



TP n°4: Classification des images en utilisant le CNN

Nous classifions les images cifar10 dataset from tensorflow keras datasets. Il y a au total 10 classes. Nous utilisons CNN pour la classification

airplane



automobile



bird



cat



deer



dog



frog



horse



ship



truck



1. Importer les modules nécessaires

```
import tensorflow as tf
from tensorflow.keras import datasets, layers, models
import matplotlib.pyplot as plt
import numpy as np
```

2. Utiliser la méthode load_data() pour charger le dataset.

load_data() retourne le tuple: (x_train, y_train), (x_test, y_test).

Interpréter les shapes de x_train, y_train, x_test, y_test

3. Afficher le contenu d'un échantillon

Afficher graphiquement le contenu du même échantillon

4. Normaliser le training data

5. Créer un réseau de neurones artificiel simple (ANN) pour entraîner les images en s'appuyant sur le code suivant:

```
ann = models.Sequential([
    layers.Flatten(input_shape=(32,32,3)),
    layers.Dense(3000, activation='relu'),
    layers.Dense(1000, activation='relu'),
    layers.Dense(10, activation='softmax')
])

ann.compile(optimizer='SGD',
            loss='sparse_categorical_crossentropy',
            metrics=['accuracy'])

ann.fit(X_train, y_train, epochs=5)
Interpréter le résultat
```

6. Créer un réseau de neurones convolutif (CNN) pour entraîner les images.

L'architecture de CNN est composée de 2 couches de convolution. chaque couche est suivie d'un max pooling. la fonction d'activation est 'relu'. Les couches de convolution utilisent respectivement 32 et 64 filtres. La taille du filtre est (3,3). Le pool_size est fixé à (2,2)

Utiliser les fonctions Conv2D et MaxPooling2D en précisant le input_shape

Pour retrouver la syntaxe d'une fonction, utiliser `help(layers.Conv2D)`

La deuxième partie de CNN, la partie de classification est composée de 2 couches: une couche de 64 neurones qui utilise la fonction d'activation 'relu' et la couche de sortie qui utilise la fonction d'activation='softmax'

Interpréter le résultat

Evaluer les performances du modèle

7. Soit `y_pred` le resultat de la prédiction des classes des échantillons `x_test`

Interpréter le shape de `y_pred`

Afficher le résultat des 5 premiers échantillons

Quelle est la classe prédite pour chaque échantillon

Comparer les classes predites avec les vraies classes

