Implementation of the Turing-Machine-Based Machine Learning Algorithm from "Discovering Problem Solutions with Low Kolmogorov Complexity and High Generalization Capability" - By Jürgen Schmidhuber (1994).

## The magnificent findings of this paper were that Input Tape restructuring the goal of machine learning from "modeling the data" to "modeling the algorithm that Program/Working Tape generated this data" allows for superior generalization ability. The paper builds from the foundation of Kolmogorov complexity and Solomonoff Induction -**Output Tape** often considered impractical, and integrates the universal and elegant Levin complexity to support a time-bounded search algorithm. **Context Manager** Main **Parameters** - Initial Program - Dataset - Generate Dataset - TM Update Algorithm - Generate Initial Program (Incremental (GA), Random, etc.) - Generate Context Manager - Stopping Conditions - Get Best Turing Machine - Performance Metric (Fitness fn) - Compute Test Loss Actions - Generate Turing Machine (TM) - Generate Executor -- Run TM on Executor - Evaluate Performance - Break Loop?

Custom Program/Working Tape -3 -2 -1 0 2 3 Program Tape Working Tape **Turing Machine Executor Parameters** - Training Data - Turing Machine - Stopping Conditions (Number of Steps) - Performance Metric (Fitness fn) Actions - Run Turing Machine - Enforce Stopping Conditions - Enforce Bounds on Tapes - Compute Training Performance - Return Performance, Stopping Status (Clean Exit/Halt, Error, Max Steps Reached) - Modify TM