

File Edit Search Source Run Debug Consoles Projects Tools View Help

C:\Users\yogay\spyder-py3\temp.py

```
temp.py* X
1 import numpy as np
2 import matplotlib.pyplot as plt
3 import pandas as pd
4
5 dataset=pd.read_csv(r'C:\Users\yogay\OneDrive\Desktop\Yogita_Yadav\Dataset\Churn.csv')
6
7 X = dataset.iloc[:, :-1].values
8 y = dataset.iloc[:, 3].values
9
10 from sklearn.impute import SimpleImputer
11 imputer = SimpleImputer(missing_values=np.nan)
12
13 imputer = imputer.fit(X[:, 1:3])
14 X[:, 1:3] = imputer.transform(X)
15
16 from sklearn.preprocessing import LabelEncoder
17 labelencoder_X = LabelEncoder()
18 labelencoder_X.fit_transform(X[:, 0])
19 X[:, 0] = labelencoder_X.transform(X[:, 0])
20
21
```

X - NumPy object array (read only)

	0	1	2
0	2	44.0	72000.0
1	0	27.0	48000.0
2	1	30.0	54000.0
3	0	38.0	61000.0
4	1	40.0	61000.0
5	2	35.0	58000.0
6	0	38.0	52000.0
7	2	48.0	79000.0
8	1	50.0	83000.0

Format Resize □ Background color

Close Table Explorer Plots Files

conda: base (Python 3.10.9) Completions: conda(base) LSP: Python Line 21, Col 1 UTF-8 CRLF RW Mem 91% 22:01

Spyder (Python 3.10)

File Edit Search Source Run Debug Consoles Projects Tools View Help

C:\Users\yogay.spyder-py3\temp.py

temp.py* X

```
1 import numpy as np
2 import matplotlib.pyplot as plt
3 import pandas as pd
4
5 dataset=pd.read_csv(r'C:\Users\yoga...
6
7 X = dataset.iloc[:, :-1].values
8 y = dataset.iloc[:, 3].values
9
10 from sklearn.impute import SimpleImputer
11 imputer = SimpleImputer(missing_value...
12
13 imputer = imputer.fit(X[:, 1:3])
14 X[:, 1:3] = imputer.transform(X[:, 1:3])
15
16 from sklearn.preprocessing import LabelEncoder
17 labelencoder_X = LabelEncoder()
18 labelencoder_X.fit_transform(X[:, 0])
19 X[:, 0] = labelencoder_X.transform(X[:, 0])
20
21 labelencoder_y = LabelEncoder()
22 y = labelencoder_y.fit_transform(y)
23
```

dataset - NumPy object array

	0	1	2	3	4	5	6	7	8
0	0	0	0	0	1	0	0	1	0
1	0	1	0	0	0	1	0	0	1
2	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0

Format Resize Background color Save and Close Close

In [17]: y = labelencoder_y.fit_transform(y)

In [18]:

IPython Console History

conda: base (Python 3.10.9) Completions: conda(base) LSP: Python Line 23, Col 1 UTF-8 CRLF RW Mem 90%

Type here to search

22:02 20°C ENG 31-10-2023

File Edit Search Source Run Debug Consoles Projects Tools View Help

C:\Users\yogay\spyder-py3\temp.py

```
temp.py* X
1 import numpy as np
2 import matplotlib.pyplot as plt
3 import pandas as pd
4
5 dataset=pd.read_csv(r'C:\Users\yogay\...
6
7 X = dataset.iloc[:, :-1].values
8 y = dataset.iloc[:, 3].values
9
10 from sklearn.impute import SimpleImputer
11 imputer = SimpleImputer(missing_val...
12
13 imputer = imputer.fit(X[:,1:3])
14 X[:, 1:3] = imputer.transform(X[:,1:3])
15
16 from sklearn.preprocessing import LabelEncoder
17 labelencoder_X = LabelEncoder()
18 labelencoder_X.fit_transform(X[:,0])
19 X[:,0] = labelencoder_X.transform(X[:,0])
20
21 labelencoder_y = LabelEncoder()
22 y = labelencoder_y.fit_transform(y)
23
24 from sklearn.model_selection import train...
25 X_train,X_test,y_train,y_test = tra...
26
```

X_test - NumPy object array (read only)

