File Edit View Run Kernel Tabs Settings Help

connecting_neurons.ipynb

Markdown

Python 3

Instructor Do: Connecting Neurons

[1]:
```
# Initial imports
import pandas as pd
import numpy as np
import hvplot.pandas
from sklearn.datasets.samples_generator import make_moons
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense
```

Creating Dummy Data

[2]:
```
# Creating dummy non-linear data
X_moons, y_moons = make_moons(n_samples=1000, noise=0.08, random_state=78)
```

[3]:
```
# Transforming y_moons to a vertical vector
y_moons = y_moons.reshape(-1, 1)
```

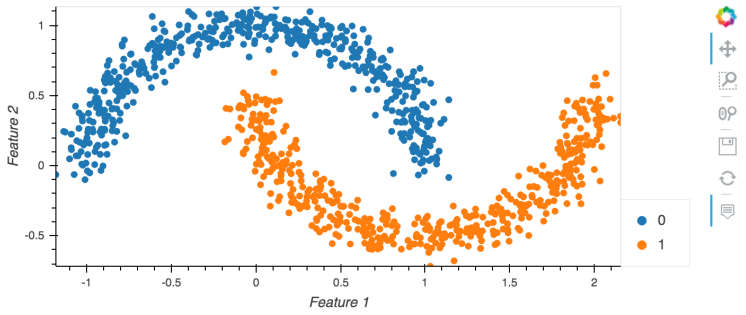
[4]:
```
# Creating a DataFrame to plot the non-linear dummy data
df_moons = pd.DataFrame(X_moons, columns=["Feature 1", "Feature 2"])
df_moons["Target"] = y_moons
df_moons.head()
```

[4]:

	Feature 1	Feature 2	Target
0	0.676217	0.815220	0
1	1.487143	-0.210519	1
2	-1.027709	0.354966	0
3	0.236012	1.025531	0
4	1.856695	-0.042783	1

[5]:
```
# Plotting the non-linear dummy data
df_moons.hvplot.scatter(x="Feature 1", y="Feature 2", by="Target")
```

[5]:



Data Preprocessing

[6]:
```
# Create training and testing sets
X_moon_train, X_moon_test, y_moon_train, y_moon_test = train_test_split(
    X_moons, y_moons, random_state=78
)
```

[7]:
```
# Create the scaler instance
X_moon_scaler = StandardScaler()
```

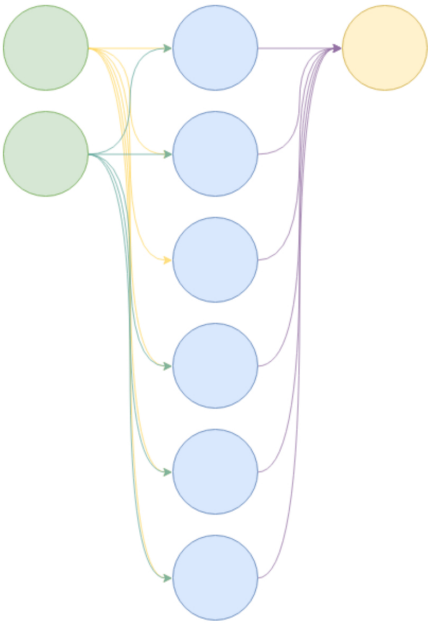
[8]:
```
# Fit the scaler
X_moon_scaler.fit(X_moon_train)
```

[8]:
```
StandardScaler(copy=True, with_mean=True, with_std=True)
```

[9]:
```
# Scale the data
X_moon_train_scaled = X_moon_scaler.transform(X_moon_train)
X_moon_test_scaled = X_moon_scaler.transform(X_moon_test)
```

