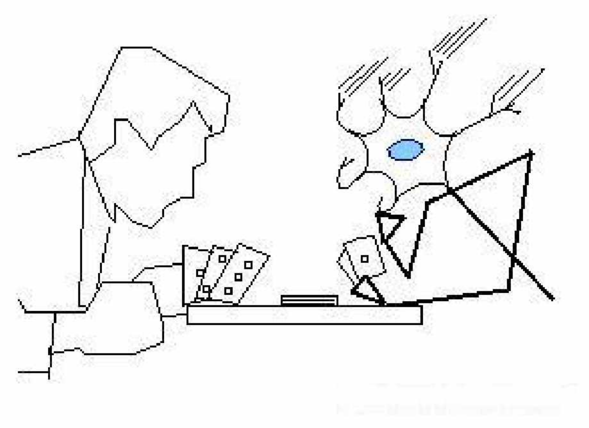


Blackjack Game:

Artificial Neural Network Implementation of the Blackjack Dealer with a Reward mechanism. (Reinforcement signal)

CSE 5360 Project

Vikramaditya Reddy J

Vikramaditya.jakkula@uta.edu

Index

Abstract

Blackjack is a card game where the player attempts to beat the dealer by having the total points higher than the dealer's hand but less than or equal to 21. The probabilistic nature of Blackjack makes it an illustrative application for learning algorithms. This project explores the method of Artificial Neural Network in Blackjack for learning strategies for the dealer, and the reinforcement signal or reward mechanism which can also be used to enhance the performance of the learning strategies. Reinforcement learning is a process to map situations to actions when the learner is not told which actions to take but to discover which actions yield the highest reward by trial and error. The ANN will be learning to play Blackjack. Furthermore, the efficiency of the ANN and the results from the learning will be investigated and interpreted as different strategies for playing Blackjack.

Blackjack or **twenty-one** is a card game where the player attempts to beat the dealer, by obtaining a sum of card values that is equal to or less than 21 so that his total is higher than the dealer's. The probabilistic nature of the game makes it an interesting test bed problem for learning algorithms, though the problem of learning a good playing strategy is not obvious. Learning with a teacher system is not very useful since the target outputs for a given stage of the game are not known. Instead, the learning system has to explore different actions and develop a certain strategy by selectively retaining the actions that maximize the player's performance. We have explored the use of blackjack as a test bed for learning strategies in neural networks.

Goals:

One of the goals of this project is to develop a better strategy for the dealer with ANN and reinforcement signaling (reward mechanism) that makes the Dealer's learn its environment rather than simple standard rule, and, the new strategy will have a higher wining percentage. We also include two ways of dealer exploring its strategies and also compare them.

Approach

The approach consist of two ways of learning as its moves, one is a simple way of learning by adjusting the weights of the ANN and the other is through a reward and punish mechanism to adjust the weights of the ANN. As a result, we compare both of these methodologies and find the reward mechanism has a better learning and has a higher rate of winning games.

Background of the game

Initially with two cards to the player and the dealer, the object of Blackjack is to draw cards to a total value of 21. The player can choose from the following actions:

- **Stand** to stay with the current hand and take no card.
- **Hit** to add a card to the hand to make the total card value closer to 21
- **Double Down** When the player is holding 2 cards, the player can double his bet by hitting with only one more card and stand after that.
- **Split** Having the pair of cards with the same values, the player can split his hand into two hands. The player may split up to 3 times in a game.

For simplicity of this project, Double Down and Split will not be considered in this project. We will consider the Hit and the Stand The value of a hand is the sum of the values of each card in the hand, where each card from 2 to 10 is valued according, with J, Q, and K of the value of 10. Aces can be either 1 or 11. The player plays against the dealer, and the goal is to obtain a hand with a greater value than the dealer's hand but less than or equal to 21. A player may hit as many times as he wishes as long as it is not over 21. The player can also win by having 5 cards in hand with total points less than 21. When the player is dealt 21 points in the first 2 cards, he automatically wins his bet if the dealer is not dealt 21. If the dealer has blackjack (21 points), the game is over and the dealer wins all bets, or ties with any player with blackjack (21 points). The player needs to press the hit or stand button each turn, and the total dealer and player points are calculated and who ever reaches 21 first wins and who crosses 21 losses and also if they get tied if they stand at the same value

Premise of the game

The basic premise of the game is that you want to have a hand value that is closer to 21 than that of the dealer, without going over 21. Other players at the table are of no concern. Your hand is strictly played out against the hand of the dealer. The rules of play for the dealer are strictly dictated, leaving no decisions up to the dealer. Therefore, there is not a problem with the dealer or any of the other players at the table seeing the cards in your hand. Indeed, if you're playing at a shoe game, the player cards are all dealt face up.

