

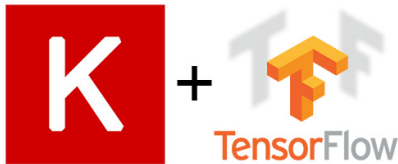
# Deep Learning par la Pratique

Introduction à Keras

---

# Introduction à Keras

Keras est une API de haut niveau qui permet de prototyper des réseaux de neurones de toute sorte.



User-friendly :

- Interface simple
- Accès facilité aux métriques d'évaluation

Modulaire :

- Les réseaux se 'branchent' facilement les uns avec les autres
- Tous les réseaux se configurent facilement

Etat de l'art :

- Les modèles et optimiseurs pertinents sont rapidement ajouté à Keras
- Reproduction facile de résultats récents

Facile de développer de nouvelles :

- Couche de réseau (Layers)
- Fonction de perte (Loss)
- Métriques d'évaluation

# Réseau de neurones en Keras

```
1 from tensorflow.keras import layers as kl
2
3 model = tf.keras.Sequential()
4 # Il est impératif de spécifier input_shape pour la première couche :
5 model.add(layers.Dense(64, activation='sigmoid', input_shape=(32,)))
6 # Ajouter une autre couche :
7 model.add(layers.Dense(64, activation='relu'))
8 # Une dernière couche de classification avec softmax ;
9 model.add(layers.Dense(10, activation='softmax'))
```

# Affichage du modèle

```
1 model.summary()
```

```
1 Model: "sequential"
```

```
2
```

```
3 Layer (type)
```

```
Output Shape
```

```
Param #
```

```
4
```

```
5 dense (Dense)
```

```
(None, 64)
```

```
2112
```

```
6
```

```
7 dense_1 (Dense)
```

```
(None, 64)
```

```
4160
```

```
8
```

```
9 dense_2 (Dense)
```

```
(None, 10)
```

```
650
```

```
10
```

```
11 Total params: 6,922
```

```
12 Trainable params: 6,922
```

```
13 Non-trainable params: 0
```

```
14
```



```
1 import numpy as np
2
3 data = np.random.random((1000, 32))
4 labels = np.random.random((1000, 10))
5
6 model.fit(data, labels, epochs=10, batch_size=32)
```