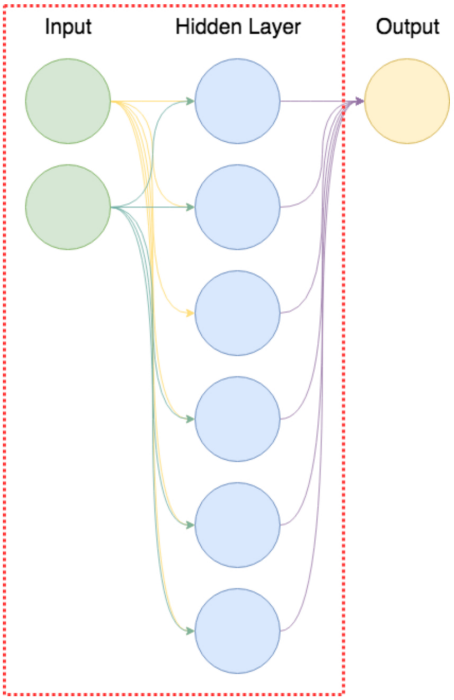
Neural Network Model Creation

Input Hidden Layer Output



[10]:
```
# Create the sequential model
nn = Sequential()
```

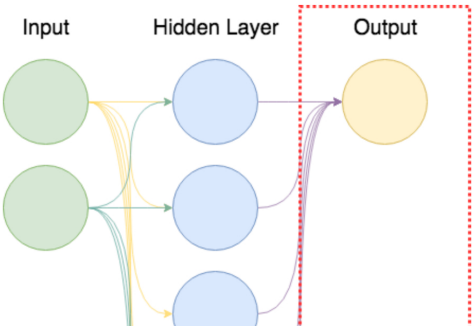
Fisrt Layer

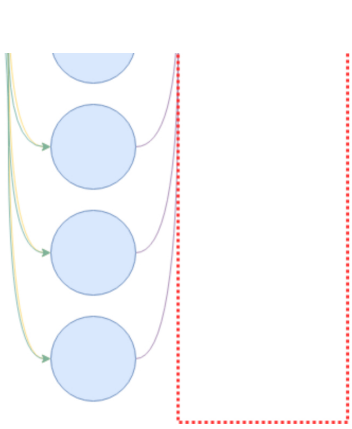


[11]:
```
# First layer
number_inputs = 2
number_hidden_nodes = 6

nn.add(Dense(units=number_hidden_nodes, activation="relu", input_dim=number_inputs))
```

Output Layer





[12]:
```
# Output layer
number_classes = 1

nn.add(Dense(units=number_classes, activation="sigmoid"))
```

[13]:
```
# Model summary
nn.summary()
```

```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
dense (Dense)                (None, 6)                 18
_________________________________________________________________
dense_1 (Dense)              (None, 1)                 7
=================================================================
Total params: 25
Trainable params: 25
Non-trainable params: 0
_________________________________________________________________
```

Compile the Model

[14]:
```
# Compile model
nn.compile(loss="binary_crossentropy", optimizer="adam", metrics=["accuracy"])
```

