pycbr Package Documentation

Release 0.1.1

Dih5

Mar 18, 2022

Contents:

CHAPTER 1

Iris CBR demo

Load the Iris dataset to build the example

[1]:
```
import pandas as pd
from sklearn import datasets

iris = datasets.load_iris()
df=pd.DataFrame(iris["data"], columns=iris["feature_names"])
df["species"]=iris["target"]
df["species"]=df["species"].apply(lambda x:iris["target_names"][x])
df
```

[1]:
```
     sepal length (cm)  sepal width (cm)  petal length (cm)  petal width (cm)  \
0                  5.1               3.5                1.4               0.2
1                  4.9               3.0                1.4               0.2
2                  4.7               3.2                1.3               0.2
3                  4.6               3.1                1.5               0.2
4                  5.0               3.6                1.4               0.2
..                 ...               ...                ...               ...
145                6.7               3.0                5.2               2.3
146                6.3               2.5                5.0               1.9
147                6.5               3.0                5.2               2.0
148                6.2               3.4                5.4               2.3
149                5.9               3.0                5.1               1.8

       species
0       setosa
1       setosa
2       setosa
3       setosa
4       setosa
..         ...
145  virginica
146  virginica
147  virginica
148  virginica
```

(continues on next page)

(continued from previous page)

```
149  virginica

[150 rows x 5 columns]
```

[2]:
```
# Store it in a temporal file
import tempfile
f=tempfile.NamedTemporaryFile(suffix=".csv")
df.to_csv(f.name, index=False)
f.name
```

[2]:
```
'/tmp/tmp6expcq6j.csv'
```

Build a CBR

[3]:
```
import pycbr

# Define a case base from the csv file
case_base=pycbr.casebase.SimpleCSVCaseBase(f.name)
# Define the set of similarity functions
recovery=pycbr.recovery.Recovery([(x, pycbr.models.QuantileLinearAttribute()) for x␣
↪in iris["feature_names"]])
# Define the aggregation method
aggregation = pycbr.aggregate.MajorityAggregate("species")

# Create a CBR instance
cbr=pycbr.CBR(case_base, recovery, aggregation)
```

```
Unable to load a logging configuration file. Using the default settings.
```

```
/usr/lib/python3.8/site-packages/sklearn/preprocessing/_data.py:2344: UserWarning: n_
↪quantiles (1000) is greater than the total number of samples (150). n_quantiles is␣
↪set to n_samples.
  warnings.warn("n_quantiles (%s) is greater than the total number "
/usr/lib/python3.8/site-packages/sklearn/preprocessing/_data.py:2344: UserWarning: n_
↪quantiles (1000) is greater than the total number of samples (150). n_quantiles is␣
↪set to n_samples.
  warnings.warn("n_quantiles (%s) is greater than the total number "
/usr/lib/python3.8/site-packages/sklearn/preprocessing/_data.py:2344: UserWarning: n_
↪quantiles (1000) is greater than the total number of samples (150). n_quantiles is␣
↪set to n_samples.
  warnings.warn("n_quantiles (%s) is greater than the total number "
/usr/lib/python3.8/site-packages/sklearn/preprocessing/_data.py:2344: UserWarning: n_
↪quantiles (1000) is greater than the total number of samples (150). n_quantiles is␣
↪set to n_samples.
  warnings.warn("n_quantiles (%s) is greater than the total number "
```

The Flask WSGI application is available as the app parameter.

[4]:
```
# Start the development server
cbr.app.run()
```

```
 * Serving Flask app "pycbr" (lazy loading)
 * Environment: production
   WARNING: This is a development server. Do not use it in a production deployment.
   Use a production WSGI server instead.
 * Debug mode: off
```

```
2020-03-05 01:38:07 ophelia werkzeug[17052] INFO  * Running on http://127.0.0.1:5000/␣
↪(Press CTRL+C to quit)
```

[ ]:

CHAPTER 2

Indices and tables

- genindex
- modindex
- search