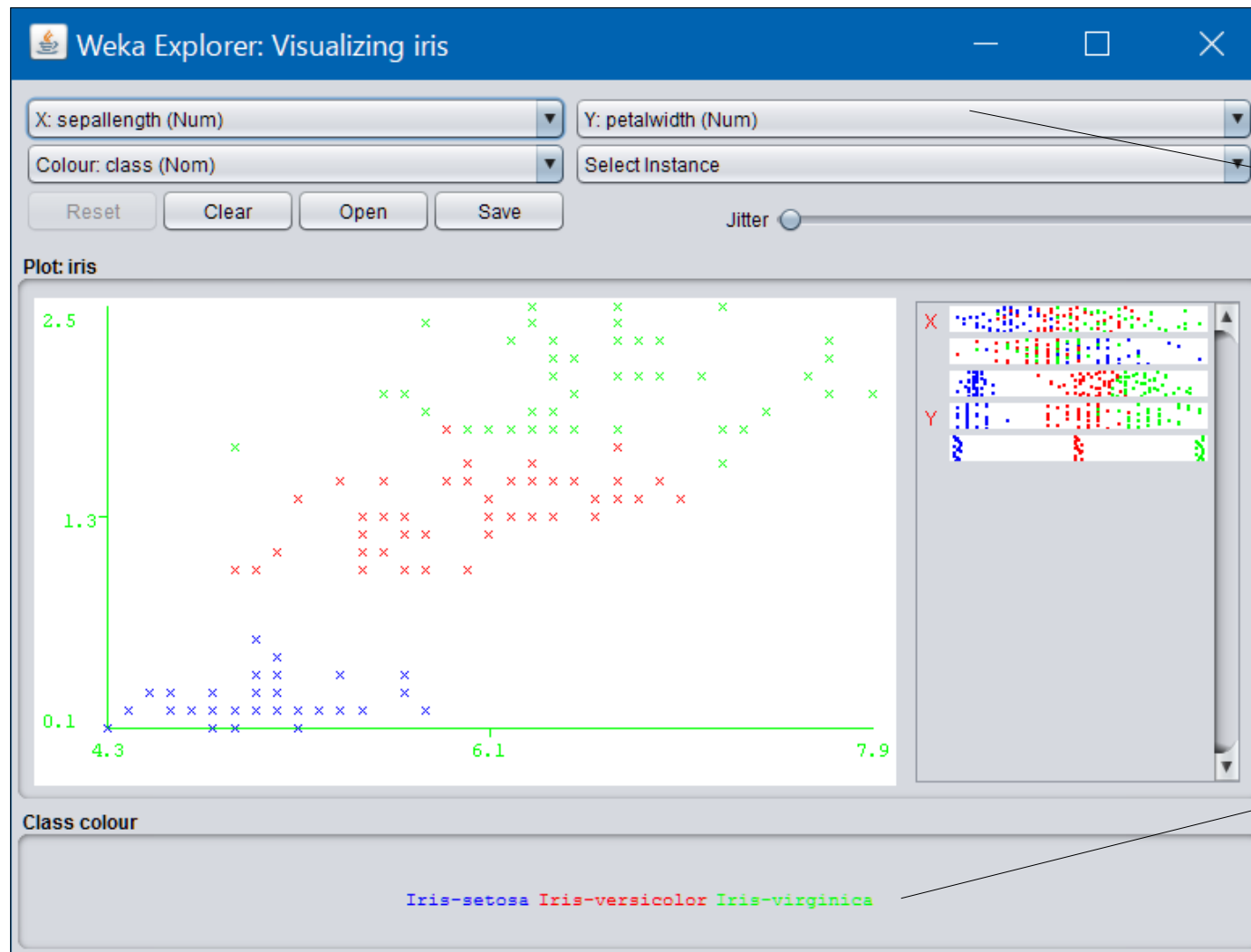
- 累積寄与率 (Cumulative Contribution Rate) points to the cumulative value 0.95801.
- 1次元目 (1st Dimension) points to the first ranked attribute.
- 2次元目 (2nd Dimension) points to the second ranked attribute.

