National University of Computer and Emerging Sciences



Lab Exercise 06

CL461-Artificial Intelligence Lab

|  |  |
| --- | --- |
| Course Instructor | Dr. Kashif Zafar |
| Lab Instructor (s) | Saddam Khalil  Ahmad Abdullah |
| Section | D |
| Semester | Spring 2021 |

Department of Computer Science

FAST-NU, Lahore, Pakistan

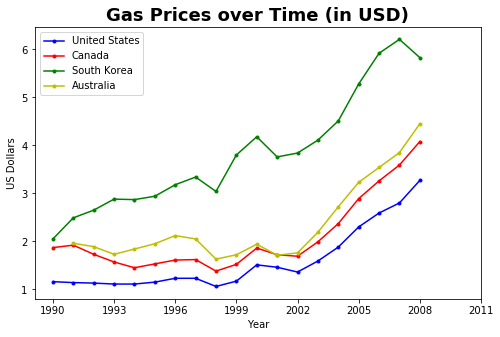
# Exercise (20 Marks)

Attempt all tasks given below.

## Make a Gas Price Line Graph (10 Marks)

Consider the following dataset: **gas\_prices.csv**

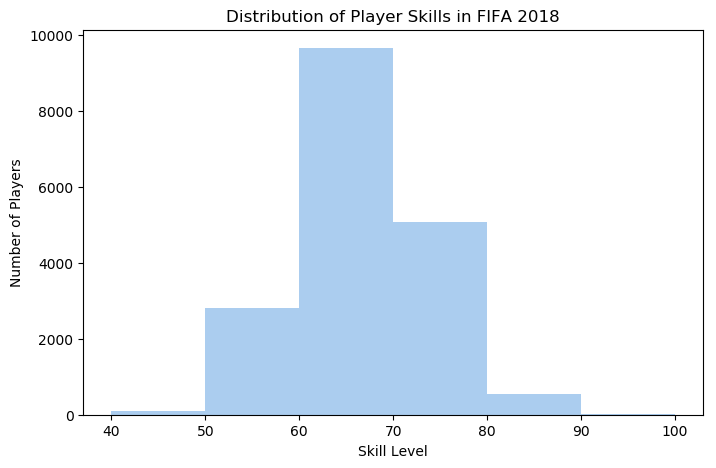
Make a line graph using matplotlib similar to the one shown below:



## Make a FIFA Player Skill Histogram (10 Marks)

Consider the following dataset: **fifa\_data.csv**

Make a histogram using matplotlib similar to the one shown below:



## Minmax (Tic Tac Toe, 3x3)

Tic Tac Toe is a game in which two players seek in alternate turns to complete a row, a column, or a diagonal with either three O's or three X's drawn in the spaces of a grid of nine squares (we take 2d array of 3x3) for this purpose. Print the grid as:

O | O | X

-----------------

X | O | X

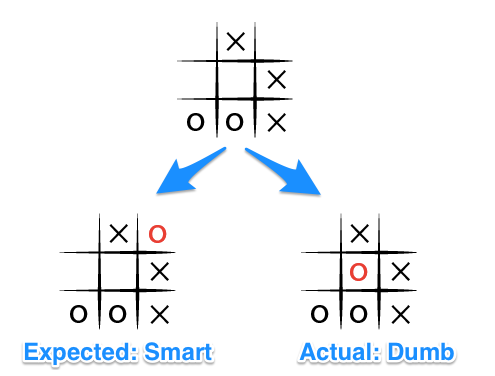
-----------------

X | X | O

O-Wins!