Values of the cards

In blackjack, the cards are valued as follows:

An Ace can count as either 1 or 11, as demonstrated below.

The cards from 2 through 9 are valued as indicated.

The 10, Jack, Queen, and King are all valued at 10.

The suits of the cards do not have any meaning in the game.

The value of a hand is simply the sum of the point counts of each card in the hand.

For example, a hand containing (5,7,9) has the value of 21. The Ace can be counted as either 1 or 11. You need not specify which value the Ace has. It's assumed to always have the value that makes the best hand. An example will illustrate: Suppose that you have the beginning hand (Ace, 6). This hand can be either 7 or 17. If you stop there, it will be 17. Let's assume that you draw another card to the hand and now have (Ace, 6, 3).

Your total hand is now 20, counting the Ace as 11. Let's backtrack and assume that you had instead drawn a third card which was an 8. The hand is now (Ace, 6, 8) which totals 15.Notice that now the Ace must be counted as only 1 to avoid going over 21.

Applying ANN to Blackjack

What is a neural net?

A neural net is a collection of ***nodes*** which collectively perform a particular kind of computation. Each node is a small computational module and the nodes can operate in parallel. The operation of the network depends on the interactions between the nodes and the adaptation of their connective .strengths Some of the key features of neural nets includes:

- Nonlinear computation at each node
- Variable weights between nodes
- A learning rule by which the weights are adapted

What does a neural net do?

In particular, this project examines the ability of neural networks to perform learning of blackjack dealer strategies. The input to the network is a not so optimal but a naïve neural input. The job of the network is to learn the strategy by playing numerous games with a random player and later those can be played with an opponent. The present neural network would modify its weight according to two strategies. The first one would be simple adjustment to the weights and this incorporates a hit and trial method to modify the weights or adjust them when the dealer win or lose. The other mechanism involves learning rate and the reward and punish mechanism. The simpler model plays against a random player. The later model is made to play with a player who follows the same neural inputs.

Applying Reward Mechanism (Reinforcement Signal) to Blackjack

During Reward:

W(s) à (W(s) * alpha) + R(s) …………………Equation (1)

During Punish:

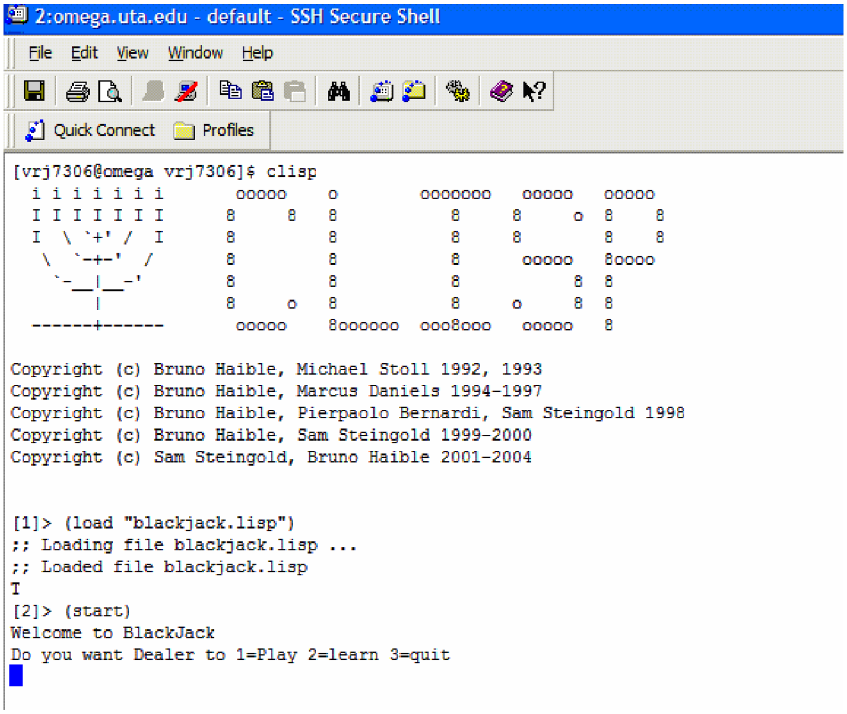
W(s) à (W(s) *alpha) – L(s) ……………………Equation(2)

