# Laboratorium: Drzewa Decyzyjne

## 1 Cel/Zakres

- Drzewa decyzyjne: klasyfikacja i regresja.
- Wizualizacja drzew.

## 2 Przygotowanie danych

Dane sa poniższe zbiory danych: `data_breast_cancer` i `df`.

```
from sklearn import datasets
data_breast_cancer = datasets.load_breast_cancer(as_frame=True)
print(data_breast_cancer['DESCR'])
```

```
.. _breast_cancer_dataset:

Breast cancer wisconsin (diagnostic) dataset
--------------------------------------------

**Data Set Characteristics:**

:Number of Instances: 569

:Number of Attributes: 30 numeric, predictive attributes and the class

:Attribute Information:
    - radius (mean of distances from center to points on the perimeter)
    - texture (standard deviation of gray-scale values)
    - perimeter
    - area
    - smoothness (local variation in radius lengths)
    - compactness (perimeter^2 / area - 1.0)
    - concavity (severity of concave portions of the contour)
    - concave points (number of concave portions of the contour)
    - symmetry
    - fractal dimension ("coastline approximation" - 1)

    The mean, standard error, and "worst" or largest (mean of the three
    worst/largest values) of these features were computed for each image,
```

resulting in 30 features. For instance, field 0 is Mean Radius, field 10 is Radius SE, field 20 is Worst Radius.

- class:
  - WDBC-Malignant
  - WDBC-Benign

:Summary Statistics:

| | Min | Max |
|---|---|---|
| radius (mean): | 6.981 | 28.11 |
| texture (mean): | 9.71 | 39.28 |
| perimeter (mean): | 43.79 | 188.5 |
| area (mean): | 143.5 | 2501.0 |
| smoothness (mean): | 0.053 | 0.163 |
| compactness (mean): | 0.019 | 0.345 |
| concavity (mean): | 0.0 | 0.427 |
| concave points (mean): | 0.0 | 0.201 |
| symmetry (mean): | 0.106 | 0.304 |
| fractal dimension (mean): | 0.05 | 0.097 |
| radius (standard error): | 0.112 | 2.873 |
| texture (standard error): | 0.36 | 4.885 |
| perimeter (standard error): | 0.757 | 21.98 |
| area (standard error): | 6.802 | 542.2 |
| smoothness (standard error): | 0.002 | 0.031 |
| compactness (standard error): | 0.002 | 0.135 |
| concavity (standard error): | 0.0 | 0.396 |
| concave points (standard error): | 0.0 | 0.053 |
| symmetry (standard error): | 0.008 | 0.079 |
| fractal dimension (standard error): | 0.001 | 0.03 |
| radius (worst): | 7.93 | 36.04 |
| texture (worst): | 12.02 | 49.54 |
| perimeter (worst): | 50.41 | 251.2 |
| area (worst): | 185.2 | 4254.0 |
| smoothness (worst): | 0.071 | 0.223 |
| compactness (worst): | 0.027 | 1.058 |
| concavity (worst): | 0.0 | 1.252 |
| concave points (worst): | 0.0 | 0.291 |
| symmetry (worst): | 0.156 | 0.664 |
| fractal dimension (worst): | 0.055 | 0.208 |

:Missing Attribute Values: None

:Class Distribution: 212 - Malignant, 357 - Benign

:Creator:  Dr. William H. Wolberg, W. Nick Street, Olvi L. Mangasarian

:Donor: Nick Street

:Date: November, 1995

This is a copy of UCI ML Breast Cancer Wisconsin (Diagnostic) datasets.
https://goo.gl/U2Uwz2

Features are computed from a digitized image of a fine needle
aspirate (FNA) of a breast mass.  They describe
characteristics of the cell nuclei present in the image.

Separating plane described above was obtained using
Multisurface Method-Tree (MSM-T) [K. P. Bennett, "Decision Tree
Construction Via Linear Programming." Proceedings of the 4th
Midwest Artificial Intelligence and Cognitive Science Society,
pp. 97-101, 1992], a classification method which uses linear
programming to construct a decision tree.  Relevant features
were selected using an exhaustive search in the space of 1-4
features and 1-3 separating planes.

The actual linear program used to obtain the separating plane
in the 3-dimensional space is that described in:
[K. P. Bennett and O. L. Mangasarian: "Robust Linear
Programming Discrimination of Two Linearly Inseparable Sets",
Optimization Methods and Software 1, 1992, 23-34].

This database is also available through the UW CS ftp server:

