# Algebraic Machine Learning

Fundação Champalimaud

David Méndez Martínez

AITP 2024

ALGEBRAIC AI

# What is Algebraic Machine Learning?

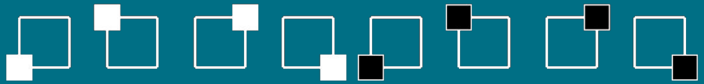
**Algebraic Machine Learning** (Martin-Maroto, de Polavieja, arXiv:1803.05252)

- Learning system inspired by Model Theory and Universal Algebra
- Data and prior knowledge are formally embedded as generating elements and relations in an algebraic structure
- Learning from data via iterative modification of the algebraic structure
  - No error minimization
  - Mathematically transparent
  - Capable of finding formal knowledge
  - No hyperparameter tuning

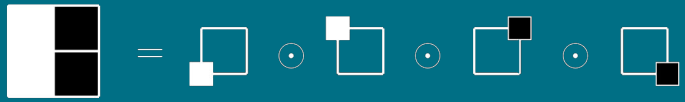
Let's give a short introduction!

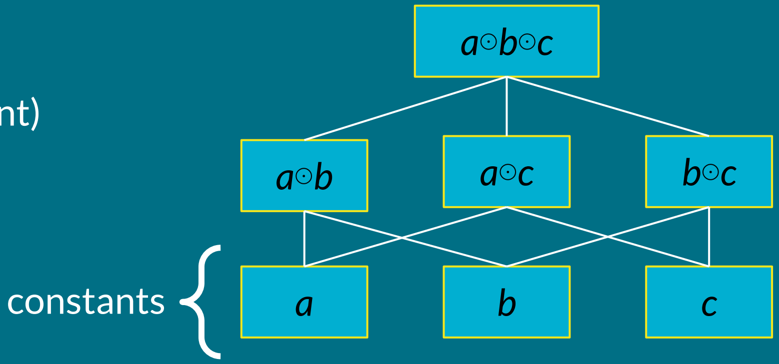
# AML components

- Algebraic structure: semilattices
  - Operation: ⊙ (associative, commutative, idempotent)
  - Order relation: $a \leq b \Leftrightarrow a \odot b = b$
- Generating elements: constants
  - Components of data



Datapoints can be represented using the operation

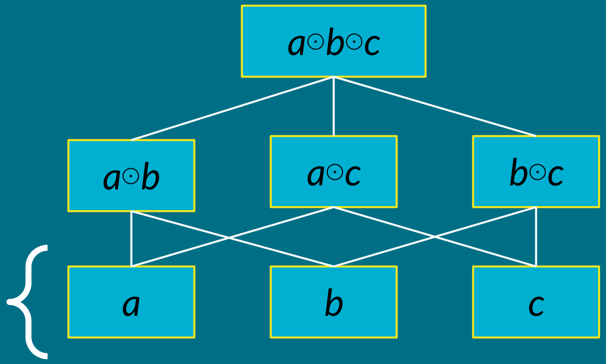




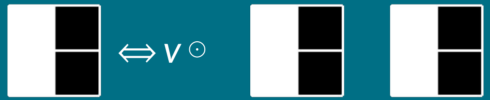
# AML components

- Algebraic structure: semilattices
  - Operation: $\odot$ (associative, commutative, idempotent)
  - Order relation: $a \leq b \Leftrightarrow a \odot b = b$
- Generating elements: constants
  - Components of data
  - Concepts (e.g. labels)

