# Machine Learning for Chess

By Ori Yonay

# Machine Learning for Chess: Motivation

- Chess has been at the heart of AI and a common benchmark of its progress
  - A complex game with many rules and difficult positions (2,000 years old and we're still learning a lot about it!)
- Traditionally, chess programs employ a brute-force approach
- Chess is a much more nuanced game that needs a more human, hands-free approach
  - Human knowledge is a **limiting factor** in chess engines!
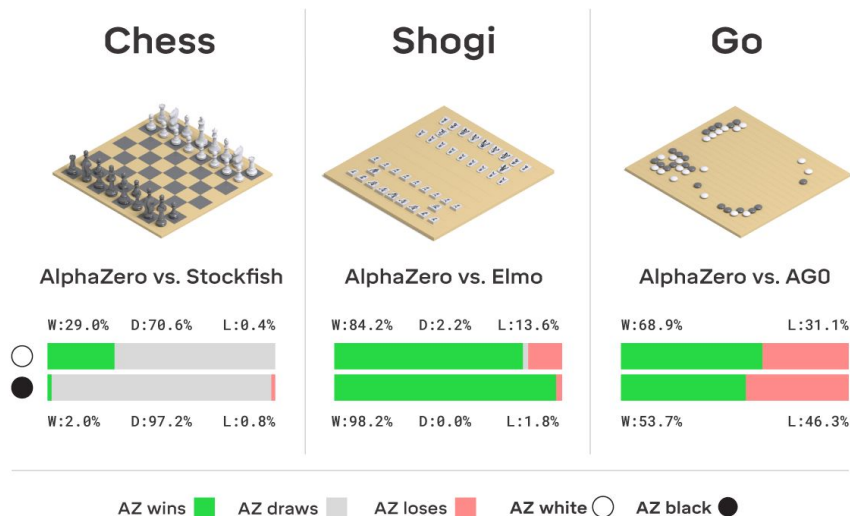
# Traditional Methods for Computer Chess

- Chess is traditionally approached as a **tree** rather than as a **function**
- Minimax/Alpha-beta pruning algorithm
- Flaw: requires human knowledge of the game!
- Another flaw: not truly 'intelligent'!
  - Acts more as a giant calculator than an intelligent agent
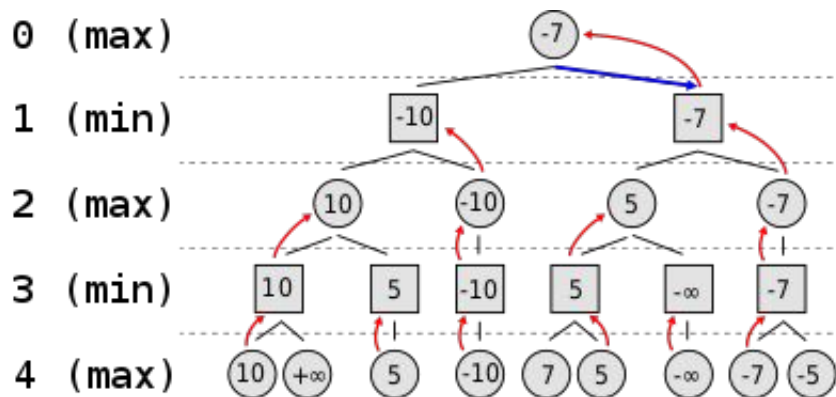
# AlphaZero and LeelaChess

- In 2017, Google's **DeepMind** revolutionized computer chess (and the ML field altogether) with AlphaGo
- AlphaZero generalized AlphaGo's original algorithm to perform on many types of board games!
- LeelaChess aims to be an open-source version of AlphaZero for chess

**Chess** | **Shogi** | **Go**

AlphaZero vs. Stockfish

W:29.0%   D:70.6%   L:0.4%

W:2.0%   D:97.2%   L:0.8%

AlphaZero vs. Elmo

W:84.2%   D:2.2%   L:13.6%

W:98.2%   D:0.0%   L:1.8%

AlphaZero vs. AG0

W:68.9%   L:31.1%

W:53.7%   L:46.3%

AZ wins   AZ draws   AZ loses   AZ white ○   AZ black ●

# AlphaZero and LeelaChess: Network Architecture

- Both AlphaZero and LeelaChess use a convolutional neural network
  - Why?
- Training: Monte Carlo Tree Search
- AlphaZero vs. Stockfish
  - 80,000 vs. 70,000,000 positions per second. Who will win?
  - Why? AlphaZero works **smarter**, not harder

# My own Chess Neural Network

- As a culmination of my research regarding machine learning for chess, I have designed my own machine learning model
- **Approach**: combine classic chess-engine methods with a supervised machine learning approach for evaluating static positions (leaf nodes)
- **Dataset**: CSV file containing 13-million chess positions and their respective evaluations (using Stockfish with a 22-move lookahead)

# Demo! (Code walkthrough)

# Conclusion

- **Machine Learning vs. Traditional AI methods**

- **Importance of convolutional neural networks**

- **Future Directions of my chess 'engine'**

- **The future of computer chess?**

Thank you!