	0	1	2
0	1	30.0	54000.0
1	1	50.0	83000.0

Name Type Size Value

dataset DataFrame (10, 4) Column names: State, Age, Salary, Purchased

SimpleImputer object of sklearn.impute._base module

LabelEncoder object of sklearn.preprocessing module

LabelEncoder object of sklearn.preprocessing module

In [19]: X_train,X_test, 0.2,random_state=0)

In [20]:

Quick Heal Internet Security

You are no more protected against new threats

Quick Heal subscription has expired

Renew It Now

conda: base (Python 3.10.9) Col 20°C ENG 22:03

The screenshot shows the Spyder IDE interface. The top menu bar includes File, Edit, Search, Source, Run, Debug, Consoles, Projects, Tools, View, and Help. The toolbar below has icons for file operations like Open, Save, and Run. The main area has tabs for 'temp.py*' and 'X_test - NumPy object array (read only)'. The code editor contains Python code for data preprocessing:

```
1 import numpy as np
2 import matplotlib.pyplot as plt
3 import pandas as pd
4
5 dataset=pd.read_csv(r'C:\Users\yogay\dataset.csv')
6
7 X = dataset.iloc[:, :-1].values
8 y = dataset.iloc[:, 3].values
9
10 from sklearn.impute import SimpleImputer
11 imputer = SimpleImputer(missing_values=np.nan)
12
13 imputer = imputer.fit(X[:, 1:3])
14 X[:, 1:3] = imputer.transform(X[:, 1:3])
15
16 from sklearn.preprocessing import LabelEncoder
17 labelencoder_X = LabelEncoder()
18 labelencoder_X.fit_transform(X[:, 0])
19 X[:, 0] = labelencoder_X.transform(X[:, 0])
20
21 labelencoder_y = LabelEncoder()
22 y = labelencoder_y.fit_transform(y)
23
24 from sklearn.model_selection import train_test_split
25 X_train,X_test,y_train,y_test = train_test_split(X, y, test_size=0.2, random_state=0)
```

A variable browser window is open, showing the 'dataset' DataFrame with columns State, Age, and Salary. The IPython console at the bottom shows the command for splitting the data into training and testing sets.

File Edit Search Source Run Debug Consoles Projects Tools View Help

C:\Users\yogay\spyder-py3\temp.py

```

1 import numpy as np
2 import matplotlib.pyplot as plt
3 import pandas as pd
4
5 dataset=pd.read_csv(r'C:\Users\yogay\yoga.csv')
6
7 X = dataset.iloc[:, :-1].values
8 y = dataset.iloc[:, 3].values
9
10 from sklearn.impute import SimpleImputer
11 imputer = SimpleImputer(missing_values=np.nan)
12
13 imputer = imputer.fit(X[:,1:3])
14 X[:, 1:3] = imputer.transform(X[:,1:3])
15
16 from sklearn.preprocessing import LabelEncoder
17 labelencoder_X = LabelEncoder()
18 labelencoder_X.fit_transform(X[:,0])
19 X[:,0] = labelencoder_X.transform(X[:,0])
20
21 labelencoder_y = LabelEncoder()
22 y = labelencoder_y.fit_transform(y)
23
24 from sklearn.model_selection import train_test_split
25 X_train,X_test,y_train,y_test = train_test_split(X,y,test_size=0.2,random_state=0)
26

```

X_train - NumPy object array (read only)

	0	1	2
0	1	40.0	61000.0
1	2	37.0	67000.0
2	0	27.0	48000.0
3	0	38.0	52000.0
4	2	48.0	79000.0
5	0	38.0	61000.0
6	2	44.0	72000.0
7	2	35.0	58000.0

Format Resize Background color Close In [19]: X_train,X_test,y_train,y_test = train_test_split(X,y,test_size=0.2,random_state=0)

In [20]:

IPython Console History

conda: base (Python 3.10.9) Completions: conda(base) LSP: Python Line 26, Col 1 UTF-8 CRLF RW Mem 91% 20°C ENG 22:03

Spyder (Python 3.10)