constants




*v*: vertical bar

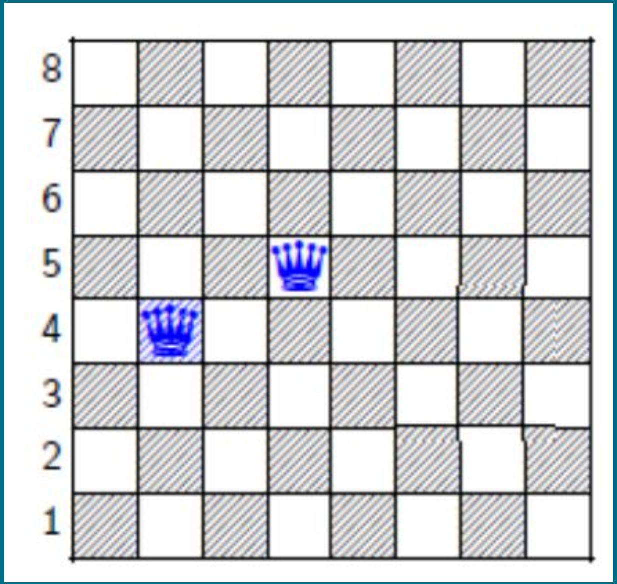
$v \leq$  $\Leftrightarrow v \odot$ "*v* is in the image"

$v \not\leq$  "*v* is not in the image"

# AML components

- Algebraic structure: semilattices
  - Operation: ⊙ (associative, commutative, idempotent)
  - Order relation: $a \leq b \Leftrightarrow a \odot b = b$
- Generating elements: constants
  - Components of data
  - Concepts (e.g. labels)

$$\forall x \forall y \, (R_x \odot C_y < Q_{xy})$$



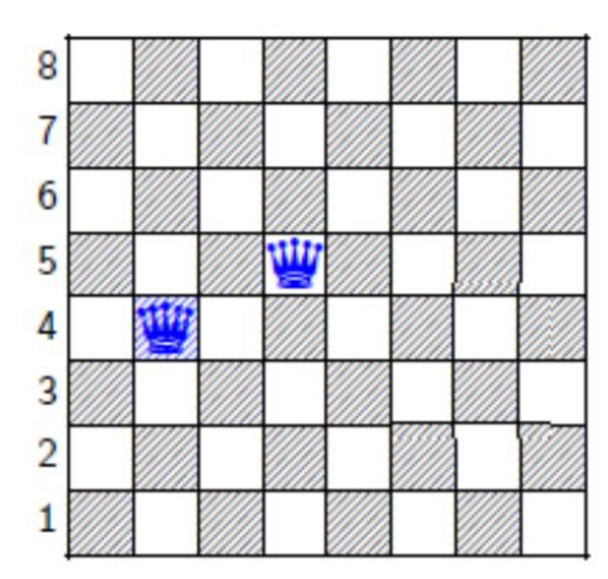

*N*-queens problem

# AML components

- Algebraic structure: semilattices
  - Operation: $\odot$ (associative, commutative, idempotent)
  - Order relation: $a \leq b \Leftrightarrow a \odot b = b$
- Generating elements: constants
  - Components of data
  - Concepts (e.g. labels)
- Prior knowledge and data: relations

Martin-Maroto - Polavieja, Semantic embeddings in Semilattices, arXiv:2205.12618

$$\forall x \forall y \, (R_x \odot C_y < Q_{xy})$$




*N*-queens problem

But how are they enforced?

# Atomized semilattices and crossing

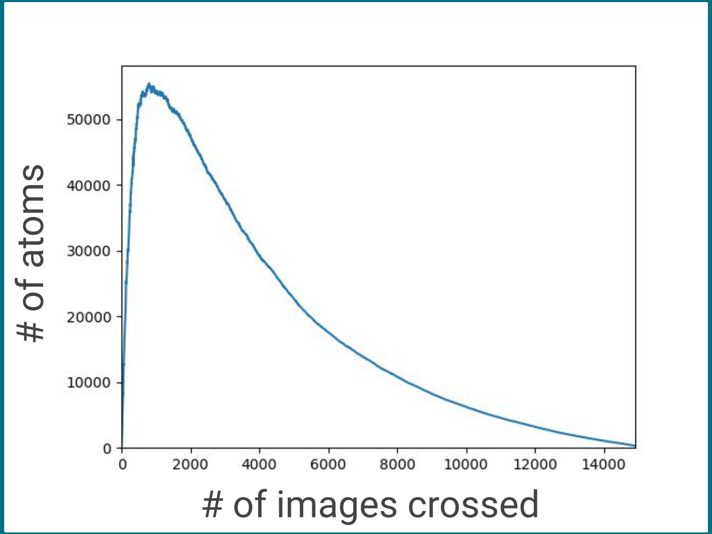
- **Atomization:** a set theoretical based representation of a semilattice
  Martin-Maroto - de Polavieja, Finite Atomized Semilattices, arXiv:2102.08050
  - Atoms $\phi$ are determined by a set of constants
  - $\phi < a \Leftrightarrow \exists\, c$ constant, $\phi < c \leq a$
  - $a \leq b \Leftrightarrow \nexists \phi,\ \phi < a$ and $\phi \not< b$ ($\phi$ discriminates $(a,b)$)

