データベース演習 （Advanced Database Exercise）

８．クラスタリング

URL: http://www.kkaneko.jp/cc/dbenshu/index.html

**概要 Abstract**

今日の授業では、オブジェクト指向データベース管理システムZODB のデータを使い、簡単なデータ解析の演習を行う

Today's class is exercise on data analysis using an object-oriented database management system 'ZODB'.

■ 分類 (classification)

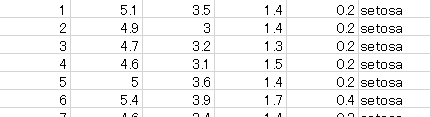
特徴空間中のベクトル集合　(a set of vectors in a feature space) ：　 {*x*1, *x*2, …, *x*n}

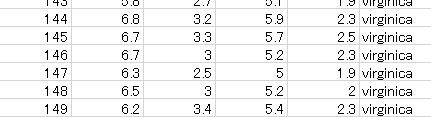
分類ラベル集合 (a set of classification labels)： {*y*1, *y*2, …, *y*n} 但し yi in {y1, y2, …,ym}

が与えられたとする. この授業では、特徴空間中のベクトル集合と、分類ラベル集合とから、特徴空間中のベクトルを分類ラベルにマップする識別機 (classifier) を生成することを、分類 (classification) という。

In this class, "classification" is to generate a classifier that maps a feature vector onto a classification label, and the classifier is genereted from a set of vectors in a feature space and a set of classification labels)

例えば、iris データセット（全部で150行）の0行目から148行目までのデータを使い識別機を生成したとする。この識別機は、iris データセットの149行目のデータにある特徴ベクトル「(5.9, 3, 5.1, 1.8)」を、分類ラベル "virginica" にマップできる識別機であることが期待できる。





分類ラベル

特徴ベクトル

図．iris データセットの 0 行目から 148 行目までのデータ

■ 訓練データ (training data)

分類を行うための特徴空間中のベクトル集合と、分類ラベル集合のことを訓練データ (training data)、訓練例 (training set) という。

■ 教師なしの分類

この授業では、特徴空間中のベクトル集合だけを使う（分類ラベル集合を使わない）ことで分類を行うことを教師なしの分類という。

　　分類：　特徴空間中のベクトル集合、分類ラベル集合を使う

　　教師なしの分類：　特徴空間中のベクトル集合だけを使う

* **ステップ1 (Step 1)**

**課題：以下の手順で Spyderの設定を行いなさい　(Setup the Spyder software)**

1. Spyder の起動 (execute the Spyder)



1. Interpreters で 「**Open an IPython Console**」を選ぶ. (Select Interpreters, and Select “Open an IPython console”)



1. 「IPython console」をクリック. (Click “IPython console”)



* **ステップ2 (Step 2)**

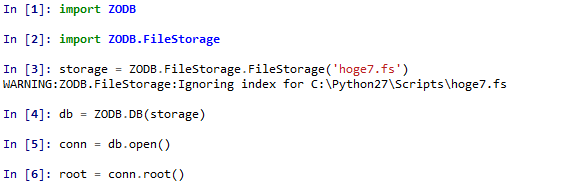
**課題：以下の手順で ZODB データベースに接続しなさい (Connect to a ZODB database)**

**◆** 今日の授業では、ZODB データベースファイル名は **hoge7.fs**

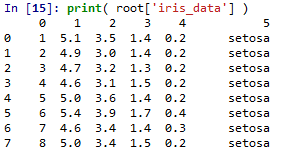
　　　(The ZODB database file name in the previous class was '**hoge7.fs**'. Let us use it again here)