File Edit Search Source Run Debug Consoles Projects Tools View Help

C:\Users\yogay\spyder-py3\temp.py

temp.py* X

```
1 import numpy as np
2 import matplotlib.pyplot as plt
3 import pandas as pd
4 dataset=pd.read_csv(r'C:\Users\yoga...
5 X = dataset.iloc[:, :-1].values
6 y = dataset.iloc[:, 3].values
7 from sklearn.impute import SimpleImputer
8 imputer = SimpleImputer(missing_val...
9 imputer = imputer.fit(X[:, 1:3])
10 X[:, 1:3] = imputer.transform(X[:, ...
11 from sklearn.preprocessing import L...
12 labelencoder_X = LabelEncoder()
13 labelencoder_X.fit_transform(X[:, 0])
14 X[:, 0] = labelencoder_X.transform(X...
15 labelencoder_y = LabelEncoder()
16 y = labelencoder_y.fit_transform(y)
17 #splitting 80% 20%
18 from sklearn.model_selection import ...
19 X_train,X_test,y_train,y_test = tra...
20 #splitting 70% 30%
21 from sklearn.model_selection import ...
22 X_train,X_test,y_train,y_test = tra...
23
24
25
26
27
28
29
30
```

X_test - NumPy object array (read only)

	0	1	2
0	1	30.0	54000.0
1	1	50.0	83000.0
2	1	40.0	61000.0

imputer impute._base.SimpleImputer 1 SimpleImputer object of sklearn.impute._base module

labelencoder_1 LabelEncoder object of sklearn.p... 1 LabelEncoder object of sklearn.p...

(10, 3) ndarray object of numpy module 1 (10, 3) ndarray object of numpy module

(3, 3) ndarray object of numpy module 1 (3, 3) ndarray object of numpy module

(7, 3) ndarray object of numpy module 1 (7, 3) ndarray object of numpy module

(10,) [0 1 0 0 1 1 0 1 0 1] 1 [0 1 0 0 1 1 0 1 0 1]

explorer Plots Files

In [21]: X_train,X_test,y_train,y_test = train_test_split(X, y, test_size = 0.3,random_state=0)

In [22]:

IPython Console History

conda: base (Python 3.10.9) Completions: conda(base) LSP: Python Line 29, Col 1 UTF-8 CRLF RW Mem 90%

Type here to search

20°C 22:06 ENG

The screenshot shows the Spyder IDE interface with a Python script named `temp.py` open. The code reads a CSV file, performs missing value imputation, and splits the data into training and testing sets.

```
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
dataset=pd.read_csv(r'C:\Users\yogay\yoga.csv')
X = dataset.iloc[:, :-1].values
y = dataset.iloc[:, 3].values
from sklearn.impute import SimpleImputer
imputer = SimpleImputer(missing_values=np.nan)
imputer = imputer.fit(X[:, 1:3])
X[:, 1:3] = imputer.transform(X[:, 1:3])
from sklearn.preprocessing import LabelEncoder
labelencoder_X = LabelEncoder()
labelencoder_X.fit_transform(X[:, 0])
X[:, 0] = labelencoder_X.transform(X[:, 0])
labelencoder_y = LabelEncoder()
y = labelencoder_y.fit_transform(y)
#splitting 80% 20%
from sklearn.model_selection import train_test_split
X_train,X_test,y_train,y_test = train_test_split(X, y, test_size = 0.2, random_state=0)
#splitting 70% 30%
from sklearn.model_selection import train_test_split
X_train,X_test,y_train,y_test = train_test_split(X, y, test_size = 0.3, random_state=0)
```

A variable inspector window is open, showing the `imputer` object as a `SimpleImputer` from `sklearn.impute._base`. The `X_train` variable is displayed as a NumPy array with 10 rows and 3 columns, containing values like 2, 37.0, 67000.0, etc.

	0	1	2
0	2	37.0	67000.0
1	0	27.0	48000.0
2	0	38.0	52000.0
3	2	48.0	79000.0
4	0	38.0	61000.0
5	2	44.0	72000.0
6	2	35.0	58000.0

The IPython console shows the execution of the `train_test_split` command:

```
In [21]: X_train,X_test,y_train,y_test = train_test_split(X, y, test_size = 0.3,random_state=0)
In [22]:
```

The status bar at the bottom indicates the environment is `conda: base (Python 3.10.9)`, with 90% memory usage.