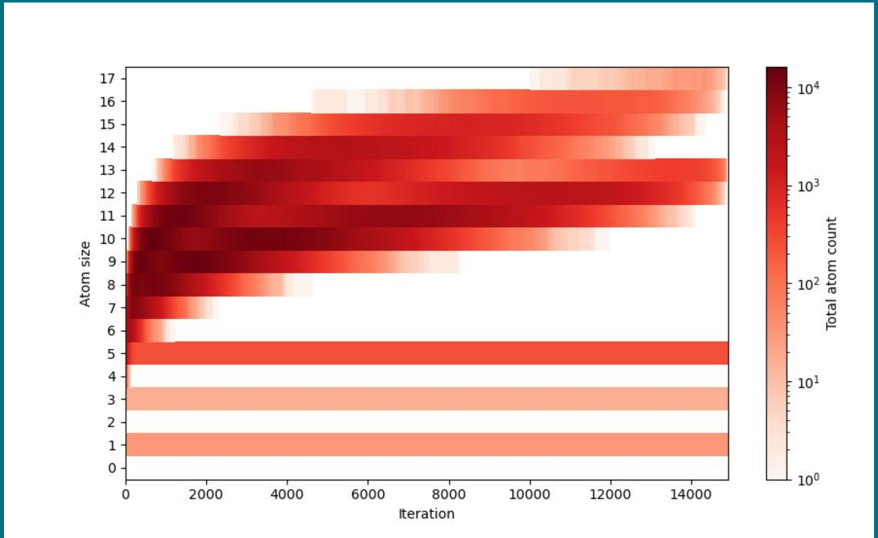
**Proposition.** Every finite semilattice can be represented through an atomization.

# Atomized semilattices and crossing

- **Atomization**: a set theoretical based representation of a semilattice
  Martin-Maroto - de Polavieja, Finite Atomized Semilattices, arXiv:2102.08050
  - Atoms $\phi$ are determined by a set of constants
  - $\phi < a \Leftrightarrow \exists\, c$ constant, $\phi < c \leq a$
  - $a \leq b \Leftrightarrow \nexists \phi,\ \phi < a$ and $\phi \not< b$ ($\phi$ discriminates $(a,b)$)
- **Crossing:** obtain a new semilattice verifying a new relation
  - Full crossing
    Freest semilattice → guaranteed to find rule in data, slower

