Note) Slides and example codes are available:
https://github.com/mint-lab/dl_tutorial

ML Lab #1:

Breast Cancer Classification

Sunglok Choi, Assistant Professor, Ph.D.
Computer Science and Engineering Department, SEOULTECH
sunglok@seoultech.ac.kr | https://mint-lab.github.io/

Overview

- **Prerequisite**
  - Anacodna (Individual Edition)

- **Practice) Breast Cancer Classification**
  - The given data
  - Expected results
  - Practice with the skeleton code
    - Step #1) Load the dataset
    - Step #2) Find any better classifier
    - Step #3) Visualize the confusion matrix

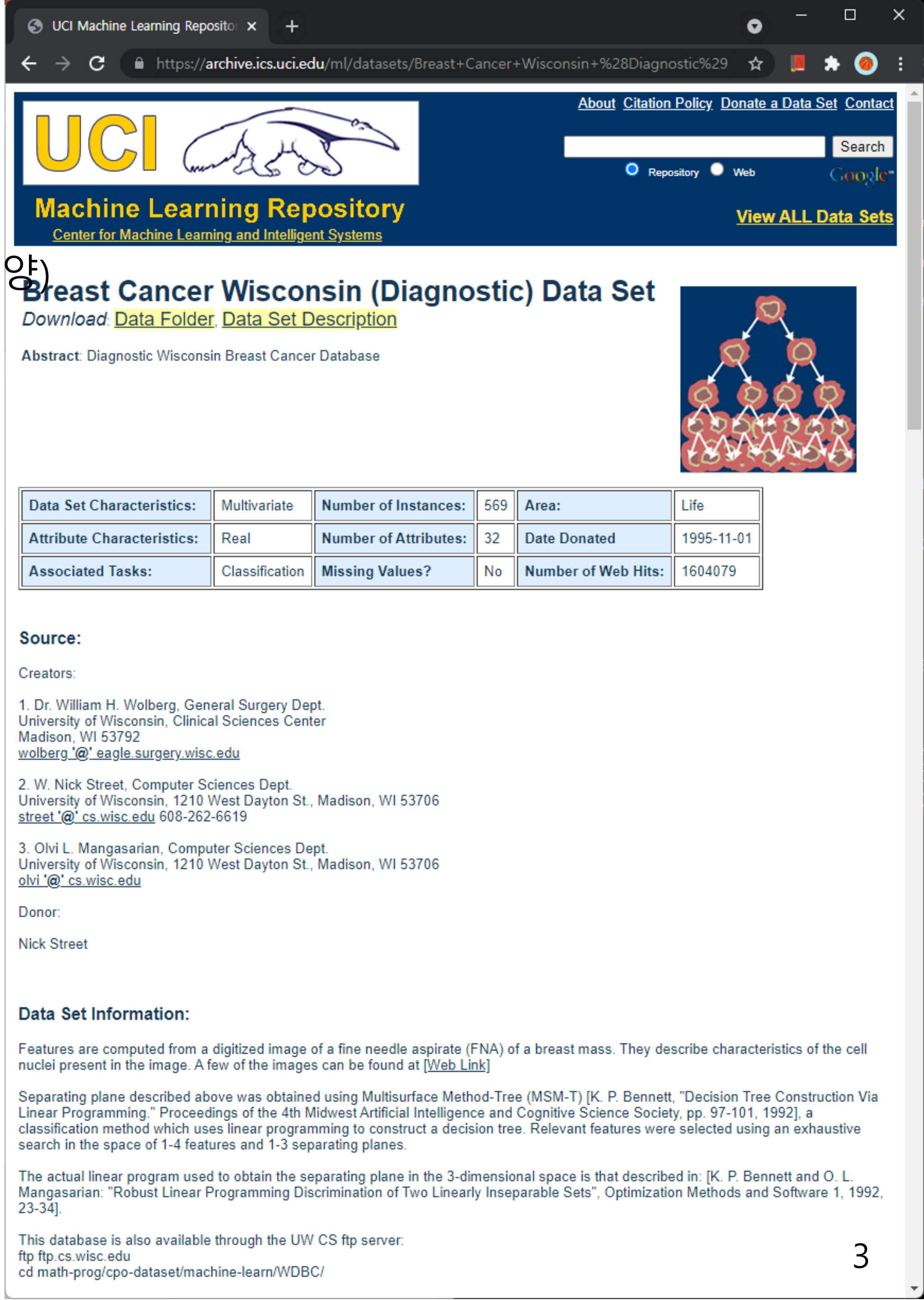
Pinkwashing

- **Assignment**
  - Mission: Complete the given skeleton code

Practice) Breast Cancer Classification

- The given data: Breast Cancer Wisconsin (Diagnostic) Data Set
  - Classes (#: **2**): *Malignant* (M; 악성종양 in Korean), *Benign* (B; 양성종양)
  - Attributes: **30** real numbers (except ID and target class)
    - Radius
    - Texture
    - Perimeter
    - Area
    - …
  - The number of data: **569** (M: 212, B: 357)
  - Note) Load the dataset using scikit-learn [API]

```
from sklearn import datasets
wdbc = datasets.load_breast_cancer()
```



