

Thực hành Nguyên Lý Máy Học

Buổi 2: Giải thuật cây quyết định

Mục tiêu:

- Củng cố lý thuyết và cài đặt giải thuật cây quyết định
- Kiểm thử và đánh giá theo nghi thức hold-out

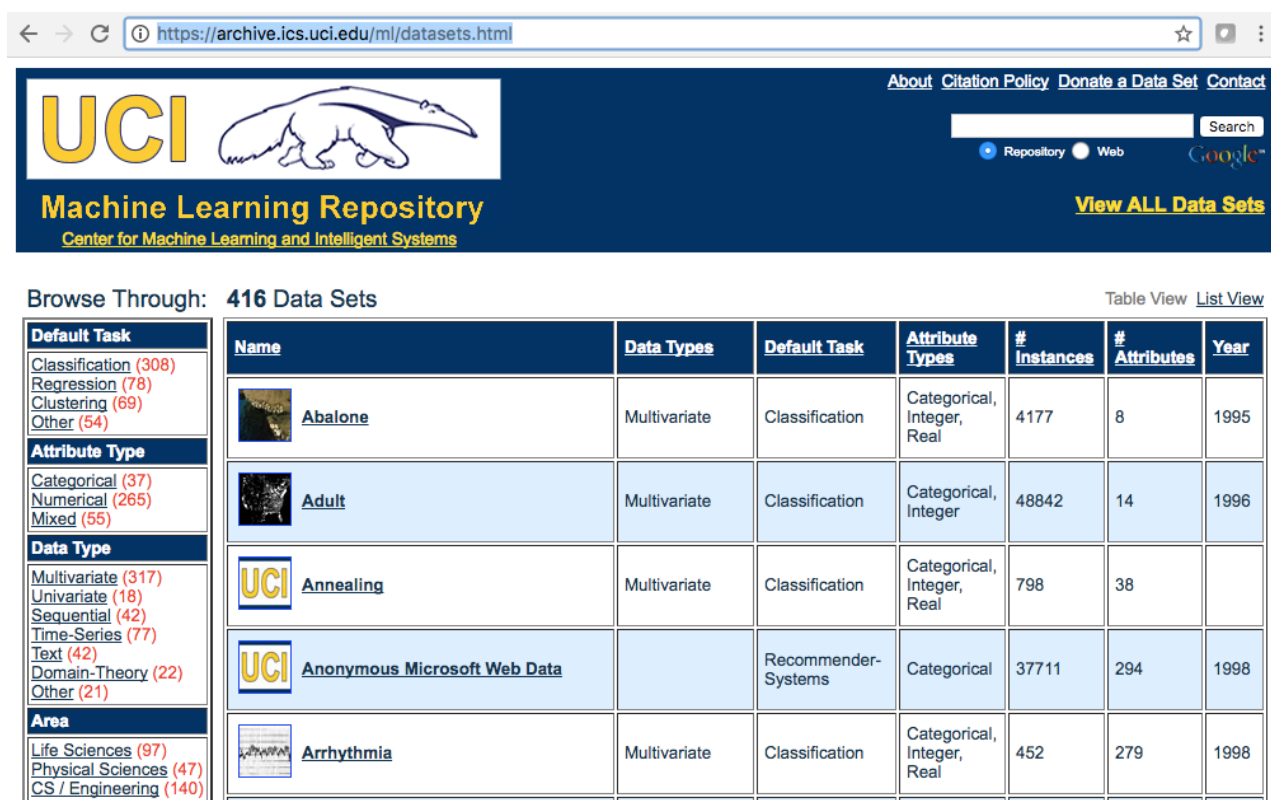
1. HƯỚNG DẪN THỰC HÀNH

Cách cài đặt một số thư viện cần thiết






- Cài đặt một số thư viện phục vụ cho bài thực hành: pandas, sklearn
 - `pip install pandas // đọc file csv`
 - `pip install sklearn`

Trang web lưu trữ các tập dữ liệu sử dụng trong quá trình thực hành

<https://archive.ics.uci.edu/ml/datasets.html>



The screenshot shows the UCI Machine Learning Repository website. The header includes the UCI logo, a search bar, and navigation links. The main content area displays a table of datasets with columns: Name, Data Types, Default Task, Attribute Types, # Instances, # Attributes, and Year. A sidebar on the left provides filters for Default Task, Attribute Type, Data Type, and Area.