# Atomized semilattices and crossing

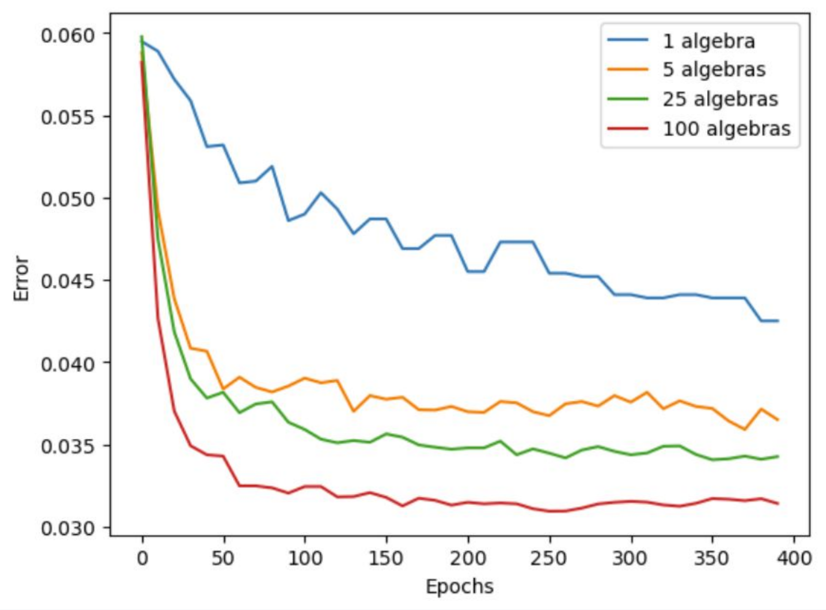


# Atomized semilattices and crossing

- **Atomization**: a set theoretical based representation of a semilattice
  Martin-Maroto - de Polavieja, Finite Atomized Semilattices, arXiv:2102.08050
  - Atoms $\phi$ are determined by a set of constants
  - $\phi < a \Leftrightarrow \exists\, c$ constant, $\phi < c \leq a$
  - $a \leq b \Leftrightarrow \nexists \phi$, $\phi < a$ and $\phi \nless b$ ($\phi$ discriminates $(a,b)$)
- **Crossing:** obtain a new semilattice verifying a new relation
  - Full crossing
    Freest semilattice → guaranteed to find rule in data, slower
  - Sparse crossing
    Less free, more logical consequences → better generalization
    Retain only some of the atoms (stochastic) → faster learning from data
    No overfitting
    Very parallelizable

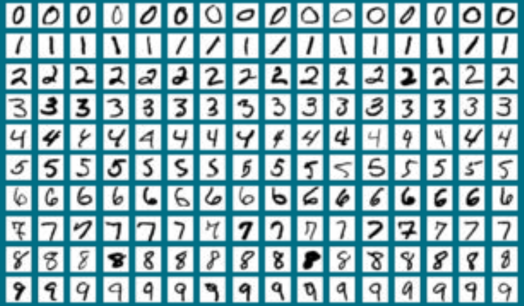
# Atomized semilattices and crossing




But does it work?

# Some benchmarks

MNIST as a benchmark: Handwritten digit classification



Binary classification: is it or is it not a given number?

- ➔ AML with no prior knowledge: < 0.5% error
- ➔ Multi-layer perceptron: < 0.6% error

General classification task

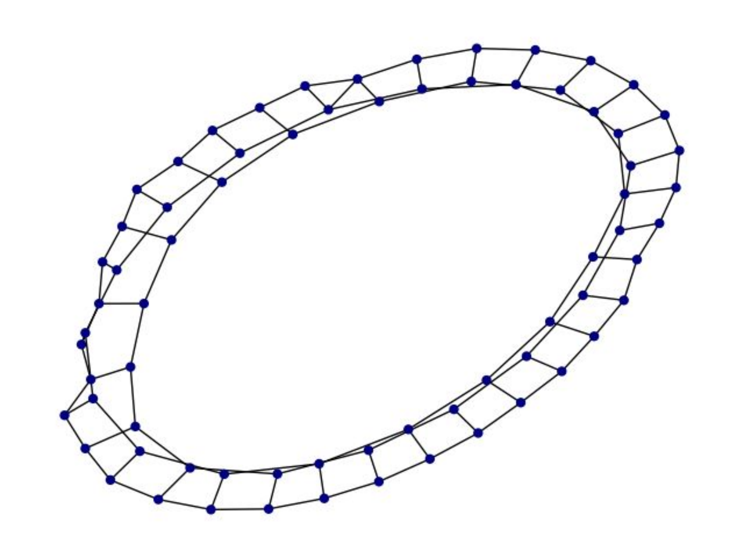
- ➔ AML with simple thresholding: ~97.6% accuracy
- ➔ AML with optimization: ~98.2% accuracy
- ➔ Multi-layer perceptron: ~98.3% accuracy
- ➔ Convolutional neural network: 99%+ accuracy

Open question: how can we effectively inject prior knowledge?