UCI Machine Learning Reposito ×
https://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+%28Diagnostic%29
About Citation Policy Donate a Data Set Contact
Search
○ Repository ○ Web
Google
UCI
Machine Learning Repository
Center for Machine Learning and Intelligent Systems
View ALL Data Sets

Breast Cancer Wisconsin (Diagnostic) Data Set

Download: Data Folder, Data Set Description

Abstract: Diagnostic Wisconsin Breast Cancer Database

Data Set Characteristics:	Multivariate	Number of Instances:	569	Area:	Life
Attribute Characteristics:	Real	Number of Attributes:	32	Date Donated	1995-11-01
Associated Tasks:	Classification	Missing Values?	No	Number of Web Hits:	1604079

Source:

Creators:

1. Dr. William H. Wolberg, General Surgery Dept.
University of Wisconsin, Clinical Sciences Center
Madison, WI 53792
wolberg '@' eagle.surgery.wisc.edu

2. W. Nick Street, Computer Sciences Dept.
University of Wisconsin, 1210 West Dayton St., Madison, WI 53706
street '@' cs.wisc.edu 608-262-6619

3. Olvi L. Mangasarian, Computer Sciences Dept.
University of Wisconsin, 1210 West Dayton St., Madison, WI 53706
olvi '@' cs.wisc.edu

Donor:

Nick Street

Data Set Information:

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. A few of the images can be found at [Web Link]

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/

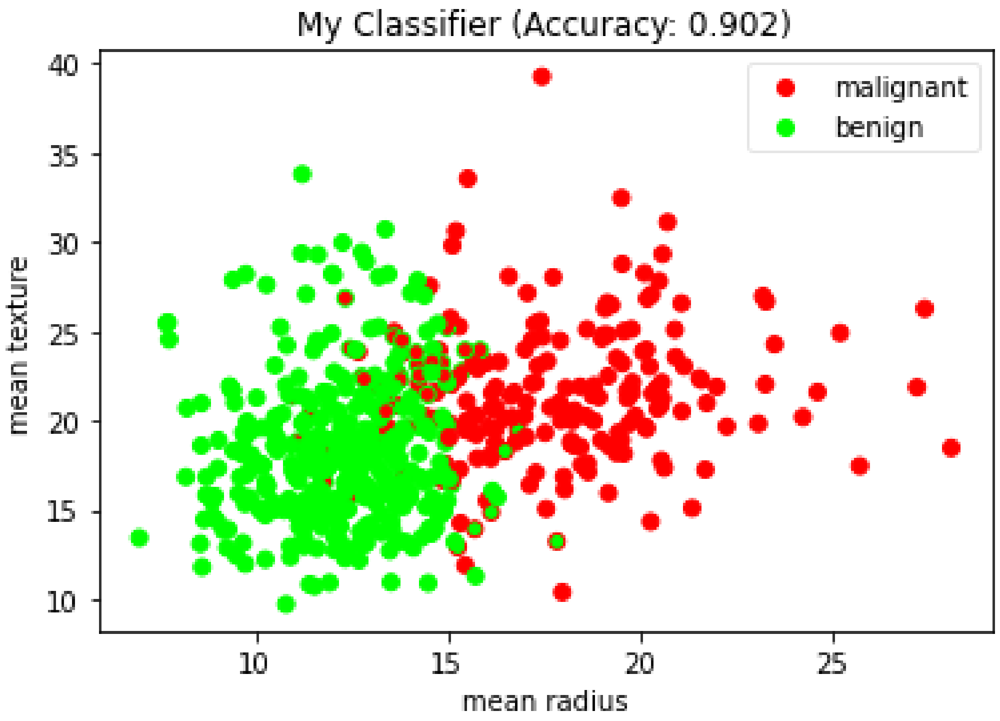
Practice) Breast Cancer Classification

- The given data (file: `data/wdbc.data`)
  - File format: CSV (comma-separated values)
    - `ID,target class (M or F),radius,texture,perimeter,area, ...`
  - Example

