Neat Process

The user will see a collection of species members on the screen. They will be able to move forward and rotate clockwise or anti-clockwise. The player will not be able to control this movement. This movement will be controlled by the NEAT algorithm. The player will have the ability to draw an area on the screen that is considered the goal. If a member is in that area by the end of a set amount of time, they will get a positive score, else they will get a negative score. The NEAT algorithm will run once per frame and will either tell the member to rotate or move forward. The inputs to the algorithm will be the members current facing direction, current position in the world and the position of other members within a set radius. The NEAT algorithm will then train over a set number of EPOCHS and the player will be able to see the results at the end. The algorithm should also save the best models of every X generation. A genome consists of a list of Nodes and a List of connections.

Table

Description automatically generated

[*https://towardsdatascience.com/neat-an-awesome-approach-to-neuroevolution-3eca5cc7930f*](https://towardsdatascience.com/neat-an-awesome-approach-to-neuroevolution-3eca5cc7930f)

Each Node member in the list consists of an Id number and the type of node it is. These types can either be Sensor (Input), Hidden (Middle Layers) or Output.

Each Connection lists the In and Out node, the weight applied to the connection, if it is enabled or not and the Innovation number.

During generations, the genome of a member will mutate. This can happen in three ways. Firstly, a new connection can be added between two existing nodes. Secondly, a new node can be added in between an existing connection. In the second case, the original connection is disabled, and two new connections are created linking the new node to the originals in the same direction. Finally, the weights on an existing connection can change.

When a new mutation is created, it is given an innovation number. This will be tracked by the NEAT class.

Chart, waterfall chart

Description automatically generated

A rough idea of what the NEAT algorithm will look like with NO mutation and NO loss function calculation.

IN,HIDDEN,OUT,Connections\_Count,Con1\_In ,Con1\_Out, ,Con1\_Weight, ,Con1\_Enabled, ,Con1\_Innovation

20361241114712