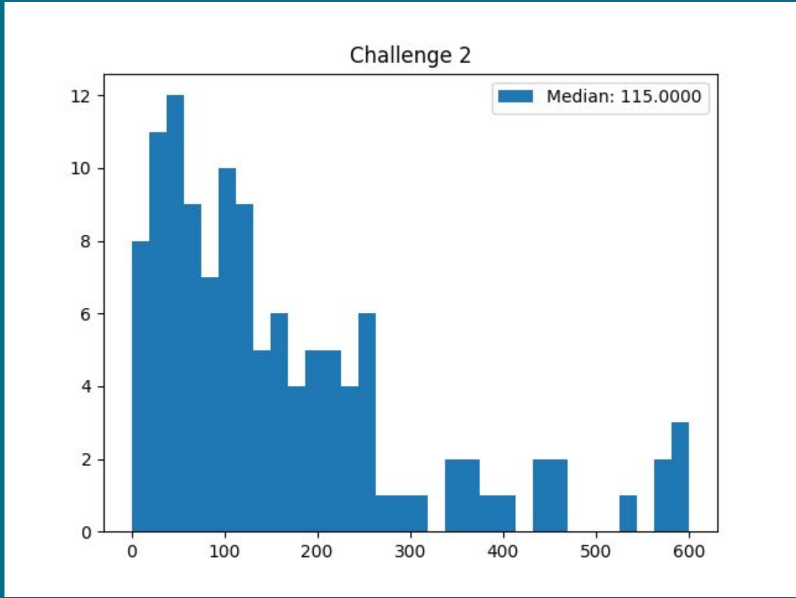
# Some benchmarks

Hamiltonian cycles

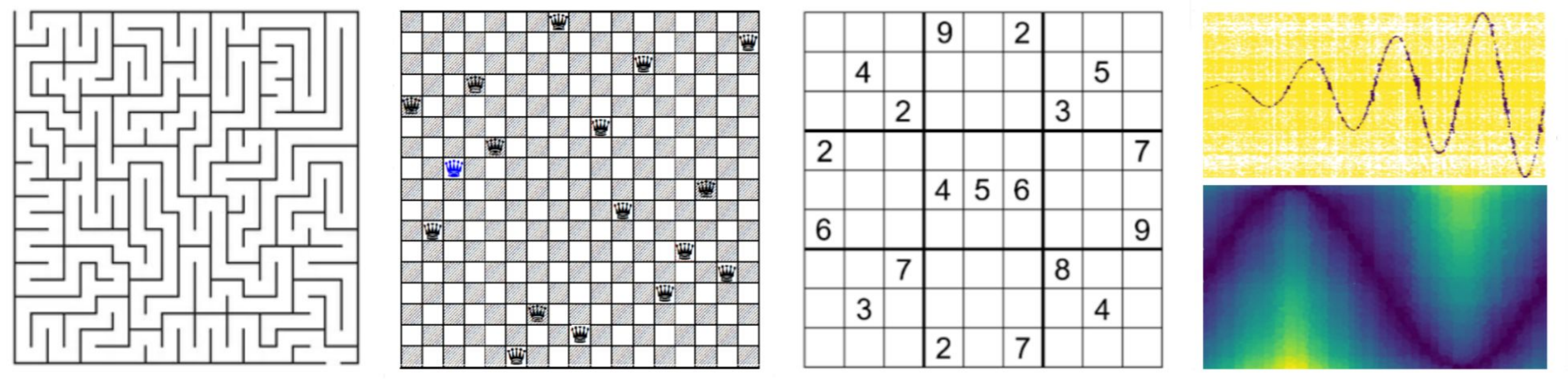
- Snakes and ladders: 5078 transforms
- Algebraic Machine Learning



Flinders Hamiltonian Cycle Project, graph 2



# Researching further applications



# Further research

- Real values/inference
  - Infinite Atomized Semilattices (F. Martin-Maroto, A. Ricciardo, D. Mendez, G. de Polavieja, arXiv:2311.01389)
  - Extended semilattices
- Automatic discovery of constants
- Statistical study of atom weights
- Training performance analysis/distances between models
- Automatic tools for model explanation
- Development of more user-friendly software

# Thank you for your attention

Fundação Champalimaud

ALGEBRAIC AI