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

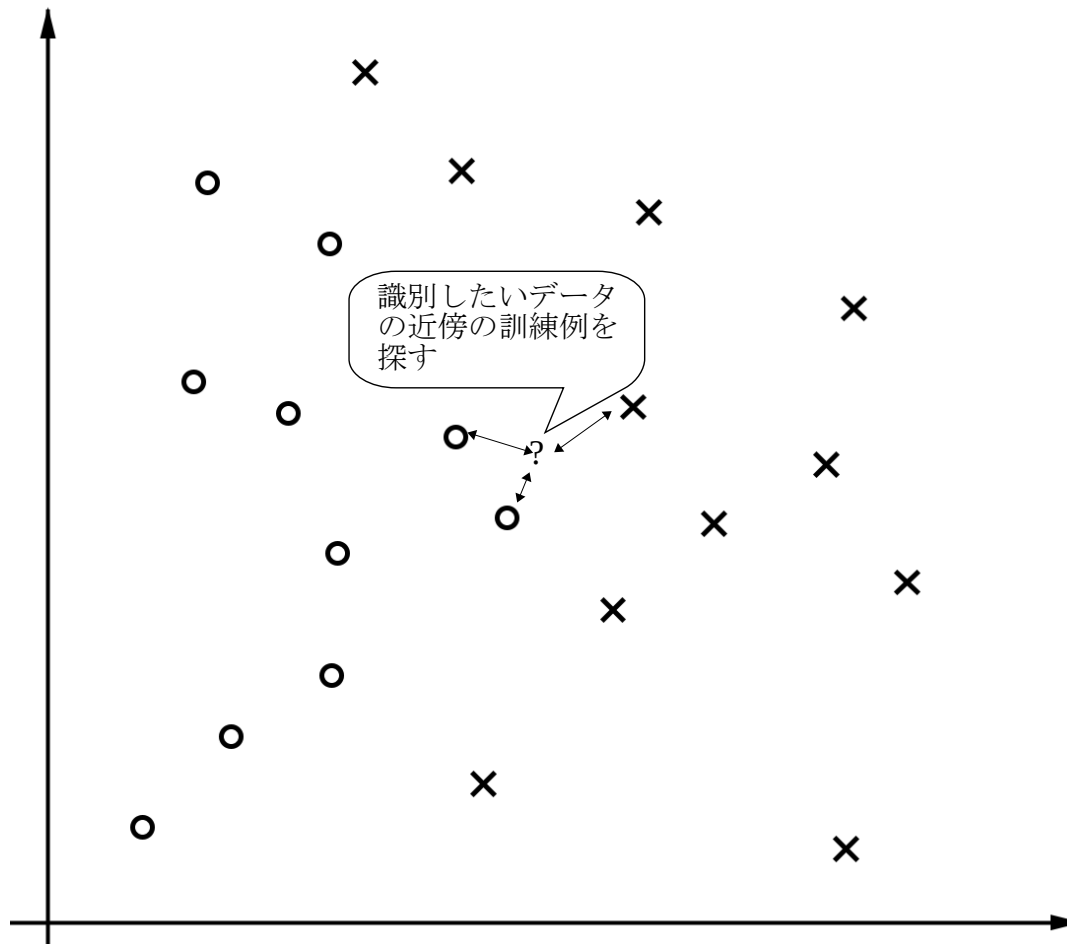


x 軸、y 軸、色の
基準が選べる

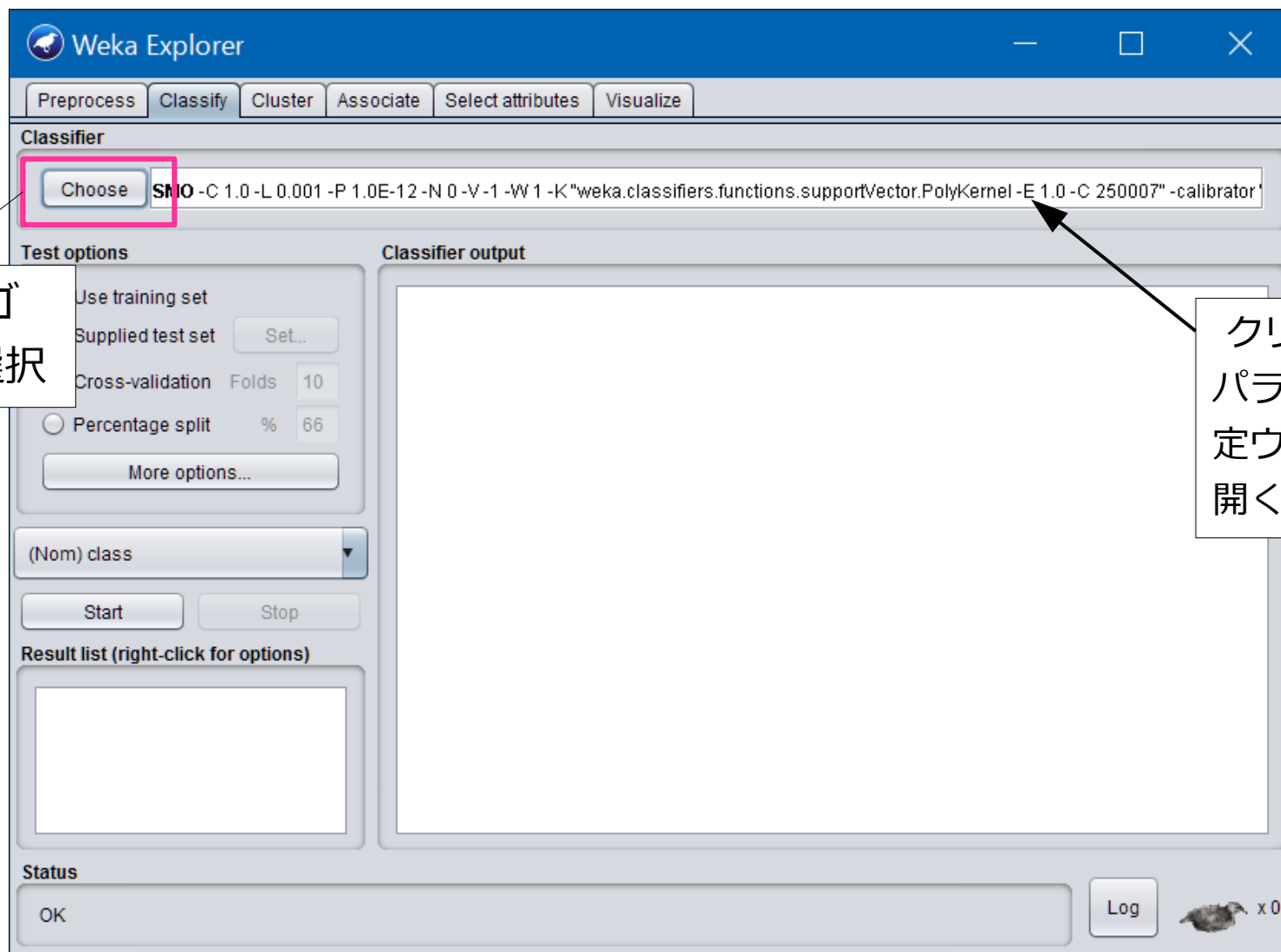
クリックで
色を変更可能

2.1.4 学習 k-NN 法