W(s) à Weight for particular state (here state is sum of cards);

Alpha à learning rate

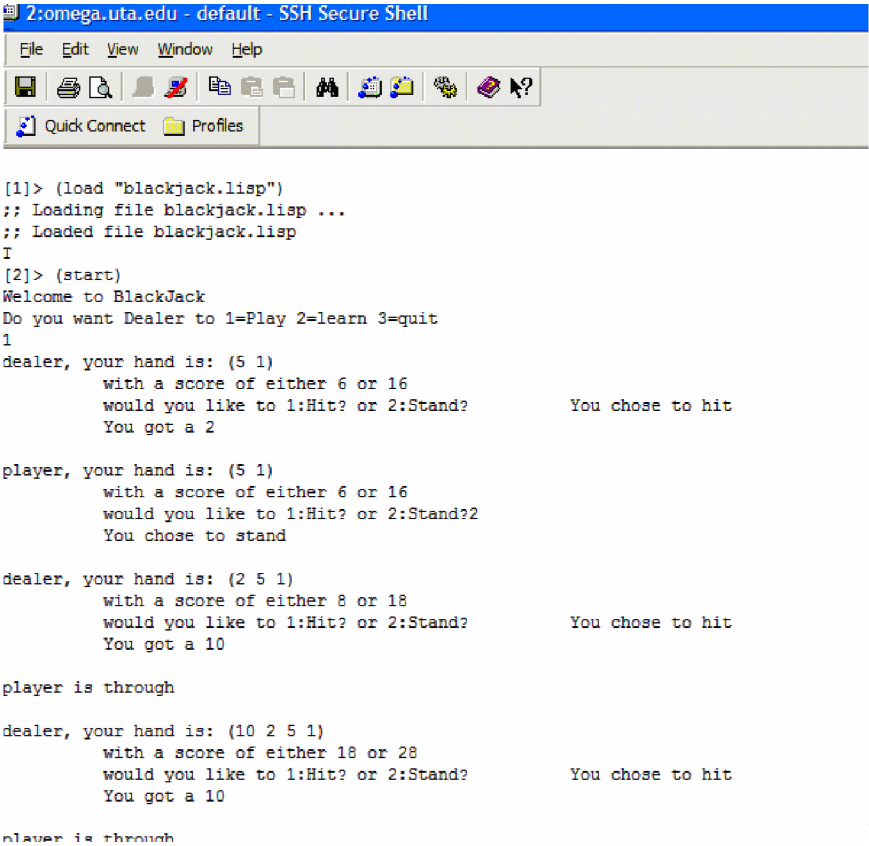
R(s) à Reward ; L(s) à lose

Sample Outputs:



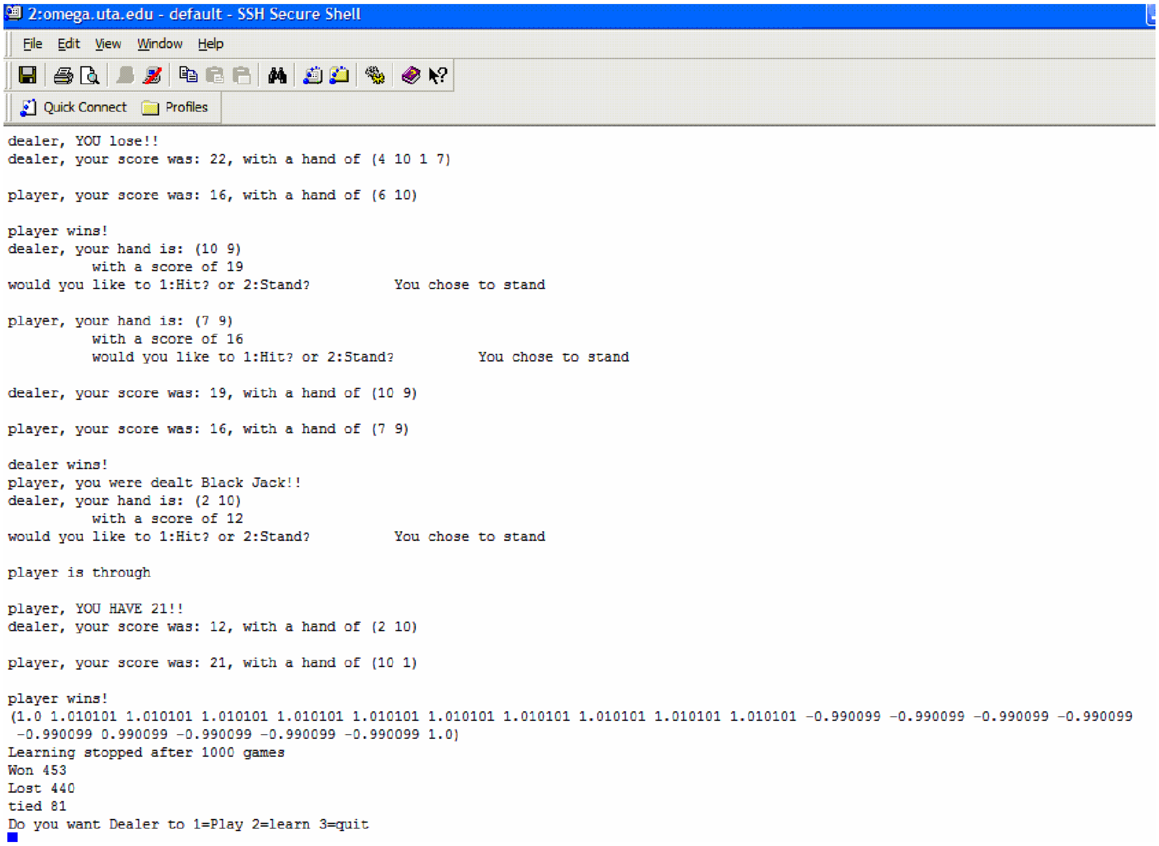
```
[vrj7306@omega vrj7306]$ clisp
  i i i i i i i       ooooo    o        ooooooo   ooooo   ooooo
  I I I I I I I      8     8   8           8     8     o  8    8
  I  \ `+' /  I      8         8           8     8        8    8
   \  `-+-'  /       8         8           8      ooooo   8oooo
    `-__|__-'        8         8           8           8  8
        |            8     o   8           8     o     8  8
  ------+------       ooooo    8oooooo  ooo8ooo   ooooo   8

Copyright (c) Bruno Haible, Michael Stoll 1992, 1993
Copyright (c) Bruno Haible, Marcus Daniels 1994-1997
Copyright (c) Bruno Haible, Pierpaolo Bernardi, Sam Steingold 1998
Copyright (c) Bruno Haible, Sam Steingold 1999-2000
Copyright (c) Sam Steingold, Bruno Haible 2001-2004


[1]> (load "blackjack.lisp")
;; Loading file blackjack.lisp ...
;; Loaded file blackjack.lisp
T
[2]> (start)
Welcome to BlackJack
Do you want Dealer to 1=Play 2=learn 3=quit
```

Figure 1: Shows a successful loading of the file



```
[1]> (load "blackjack.lisp")
;; Loading file blackjack.lisp ...
;; Loaded file blackjack.lisp
T
[2]> (start)
Welcome to BlackJack
Do you want Dealer to 1=Play 2=learn 3=quit
1
dealer, your hand is: (5 1)
          with a score of either 6 or 16
          would you like to 1:Hit? or 2:Stand?          You chose to hit
          You got a 2

player, your hand is: (5 1)
          with a score of either 6 or 16
          would you like to 1:Hit? or 2:Stand?2
          You chose to stand

dealer, your hand is: (2 5 1)
          with a score of either 8 or 18
          would you like to 1:Hit? or 2:Stand?          You chose to hit
          You got a 10

player is through

dealer, your hand is: (10 2 5 1)
          with a score of either 18 or 28
          would you like to 1:Hit? or 2:Stand?          You chose to hit
          You got a 10

player is through
```

Figure2: Sample Play.