The screenshot shows the Spyder IDE interface with a Python script named 'temp.py' open. The code imports numpy, matplotlib.pyplot, and pandas, reads a dataset from a CSV file, and performs various data transformations using sklearn's SimpleImputer and LabelEncoder.

```
1 import numpy as np
2 import matplotlib.pyplot as plt
3 import pandas as pd
4
5 dataset=pd.read_csv(r'C:\Users\yogay\spyder-py3\temp.csv')
6
7 X = dataset.iloc[:, :-1].values
8 y = dataset.iloc[:, 3].values
9
10 from sklearn.impute import SimpleImputer
11 imputer = SimpleImputer(missing_values=np.nan)
12
13 imputer = imputer.fit(X[:, 1:3])
14 X[:, 1:3] = imputer.transform(X[:, 1:3])
15
16 from sklearn.preprocessing import LabelEncoder
17 labelencoder_X = LabelEncoder()
18 labelencoder_X.fit_transform(X[:, 0])
19 X[:, 0] = labelencoder_X.fit_transform(X[:, 0])
20
21 labelencoder_y = LabelEncoder()
22 y = labelencoder_y.fit_transform(y)
23 #splitting 80% 20%
24 from sklearn.model_selection import train_test_split
25 X_train,X_test,y_train,y_test = train_test_split(X, y, test_size=0.2, random_state=0)
26 #splitting 70% 30%
27 from sklearn.model_selection import train_test_split
28 X_train,X_test,y_train,y_test = train_test_split(X, y, test_size=0.3, random_state=0)
29 #splitting 60% 40%
30 from sklearn.model_selection import train_test_split
31 X_train,X_test,y_train,y_test = train_test_split(X, y, test_size=0.4, random_state=0)
```

A variable inspection window is open, showing the state of 'imputer' as a 'SimpleImputer' object and 'X_test' as a NumPy array. The IPython console at the bottom shows the execution of code to split the dataset into training and testing sets.

Name	Type	Size	Value
imputer	impute._base.SimpleImputer	1	SimpleImputer object of sklearn.impute._base module
X_test	ndarray	(10, 3)	NumPy object array (read only)
labelencoder_X	LabelEncoder	1	LabelEncoder object of sklearn.preprocessing
labelencoder_y	LabelEncoder	1	LabelEncoder object of sklearn.preprocessing

	0	1	2
0	1	30.0	54000.0
1	1	50.0	83000.0
2	1	40.0	61000.0
3	2	37.0	67000.0

```
In [23]: X_train,X_test,y_train,y_test = train_test_split(X, y, test_size = 0.4,random_state=0)
In [24]:
```

Spyder (Python 3.10)

File Edit Search Source Run Debug Consoles Projects Tools View Help

C:\Users\yogay\spyder-py3\temp.py

```

1 import numpy as np
2 import matplotlib.pyplot as plt
3 import pandas as pd
4
5 dataset=pd.read_csv(r'C:\Users\yogay\yoga.csv')
6
7 X = dataset.iloc[:, :-1].values
8 y = dataset.iloc[:, 3].values
9
10 from sklearn.impute import SimpleImputer
11 imputer = SimpleImputer(missing_valu
12
13 imputer = imputer.fit(X[:,1:3])
14 X[:, 1:3] = imputer.transform(X[:,1
15
16 from sklearn.preprocessing import L
17 labelencoder_X = LabelEncoder()
18 labelencoder_X.fit_transform(X[:,0]
19 X[:,0] = labelencoder_X.transform(
20
21 labelencoder_y = LabelEncoder()
22 y = labelencoder_y.fit_transform(y)
23 #splitting 80% 20%
24 from sklearn.model_selection import
25 X_train,X_test,y_train,y_test = tra
26 #splitting 70% 30%
27 from sklearn.model_selection import
28 X_train,X_test,y_train,y_test = tra
29 #splitting 60% 40%
30 from sklearn.model_selection import train_test_split
31 X_train,X_test,y_train,y_test = train_test_split(X, y, test_size = 0.4,random_
32

```

X_train - NumPy object array (read only)

	0	1	2
0	0	27.0	48000.0
1	0	38.0	52000.0
2	2	48.0	79000.0
3	0	38.0	61000.0
4	2	44.0	72000.0
5	2	35.0	58000.0

imputer impute._base.SimpleImputer 1

labelencoder_X LabelEncoder 1

(10, 3) ndarray object of numpy module

(4, 3) ndarray object of numpy module

(6, 3) ndarray object of numpy module

(10,) [0 1 0 0 1 1 0 1 0 1]

In [23]: X_train,X_test,y_train,y_test = train_test_split(X, y, test_size = 0.4,random_
In [24]:

22:08 20°C ENG 31-10-2023

File Edit Search Source Run Debug Consoles Projects Tools View Help

C:\Users\yogay\spyder-py3\temp.py

```

5 dataset=pd.read_csv(r'C:\Users\yogay\OneDrive\Desktop\Yogita_Yadav\Data Scie
6 X = dataset.iloc[:, :-1].values
7 y = dataset.iloc[:, 3].values
8
9 from sklearn.impute import SimpleImputer
10 imputer = SimpleImputer(missing_val
11 imputer = imputer.fit(X[:, 1:3])
12 X[:, 1:3] = imputer.transform(X[:, 1:3])
13
14 from sklearn.preprocessing import LabelEncoder
15 labelencoder_X = LabelEncoder()
16 labelencoder_X.fit_transform(X[:, 0])
17 X[:, 0] = labelencoder_X.fit_transform(X[:, 0])
18
19 labelencoder_y = LabelEncoder()
20 y = labelencoder_y.fit_transform(y)
21 #splitting 80% 20%
22 from sklearn.model_selection import train_test_split
23 X_train,X_test,y_train,y_test = train_test_split(X, y, test_size = 0.2,random_state=0)
24 #splitting 70% 30%
25 from sklearn.model_selection import train_test_split
26 X_train,X_test,y_train,y_test = train_test_split(X, y, test_size = 0.3,random_state=0)
27 #splitting 60% 40%
28 from sklearn.model_selection import train_test_split
29 X_train,X_test,y_train,y_test = train_test_split(X, y, test_size = 0.4,random_state=0)
30
31 X_train,X_test,y_train,y_test = train_test_split(X, y, test_size = 0.25,random_state=0)
32 #splitting 75% 25%
33 from sklearn.model_selection import train_test_split
34 X_train,X_test,y_train,y_test = train_test_split(X, y, test_size = 0.25,random_state=0)
35
36

```

X - NumPy object array (read only)

	0	1	2
0	1	30.0	54000.0
1	1	50.0	83000.0
2	1	40.0	61000.0

imputer

Name	Type	Size	Value
imputer	impute._base.SimpleImputer	1	SimpleImputer object of sklearn.impute._base module
labelencoder_X	LabelEncoder object of sklearn.preprocessing	1	LabelEncoder object of sklearn.preprocessing
X	ndarray object of numpy module	(10, 3)	[10, 3]
y	ndarray object of numpy module	(3, 3)	(3, 3)
labelencoder_y	ndarray object of numpy module	(7, 3)	[7, 3]
train	ndarray object of numpy module	(10,)	[0 1 0 0 1 1 0 1 0 1]

In [25]: X_train,X_test,y_train,y_test = train_test_split(X, y, test_size = 0.25,random_state=0)

In [26]:

IPython Console History

conda: base (Python 3.10.9) ✓ Completions: conda(base) ✓ LSP: Python Line 35, Col 1 UTF-8 CRLF RW Mem 90%

20°C 22:11 ENG 21-10-2023

Spyder (Python 3.10)

File Edit Search Source Run Debug Consoles Projects Tools View Help

C:\Users\yogay\spyder-py3\temp.py

```

5 dataset=pd.read_csv(r'C:\Users\yogay\OneDrive\Desktop\Yogita_Yadav\Data_Scie
6
7 X = dataset.iloc[:, :-1].values
8 y = dataset.iloc[:,3].values
9
10 from sklearn.impute import SimpleImputer
11 imputer = SimpleImputer(missing_values=np.nan, strategy="median")
12
13 imputer = imputer.fit(X[:,1:3])
14 X[:, 1:3] = imputer.transform(X[:,1:3])
15
16 from sklearn.preprocessing import LabelEncoder
17 labelencoder_X = LabelEncoder()
18 labelencoder_X.fit_transform(X[:,0])
19 X[:,0] = labelencoder_X.fit_transform(X[:,0])
20
21 labelencoder_y = LabelEncoder()
22 y = labelencoder_y.fit_transform(y)
23 #splitting 80% 20%
24 from sklearn.model_selection import train_test_split
25 X_train,X_test,y_train,y_test = train_test_split(X, y, test_size = 0.2, i
26 #splitting 70% 30%
27 from sklearn.model_selection import train_test_split
28 X_train,X_test,y_train,y_test = train_test_split(X, y, test_size = 0.3,i
29 #splitting 60% 40%
30 from sklearn.model_selection import train_test_split
31 X_train,X_test,y_train,y_test = train_test_split(X, y, test_size = 0.4,i
32 #splitting 75% 25%
33 from sklearn.model_selection import train_test_split
34 X_train,X_test,y_train,y_test = train_test_split(X, y, test_size = 0.25.
35
36