- 識別したいデータの近傍の k 個の学習データを探し、属するクラスの多数決で識別



識別器の学習 (Classify)

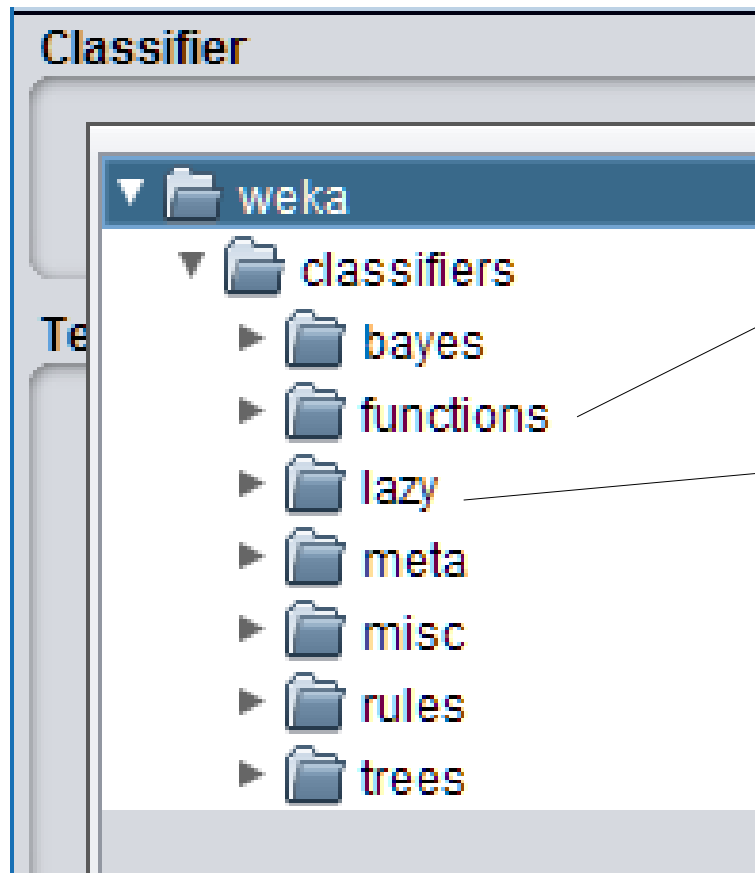


識別アルゴ
リズムの選択

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)