Name	Data Types	Default Task	Attribute Types	# Instances	# Attributes	Year
 Abalone	Multivariate	Classification	Categorical, Integer, Real	4177	8	1995
 Adult	Multivariate	Classification	Categorical, Integer	48842	14	1996
 Annealing	Multivariate	Classification	Categorical, Integer, Real	798	38	
 Anonymous Microsoft Web Data		Recommender-Systems	Categorical	37711	294	1998
 Arrhythmia	Multivariate	Classification	Categorical, Integer, Real	452	279	1998

Default Task
Classification (308)
Regression (78)
Clustering (69)
Other (54)

Attribute Type
Categorical (37)
Numerical (265)
Mixed (55)

Data Type
Multivariate (317)
Univariate (18)
Sequential (42)
Time-Series (77)
Text (42)
Domain-Theory (22)
Other (21)

Area
Life Sciences (97)
Physical Sciences (47)
CS / Engineering (140)
Social Sciences (94)

Tập dữ liệu rượu vang sẽ sử dụng trong phần bài tập

Wine Quality Data Set

Download: [Data Folder](#), [Data Set Description](#)

Abstract: Two datasets are included, related to red and white vinho verde wine samples, from the north of Portugal. The goal is to model wine quality based on physicochemical tests (see [Cortez et al., 2009], [Web Link]).



Data Set Characteristics:	Multivariate	Number of Instances:	4898	Area:	Business
Attribute Characteristics:	Real	Number of Attributes:	12	Date Donated	2009-10-07
Associated Tasks:	Classification, Regression	Missing Values?	N/A	Number of Web Hits:	578954

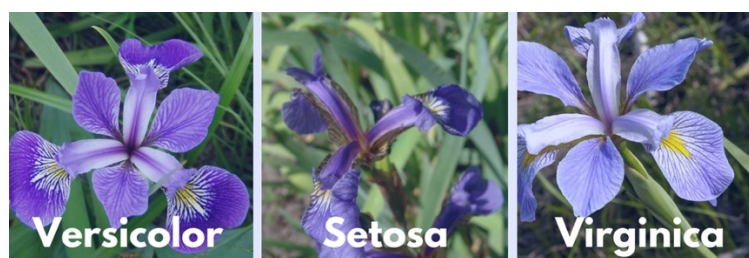
Index of /ml/machine-learning-databases/wine-quality

Name	Last modified	Size	Description
 Parent Directory		-	
 winequality-red.csv	16-Oct-2009 14:36	82K	
 winequality-white.csv	16-Oct-2009 14:36	258K	
 winequality.names	21-Oct-2009 11:00	3.2K	

Apache/2.2.15 (CentOS) Server at archive.ics.uci.edu Port 443

• Tập dữ liệu Iris

Xét bài toán phân loại hoa IRIS dựa trên thông tin về kích thước của cánh hoa và đài hoa. Tập dữ liệu này có 150 phần tử, mỗi loại hoa có 50 phần tử. Dữ liệu có 4 thuộc tính (sepal length, sepal width, petal length, petal width) và 3 lớp (3 loại hoa Iris: Setosa, Versicolour, Virginica)



Tập dữ liệu này có thể download từ trang UCI (<https://archive.ics.uci.edu/ml/datasets/iris>) rồi đọc dữ liệu bằng lệnh `read_csv` của thư viện **Pandas** hoặc có thể nạp dữ liệu có sẵn bởi thư viện **Sklearn**

	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
1	5.1	3.5	1.4	0.2	setosa
2	4.9	3.0	1.4	0.2	setosa
3	4.7	3.2	1.3	0.2	setosa
4	4.6	3.1	1.5	0.2	setosa
5	5.0	3.6	1.4	0.2	setosa
6	5.4	3.9	1.7	0.4	setosa
7	4.6	3.4	1.4	0.3	setosa
8	5.0	3.4	1.5	0.2	setosa

A. Bài toán phân lớp – chỉ số Gini

- Sử dụng tập dữ liệu có sẵn “iris”

```
#Lay file iris truc tiep tu sklearn
from sklearn.datasets import load_iris
iris_dt = load_iris()
iris_dt.data[1:5] # thuoc tinh cua tap iris
iris_dt.target[1:5] #gia tri cua nhan /class
```

