# ANALYSIS OF MACHINE LEARNING APPLIED TO BOARD GAMES

Adam Amanbaev, Jonathan Hallström, Alvar Edvardsson
Hugo Åkerfeldt, Romeo Patzer

Supervisor: Ulf Backlund

**ABSTRACT**

In late 2017 DeepMind announced a groundbreaking system in a preprint [1] and the results were astonishing. The system was called AlphaZero and utilized *artificial neural networks* in combination with *heuristic algorithms* in order to teach itself the game chess without any proprietary knowledge. After approximately 9 hours it was able to beat the strongest handcrafted engines, such as Stockfish and it had learned centuries of human knowledge of chess. In this paper we aim to study the effectiveness of different *neural networks* and *heuristic algorithms* such as the one used in AlphaZero. More precisely, we intend to analyze the efficiency of those networks and algorithms in combination with varying *optimizations, parameters, hyperparameters* and *architectures* applied to the classic games *Connect Four* and *Othello*.

**Keywords** — Machine Learning, Supervised Learning, Reinforcement Learning, Neural Network, Deep Learning

# Contents

# Summary of Notation

> I am a forest, and a night of dark trees: but he who is not afraid of my darkness, will find banks full of roses under my cypresses.
>
> — Friedrich Nietzsche, Thus Spoke Zarathustra

| | |
|---|---|
| $\text{argmax}_x f(x)$ | $\{x \mid f(x) = \max_{x'} f(x')\}$ |
| $\text{argmin}_x f(x)$ | $\{x \mid f(x) = \min_{x'} f(x')\}$ |
| $\leftarrow$ | assignment |
| s, s’ | states |
| $\varepsilon$ | probability of taking a random action in an $\varepsilon$-greedy policy |
| $\gamma$ | discount-rate parameter |
| $S$ | set of all nonterminal states |

# Todo

Regularization: Tikhonov
Network Architecture: Depth-Breadth
Ensamble methods: Dropout
Vanishing/Exploding Gradient: ReLU, Sigmoid
Convergence:
Local Optima: Pretraining
Prove nonlinearity's importance: identity function
Parameter Reduction: Deeper networks
ResNet: Skip Connections
Long Short-Term Memory: "forgetting"

ConvNet: "forgetting", feature maps, pooling, ReLU
RecurrentNet: sequence of moves
Stochastic Gradient Descent: Mini-Batch, Point
Learning Rate Decay: Exponential and Inverse Decay
Momentum-Based Learning: Nesterov Momentum, AdaGrad, RMSProp,
AdaDelta, Adam
Acceleration and Data Compression: GPU, SIMD ...
Overfitting: Penalty Regularization, Ensamble Methods,
Early Stopping, Pretraining, Continuation Methods Imitation Learning: Supervised

# §1. Introduction

§1.1 Background

§1.2 Analysis

## References

[1] David Silver, Thomas Hubert, Julian Schrittwieser, Ioannis Antonoglou, Matthew Lai, Arthur Guez, Marc Lanctot, Laurent Sifre, Dharshan Kumaran, Thore Graepel, Timothy Lillicrap, Karen Simonyan, and Demis Hassabis. Mastering chess and shogi by self-play with a general reinforcement learning algorithm. *arXiv*, 2017.