- 識別器の選択

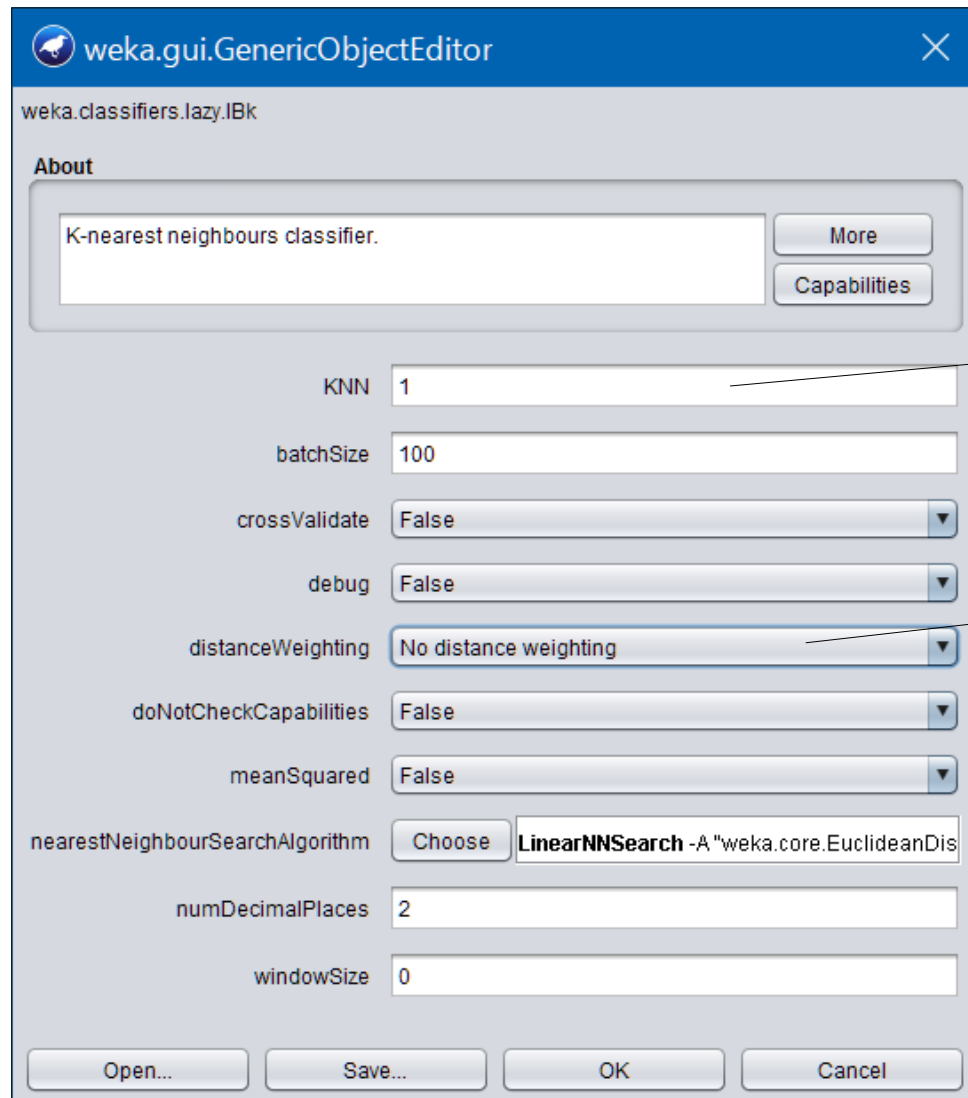


- MultilayerPerceptron (ニューラルネット)
- SMO (SVM)

- IBk (k-NN)

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.EuclideanDis
- numDecimalPlaces: 2
- windowSize: 0

