

# Formulaire de réponse Projet ML

## 1. Algorithme Perceptron Simple

- Donner quelques éléments de commentaires sur la stratégie que vous avez utilisée pour développer le perceptron simple.
- Montrer que votre code fonctionne dans le cas du OU
- Afficher la figure individu/frontière et commenter

## 2. Apprentissage Widrow : ensemble Test 1 / Test 2

- Donner quelques éléments de commentaires sur la stratégie que vous avez utilisée pour développer l'apprentissage Widrow.
- Pour l'ensemble Test 1,
  - Indiquez graphiquement quelques étapes de l'apprentissage (ensemble et droite de séparation).
  - Représenter l'erreur en fonction des itérations.
  - Faites quelques tests avec des initialisations différentes, commenter le résultat après convergence.
- Pour l'ensemble Test 2,
  - Indiquez graphiquement quelques étapes de l'apprentissage (ensemble et droite de séparation).
  - Représenter l'erreur en fonction des itérations. Comparer avec l'opération sur test 1
  - Faites quelques tests avec des initialisations différentes, commenter le résultat après convergence.

## 3. Mise en place d'un perceptron Multicouche

- 4. Donner quelques éléments de commentaires sur la stratégie que vous avez utilisée pour développer le perceptron Multicouche.
- Indiquer le résultat numérique et par calcul (en donnant le détail) pour le test demandé

## 5. Apprentissage Multicouches

- Donner quelques éléments de commentaires sur la stratégie que vous avez utilisée pour développer l'apprentissage Multicouche.
- Représenter l'erreur en fonction des itérations. Commenter
- Tester votre structure après apprentissage et montrer que c'est bien un XOR
- Représenter les trois droites séparatrices et l'ensemble d'apprentissage. Expliquer le fonctionnement
- Question bonus : si vous avez fait une version On-line la transformer en Batch et comparer l'évolution de l'erreur en fonction des itérations.

# Formulaire de réponse Projet ML – volet classification

## 6. Volet Chargement des descripteurs

- Expliciter comment sous Python vous avez importé et créer votre corpus : vecteur de labels, ensemble d'apprentissage, ensemble de tests.

## 7. Classification par Full Connected

Pour tous les tests et comparaisons qui suivent, vous vous appuyerez sur le taux d'erreur et la matrice de confusion :

- Donner quelques éléments de commentaires sur la stratégie que vous avez utilisée pour développer le perceptron (Full-connected).
- Comparer la qualité de la discrimination en fonction de la caractéristique utilisée ou du cumul de toutes les mesures
- Avec les meilleures combinaisons de mesures, paramétrer au mieux l'algorithme Full Connected
- Avec le meilleur discriminateur évaluer la qualité de la procédure de discrimination selon la classe d'images.
- Comparer avec l'algorithme KppV

## 8. Classification par Deep

Pour tous les tests et comparaisons qui suivent, vous vous appuyerez sur le taux d'erreur et la matrice de confusion :

- Donner quelques éléments de commentaires sur la stratégie que vous avez utilisée pour développer votre structure Deep.
- Comparer la qualité de la discrimination en fonction des caractéristiques du réseau. Etudier notamment l'évolution de la fonction de cout/ Accuracy. Bonus : proposer une stratégie pour optimiser les hyperparamètres.
- Avec la meilleure combinaison comparer avec le Full Connected
- Avec le meilleur discriminateur évaluer la qualité de la procédure de discrimination selon la classe d'images.
- Faire progresser les résultats avec une procédure de Data Augmentation.
- Evaluer les performances avec les stratégies limitant l'overfitting si besoin ou la "disparition" du gradient
- Evaluer les performances par transfert Learning