Practical – 1 Smart sweeper

CSC3022H – C++ with Applications

8/30/2013
University of Cape Town
Shirin Patel – PTLSHI004 and Victor Soudien – SDNVIC001

Learning method: Supervised learning

We chose an artificial neural network (ANN) to solve the problem. Our ANN consists of a single perceptron with a linear activation function. The properties of such a network are that they are able to classify input data as being on either side of a boundary and the output is either 0 or 1. This led us to the conclusion that it would be suitable for the problem since we have two sensor inputs from the minesweeper i.e. the distance to the closest object and the angle between the minesweeper. The second parameter, the angle, was calculated by taking the dot product of the minesweepers direction vector and the vector to the closest mine.

One of the disadvantages of our approach much like any supervised learning algorithm, is that we need to provide training data. The accuracy of our training data was vital since perceptron's are not able to handle erroneous data correctly. We specified target output as being 1 to turn and 0 if the minesweeper should continue along its path. The training data was of the form:

$$\{distance_1, \theta_1, t_1\}, \{distance_2, \theta_2, t_2\}, \ldots, \{distance_Q, \theta_Q, t_Q\},$$

where $distance_Q(how\ far\ is\ the\ mine\ )\ and\ \theta_Q\ (angle\ between\ the\ sweeper\ and\ mine)$ is an input to the network and $t_Q$is the corresponding target output (binary value).

As the inputs are applied to the network, the network outputs are compared to the targets. The learning rule is used to adjust the weights of the network in order to move the network outputs closer to the target.

Like most machine learning algorithms, the solution is not perfect. The minesweepers do sometimes collide with the mines.

Steps:

1. Initialize threshold, learning rate and weights:

   ```
   Perceptron p = *(new Perceptron (-0.8, 0.1, 0.5, 0.5));
   ```

2. We provided the ANN with twenty one training examples. They were made up as follows, distance between 0 and 60 in increments of 5 and an angle between 0 and 90 in increments of 10 for each of the distances.

3. Once the ANN has been trained, we provide the inputs to it on each iteration and based on the output, it takes the appropriate action.

Pseudo code

```
To train the ANN:

Initialize perceptron as shown in step 1

For each distance to the mine in specific range

    For each angle to the mine in range

        If distance < 15 and angle <= 25

            Train the perceptron to turn away

        Else

            Train it to continue moving in same direction


Minesweeper uses ANN:

If current closest mine and the dot product of (closest mine and angle to it) returns 1

    Turn away from it

Else do nothing // continues
```