ftp ftp.cs.wisc.edu
cd math-prog/cpo-dataset/machine-learn/WDBC/

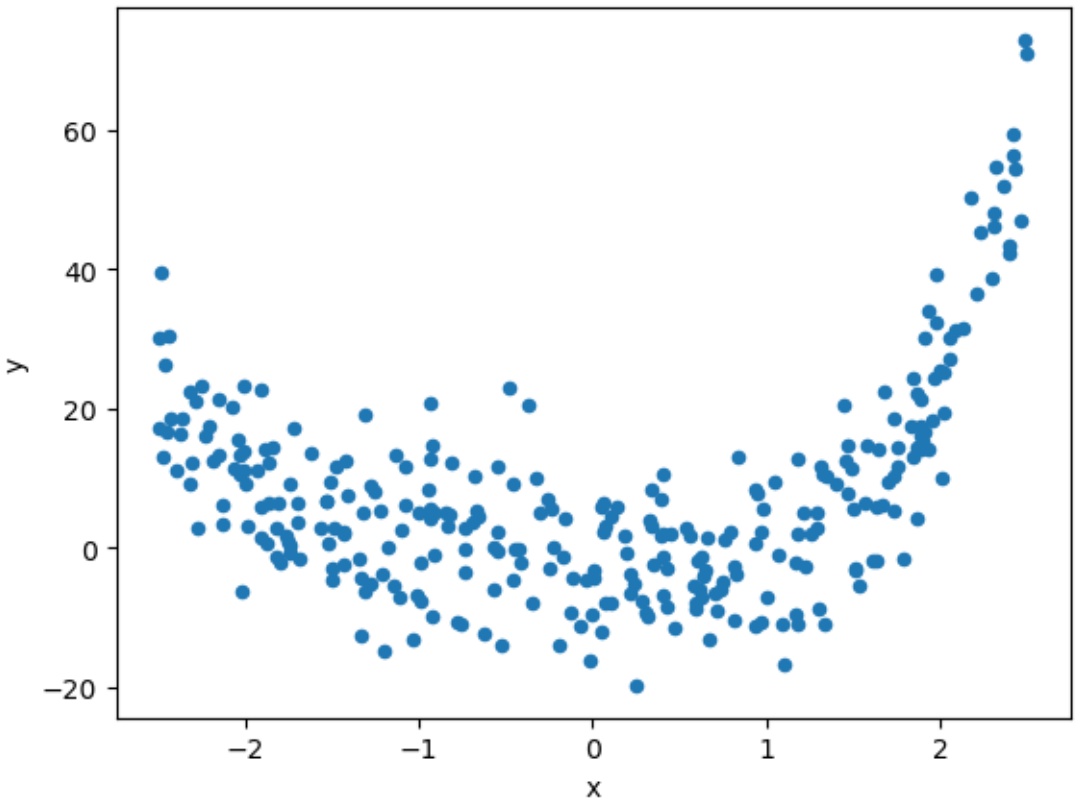
|details-start|
**References**
|details-split|


- W.N. Street, W.H. Wolberg and O.L. Mangasarian. Nuclear feature extraction
  for breast tumor diagnosis. IS&T/SPIE 1993 International Symposium on
  Electronic Imaging: Science and Technology, volume 1905, pages 861-870,
  San Jose, CA, 1993.
- O.L. Mangasarian, W.N. Street and W.H. Wolberg. Breast cancer diagnosis and
  prognosis via linear programming. Operations Research, 43(4), pages 570-577,
  July-August 1995.
- W.H. Wolberg, W.N. Street, and O.L. Mangasarian. Machine learning techniques
  to diagnose breast cancer from fine-needle aspirates. Cancer Letters 77 (1994)
  163-171.

|details-end|

```
import numpy as np
import pandas as pd
size = 300
X = np.random.rand(size)*5-2.5
w4, w3, w2, w1, w0 = 1, 2, 1, -4, 2
y = w4*(X**4) + w3*(X**3) + w2*(X**2) + w1*X + w0 + np.random.randn(size)*8-4
df = pd.DataFrame({'x': X, 'y': y})
df.plot.scatter(x='x',y='y')
```

```
<Axes: xlabel='x', ylabel='y'>
```




## 3 Klasyfikacja

1. Użyj drzew decyzyjnych do klasyfikacji zbioru danych `data_breast_cancer` dla cech `mean texture` i `mean symmetry`.
2. Podziel ww. zbiór na uczący i testujący w proprcjach 80:20.
3. Znajdź odpowiednią głębokosc drzewa decyzyjnego, tak aby osiągnąć maksymalną wartość f1 (uwaga: sprawdź dla zbioru uczącego i testowego).

4. Wygenruj rysunek drzewa decyzyjnego w pliku `bc.png`.

   1 pkt

5. Zapisz w pliku Pickle `f1acc_tree.pkl` listę zawierającą: głebokość drzewa, f1 dla zbioru uczącego, f1 dla zbioru testowego, dokładność (accuracy) dla zbioru uczącego, dokładność (accuracy) dla zbioru testowego.

   5 pkt

## 4 Regresja

1. Użyj drzew decyzyjnych do budowy regresora na zbiorze danych `df`.
2. Podziel w/w zbiór na uczący i testujący w proprcjach 80/20.
3. Znajdź odpowiednią głębokosc drzewa decyzyjnego, tak aby wartość błędu średniokwadratowego (MSE), zarówno dla zbioru uczącego i testującego, były jak najmniejsze (uwaga na *overfitting*).
4. Sporządź wykres wszystkich danych z `df` oraz predykcji regresora, porównaj wyniki z tymi osiągniętymi dla regresji wielomianowej i KNN z poprzednich ćwiczeń.
5. Wygenruj rysunek drzewa decyzyjnego w pliku `reg.png`.

   1 pkt

6. Zapisz w pliku Pickle `mse_tree.pkl` listę zawierającą: głebokość drzewa, MSE dla zbioru uczącego, MSE dla zbioru testowego.

   5 pkt

## 5 Prześlij raport

Prześlij plik o nazwie `lab05/lab05.py` realizujący ww. ćwiczenia.

Sprawdzane będzie, czy skrypt Pythona tworzy wszystkie wymagane pliki oraz czy ich zawartość jest poprawna.