

Simulation de Keras sous VBA

Page source : <https://machinelearningmastery.com/tutorial-first-neural-network-python-keras>



Chargement des bibliothèques et import des données

```
1 # first neural network with keras tutorial
2 from numpy import loadtxt
3 from keras.models import Sequential
4 from keras.layers import Dense
5 ...
```

```
1 ...
2 # load the dataset
3 dataset = loadtxt('pima-indians-diabetes.csv', delimiter=',')
4 # split into input (X) and output (y) variables
5 X = dataset[:,0:8]
6 y = dataset[:,8]
7 ...
```

Définition de l'architecture du modèle

```
1 ...  
2 # define the keras model  
3 model = Sequential()  
4 model.add(Dense(12, input_dim=8, activation='relu'))  
5 model.add(Dense(8, activation='relu'))  
6 model.add(Dense(1, activation='sigmoid'))  
7 ...
```

```
1 ...  
2 # compile the keras model  
3 model.compile(loss='binary_crossentropy', optimizer='adam', metrics=['accuracy'])  
4 ...
```

Apprentissage et validation du modèle

```
1 ...  
2 # fit the keras model on the dataset  
3 model.fit(X, y, epochs=150, batch_size=10)  
4 ...
```

```
1 ...  
2 # evaluate the keras model  
3 _, accuracy = model.evaluate(X, y)  
4 print('Accuracy: %.2f' % (accuracy*100))
```

Utilisation du modèle

```
1 ...  
2 # make probability predictions with the model  
3 predictions = model.predict(X)  
4 # round predictions  
5 rounded = [round(x[0]) for x in predictions]
```

```
1 ...  
2 # make class predictions with the model  
3 predictions = model.predict_classes(X)
```

Le tout tient sur 23 lignes !!!!

```
1 # first neural network with keras make predictions
2 from numpy import loadtxt
3 from keras.models import Sequential
4 from keras.layers import Dense
5 # load the dataset
6 dataset = loadtxt('pima-indians-diabetes.csv', delimiter=',')
7 # split into input (X) and output (y) variables
8 X = dataset[:,0:8]
9 y = dataset[:,8]
10 # define the keras model
11 model = Sequential()
12 model.add(Dense(12, input_dim=8, activation='relu'))
13 model.add(Dense(8, activation='relu'))
14 model.add(Dense(1, activation='sigmoid'))
15 # compile the keras model
16 model.compile(loss='binary_crossentropy', optimizer='adam', metrics=['accuracy'])
17 # fit the keras model on the dataset
18 model.fit(X, y, epochs=150, batch_size=10, verbose=0)
19 # make class predictions with the model
20 predictions = model.predict_classes(X)
21 # summarize the first 5 cases
22 for i in range(5):
23     print('%s => %d (expected %d)' % (X[i].tolist(), predictions[i], y[i]))
```