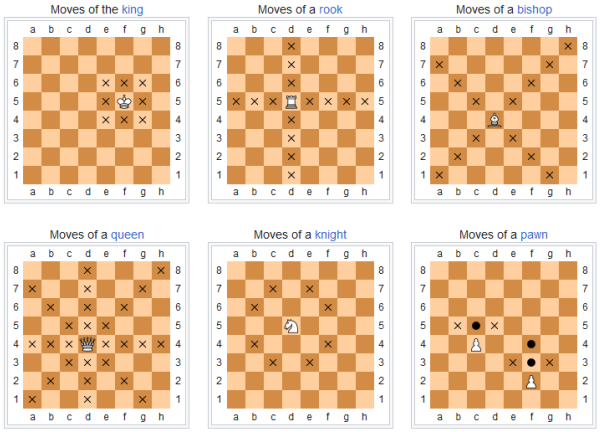
Reinforcement Learning Project

Welcome to your Reinforcement Learning project! **Join in groups of a maximum of 4 students** on a project focused on developing an RL agent capable of playing chess at a strategic level. **The project deadline has been set to the 18th of June.**

Chess has long been considered a benchmark for measuring AI capabilities, and this project aims to leverage the power of RL to create an intelligent agent that can make optimal decisions in complex chess positions. By combining the principles of reinforcement learning with the rich strategic domain of chess, you will explore new approaches to create the most effective chess player.



Project Objectives:

- Train an RL agent to play chess: The primary objective of this project is to develop an RL agent that can play chess at a high level of proficiency. The agent should be capable of evaluating chess positions and making strategic decisions.
- Optimize decision-making using RL algorithms: Explore different RL algorithms, as seen in class, to train the agent. Compare and analyse their effectiveness in learning and decision-making capabilities in the context of chess.
- Use a challenging chess environment: Use a comprehensive environment for the agent to interact with, representing the rules and dynamics of chess. This environment will provide a realistic and challenging setting for the agent's training and evaluation.
- Evaluate and benchmark performance: Assess the performance of the RL agent against different benchmarks from an existing chess engine (Stockfish).

Extra Objectives:

- Investigate transfer learning and generalization: Explore techniques for transfer learning to leverage knowledge acquired in related domains or from pre-training on large chess datasets. Investigate the agent's ability to generalize its knowledge.
- Enhance interpretability and analysis: Develop methods to analyse the agent's decision-making process and provide insights into its strategic thinking. Investigate techniques to visualize the agent's evaluation of chess positions and understand its reasoning behind specific moves.

Deliverables

Code

- RL Agent: The primary deliverable of this project is a fully trained RL agent capable of playing chess at a strategic level. This deliverable consists of a Jupyter notebook.
- Documentation: Provide a well-documented Jupyter notebook that includes the implementation of the RL agent, chess environment, and any supporting scripts or utilities attached. The documentation should cover the setup instructions, usage guidelines, and explanations of key components and algorithms used in the project.

Report

A report consisting of a maximum of 5 pages (without cover or bibliography) with:

- An introduction describing your overall approach to the problem, taking into consideration that the chess environment is tabular in nature, the issues associated with the overall sheer number of states and more.
- A methodological description of all different learning algorithms used and the respective reasoning for their choice.
- A comprehensive evaluation that analyses the performance of the RL agent against the different learning algorithms. You are given in a Jupyter notebook the functions required to benchmark your algorithm against Stockfish, a chess engine.
- Don't forget to develop useful visualizations to provide a better understanding of the agent's decision-making process.

Project Evaluation

- The usage and argumentation of different learning methods will be the main focus of the project's evaluation process. Choose an appropriate number of learning algorithms (does not need to be more than 3) and argue their relevance and usefulness in solving the problem.
- While the overall performance of your RL agent is not to be part of the evaluation, it is rather essential to understand how different learning methods improve the performance of your agent.
- It is important that you develop a comprehensive evaluation to the extent to which the agent's reasoning can be understood. Assess the clarity and interpretability with visualizations.
- You can achieve the maximum score without the Extra Objectives, which means in practice groups can have "more than 20" in the project.

Useful Materials

- Most of the inspiration for this project comes from DeepMind's AlphaZero project (https://www.deepmind.com/blog/alphazero-shedding-new-light-on-chess-shogi-and-go) a single system that taught itself from scratch how to master the games of chess, shogi (Japanese chess), and Go, beating a world-champion program in each case. You can see in the link provided, several materials and research that might inspire you as well.
- We probably will not have time throughout the classes to introduce transfer learning in RL, nevertheless for those interested in going for the extra materials you can find materials in the following link (https://www.deepmind.com/open-source/alphazero-resources).