

Projekt 3

Klasyfikacyjna sieć neuronowa

Autor: Jakub Smyk

Numer albumu: 11336

Kierunek studiów: Zarządzanie

Specjalizacja: Analityka danych, big data i kodowanie

Nazwa przedmiotu: Wprowadzenie do sieci neuronowych i uczenia maszynowego

Wykładowca: mgr inż. Karol Struniawski

Uczelnia: Collegium Civitas

Celem tej pracy jest przetestowanie różnych topologii sieci neuronowej i ich funkcji aktywacji, aby zmaksymalizować miarę dopasowania *Accuracy*. Poniższy kod normalizuje dane, dzieli je na zbiór testowy i treningowy z walidacją krzyżową 10%, losowym ziarnem równym 150, optymalizacją SGD i funkcją kosztu będącą błędem średniokwadratowym.

Maksymalne dopasowanie jakie udało się uzyskać to 1.00 w drugiej topologii (Warstwy 24, 12, 1)

```
X = df.drop(['y'], axis = 1)
y = df['y']
```

```
scaler = StandardScaler()
X = scaler.fit_transform(X)
```

```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.1, random_state = 150)
```

```
X_train = np.reshape(X_train, (X_train.shape[0], 10))
X_test = np.reshape(X_test, (X_test.shape[0], 10))
```

```
model = tf.keras.Sequential()
model.add(tf.keras.layers.Dense(12, input_dim=10, activation='gelu'))
model.add(tf.keras.layers.Dense(8, activation='gelu'))
model.add(tf.keras.layers.Dense(1, activation='gelu'))

model.compile(loss='mean_squared_error', optimizer=tf.keras.optimizers.SGD(), metrics=['Accuracy'])
model.fit(X_train, y_train, epochs=15, batch_size=32, validation_data=(X_test, y_test))
```

```
Epoch 1/15
16/16 [=====] - 2s 31ms/step - loss: 0.3612 - Accuracy: 0.6992 - val_loss: 0.4056 - val_Accuracy: 0.64
91
Epoch 2/15
16/16 [=====] - 0s 9ms/step - loss: 0.3597 - Accuracy: 0.6992 - val_loss: 0.4042 - val_Accuracy: 0.649
1
Epoch 3/15
16/16 [=====] - 0s 8ms/step - loss: 0.3583 - Accuracy: 0.6992 - val_loss: 0.4029 - val_Accuracy: 0.649
1
Epoch 4/15
16/16 [=====] - 0s 8ms/step - loss: 0.3569 - Accuracy: 0.6992 - val_loss: 0.4018 - val_Accuracy: 0.649
1
Epoch 5/15
16/16 [=====] - 0s 9ms/step - loss: 0.3556 - Accuracy: 0.6992 - val_loss: 0.4008 - val_Accuracy: 0.649
1
Epoch 6/15
16/16 [=====] - 0s 9ms/step - loss: 0.3544 - Accuracy: 0.6992 - val_loss: 0.3999 - val_Accuracy: 0.649
1
Epoch 7/15
16/16 [=====] - 0s 7ms/step - loss: 0.3533 - Accuracy: 0.6992 - val_loss: 0.3991 - val_Accuracy: 0.649
1
Epoch 8/15
16/16 [=====] - 0s 10ms/step - loss: 0.3522 - Accuracy: 0.6992 - val_loss: 0.3983 - val_Accuracy: 0.64
91
Epoch 9/15
16/16 [=====] - 0s 8ms/step - loss: 0.3512 - Accuracy: 0.6992 - val_loss: 0.3977 - val_Accuracy: 0.649
1
Epoch 10/15
16/16 [=====] - 0s 7ms/step - loss: 0.3503 - Accuracy: 0.6992 - val_loss: 0.3970 - val_Accuracy: 0.649
1
Epoch 11/15
16/16 [=====] - 0s 9ms/step - loss: 0.3493 - Accuracy: 0.6992 - val_loss: 0.3965 - val_Accuracy: 0.649
1
Epoch 12/15
16/16 [=====] - 0s 7ms/step - loss: 0.3485 - Accuracy: 0.6992 - val_loss: 0.3959 - val_Accuracy: 0.649
1
Epoch 13/15
16/16 [=====] - 0s 7ms/step - loss: 0.3477 - Accuracy: 0.6992 - val_loss: 0.3955 - val_Accuracy: 0.649
1
Epoch 14/15
16/16 [=====] - 0s 10ms/step - loss: 0.3469 - Accuracy: 0.6992 - val_loss: 0.3950 - val_Accuracy: 0.64
91
Epoch 15/15
16/16 [=====] - 0s 9ms/step - loss: 0.3462 - Accuracy: 0.6992 - val_loss: 0.3946 - val_Accuracy: 0.649
1
```