2:omega.uta.edu - default - SSH Secure Shell
File Edit View Window Help
Quick Connect
Profiles

```
dealer, YOU lose!!
dealer, your score was: 22, with a hand of (4 10 1 7)

player, your score was: 16, with a hand of (6 10)

player wins!
dealer, your hand is: (10 9)
          with a score of 19
would you like to 1:Hit? or 2:Stand?          You chose to stand

player, your hand is: (7 9)
          with a score of 16
          would you like to 1:Hit? or 2:Stand?          You chose to stand

dealer, your score was: 19, with a hand of (10 9)

player, your score was: 16, with a hand of (7 9)

dealer wins!
player, you were dealt Black Jack!!
dealer, your hand is: (2 10)
          with a score of 12
would you like to 1:Hit? or 2:Stand?          You chose to stand

player is through

player, YOU HAVE 21!!
dealer, your score was: 12, with a hand of (2 10)

player, your score was: 21, with a hand of (10 1)

player wins!
 (1.0 1.010101 1.010101 1.010101 1.010101 1.010101 1.010101 1.010101 1.010101 1.010101 1.010101 -0.990099 -0.990099 -0.990099 -0.990099
  -0.990099 0.990099 -0.990099 -0.990099 -0.990099 1.0)
Learning stopped after 1000 games
Won 453
Lost 440
tied 81
Do you want Dealer to 1=Play 2=learn 3=quit
```

Figure 3: Shows after a learning the modifications in the weights.

Experimental Results

The implementation is a simplified version of the blackjack game. The cards are shuffled before each hand. An Ace is automatically valued as 11, unless it would cause a bust. In that case it is valued as 1. The efficiency can be observed from the percentage of winning and the percentage of draw games. The comparison of the play's random moves versus the dealer's learned moves is also included in the game. The learned strategy with the normal mechanism implements the traditional over 17 rule which would hit if the sum of the cards in hand is less than 17 or else it would stand. We can also experiment with the variations in the learning rate.

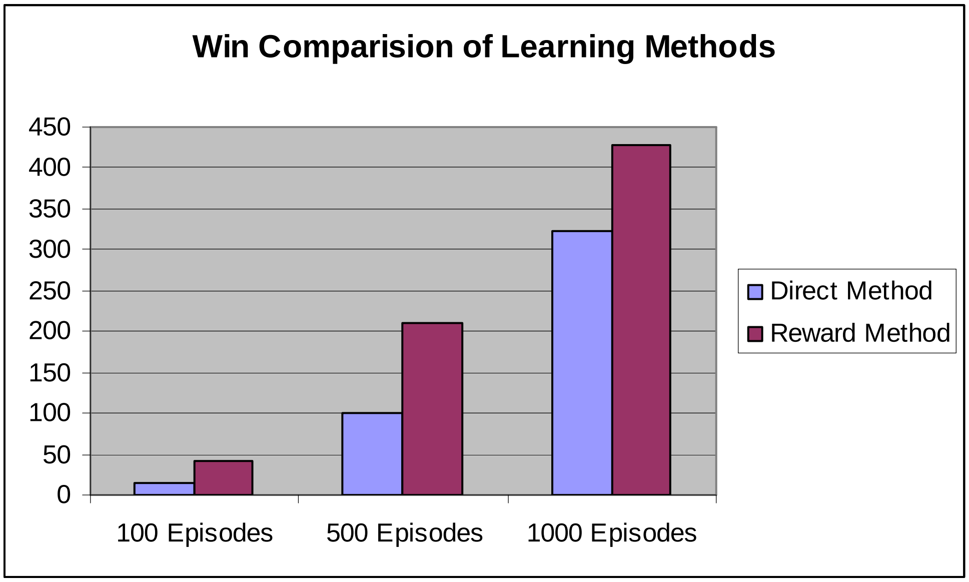
The learned strategy with the reward mechanism is very conservative, i.e., not to hit if the score is larger than 11. However, it is interesting to note how the algorithm determines such threshold without knowing the rules of the game, nor the goal of the game (just by experience and the reinforcement signal at the end of each hand).