```

C:\Users\yogay

Name	Type	Size	Value
imputer	impute._base.SimpleImputer	1	SimpleImputer object of sklearn.impute._base module
labelencoder_X	preprocessing._label.LabelEncoder	1	LabelEncoder object of sklearn.p...
labelencoder_y	preprocessing._label.LabelEncoder	1	LabelEncoder object of sklearn.p...
X	Array of object	(10, 3)	ndarray object of numpy module
X_test	Array of object	(3, 3)	ndarray object of numpy module

X_train - NumPy object array (read only)

	0	1	2
0	2	37.0	67000.0
1	0	27.0	48000.0
2	0	38.0	52000.0
3	2	48.0	79000.0
4	0	38.0	61000.0
5	2	44.0	72000.0
6	2	35.0	58000.0

Type here to search

Format Resize Background color Close

20°C 22:11 ENG 31-10-2023

The screenshot shows the Spyder Python IDE interface. The top menu bar includes File, Edit, Search, Source, Run, Debug, Consoles, Projects, Tools, View, and Help. The main area has tabs for temp.py*, IPython Console, and History. The IPython Console tab is active, showing the command `In [27]: X_train,X_test,y_train,y_test = train_test_split(X, y, test_size = 0.2)`. The History tab shows the command `In [28]:`. The code editor on the left contains Python code for data preprocessing and splitting. A variable viewer window is open on the right, showing the state of the variable `labelencoder_y` as a `LabelEncoder object of sklearn.p...` with the following details:

Name	Type	Size	Value
labelencoder_y	preprocessing._label.LabelEncoder	1	LabelEncoder object of sklearn.p...
		(10, 3)	ndarray object of numpy module
		(2, 3)	ndarray object of numpy module
		(8, 3)	ndarray object of numpy module
		(10,)	[0 1 0 0 1 1 0 1 0 1]
		(2,)	[1 0]
		(8,)	[0 1 1 0 0 0 1 1]

Spyder (Python 3.10)

File Edit Search Source Run Debug Consoles Projects Tools View Help

C:\Users\yogay\spyder-py3\temp.py

```

7 X = dataset.iloc[:, :-1].values
8 y = dataset.iloc[:, 3].values
9
10 from sklearn.impute import SimpleImputer
11 imputer = SimpleImputer(missing_val...
12
13 imputer = imputer.fit(X[:, 1:3])
14 X[:, 1:3] = imputer.transform(X[:, 1:3])
15
16 from sklearn.preprocessing import LabelEncoder
17 labelencoder_X = LabelEncoder()
18 labelencoder_X.fit_transform(X[:, 0])
19 X[:, 0] = labelencoder_X.transform(X[:, 0])
20
21 labelencoder_y = LabelEncoder()
22 y = labelencoder_y.fit_transform(y)
23 #splitting 80% 20%
24 from sklearn.model_selection import train_test_split
25 X_train, X_test, y_train, y_test = train...
26 #splitting 70% 30%
27 from sklearn.model_selection import train...
28 X_train, X_test, y_train, y_test = train...
29 #splitting 60% 40%
30 from sklearn.model_selection import train...
31 X_train, X_test, y_train, y_test = train...
32 #splitting 75% 25%
33 from sklearn.model_selection import train...
34 X_train, X_test, y_train, y_test = train...
35
36 #set random state=0
37 from sklearn.model_selection import train...
38 X_train, X_test, y_train, y_test = train...
39

```

labelencoder_y

Name	Type	Size	Value
labelencoder_y	preprocessing._label.LabelEncoder	1	LabelEncoder object of sklearn.p...
		(10, 3)	ndarray object of numpy module
		(2, 3)	ndarray object of numpy module
		(8, 3)	ndarray object of numpy module
		(10,)	[0 1 0 0 1 1 0 1 0 1]
		(2,)	[1 0]
		(8,)	[0 1 1 0 0 0 1 1]

0 0
1 1
2 1
3 0
4 0
5 0
6 1
7 1

Format Resize Background color Save and Close Close

In [27]: X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.2)

In [28]:

IPython Console History

conda: base (Python 3.10.9) Completions: conda(base) LSP: Python Line 39, Col 1 UTF-8 CRLF RW Mem 91%

22:14 31-10-2023

X_test - NumPy object array (read o...)