Fit the Model

[15]:
```
# Training the model with the non-linear data
model_moon = nn.fit(X_moon_train_scaled, y_moon_train, epochs=100)
```

```
Train on 750 samples
Epoch 1/100
750/750 [==============================] - 1s 887us/sample - loss: 0.4574 - accuracy: 0.8747
Epoch 2/100
750/750 [==============================] - 0s 38us/sample - loss: 0.4341 - accuracy: 0.8773
Epoch 3/100
750/750 [==============================] - 0s 38us/sample - loss: 0.4125 - accuracy: 0.8773
Epoch 4/100
750/750 [==============================] - 0s 39us/sample - loss: 0.3928 - accuracy: 0.8787
Epoch 5/100
750/750 [==============================] - 0s 37us/sample - loss: 0.3753 - accuracy: 0.8773
Epoch 6/100
750/750 [==============================] - 0s 41us/sample - loss: 0.3596 - accuracy: 0.8760
Epoch 7/100
750/750 [==============================] - 0s 38us/sample - loss: 0.3460 - accuracy: 0.8773
Epoch 8/100
750/750 [==============================] - 0s 38us/sample - loss: 0.3334 - accuracy: 0.8773
Epoch 9/100
750/750 [==============================] - 0s 37us/sample - loss: 0.3225 - accuracy: 0.8773
Epoch 10/100
750/750 [==============================] - 0s 38us/sample - loss: 0.3130 - accuracy: 0.8787
Epoch 11/100
750/750 [==============================] - 0s 38us/sample - loss: 0.3046 - accuracy: 0.8787
Epoch 12/100
750/750 [==============================] - 0s 36us/sample - loss: 0.2973 - accuracy: 0.8787
Epoch 13/100
750/750 [==============================] - 0s 33us/sample - loss: 0.2909 - accuracy: 0.8800
Epoch 14/100
750/750 [==============================] - 0s 41us/sample - loss: 0.2853 - accuracy: 0.8800
Epoch 15/100
750/750 [==============================] - 0s 38us/sample - loss: 0.2803 - accuracy: 0.8800
Epoch 16/100
750/750 [==============================] - 0s 36us/sample - loss: 0.2758 - accuracy: 0.8800
Epoch 17/100
750/750 [==============================] - 0s 37us/sample - loss: 0.2718 - accuracy: 0.8800
Epoch 18/100
750/750 [==============================] - 0s 41us/sample - loss: 0.2685 - accuracy: 0.8813
Epoch 19/100
750/750 [==============================] - 0s 37us/sample - loss: 0.2654 - accuracy: 0.8813
Epoch 20/100
750/750 [==============================] - 0s 37us/sample - loss: 0.2623 - accuracy: 0.8813
Epoch 21/100
750/750 [==============================] - 0s 35us/sample - loss: 0.2597 - accuracy: 0.8827
Epoch 22/100
750/750 [==============================] - 0s 40us/sample - loss: 0.2573 - accuracy: 0.8827
Epoch 23/100
750/750 [==============================] - 0s 36us/sample - loss: 0.2550 - accuracy: 0.8840
Epoch 24/100
750/750 [==============================] - 0s 33us/sample - loss: 0.2530 - accuracy: 0.8840
Epoch 25/100
```

```
750/750 [==============================] - 0s 36us/sample - loss: 0.2510 - accuracy: 0.8853
Epoch 26/100
750/750 [==============================] - 0s 37us/sample - loss: 0.2492 - accuracy: 0.8880
Epoch 27/100
750/750 [==============================] - 0s 41us/sample - loss: 0.2473 - accuracy: 0.8893
Epoch 28/100
750/750 [==============================] - 0s 40us/sample - loss: 0.2456 - accuracy: 0.8893
Epoch 29/100
750/750 [==============================] - 0s 39us/sample - loss: 0.2440 - accuracy: 0.8893
Epoch 30/100
750/750 [==============================] - 0s 38us/sample - loss: 0.2423 - accuracy: 0.8907
Epoch 31/100
750/750 [==============================] - 0s 38us/sample - loss: 0.2408 - accuracy: 0.8907
Epoch 32/100
750/750 [==============================] - 0s 38us/sample - loss: 0.2392 - accuracy: 0.8933
Epoch 33/100
750/750 [==============================] - 0s 38us/sample - loss: 0.2377 - accuracy: 0.8947
Epoch 34/100
750/750 [==============================] - 0s 38us/sample - loss: 0.2363 - accuracy: 0.8960
