

```

from google.colab import drive
import matplotlib.pyplot as plt
import pandas as pd
from keras.models import Sequential
from keras.layers import Dense, Activation
from keras.optimizers import SGD
import numpy as np

# Montowanie dysku Google Drive
drive.mount('/content/drive')

# Wczytywanie danych z pliku CSV
df = pd.read_csv('/content/drive/MyDrive/xor.csv')

# Wyświetlanie danych
print(df)

# Przygotowanie danych wejściowych (x) i wyjściowych (y)
x = df[['x1', 'x2']].to_numpy()
y = df['y'].to_numpy()

# Wykres punktów dla dwóch klas
plt.plot(x[y == 0][:, 0], x[y == 0][:, 1], 'bo', label='Klasa 0')
plt.plot(x[y == 1][:, 0], x[y == 1][:, 1], 'ro', label='Klasa 1')
plt.legend()
plt.title('Prezentacja umiejscowienia punktów 2D z 2 klasami.')
plt.show()

# Definicja modelu sieci neuronowej
model = Sequential()
model.add(Dense(2, activation='sigmoid', input_dim=2, name="first"))
model.add(Dense(1, activation='sigmoid', name="second"))

# Kompilacja modelu
model.compile(optimizer=SGD(learning_rate=0.1), loss='mse', metrics=['accuracy'])

# Wstępna predykcja i wyświetlenie wyników
pred = np.round(model.predict(x).flatten())
for i in range(len(x)):
    print(x[i], y[i], pred[i])

# Wykres punktów poprawnie i niepoprawnie zaklasyfikowanych
plt.plot(x[y == pred][:, 0], x[y == pred][:, 1], 'go', label='Poprawne')
plt.plot(x[y != pred][:, 0], x[y != pred][:, 1], 'ro', label='Niepoprawne')
plt.legend()
plt.title("Rozpoznanie klas punktów przed nauką")
plt.show()

# Trenowanie modelu
history = model.fit(x, y, epochs=1000, batch_size=2, verbose=0)

# Wykres straty modelu
plt.plot(history.history['loss'])
plt.title('Model loss')
plt.ylabel('Loss')
plt.xlabel('Epoch')
plt.legend(['Train'], loc='upper right')
plt.show()

# Wykres dokładności modelu
plt.plot(history.history['accuracy'])
plt.title('Model accuracy')
plt.ylabel('Accuracy')
plt.xlabel('Epoch')
plt.legend(['Train'], loc='upper right')
plt.show()

# Po nauczaniu modelu
pred2 = np.round(model.predict(x).flatten())
for i in range(len(x)):
    print(x[i], y[i], pred2[i])

# Wykres punktów poprawnie i niepoprawnie zaklasyfikowanych po nauczaniu
plt.plot(x[y == pred2][:, 0], x[y == pred2][:, 1], 'go', label='Poprawne')
plt.plot(x[y != pred2][:, 0], x[y != pred2][:, 1], 'ro', label='Niepoprawne')
plt.legend()

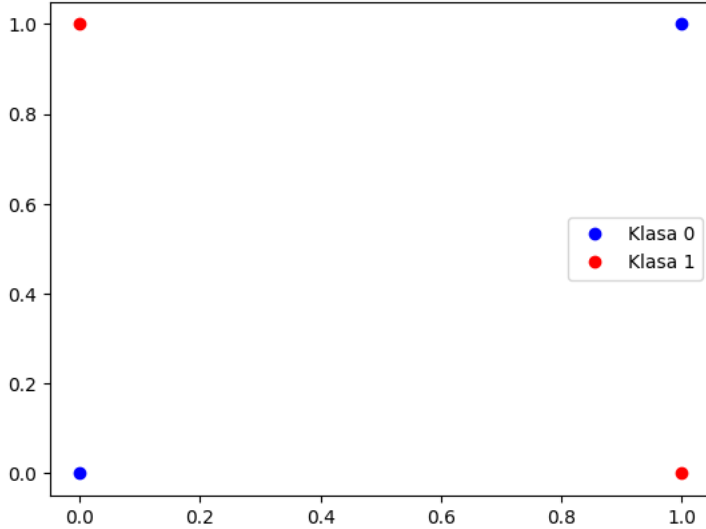
```

```
plt.title("Rozpoznanie klas punktów po nauczaniu")  
plt.show()
```

Drive already mounted at /content/drive; to attempt to forcibly remount, call `drive.mount("/content/drive", force_remount=True)`.

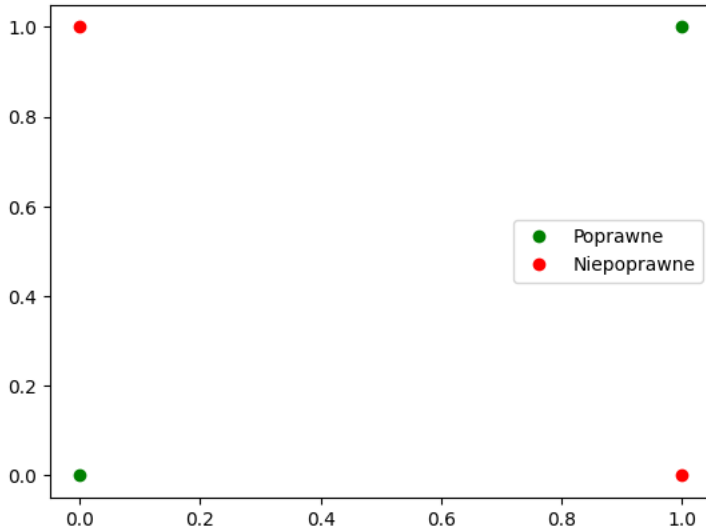
```
x1 x2 y
0 0 0 0
1 0 1 1
2 1 0 1
3 1 1 0
```

Prezentacja umiejscowienia punktów 2D z 2 klasami.



```
1/1 [=====] - 0s 56ms/step
[0 0] 0 0.0
[0 1] 1 0.0
[1 0] 1 0.0
[1 1] 0 0.0
```

Wizualizacja poprawności rozpoznania klas punktów przed nauką



Model loss