W pierwszej topologii, która składa się z trzech warstw o głębokości kolejno 12, 8 i 1 neuron, accuracy wynosi w przybliżeniu 0.69. Funkcja aktywacji to GELU (Gaussian Error Linear Unit). Sieć nie poprawia swojej efektywności klasyfikacji. W pierwszej epoce accuracy wynosiła 0.6992 a w ostatniej jest taka sama. Błąd średniokwadratowy, również utrzymuje się na podobnym poziomie. W ciągu całego procesu trenowania wynosi około 0.35.

```

: model = tf.keras.Sequential()
model.add(tf.keras.layers.Dense(24, input_dim=10, activation='linear'))
model.add(tf.keras.layers.Dense(12, activation='linear'))
model.add(tf.keras.layers.Dense(1, activation='linear'))

model.compile(loss='mean_squared_error', optimizer=tf.keras.optimizers.SGD(), metrics=['Accuracy'])
model.fit(X_train, y_train, epochs=15, batch_size=32, validation_data=(X_test, y_test))

Epoch 1/15
16/16 [=====] - 1s 31ms/step - loss: 0.3045 - Accuracy: 0.8086 - val_loss: 0.0516 - val_Accuracy: 0.98
25
Epoch 2/15
16/16 [=====] - 0s 7ms/step - loss: 0.0344 - Accuracy: 0.9902 - val_loss: 0.0097 - val_Accuracy: 1.000
0
Epoch 3/15
16/16 [=====] - 0s 9ms/step - loss: 0.0071 - Accuracy: 1.0000 - val_loss: 0.0027 - val_Accuracy: 1.000
0
Epoch 4/15
16/16 [=====] - 0s 10ms/step - loss: 0.0019 - Accuracy: 1.0000 - val_loss: 8.6215e-04 - val_Accuracy:
1.0000
Epoch 5/15
16/16 [=====] - 0s 7ms/step - loss: 5.7392e-04 - Accuracy: 1.0000 - val_loss: 3.2349e-04 - val_Accurac
y: 1.0000
Epoch 6/15
16/16 [=====] - 0s 7ms/step - loss: 1.9872e-04 - Accuracy: 1.0000 - val_loss: 1.3671e-04 - val_Accurac
y: 1.0000
Epoch 7/15
16/16 [=====] - 0s 7ms/step - loss: 8.0943e-05 - Accuracy: 1.0000 - val_loss: 6.5981e-05 - val_Accurac
y: 1.0000
Epoch 8/15
16/16 [=====] - 0s 7ms/step - loss: 3.7222e-05 - Accuracy: 1.0000 - val_loss: 3.4309e-05 - val_Accurac
y: 1.0000
Epoch 9/15
16/16 [=====] - 0s 10ms/step - loss: 1.8990e-05 - Accuracy: 1.0000 - val_loss: 1.8904e-05 - val_Accura
cy: 1.0000
Epoch 10/15
16/16 [=====] - 0s 8ms/step - loss: 1.0353e-05 - Accuracy: 1.0000 - val_loss: 1.0679e-05 - val_Accurac
y: 1.0000
Epoch 11/15
16/16 [=====] - 0s 10ms/step - loss: 5.8274e-06 - Accuracy: 1.0000 - val_loss: 6.0386e-06 - val_Accura
cy: 1.0000
Epoch 12/15
16/16 [=====] - 0s 10ms/step - loss: 3.3620e-06 - Accuracy: 1.0000 - val_loss: 3.5284e-06 - val_Accura
cy: 1.0000
Epoch 13/15
16/16 [=====] - 0s 11ms/step - loss: 1.9545e-06 - Accuracy: 1.0000 - val_loss: 2.0333e-06 - val_Accura
cy: 1.0000
Epoch 14/15
16/16 [=====] - 0s 8ms/step - loss: 1.1406e-06 - Accuracy: 1.0000 - val_loss: 1.1956e-06 - val_Accurac
y: 1.0000
Epoch 15/15
16/16 [=====] - 0s 10ms/step - loss: 6.6702e-07 - Accuracy: 1.0000 - val_loss: 6.9221e-07 - val_Accura
cy: 1.0000