1. **Implement simple Tic Tac Toe for 2 players**
2. **AI-based Toc Tac Toe (Computer vs Human)**

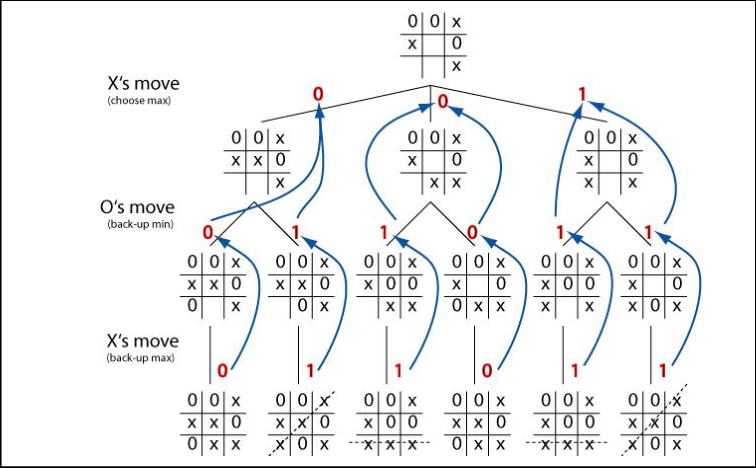
To make the game unbeatable, it is necessary to create an algorithm that could calculate all the possible moves available for the computer player and use some metric to determine the best possible move. So, the Minimax algorithm is a good step to start working.



**We expect Computer to choose smartly.**

**The execution of the Minmax algorithm for solving Tic-Tac-Toe:**

it works by visualizing all future possible states of the board and constructs it in the form of a tree. When the current board state is given to the algorithm (the root of the tree), it splits into ’n’ branches (where n denotes the number of moves that can be chosen by the AI/number of empty cells where the AI can be placed). If any of these new states is a terminal state, no further splits are performed for this state.



If there are no terminal states, each of these new states is considered a new root and they give rise to a tree of their own. But, there’s a catch, since this is a 2-player game and the players take turns alternatively, therefore whenever we go one layer deeper in the network, we need to change the player which would be placed in one of the empty cells. This way we can visualize what moves the other player would take as a result of our move. The algorithm evaluates the best move to take by choosing the move which has the maximum score when it is the Computer’s turn and choosing the minimum score when it is the Human’s turn. This is how min-max will work for the tic tac toe.

**This is how your game should work. (sample run)**

New Game!  
----------------  
| || || |  
----------------  
| || || |  
----------------  
| || || |  
----------------

Choose which player goes first - X (You - the petty human) or O(The mighty AI): **O**

AI plays move: 2  
----------------  
| || O || |  
----------------  
| || || |  
----------------  
| || || |  
----------------

Oye Human, your turn! Choose where to place (1 to 9): 3  
----------------  
| || O || X |  
----------------  
| || || |  
----------------  
| || || |  
----------------  
AI plays move: 9  
----------------  
| || O || X |  
----------------  
| || || |  
----------------  
| || || O |  
----------------

Oye Human, your turn! Choose where to place (1 to 9): 5  
----------------  
| || O || X |  
----------------  
| || X || |  
----------------  
| || || O |  
----------------  
AI plays move: 7  
----------------  
| || O || X |  
----------------  
| || X || |  
----------------  
| O || || O |  
----------------

Oye Human, your turn! Choose where to place (1 to 9): 6  
----------------  
| || O || X |  
----------------  
| || X || X |  
----------------  
| O || || O |  
----------------  
AI plays move: 4  
----------------  
| || O || X |  
----------------  
| O || X || X |  
----------------  
| O || || O |  
----------------

Oye Human, your turn! Choose where to place (1 to 9): 1  
----------------  
| X || O || X |  
----------------  
| O || X || X |  
----------------  
| O || || O |  
----------------  
AI plays move: 8  
----------------  
| X || O || X |  
----------------  
| O || X || X |  
----------------  
| O || O || O |  
----------------  
Draw!

# Submission Instructions

Always read the submission instructions carefully.

* Rename your Jupyter notebook to your roll number and download the notebook as **.ipynb** extension.
* To download the required file, go to **File->Download .ipynb**
* Only submit the **.ipynb** file. DO NOT **zip** or **rar** your submission file
* Submit this file on Google Classroom under the relevant assignment.
* Late submissions will not be accepted