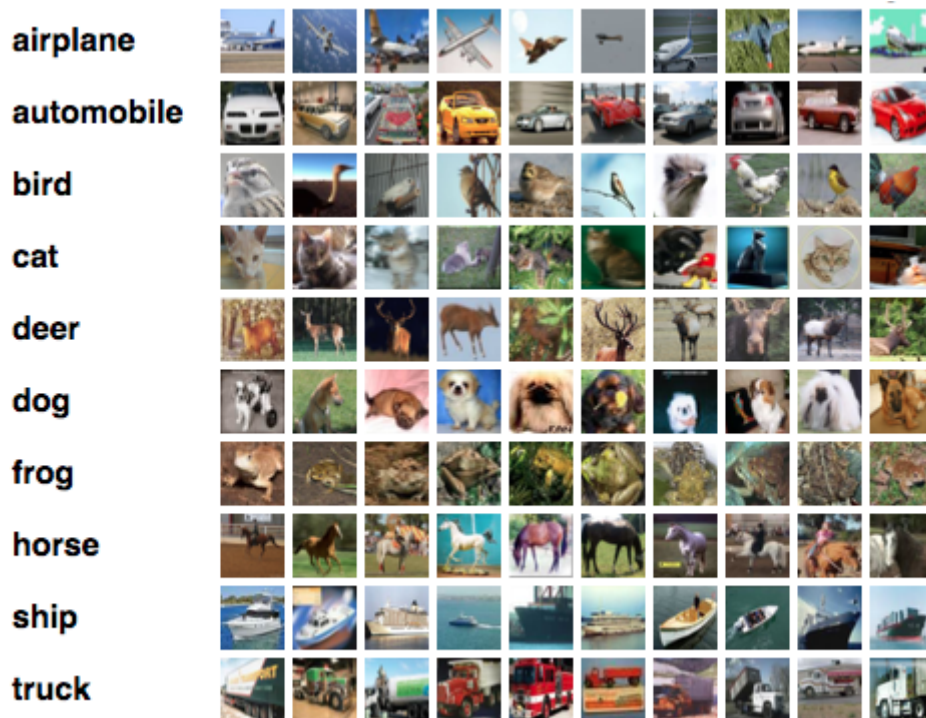


HAI923 2021/2022

PROJET MACHINE LEARNING II

Le but de ce projet est de faire une initiation à la classification d'images en utilisant l'API Keras et / ou Pytorch. La réalisation du projet sous Pytorch sera appréciée par des points bonus.

Nous allons explorer et faire une tâche de classification sur les données CIFAR-10 qui est un jeu de données de 60k images en couleur dans 10 classes de taille 32*32.

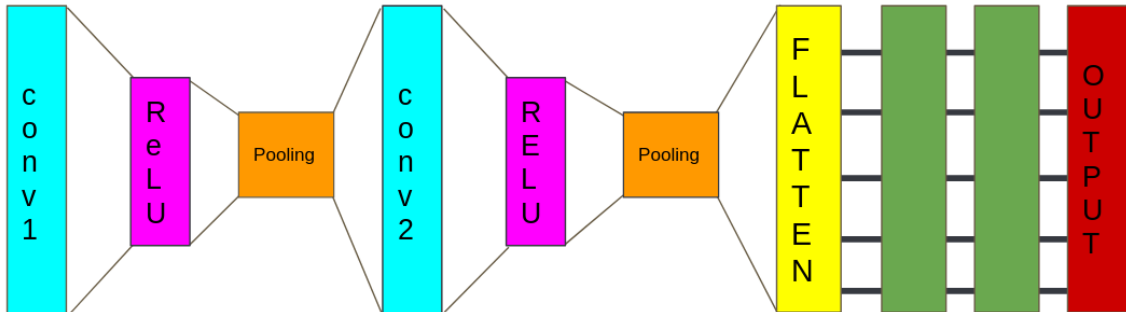


<https://www.cs.toronto.edu/~kriz/cifar.html>

Il s'agirait donc de trouver la bonne classe d'un objet de ce jeu de données en entraînant un modèle de machine/deep learning. À l'issue de l'entraînement des stratégies d'évaluation seront explorées et améliorées. Une démarche (pas obligatoire) vous est proposée ci-dessous :

1. Chargement des données de cifar-10
2. Exploration/ visualisation des données
3. Normalisation des données avant l'entraînement de 0-255 → 0-1, réduit la complexité et le temps d'apprentissage tout en conservant les informations
4. L'encodage des labels/étiquettes (one hot encoding) en binaire pour ne pas faire une tâche de régression, Notez que l'utilisation d'une fonction d'erreur comme la cross entropy permet de ne pas encoder les labels.
5. Construction du modèle :
 - a. Création du modèle
 - b. Ajout de couche de convolution

- c. Activation
- d. Pooling (si vous voulez)
- e. Normalisation par lot
- f. Flatten
- g. Couche Fully-connected avec softmax à la dernière



6. Compilation et entraînement du modèles

7. Évaluation

- a. Étude du sur/sous apprentissage:
 - i. observation de l'évolution de l'accuracy et de la fonction d'erreur
 - ii. Matrice de confusion
- b. Étude des hyperparamètres :
 - i. learning rate
 - ii. optimizer
 - iii. fonction d'erreur
 - iv. nombre d'epoch
 - v. batch_size
- c. Comparaison des différents paramètres dans un tableau et conclusion
- d. Proposition des améliorations au modèles
- e. **N'oubliez pas de sauvegarder vos modèles**

Les modèles de réseaux de neurones nécessitent un nombre important de données et par conséquent un long temps d'apprentissage et aussi de ressources de calcul, voilà pourquoi dans certains cas il est préférable de faire de l'augmentation de données et/ou d'utiliser des modèles pré-entraînés en faisant du **transfert learning**. Nous allons donc utiliser ces différentes techniques pour comparer et améliorer notre modèle.

Pour cela nous allons choisir 3 ou 4 modèles pré-entraînés et comparer les techniques de **fine-tuning** et de **Fixe Feature Extractor**. Aidez-vous de l'augmentation des données Data augmentation pour améliorer le modèle et évaluer comme précédemment.

Faites une comparaison dans un tableau de l'ensemble de vos tests et conclure

Modalités de travail :

- 1) travail à effectuer en groupes de **3 ou 4 personnes maximum**
- 2) rendu au plus tard **le mercredi 8 décembre** sur Moodle contenant un dossier unique portant dans son nom les noms des membres du groupe avec:
 - a) un mini rapport — 5 pages maximum
 - b) le notebook Jupyter et le code des traitements automatiques
- 3) Soutenances: **vendredi 10 décembre à partir de 8h**