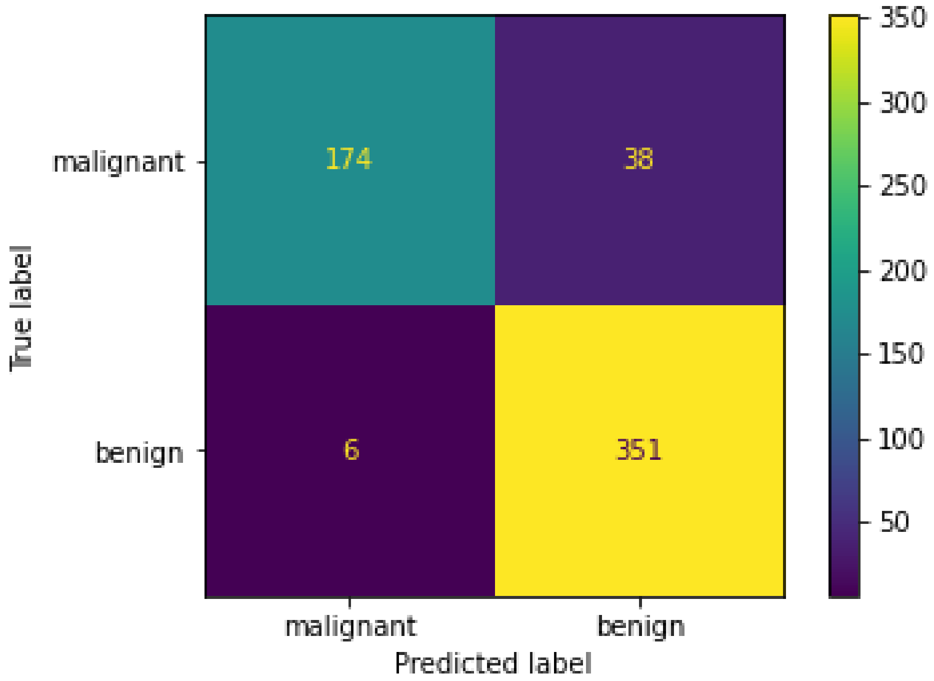
```
842302,M,17.99,10.38,122.8,1001,0.1184,0.2776,0.3001,0.1471,0.2419,0.07871,1.095,0.9053,8.589,153.4,0.006399,0.04904,0.05373,0.01587,0.03003,0.006193,25.38,17.33,184.6,2019,0.1622,0.6656,0.7119,0.2654,0.4601,0.1189
...
```

Practice) Breast Cancer Classification

- Expected results
  - The default classifier: SVM (`svm.SVC`)



My Classifier (Accuracy: 0.902)



Practice) Breast Cancer Classification

- The given skeleton code (`wdbc_classification_skeleton.py`)
  - Step #1) Load the dataset

```
def load_wdbc_data(filename):
    class WDBCData:
        data         = [] # Shape: (569, 30)
        target       = [] # Shape: (569, )
        target_names = ['malignant', 'benign']
        ...
    wdbc = WDBCData()
    with open(filename) as f:
        for line in f.readlines():
            items = line.split(',')
            wdbc.target.append(items[1])        # TODO #1) Add the true label (0 for M / 1 for others)
            wdbc.data.append(items[2:])         # TODO #1) Add 30 attributes (as floating-point numbers)
        wdbc.data = np.array(wdbc.data)
    return wdbc

if __name__ == '__main__':
    # Load a dataset
    wdbc = load_wdbc_data('data/wdbc.data')     # TODO #1) Implement 'load_wdbc_data()'
```

Practice) Breast Cancer Classification

- The given skeleton code (`wdbc_classification_skeleton.py`)
  - Step #2) Find any better classifier
  - Step #3) Visualize the confusion matrix

```
if __name__ == '__main__':
    # Load a dataset
    # wdbc = datasets.load_breast_cancer()
    wdbc = load_wdbc_data('data/wdbc.data')     # TODO #1) Implement 'load_wdbc_data()'

    # Train a model
    model = svm.SVC()                           # TODO #2) Find a better classifier (SVC accuracy: 0.902)
    model.fit(wdbc.data, wdbc.target)

    # Test the model
    predict = model.predict(wdbc.data)
    accuracy = metrics.balanced_accuracy_score(wdbc.target, predict)

    # Mission #3) Visualize the confusion matrix

    # Visualize testing results
    ...
```

Assignment

- Mission
  - Complete the following three missions using the given skeleton code (`wdbc_classification_skeleton.py`)
  - Submit your code (`wdbc_classification.py`) and its two result images (`wdbc_classification_scatter.png`, `wdbc_classification_matrix.png`)

- Condition
  - Please follow the above filename convention.
  - You can start from scratch (without using the given skeleton code).
    - However, you should use the given data.
  - You can freely change the given skeleton code if necessary.

- Submission
  - Deadline: **November 6, 2024 23:59** (firm deadline; no extension)
  - Where: e-Class > Assignments
  - Score: Max 10 points