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

k

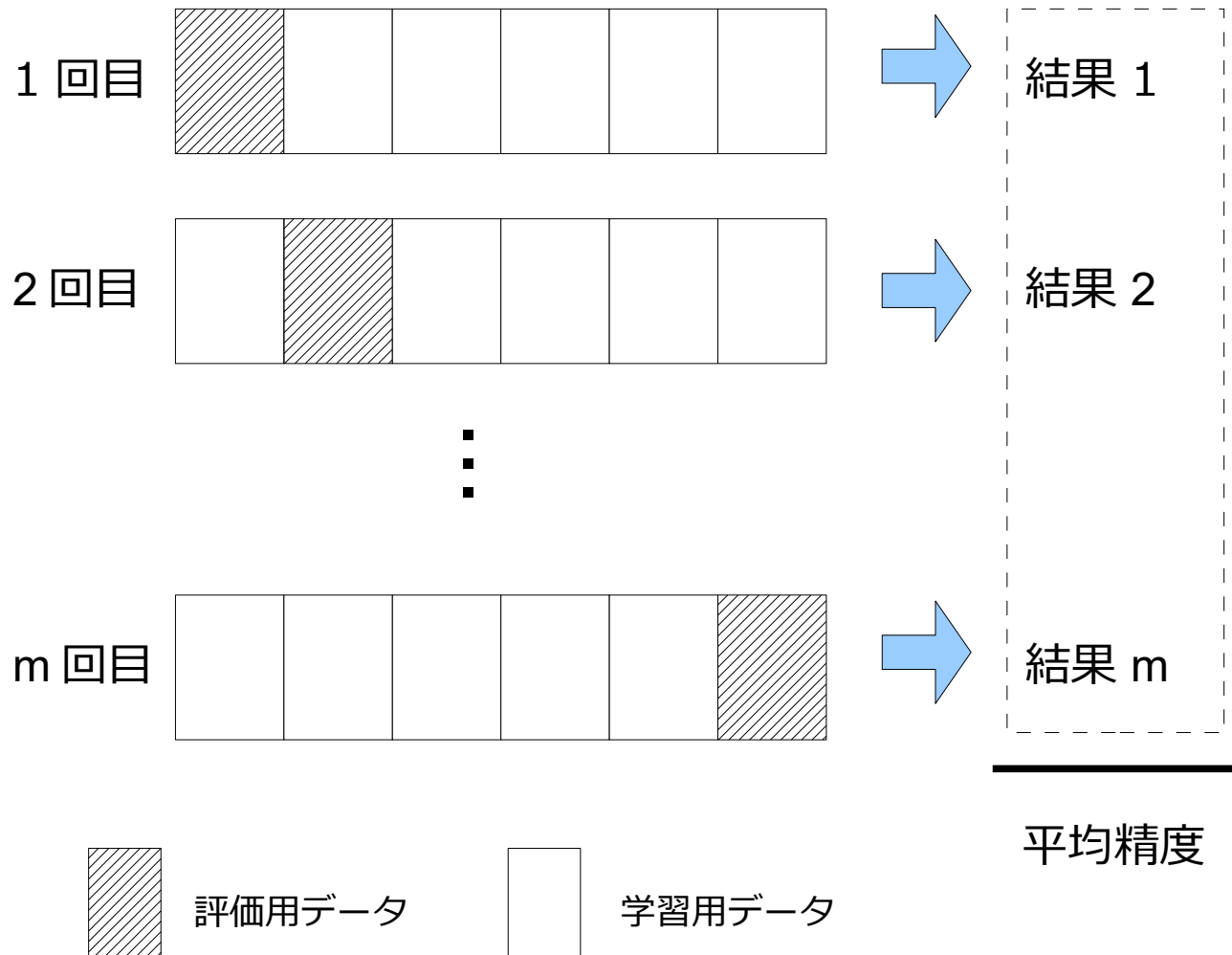
距離による重み付けの有無

2.1.3 評価基準の設定

- 分割学習法
 - データの半分を学習用、残りの半分を評価用とする
 - ハイパーパラメータを調整する場合は、学習用・検証用・評価用に分ける
- 交差確認法
 - データを m 個の集合に分割し、 $m-1$ 個の集合で学習、残りの 1 個の集合で評価を行う
 - 評価する集合を入れ替え、合計 m 回評価を行う
 - 分割数をデータ数とする場合を一つ抜き法とよぶ

