TREX BRAIN

Etat d’avancement

TODO : ajouter schéma, image et gantt + new gantt + parler des difficultés ?

# Contexte du projet

Pour rappel, le projet consiste à mettre en place une IA qui apprend d’elle-même à jouer au jeu du T-Rex de Google Chrome (Easter Egg caché lorsque l’on ouvre le navigateur Chrome sans connexion internet).

Le projet se décompose en 3 étapes :

1. D’abord, coder le jeu en javascript,
2. Ensuite, mettre en place le réseau neuronal associé à un T-Rex,
3. Enfin, faire évoluer le réseau neuronal afin que l’IA fasse un score maximal.

# Travail réalisé

## Codage du jeu

Actuellement, le jeu a été codé en Javascript. Il est jouable par un humain. Le joueur doit faire sauter le T-Rex lorsqu’il rencontre un cactus, ou bien l’accroupir lorsqu’il rencontre un ptérodactyle. La librairie p5 a été utilisée pour gérer facilement la boucle de jeu (update et draw). Cependant, cette librairie est assez contraignante et me pose des problèmes lorsque j’essaie de mettre en place l’algorithme génétique…

## Mise en place d’un réseau neuronal contrôlant l’IA

Une fois le jeu codé, il fallait que le T-Rex soit soumis à un réseau neuronal. La librairie synaptic.js a été utilisée. Elle permet de créer un perceptron très simplement, et possède des fonctions qui vont m’être utiles par la suite (notamment les exportations en JSON des perceptrons). Le T-Rex possède donc un Cerveau (Brain.js) qui contient lui-même un perceptron.

Le perceptron a 4 neurones d’entrées qui correspondent respectivement à :

* La vitesse courante du jeu
* La distance séparant le T-Rex du prochain obstacle
* La largeur du prochain obstacle
* Le type du prochain obstacle (CACTUS ou PTERODACTYL)

La valeur d’entrée doit être comprise entre 0 et 1, on effectue donc un mapping des valeurs d’entrée avant de les mettre dans les neurones d’entrée.

A chaque tour dans la boucle de jeu, le réseau neuronal est activé, et, en fonction des inputs, sort deux outputs liés aux actions Sauter et Se Baisser. Si la valeur de ces neurones est supérieure à 0,5 (la valeur varie entre 0 et 1), alors le T-Rex saute ou se baisse selon le neurone sortant.

## Squelette de l’algorithme génétique

Par ailleurs, le squelette de l’algorithme génétique est implémenté. Concrètement, il existe des signatures pour les fonctions de sélection des meilleurs réseaux neuronaux, de croisement entre les meilleurs réseaux (pour créer de nouveaux individus) et de mutation de ces nouveaux individus.

# Travail restant

## Implémentation de l’algorithme génétique

À présent, il reste à implémenter les fonctions de l’algorithme génétique.

J’appelle ‘individu’ un perceptron relié à un T-Rex.

La sélection des meilleurs individus se fera sur le nombre de cactus qu’un T-Rex sautera ;

Un crossing-over entre les meilleurs individus donnera naissance à des nouveaux individus

Des mutations seront appliquées sur ces nouveaux individus (pour ne pas se restreindre aux caractères génétiques de la première génération)

Et on reprend à partir de la sélection, avec les nouveaux individus, et ainsi de suite jusqu’à avoir une IA jouant aussi bien –voire mieux- qu’un humain.

## Affichage des données importantes

Données importantes pour suivre l’évolution + graphique

## Prier pour que ça marche