Epoch 35/100
750/750 [==============================] - 0s 36us/sample - loss: 0.2347 - accuracy: 0.8960
Epoch 36/100
750/750 [==============================] - 0s 36us/sample - loss: 0.2333 - accuracy: 0.8987
Epoch 37/100
750/750 [==============================] - ETA: 0s - loss: 0.2288 - accuracy: 0.90 - 0s 39us/sample - loss: 0.2317 - accuracy: 0.8987
Epoch 38/100
750/750 [==============================] - 0s 39us/sample - loss: 0.2303 - accuracy: 0.8987
Epoch 39/100
750/750 [==============================] - 0s 38us/sample - loss: 0.2288 - accuracy: 0.9000
Epoch 40/100
750/750 [==============================] - 0s 37us/sample - loss: 0.2275 - accuracy: 0.9013
Epoch 41/100
750/750 [==============================] - 0s 35us/sample - loss: 0.2259 - accuracy: 0.9027
Epoch 42/100
750/750 [==============================] - 0s 36us/sample - loss: 0.2244 - accuracy: 0.9040
Epoch 43/100
750/750 [==============================] - 0s 38us/sample - loss: 0.2230 - accuracy: 0.9040
Epoch 44/100
750/750 [==============================] - 0s 37us/sample - loss: 0.2215 - accuracy: 0.9040
Epoch 45/100
750/750 [==============================] - 0s 39us/sample - loss: 0.2201 - accuracy: 0.9040
Epoch 46/100
750/750 [==============================] - 0s 39us/sample - loss: 0.2187 - accuracy: 0.9040
Epoch 47/100
750/750 [==============================] - 0s 37us/sample - loss: 0.2171 - accuracy: 0.9040
Epoch 48/100
750/750 [==============================] - 0s 36us/sample - loss: 0.2156 - accuracy: 0.9067
Epoch 49/100
750/750 [==============================] - 0s 39us/sample - loss: 0.2140 - accuracy: 0.9067
Epoch 50/100
750/750 [==============================] - 0s 37us/sample - loss: 0.2125 - accuracy: 0.9067
Epoch 51/100
750/750 [==============================] - 0s 38us/sample - loss: 0.2110 - accuracy: 0.9080
Epoch 52/100
750/750 [==============================] - 0s 40us/sample - loss: 0.2095 - accuracy: 0.9080
Epoch 53/100
750/750 [==============================] - 0s 38us/sample - loss: 0.2080 - accuracy: 0.9120
Epoch 54/100
750/750 [==============================] - 0s 36us/sample - loss: 0.2063 - accuracy: 0.9120
Epoch 55/100
750/750 [==============================] - 0s 38us/sample - loss: 0.2048 - accuracy: 0.9120
Epoch 56/100
750/750 [==============================] - 0s 38us/sample - loss: 0.2032 - accuracy: 0.9147
Epoch 57/100
750/750 [==============================] - 0s 39us/sample - loss: 0.2016 - accuracy: 0.9147
Epoch 58/100
750/750 [==============================] - 0s 39us/sample - loss: 0.2001 - accuracy: 0.9160
Epoch 59/100
750/750 [==============================] - 0s 37us/sample - loss: 0.1984 - accuracy: 0.9160
Epoch 60/100
750/750 [==============================] - 0s 37us/sample - loss: 0.1968 - accuracy: 0.9187
Epoch 61/100
750/750 [==============================] - 0s 36us/sample - loss: 0.1952 - accuracy: 0.9200
Epoch 62/100
750/750 [==============================] - 0s 38us/sample - loss: 0.1936 - accuracy: 0.9213
Epoch 63/100
750/750 [==============================] - 0s 36us/sample - loss: 0.1920 - accuracy: 0.9227
Epoch 64/100
750/750 [==============================] - 0s 38us/sample - loss: 0.1902 - accuracy: 0.9227
Epoch 65/100
750/750 [==============================] - 0s 38us/sample - loss: 0.1886 - accuracy: 0.9227
Epoch 66/100
750/750 [==============================] - 0s 39us/sample - loss: 0.1870 - accuracy: 0.9227
Epoch 67/100
750/750 [==============================] - 0s 38us/sample - loss: 0.1855 - accuracy: 0.9227
Epoch 68/100
750/750 [==============================] - 0s 36us/sample - loss: 0.1838 - accuracy: 0.9240
Epoch 69/100