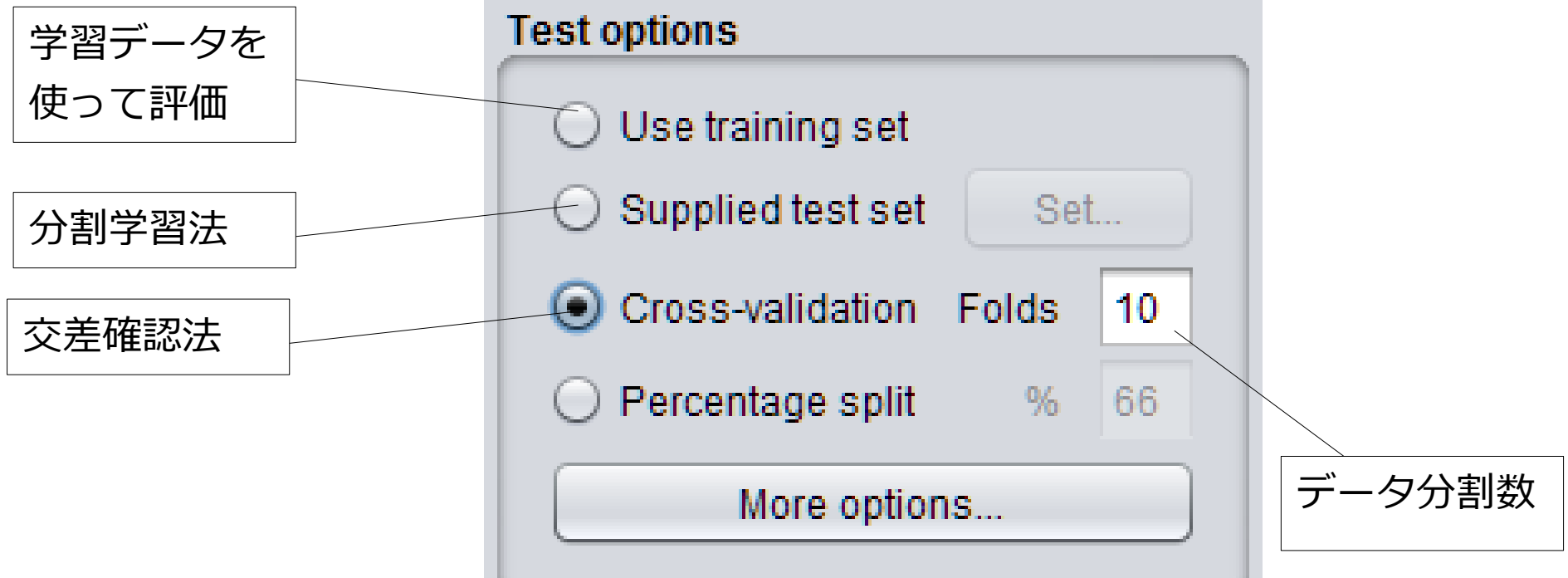
2.1.3 評価基準の設定

- 交差確認法



識別器の学習 (Classify)

- 評価法の設定



識別器の学習 (Classify)

- 学習結果の見方

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

```
Correctly Classified Instances      14
Incorrectly Classified Instances    1
Kappa statistic                     0.9167
Mean absolute error                 0.1051
Root mean squared error             0.1645
Relative absolute error             31.4161 %
Root relative squared error         39.3051 %
Total Number of Instances          15
```

...

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

```
a b c d e    <-- classified as
3 0 0 0 0 | a = a
0 3 0 0 0 | b = i
0 0 3 0 0 | c = u
0 0 0 3 0 | d = e
1 0 0 0 2 | e = o
```

正解率

93.3333 %
6.6667 %

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

- 学習したモデル
 - 式、木構造、ネットワークの重み、 etc.
- 性能
 - 正解率、精度、再現率、 F 値
 - グラフ
 - パラメータを変えたときの性能の変化
 - 異なるモデルの性能比較

