Spyder (Python 3.8)

File Edit Search Source Run Debug Consoles Projects Tools View Help

D:\AI Programs

D:\AI Programs\training.py

perceptron.py | training.py

```
1  import numpy as np
2  from perceptron import Perceptron
3
4  training_inputs=[]
5  training_inputs.append(np.array([0,0,0]))
6  training_inputs.append(np.array([0,0,1]))
7  training_inputs.append(np.array([0,1,0]))
8  training_inputs.append(np.array([0,1,1]))
9  training_inputs.append(np.array([1,1,0]))
10 training_inputs.append(np.array([1,1,1]))
11
12 lables=np.array([0,0,0,0,0,1])
13 perceptron=Perceptron(3)
14 perceptron.train(training_inputs,lables)
15 inputs= np.array([1,0,0])
16 print(perceptron.pridict(inputs))
17
```

D:\AI Programs\perceptron.py

perceptron.py | training.py

```
2
3  class Perceptron(object):
4      def __init__(self,no_of_inputs,threshold=100,learning_rate=0.0001):
5          self.threshold=threshold
6          self.learning_rate=learning_rate
7          self.weights=np.zeros(no_of_inputs+1)
8
9      def pridict(self,inputs):
10         summation=np.dot(inputs,self.weights[1:])+self.weights[0]   #'''w.x=b'''
11         if summation>0:
12             activation=1
13         else:
14             activation=0
15         return activation
16     def train(self,training_inputs,lables):
17         for _ in range(self.threshold):
18             for inputs,lable in zip(training_inputs,lables):
19                 pridiction = self.pridict(inputs)
20                 self.weights[1:]+=self.learning_rate*(lable-pridiction)*inputs
21                 self.weights[0]+=self.learning_rate*(lable-pridiction)
22
```

Source Console Object

Usage

Here you can get help of any object by pressing **Ctrl+I** in front of it, either on the Editor or the Console.

Help can also be shown automatically after writing a left parenthesis next to an object. You can activate this behavior in *Preferences > Help*.

New to Spyder? Read our tutorial

Variable explorer Help Plots Files

Console 5/A

```
Python 3.8.3 (default, Jul  2 2020, 17:30:36) [MSC v.1916 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.

IPython 7.16.1 -- An enhanced Interactive Python.

In [1]: runfile('D:/AI Programs/training.py', wdir='D:/AI Programs')
0

In [2]:
```

IPython console History

LSP Python: ready | Kite: ready | conda: base (Python 3.8.3) | Line 15, Col 24 | ASCII | CRLF | RW | Mem 61%

Type here to search

ENG 19:21 09-08-2020

Spyder (Python 3.8)

File Edit Search Source Run Debug Consoles Projects Tools View Help

D:\AI Programs

D:\AI Programs\training.py

perceptron.py | training.py

```
1  import numpy as np
2  from perceptron import Perceptron
3
4  training_inputs=[]
5  training_inputs.append(np.array([0,0,0]))
6  training_inputs.append(np.array([0,0,1]))
7  training_inputs.append(np.array([0,1,0]))
8  training_inputs.append(np.array([0,1,1]))
9  training_inputs.append(np.array([1,1,0]))
10 training_inputs.append(np.array([1,1,1]))
11
12 lables=np.array([0,0,0,0,0,1])
13 perceptron=Perceptron(3)
14 perceptron.train(training_inputs,lables)
15 inputs= np.array([1,0,1])
16 print(perceptron.pridict(inputs))
17
```

D:\AI Programs\perceptron.py

perceptron.py | training.py

```
2
3  class Perceptron(object):
4      def __init__(self,no_of_inputs,threshold=1000,learning_rate=1):
5          self.threshold=threshold
6          self.learning_rate=learning_rate
7          self.weights=np.zeros(no_of_inputs+1)
8
9      def pridict(self,inputs):
10         summation=np.dot(inputs,self.weights[1:])+self.weights[0]   #'''w.x=b'''
11         if summation>0:
12             activation=1
13         else:
14             activation=0
15         return activation
16     def train(self,training_inputs,lables):
17         for _ in range(self.threshold):
18             for inputs,lable in zip(training_inputs,lables):
19                 pridiction = self.pridict(inputs)
20                 self.weights[1:]+=self.learning_rate*(lable-pridiction)*inputs
21                 self.weights[0]+=self.learning_rate*(lable-pridiction)
22
```

Source Console Object

Usage

Here you can get help of any object by pressing **Ctrl+I** in front of it, either on the Editor or the Console.

Help can also be shown automatically after writing a left parenthesis next to an object. You can activate this behavior in *Preferences > Help*.

New to Spyder? Read our tutorial

Variable explorer Help Plots Files

Console 5/A

```
In [18]: runfile('D:/AI Programs/training.py', wdir='D:/AI Programs')
Reloaded modules: perceptron
0

In [19]: runfile('D:/AI Programs/training.py', wdir='D:/AI Programs')
Reloaded modules: perceptron
0

In [20]:
```

IPython console History

LSP Python: ready | Kite: ready | conda: base (Python 3.8.3) | Line 15, Col 24 | ASCII | CRLF | RW | Mem 58%

Type here to search

ENG 19:24 09-08-2020