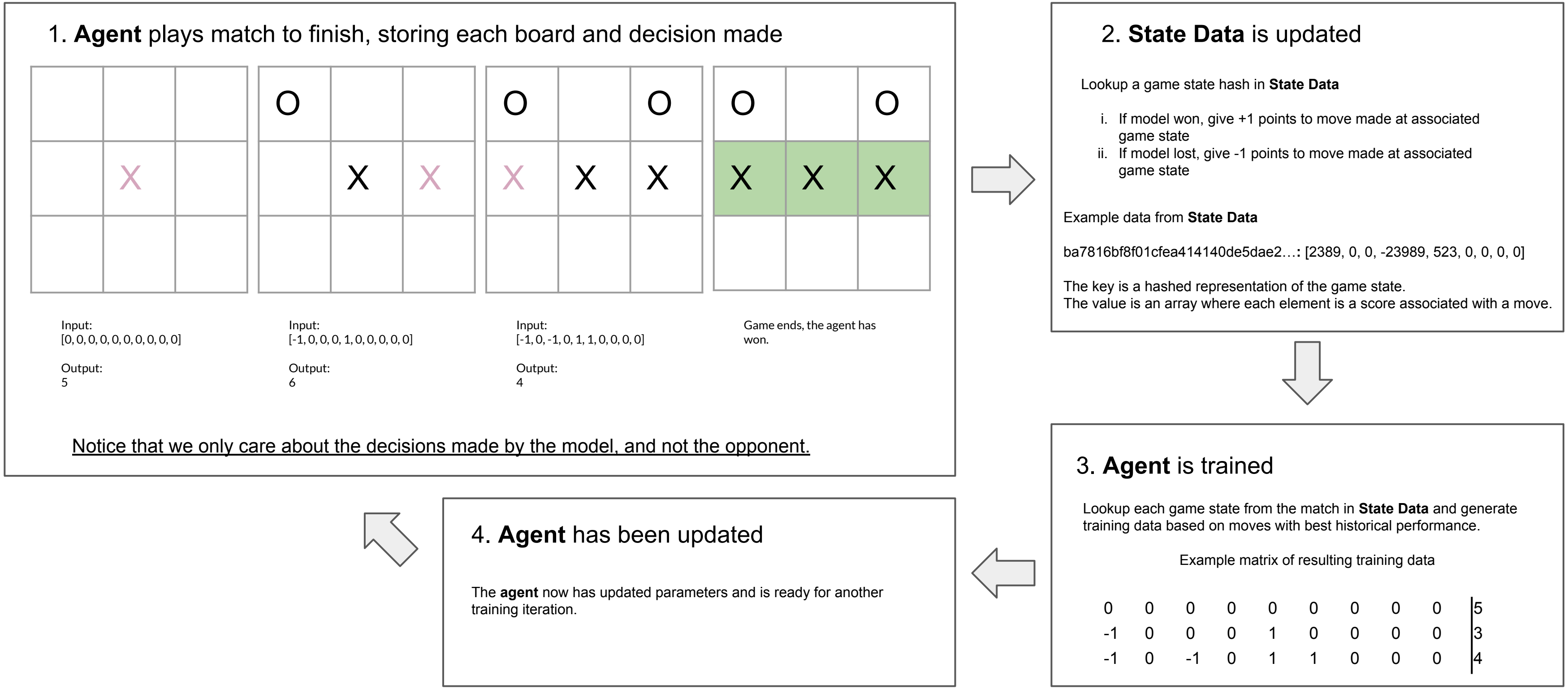
1. **Agent** competes against random moves until completion of match
2. Results of match are used to update the **State Data**
    a. If **agent** wins, each move made is given +1 points
    b. If **agent** loses or draws, each moves made is given -1 points
3. Data is generated to incrementally train the **agent**
    a. A matrix of all the game states seen during the iteration is generated to be used as **features**
    b. A lookup into **State Data** is conducted to find the moves associated with each game state that have historically had the best performance. These moves are converted into a vector to be used as **labels**
    c. The **features** and **labels** are passed to the **agent** for partial fitting
4. The **agent**'s parameters have been updated to perform better.



1. **Agent** plays match to finish, storing each board and decision made

Input:
[0, 0, 0, 0, 0, 0, 0, 0, 0]

Output:
5

Input:
[-1, 0, 0, 0, 1, 0, 0, 0, 0]

Output:
6

Input:
[-1, 0, -1, 0, 1, 1, 0, 0, 0]

Output:
4

Game ends, the agent has won.

Notice that we only care about the decisions made by the model, and not the opponent.

2. **State Data** is updated

Lookup a game state hash in **State Data**

i. If model won, give +1 points to move made at associated game state
ii. If model lost, give -1 points to move made at associated game state

Example data from **State Data**

ba7816bf8f01cfea414140de5dae2…: [2389, 0, 0, -23989, 523, 0, 0, 0, 0]

The key is a hashed representation of the game state.
The value is an array where each element is a score associated with a move.

3. **Agent** is trained

Lookup each game state from the match in **State Data** and generate training data based on moves with best historical performance.

Example matrix of resulting training data

$$\begin{array}{ccccccccc|c} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 5 \\ -1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 3 \\ -1 & 0 & -1 & 0 & 1 & 1 & 0 & 0 & 0 & 4 \end{array}$$

4. **Agent** has been updated

The **agent** now has updated parameters and is ready for another training iteration.