2.1.5 結果の可視化

- 混同行列

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

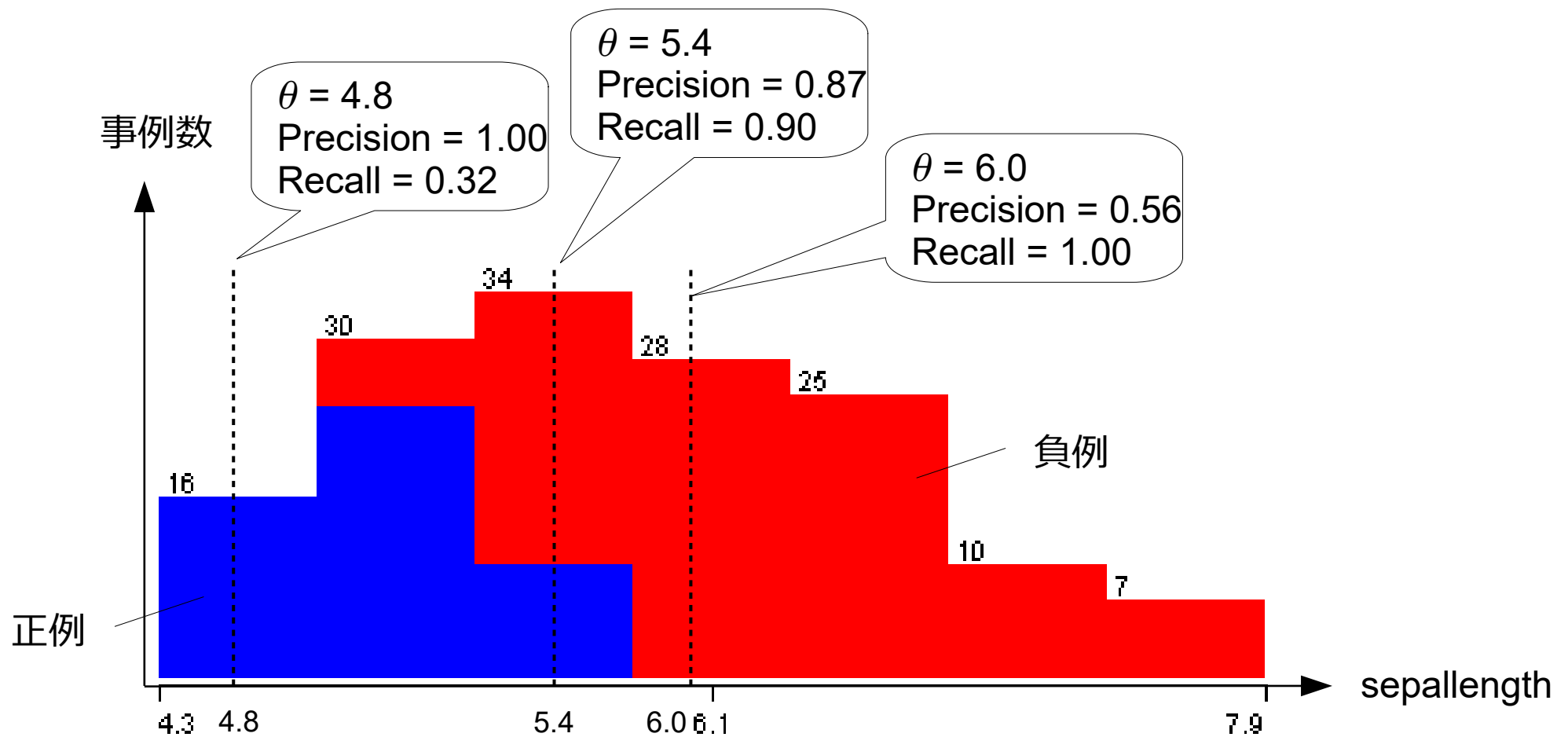
正しく判定された
正例の割合

↑
トレードオフ
↓

精度と再現率の
調和平均

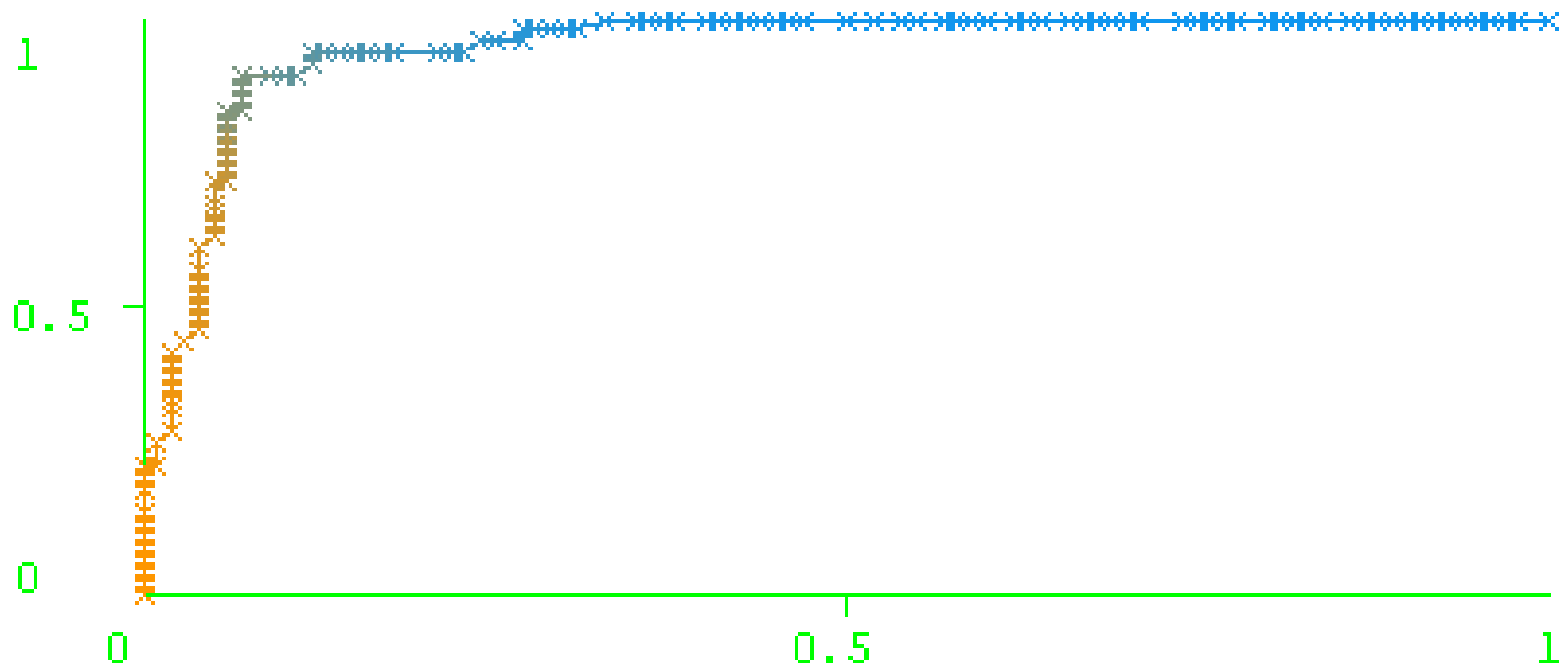
2.1.5 結果の可視化

- 識別のための閾値の設定
 - sepallength 特徴による Iris-setosa の識別