Comparison of Win & Loss for the learning methods:

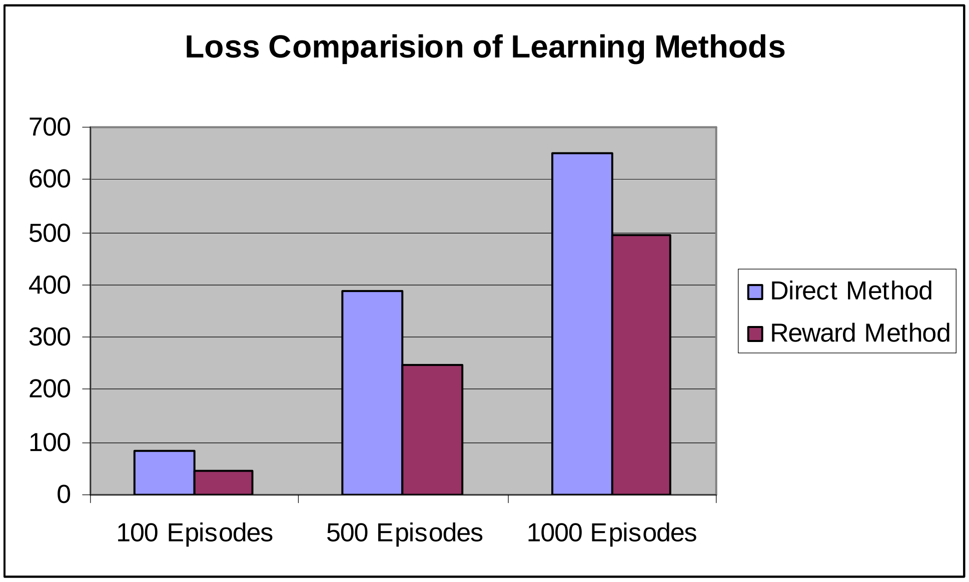
The learning was done with the simple method and the reward mechanism method. The observed the following win and loss while learning.

Strategy	100 episodes	500 episodes	1000 episodes
win			
Direct	14	100	322
Reward	42	210	428
Loss			
Direct	85	387	649
Reward	47	247	493
Tie			
Direct	2	14	30
Reward	12	44	80

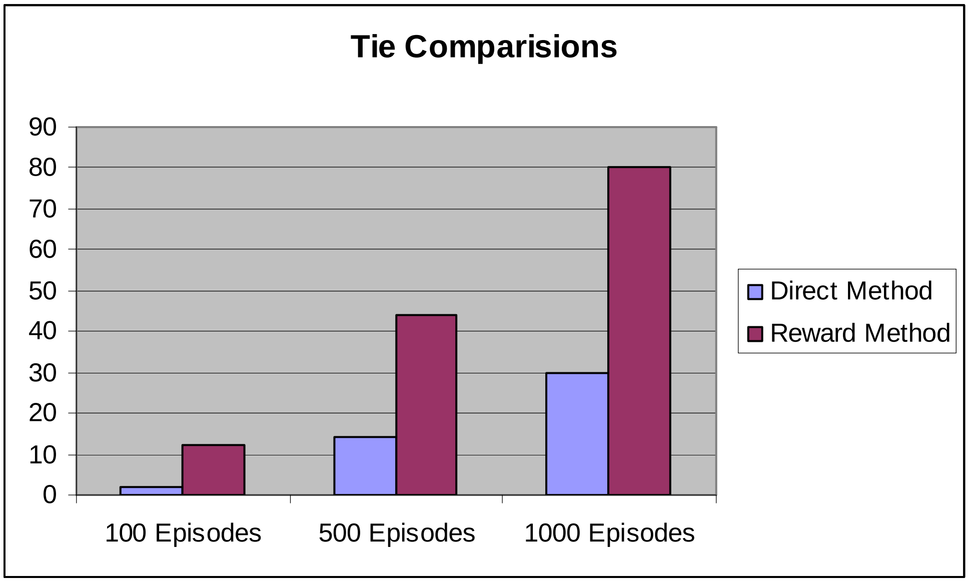
Graphical View:



Graph 1: Shows the comparison of the two learning methods with the number of wins over 100,500 and 1000 episodes of play.



Graph 2: Shows the comparison of the two learning methods with the number of loses over 100,500 and 1000 episodes of play.