	0	1	2
0	2	35.0	58000.0
1	1	30.0	54000.0

Format Resize Background color Close

```

17 labelencoder_X = LabelEncoder()
18 labelencoder_X.fit_transform(X[:,0])
19 X[:,0] = labelencoder_X.fit_transform(X[:,0])
20
21 labelencoder_y = LabelEncoder()
22 y = labelencoder_y.fit_transform(y)
23 #splitting 80% 20%
24 from sklearn.model_selection import train_test_split
25 X_train,X_test,y_train,y_test = train_test_split(X,y,test_size=0.2)
26 #splitting 70% 30%
27 from sklearn.model_selection import train_test_split
28 X_train,X_test,y_train,y_test = train_test_split(X,y,test_size=0.3)
29 #splitting 60% 40%
30 from sklearn.model_selection import train_test_split
31 X_train,X_test,y_train,y_test = train_test_split(X,y,test_size=0.4,random_state=0.4,random_state=0.4)
32 #splitting 75% 25%
33 from sklearn.model_selection import train_test_split
34 X_train,X_test,y_train,y_test = train_test_split(X,y,test_size=0.25,random_state=0.25,random_state=0.25)
35
36 #set random state!=0
37 from sklearn.model_selection import train_test_split
38 X_train,X_test,y_train,y_test = train_test_split(X,y,test_size=0.2)
39
40
41

```

X_train - NumPy object array (read o...)

	0	1	2
0	0	38.0	61000.0
1	2	37.0	67000.0
2	1	40.0	61000.0
3	2	44.0	72000.0
4	1	50.0	83000.0
5	0	38.0	52000.0
6	2	48.0	79000.0
7	0	27.0	48000.0

Format Resize Background color Close

In [24]: from

In [25]: X_tr

In [26]: from

In [27]: X_tr

In [28]:

preprocessing

Type	Size	Value
Array of object	1	y_test - NumPy object array
Array of object	1	y_train - NumPy object array
Array of object	1	y - NumPy object array
Array of int32	1	0
Array of int32	1	1

C:\Users\yogay

y_train - NumPy object array

	0
0	0
1	1
2	1
3	0
4	0
5	0
6	1
7	1

Format Resize Background color Close

conda: base (Python)

Type here to search

20°C 22:18 ENG 24.10.2023

The figure shows a Spyder IDE interface with four windows open:

- X_test - NumPy object array (read only)**: A table showing data for X_test. The columns are labeled 0, 1, and 2. The rows are labeled 0 and 1. The data is:

	0	1	2
0	1	30.0	54000.0
1	1	50.0	83000.0
- X_train - NumPy object array (read only)**: A table showing data for X_train. The columns are labeled 0, 1, and 2. The rows are labeled 0 through 7. The data is:

	0	1	2
0	1	40.0	61000.0
1	2	37.0	67000.0
2	0	27.0	48000.0
3	0	38.0	52000.0
4	2	48.0	79000.0
5	0	38.0	61000.0
6	2	44.0	72000.0
7	2	35.0	58000.0
- C:\Users\yogay**: A file browser window showing the path C:\Users\yogay.
- y_train - NumPy object array**: A table showing data for y_train. The columns are labeled 0. The rows are labeled 0 through 7. The data is:

	0
0	1
1	1
2	1
3	0
4	1
5	0
6	0
7	1

A code editor window is also visible, containing the following Python code:

```
19 X[:,0] = labelencoder_X.fit_transform(X[:,0])
20
21 labelencoder_y = LabelEncoder()
22 y = labelencoder_y.fit_transform(y)
23 #splitting 80% 20%
24 from sklearn.model_selection import train_
25 #set random state=0
26 from sklearn.model_selection import train_
27 X_train,X_test,y_train,y_test = train_
28 test_size = 0.2,random_
29 state=0
30
31
32
33
34
35
36
37
38
39
40
41
42
43
```

Below the code editor, a notebook cell displays the following output:

```
In [24]: from sklearn.model_selection import train_
In [25]: X_train,X_test,y_train,y_
test_size = 0.25,random_
state=0
In [26]: from sklearn.model_selection import train_
In [27]: X_train,X_test,y_train,y_
test_size = 0.25,random_
state=0
In [28]: from sklearn.model_selection import train_
In [29]: X_train,X_test,y_train,y_
test_size = 0.2,random_
state=0
In [30]:
```