2.1.5 結果の可視化

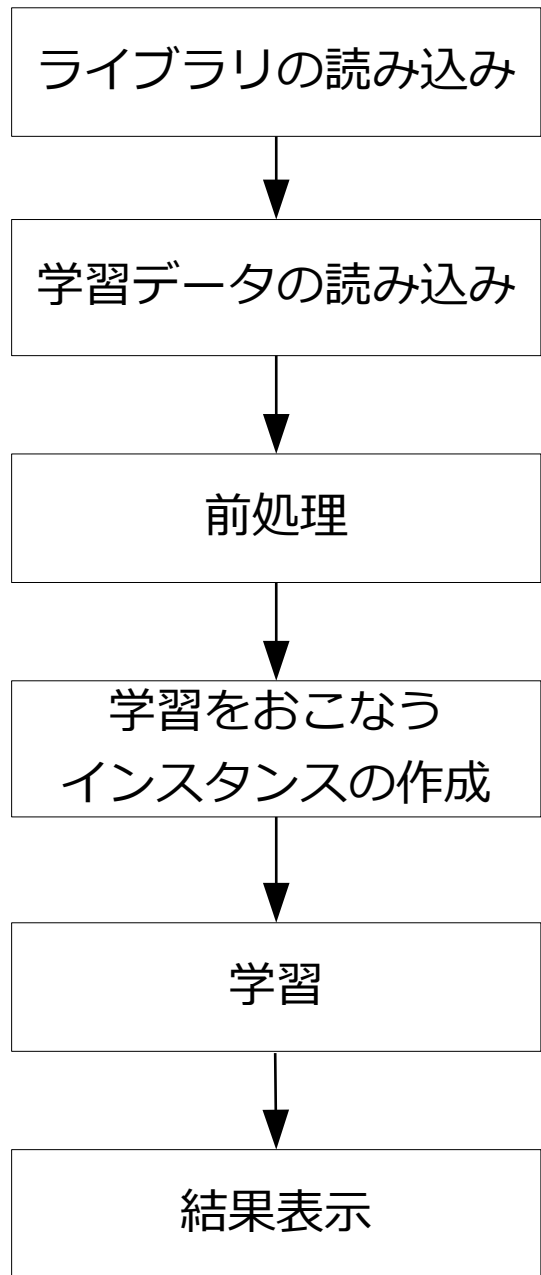
- 精度と再現率のトレードオフ
 - ROC 曲線



2.2 Python による機械学習

- Python を使うメリット
 - データ処理や機械学習のパッケージが充実
 - グラフ表示などの可視化が容易
 - Jupyter Notebook で、実行手順を記録しながらコーディングが可能

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