

▼ iris_cnn

```
%tensorflow_version 1.x
```



TensorFlow 1.x selected.

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
```

```
import tensorflow as tf
```

```
from tensorflow import keras
from tensorflow.keras import optimizers
# from tensorflow.keras.layers import Dense
from tensorflow.keras.layers import Dense, Conv1D, Flatten
```

```
!wget https://raw.githubusercontent.com/dhrim/wiset_2020_06/master/material/deep_learning/iris_with
```



```
--2020-06-22 08:24:17-- https://raw.githubusercontent.com/dhrim/wiset_2020_06/master/material/deep_learning/iris_with
Resolving raw.githubusercontent.com (raw.githubusercontent.com)... 151.101.0.133, 151.101.64.
Connecting to raw.githubusercontent.com (raw.githubusercontent.com)|151.101.0.133|:443... con
HTTP request sent, awaiting response... 200 OK
Length: 2218 (2.2K) [text/plain]
Saving to: 'iris_with_category_index.csv'
```

```
iris_with_category_ 100%[=====>] 2.17K --.-KB/s in 0s
```

```
2020-06-22 08:24:17 (11.9 MB/s) - 'iris_with_category_index.csv' saved [2218/2218]
```

```
iris = pd.read_csv("iris_with_category_index.csv")
iris.head()
```



	sepal_length	sepal_width	petal_length	petal_width	class
0	6.4	2.8	5.6	2.2	2
1	5.0	2.3	3.3	1.0	1
2	4.9	2.5	4.5	1.7	2
3	4.9	3.1	1.5	0.1	0
4	5.7	3.8	1.7	0.3	0

```
data = iris.to_numpy()
print(data.shape)
print(data[:5])
```

```
(120, 5)
[[6.4 2.8 5.6 2.2 2. ]
 [5.  2.3 3.3 1.  1. ]
 [4.9 2.5 4.5 1.7 2. ]
 [4.9 3.1 1.5 0.1 0. ]
 [5.7 3.8 1.7 0.3 0. ]]
```

```
x = data[:, :4]
y = data[:, 4:]
```

```
split_index = 100
```

```
train_x, test_x = x[:split_index], x[split_index:]
train_y, test_y = y[:split_index], y[split_index:]
```

```
print(train_x.shape)
print(train_y.shape)
print(test_x.shape)
print(test_y.shape)
```

```
train_x = np.reshape(train_x, (100, 4, 1)) # ADD
test_x = np.reshape(test_x, (20, 4, 1))    # ADD
```

```
print(train_x.shape) # ADD
print(test_x.shape)  # ADD
```

```
(100, 4)
(100, 1)
(20, 4)
(20, 1)
(100, 4, 1)
(20, 4, 1)
```

```
model = keras.Sequential()
model.add(Conv1D(4, 3, activation='relu', input_shape=(4,1))) # ADD activation='relu' 는 1D 이므로
model.add(Flatten()) # ADD
# model.add(Dense(10, activation='relu', input_shape=(4,)))
model.add(Dense(10, activation='relu')) # modify
model.add(Dense(10, activation='relu'))
model.add(Dense(3, activation="softmax")) # 10이 아니고 클래스 수 3이다
```

```
model.compile(optimizer="SGD", loss="sparse_categorical_crossentropy", metrics=["accuracy"])
model.summary()
```

```
model.fit(train_x, train_y, epochs=1000, verbose=0, batch_size=20)
```

```
loss, acc = model.evaluate(test_x, test_y)
print("loss=", loss)
print("acc=", acc)
```



WARNING:tensorflow:From /tensorflow-1.15.2/python3.6/tensorflow_core/python/ops/resource_variable_ops.py:169: tf.nn.conv2d is deprecated and will be removed in a future version. Instructions for updating:
If using Keras pass *_constraint arguments to layers.
Model: "sequential"

Layer (type)	Output Shape	Param #
conv1d (Conv1D)	(None, 2, 4)	16
flatten (Flatten)	(None, 8)	0
dense (Dense)	(None, 10)	90
dense_1 (Dense)	(None, 10)	110
dense_2 (Dense)	(None, 3)	33

Total params: 249

Trainable params: 249

Non-trainable params: 0

20/20 [=====] - 0s 2ms/sample - loss: 0.0560 - acc: 0.9500

```
y_ = model.predict(test_x)
print(y_)
print(np.argmax(y_, axis=1))
```



```
[[9.9922991e-01 7.7004876e-04 4.4361993e-11]
 [5.3430619e-03 9.9458319e-01 7.3718111e-05]
 [9.9801922e-01 1.9807674e-03 8.5626617e-10]
 [2.7312705e-04 9.9965060e-01 7.6270779e-05]
 [9.9908340e-01 9.1657741e-04 4.9820398e-11]
 [9.9956065e-01 4.3929275e-04 6.3868993e-12]
 [9.9960297e-01 3.9709502e-04 3.7614178e-12]
 [9.9945611e-01 5.4383784e-04 1.0488147e-11]
 [7.8068785e-03 9.9216652e-01 2.6653775e-05]
 [9.9838138e-01 1.6185682e-03 1.2488670e-10]
 [2.6257758e-05 6.3346744e-01 3.6650631e-01]
 [3.6416494e-04 9.9909377e-01 5.4204097e-04]
 [9.9885201e-01 1.1479990e-03 6.4390854e-11]
 [1.1358751e-06 6.6246629e-02 9.3375224e-01]
 [9.9919754e-01 8.0242421e-04 6.4661040e-11]
 [5.9844519e-05 9.8063570e-01 1.9304480e-02]
 [2.0107500e-04 9.9953341e-01 2.6544381e-04]
 [9.9810779e-01 1.8921619e-03 4.0259540e-10]
 [9.9907589e-01 9.2409586e-04 5.2082030e-11]
 [1.3248803e-03 9.9862206e-01 5.3106560e-05]]
[0 1 0 1 0 0 0 0 1 0 1 1 0 2 0 1 1 0 0 1]
```