**2. IPython コンソールで、ZODBデータベースに接続し、変数root に**ZODBのルートオブジェクトを格納する

**(connect to a ZODB database using IPython console, and store the root object into the variable named root)**



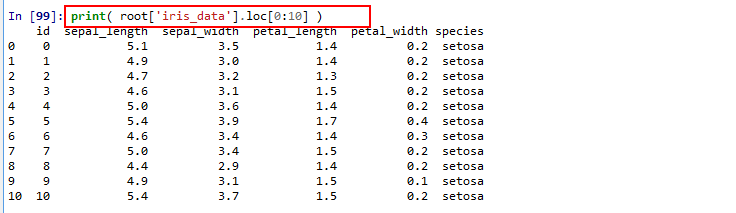
3. iris\_data の確認 Print iris\_data



* **ステップ 5 (Step 5)**
* **ステップ3 (Step 3)**

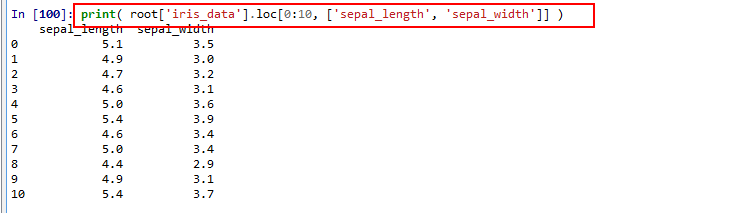
**データフレームからの行の選択と列の選択)**

1. loc を用いた行の選択. 0行目から10行目を選ぶ. Select lines from 0 to 10 using loc

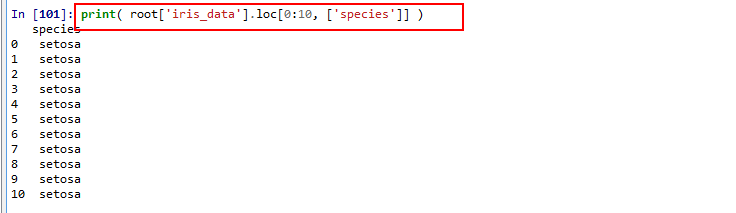


2. 列を選ぶ. Select Columns.

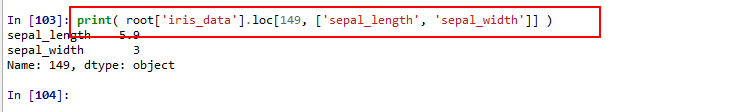
(1) 列 sepal\_length, sepal\_widthを選ぶ

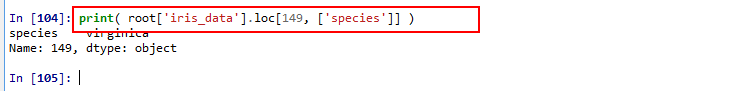


(2) 列 species を選ぶ



(3) 今度は149 行目を選ぶ. Select line 149 using loc.





* **ステップ4 (Step 4)**

**分類する．この授業では Linear SVM を用いる．Linear SVM の詳細はこの授業では触れない。**

　　Do classification using SVM

**1. 分類する。分類機は変数 classifier に格納される. An classifier is stored into variable named 'classifier'**

import sklearn

import sklearn.svm

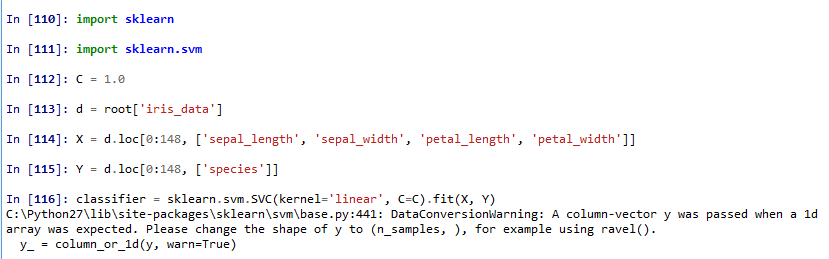
C = 1.0

d = root['iris\_data']

X = d.loc[0:148, ['sepal\_length', 'sepal\_width', 'petal\_length', 'petal\_width']]

Y = d.loc[0:148, ['species']]

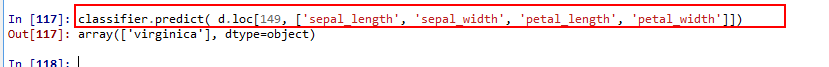
classifier = sklearn.svm.SVC(kernel='linear', C=C).fit(X, Y)



特徴ベクトル

**2. 分類機を使ってみる．Use the classifier**

　The data is mapped onto 'virginica'.



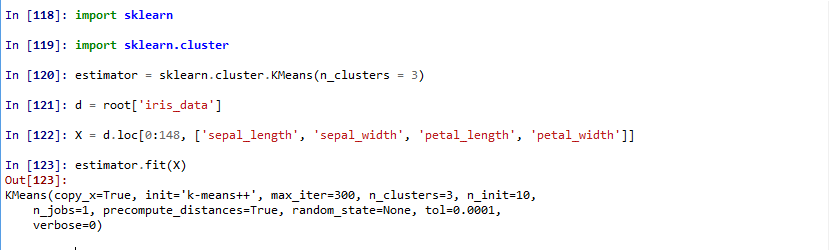
* **ステップ5 (Step 5)**

**教師なしで分類する．この授業では k-Meansを用いる．k-Meansの詳細はこの授業では触れない。**

　　Do classification using k-Means

1. 教師なしで分類する。分類機は変数 estimator に格納される. An classifier is stored into variable named 'classifier'

n\_clusters = 3 は「分類後のクラスタ数が3である」ことを指定している。このとき、分類ラベルは 0, 1, 2 の3通りになる。



特徴ベクトルだけ使用

2. 分類機を見てみる