750/750 [==============================] - 0s 35us/sample - loss: 0.1821 - accuracy: 0.9267
Epoch 70/100
750/750 [==============================] - 0s 37us/sample - loss: 0.1804 - accuracy: 0.9267
Epoch 71/100
750/750 [==============================] - 0s 37us/sample - loss: 0.1788 - accuracy: 0.9267
Epoch 72/100
750/750 [==============================] - 0s 40us/sample - loss: 0.1772 - accuracy: 0.9280
Epoch 73/100
750/750 [==============================] - 0s 36us/sample - loss: 0.1755 - accuracy: 0.9293
Epoch 74/100
750/750 [==============================] - 0s 36us/sample - loss: 0.1742 - accuracy: 0.9293
Epoch 75/100
750/750 [==============================] - 0s 37us/sample - loss: 0.1722 - accuracy: 0.9307
Epoch 76/100
750/750 [==============================] - 0s 38us/sample - loss: 0.1707 - accuracy: 0.9333
Epoch 77/100
750/750 [==============================] - 0s 39us/sample - loss: 0.1690 - accuracy: 0.9347
Epoch 78/100
750/750 [==============================] - 0s 37us/sample - loss: 0.1672 - accuracy: 0.9347
Epoch 79/100
750/750 [==============================] - 0s 36us/sample - loss: 0.1657 - accuracy: 0.9347
Epoch 80/100
750/750 [==============================] - 0s 34us/sample - loss: 0.1640 - accuracy: 0.9360
Epoch 81/100
```

```
750/750 [==============================] - 0s 34us/sample - loss: 0.1623 - accuracy: 0.9360
Epoch 82/100
750/750 [==============================] - 0s 36us/sample - loss: 0.1607 - accuracy: 0.9360
Epoch 83/100
750/750 [==============================] - 0s 38us/sample - loss: 0.1590 - accuracy: 0.9360
Epoch 84/100
750/750 [==============================] - 0s 38us/sample - loss: 0.1574 - accuracy: 0.9373
Epoch 85/100
750/750 [==============================] - 0s 36us/sample - loss: 0.1557 - accuracy: 0.9400
Epoch 86/100
750/750 [==============================] - 0s 39us/sample - loss: 0.1542 - accuracy: 0.9413
Epoch 87/100
750/750 [==============================] - 0s 36us/sample - loss: 0.1526 - accuracy: 0.9413
Epoch 88/100
750/750 [==============================] - 0s 36us/sample - loss: 0.1509 - accuracy: 0.9427
Epoch 89/100
750/750 [==============================] - 0s 37us/sample - loss: 0.1493 - accuracy: 0.9427
Epoch 90/100
750/750 [==============================] - 0s 38us/sample - loss: 0.1477 - accuracy: 0.9440
Epoch 91/100
750/750 [==============================] - 0s 39us/sample - loss: 0.1461 - accuracy: 0.9440
Epoch 92/100
750/750 [==============================] - 0s 35us/sample - loss: 0.1444 - accuracy: 0.9453
Epoch 93/100
750/750 [==============================] - 0s 36us/sample - loss: 0.1430 - accuracy: 0.9467
Epoch 94/100
750/750 [==============================] - 0s 36us/sample - loss: 0.1412 - accuracy: 0.9467
Epoch 95/100
750/750 [==============================] - 0s 37us/sample - loss: 0.1396 - accuracy: 0.9467
Epoch 96/100
750/750 [==============================] - 0s 40us/sample - loss: 0.1381 - accuracy: 0.9467
Epoch 97/100
750/750 [==============================] - 0s 38us/sample - loss: 0.1365 - accuracy: 0.9467
Epoch 98/100
750/750 [==============================] - 0s 37us/sample - loss: 0.1349 - accuracy: 0.9467
Epoch 99/100
750/750 [==============================] - 0s 37us/sample - loss: 0.1334 - accuracy: 0.9467
Epoch 100/100
750/750 [==============================] - 0s 38us/sample - loss: 0.1319 - accuracy: 0.9493
```

Model Evaluation

[16]:

```
# Evaluate the model using non-linear testing data
model_moon_loss, model_moon_accuracy = nn.evaluate(
    X_moon_test_scaled, y_moon_test, verbose=2
)
print(f"Loss: {model_moon_loss}, Accuracy: {model_moon_accuracy}")
```

```
250/1 - 0s - loss: 0.1242 - accuracy: 0.9440
Loss: 0.13856448686122894, Accuracy: 0.9440000057220459
```