- *Phân chia tập dữ liệu để xây dựng mô hình và kiểm tra theo nghi thức Hold-out*

```
from sklearn.cross_validation import train_test_split
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(iris_dt.data, iris_dt.target, test_size=1/3.0,
random_state=5)

X_train[1:6]
X_train[1:6,1:3]
y_train[1:6]
X_test[6:10]
y_test[6:10]
```

- *Xây dựng mô hình cây quyết định dựa trên chỉ số Gini với độ sâu của cây bằng 3, nút nhánh ít nhất có 5 phần tử.*

```
# Xây dựng mô hình cây quyết định dựa trên chỉ số Gini
from sklearn.tree import DecisionTreeClassifier
clf_gini = DecisionTreeClassifier(criterion = "gini", random_state = 100, max_depth=3, min_samples_leaf=5)
clf_gini.fit(X_train, y_train)
```

- *Dự đoán nhãn cho các phần tử trong tập kiểm tra*

```
# dự đoán

y_pred = clf_gini.predict(X_test)
y_test
clf_gini.predict([[4, 4, 3, 3]])
```

- *Tính độ chính xác cho giá trị dự đoán của phần tử trong tập kiểm tra*

```
# tính độ chính xác
from sklearn.metrics import accuracy_score
print ("Accuracy is ", accuracy_score(y_test,y_pred)*100)
```

Kết quả thu được
Accuracy is 96.0

- Tính độ chính xác cho giá trị dự đoán thông qua ma trận con

```
from sklearn.metrics import confusion_matrix
confusion_matrix(y_test, y_pred, labels=[2,0,1])
```

Kết quả thu được

```
[>>> confusion_matrix(y_test, y_pred, labels=[2,0,1])
array([[15,  0,  2],
       [ 0, 16,  0],
       [ 1,  0, 16]])
```

B. Một số cách đọc dữ liệu đầu vào

1. Đọc dữ liệu từ file bằng thư viện panda

Hướng dẫn đọc dữ liệu từ file bằng thư viện “pandas” và truy xuất dữ liệu theo số lượng dòng cũng như theo chỉ số; xác định độ lớn của tập dữ liệu (số record)

```
import pandas as pd
dt5 = pd.read_csv("iris_data.csv")
dt5[1:5]
len(dt5)
dt5.petalLength[1:5]
```

2. Tạo các biến lưu trữ dữ liệu

Tạo dữ liệu gồm 2 thuộc tính x1, x2 và nhãn đặt ở biến y

STT	X1	X2	Nhãn
1.	0	0	0
2.	1	0	0
3.	1	1	0
4.	2	1	1
5.	2	1	1
6.	2	0	0

```
X = [ [0, 0],
       [1, 0],
       [1, 1],
       [2, 1],
       [2, 1],
       [2, 0]]
Y = [0,0,0,1,1,0]
```

C. Bài toán hồi quy – cây hồi quy

1. Cho tập dữ liệu housing_RT.csv có dạng:

	price	lotsize	bedrooms	bathrms	stories
1	38500.0	4000	2	1	1
2	49500.0	3060	3	1	1
3	60500.0	6650	3	1	2

4 61000.0 6360 2 1 1

2. Đọc dữ liệu vào biến “dulieu”

```
import pandas as pd
dulieu = pd.read_csv("housing_RT.csv", index_col=0)
dulieu.iloc[1:5,]
```

3. Sử dụng nghi thức hold-out Phân chia tập dữ liệu huấn luyện

```
from sklearn.model_selection import train_test_split
#X_train,X_test,y_train,y_test = train_test_split( dulieu.ix[:,1:5],dulieu.ix[:,0], test_size=1/3.0, random_state=100)
X_train,X_test,y_train,y_test = train_test_split( dulieu.iloc[:,1:5],dulieu.iloc[:,0], test_size=1/3.0, random_state=100)
X_train.iloc[1:5,]
X_test[1:5]
y_test[1:5]
```

```
from sklearn.tree import DecisionTreeRegressor
regressor = DecisionTreeRegressor(random_state = 0)
regressor.fit(X_train, y_train)
```

4. Dự báo và đánh giá mô hình

- Dự đoán giá trị nhà
y_pred = regressor.predict(X_test)
y_test[1:5]
y_pred[1:5]
- Đánh giá kết quả dự đoán giá trị nhà thông qua chỉ số MSE và RMSE

```
from sklearn.metrics import mean_squared_error
err = mean_squared_error(y_test, y_pred)
err
import numpy as np
np.sqrt(err)
```