```

W drugiej topologii, która składa się z trzech warstw o głębokości kolejno 24, 12 i 1 neuron, liniowej funkcji aktywacji, accuracy wynosi w 1. Sieć poprawia swoją efektywność klasyfikacji. W pierwszej epoce accuracy wynosiła 0.8 a w ostatniej 1. Błąd średniokwadratowy spada z 0.3 do 0.006.

```

In [29]: model = tf.keras.Sequential()
model.add(tf.keras.layers.Dense(16, input_dim=10, activation='relu'))
model.add(tf.keras.layers.Dense(8, activation='relu'))
model.add(tf.keras.layers.Dense(4, activation='relu'))
model.add(tf.keras.layers.Dense(1, activation='relu'))

model.compile(loss='mean_squared_error', optimizer=tf.keras.optimizers.SGD(), metrics=['Accuracy'])
model.fit(X_train, y_train, epochs=15, batch_size=32, validation_data=(X_test, y_test))

Epoch 1/15
16/16 [=====] - 2s 27ms/step - loss: 0.2921 - Accuracy: 0.6992 - val_loss: 0.3275 - val_Accuracy: 0.64
91
Epoch 2/15
16/16 [=====] - 0s 7ms/step - loss: 0.2783 - Accuracy: 0.6992 - val_loss: 0.3021 - val_Accuracy: 0.649
1
Epoch 3/15
16/16 [=====] - 0s 7ms/step - loss: 0.2560 - Accuracy: 0.7012 - val_loss: 0.2547 - val_Accuracy: 0.649
1
Epoch 4/15
16/16 [=====] - 0s 10ms/step - loss: 0.2217 - Accuracy: 0.7051 - val_loss: 0.2044 - val_Accuracy: 0.68
42
Epoch 5/15
16/16 [=====] - 0s 7ms/step - loss: 0.1908 - Accuracy: 0.7227 - val_loss: 0.1667 - val_Accuracy: 0.684
2
Epoch 6/15
16/16 [=====] - 0s 10ms/step - loss: 0.1683 - Accuracy: 0.7676 - val_loss: 0.1408 - val_Accuracy: 0.77
19
Epoch 7/15
16/16 [=====] - 0s 9ms/step - loss: 0.1524 - Accuracy: 0.8125 - val_loss: 0.1229 - val_Accuracy: 0.824
6
Epoch 8/15
16/16 [=====] - 0s 7ms/step - loss: 0.1399 - Accuracy: 0.8281 - val_loss: 0.1082 - val_Accuracy: 0.894
7
Epoch 9/15
16/16 [=====] - 0s 6ms/step - loss: 0.1286 - Accuracy: 0.8477 - val_loss: 0.0964 - val_Accuracy: 0.912
3
Epoch 10/15
16/16 [=====] - 0s 7ms/step - loss: 0.1186 - Accuracy: 0.8633 - val_loss: 0.0864 - val_Accuracy: 0.912
3
Epoch 11/15
16/16 [=====] - 0s 9ms/step - loss: 0.1097 - Accuracy: 0.8770 - val_loss: 0.0773 - val_Accuracy: 0.947
4
Epoch 12/15
16/16 [=====] - 0s 10ms/step - loss: 0.1016 - Accuracy: 0.8906 - val_loss: 0.0697 - val_Accuracy: 0.98
25
Epoch 13/15
16/16 [=====] - 0s 9ms/step - loss: 0.0941 - Accuracy: 0.9062 - val_loss: 0.0627 - val_Accuracy: 0.982
5
Epoch 14/15
16/16 [=====] - 0s 9ms/step - loss: 0.0872 - Accuracy: 0.9160 - val_loss: 0.0565 - val_Accuracy: 0.982
5
Epoch 15/15
16/16 [=====] - 0s 7ms/step - loss: 0.0810 - Accuracy: 0.9219 - val_loss: 0.0510 - val_Accuracy: 0.982
5

```

W trzeciej topologii, która składa się z czterech warstw o głębokości kolejno 16, 8 i 4, 1 neuron, accuracy wynosi 0.92. Funkcja aktywacji to RELU (Rectified Linear Unit). Sieć poprawia swoją efektywność klasyfikacji. W pierwszej epoce accuracy wynosiła 0.69 a w ostatniej 0.92. Błąd średniokwadratowy spada z 0.2 do 0.08.