Graph 3: Shows the comparison of the two learning methods with the number of ties over 100,500 and 1000 episodes of play.

We can experiment more by running both the methods one after another.

Conclusion

This project explored the methods of Artificial Neural Network in Blackjack for learning strategies without explicitly teaching the rules of the game. From the above experiment, it can be concluded that the reward mechanism learning strategy (Reinforcement signaling) does learn better strategies than the fixed rule strategy in playing Blackjack, as it improves the winning of each game dramatically. For future work the inclusion of reinforcement learning and also including the player's action in decision making would greatly enhance the blackjack game play. Reinforcement learning algorithms would be able to suggest the best actions when the situation changes over time. It finds a mapping from different states to probability distributions over various actions. Without knowing any rules of Blackjack, the ANN is able to performance well in playing the game, and more over the randomness of the game also makes it more unpredictable.

The performance of playing with ANN has an average of 42% wining, a minimum of 46% losing and 8% of tie when simulated play with the reward mechanism in 1000 games. The reward mechanism is approximately 31 % more efficient than the normal mechanism. However, as the number of games increases, the number of un-encountered situations increases, which include the randomness of the game. Therefore, the average winning drops down as the game progresses. Continue learning is important to produce good results. As the number of game increases, the game strategies will change over time. The more the game play, the better the performance will be. Finally, we see that, the winning percentage increase as this program uses a neural net to learn how to play blackjack.

Acknowledgements

I would like to thank my Professor Drs. Diane Cook for her continuous support and suggestions during this making of the project. I would also like to thank all my friends, family and other people who have assisted me with motivation during the project.

Related Work

In the early 1970s, Bernard Widrow used the game of blackjack to make the idea of one of the first reinforcement learning algorithms called *selective bootstrap adaptation*. He demonstrated how an Adaptive Linear Element learned with a critic to play blackjack without knowing the game and the objective of play. More recently, in neural networks and Temporal Differences learning were applied to learning play games from experience. Finally, in Monte Carlo methods have been used to learn value-functions of the game of Black Jack.

References

- De Wilde, Philippe (1995). *Neural network models: an analysis,* Springer-Verlag
- M.L littman “Markov games as a framework for multi-agent reinforcement learning”, Proceedings of the Eleven International Conference on Machine Learning, 1994, pp. 157 – 163. Morgan Kaufmann.
- B. Widrow, N. Gupta and S. Maitra, “Punish/Reward: Learning with a Critic in Adaptive Threshold System”, IEEE Transactions on Systems, Man and Cybernetics, vol 3, no. 5, pp. 455 – 465, 1973.
- Haykin, Simon (1999). *Neural networks: a comprehensive foundation,*2nd edition, Prentice Hall, New Jersey
- A. Perez-Uribe and E. Sanchez, "Blackjack as a Test Bed for Learning Strategies in Neural Networks", Proceedings of the IEEE International Joint Conference on Neural Networks IJCNN'98.
- B. Widrow, N. Gupta, and S. Maitra, ``Punish/Reward: Learning with a Critic in Adaptive Threshold Systems,'', IEEE Transactions on Systems, Man and Cybernetics, vol. 3, no.5, pp. 455--465, 1973.
- K. Olson, ``Learning to play games from experience: An application of artificial neural networks and temporal difference learning,'' M.S. thesis, Pacific Lutheran University, Washington, 1993.

- J.C.H Watkins and P. Dayan, ``Technical note q-learning,'', Machine Learning , vol. 8, pp. 279--292, 1992.
- G. Rummery and M. Niranjan, ``On-line q-learning using connectionist systems,'',Tech. Rep. Technical Report CUED/F-INFENG/TR 166, Cambridge, University Engineering Department, 1994.
- M.L. Littman, `Markov games as a framework for multi-agent reinforcement learning,'', Proceedings of the Eleven International Conference on Machine Learning, 1994, pp. 157--163, Morgan Kaufmann.
- R.S. Sutton and A.G. Barto, Reinforcement Learning: An Introduction, MIT Press, 1998.
- http://hampshire.edu/lspector/ .
- http://www.geocities.com/fastiland/ for some definitions and explanations.
- Google search for images and basic terminology and definitions and other required explanations.