

Machine Learning using Descriptive Complexity and Propositional Solvers

Skip Jordan and Łukasz Kaiser

ERATO Group, JST & Hokkaido University
LIAFA, CNRS & Université Paris Diderot

Highlights '13

Why Logic in Machine Learning?

What is machine learning?

Given $(x^{(i)}, y^{(i)})_{i=1\dots m}$ **find** $h \in H$ **such that** $h(x^{(i)}) \approx y^{(i)}$

Why Logic in Machine Learning?

What is machine learning?

Given $(x^{(i)}, y^{(i)})_{i=1\dots m}$ find $h \in H$ such that $h(x^{(i)}) \approx y^{(i)}$

Issues

- Is H **abstract** (e.g. PTIME) or **concrete** (e.g. $\{\theta \cdot \text{input} \mid \theta\}$)?
- Can we get **efficient algorithms** and **theoretical guarantees**?

Why Logic in Machine Learning?

What is machine learning?

Given $(x^{(i)}, y^{(i)})_{i=1\dots m}$ find $h \in H$ such that $h(x^{(i)}) \approx y^{(i)}$

Issues

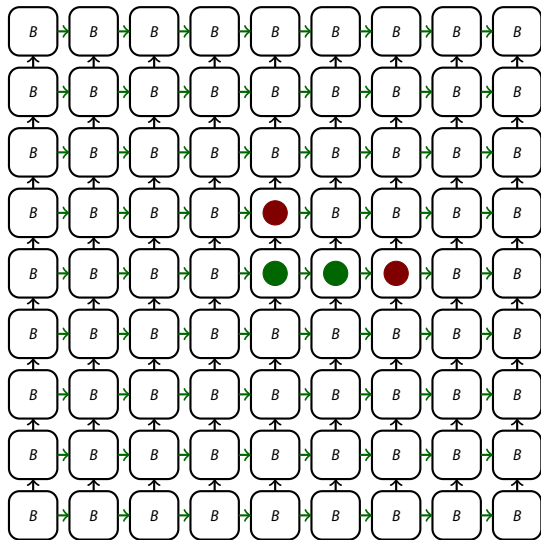
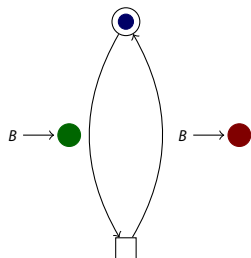
- Is H **abstract** (e.g. PTIME) or **concrete** (e.g. $\{\theta \cdot \text{input} \mid \theta\}$)?
- Can we get **efficient algorithms** and **theoretical guarantees**?

How can we use logic?

- Idea: H are **parametrized formulas**
- **Descriptive complexity** gives **theoretical guarantees**
- **Propositional solvers** used for **efficient learning**

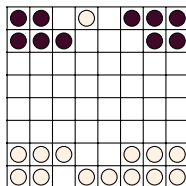
Learning Board Game Rules

Representing Board Games

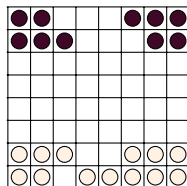


$$\exists x_1 \dots x_5 \left(\bigwedge_{1 \leq i \leq 5} G(x_i) \wedge \left(\bigwedge_{1 \leq i \leq 5} R(x_i, x_{i+1}) \vee \bigwedge_{1 \leq i \leq 5} C(x_i, x_{i+1}) \vee \right. \right. \\ \left. \left. \bigwedge_{1 \leq i \leq 5} \exists y (R(x_i, y) \wedge C(y, x_{i+1})) \vee \bigwedge_{1 \leq i \leq 5} \exists y (R(x_i, y) \wedge C(x_{i+1}, y)) \right) \right)$$

Learning Winning Conditions



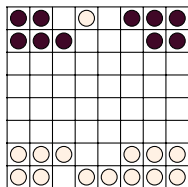
Positive Example \mathcal{A}



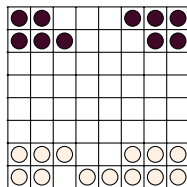
Negative Example \mathcal{B}

Find **minimal** φ such that $\mathcal{A} \models \varphi$, $\mathcal{B} \models \neg\varphi$

Learning Winning Conditions



Positive Example \mathfrak{A}



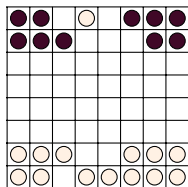
Negative Example \mathfrak{B}

Find **minimal** φ such that $\mathfrak{A} \models \varphi$, $\mathfrak{B} \models \neg\varphi$

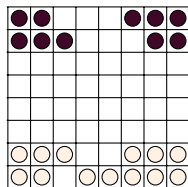
Which logic and minimality?

- Full FO, minimal quantifier rank: **PSPACE-complete** (Pezzoli '98)
- $\text{FO}^k + C$, minimal quantifier rank: **PTIME** (Grohe '99)
- $k = 16$ and $\log(n)$ quantifiers suffice for ... (Pikhurko, Verbitsky '10)

Learning Winning Conditions



Positive Example \mathfrak{A}



Negative Example \mathfrak{B}

Find **minimal** φ such that $\mathfrak{A} \models \varphi$, $\mathfrak{B} \models \neg\varphi$

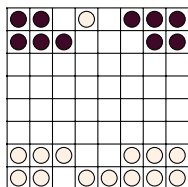
Which logic and minimality?

- Full FO, minimal quantifier rank: **PSPACE-complete** (Pezzoli '98)
- $\text{FO}^k + \text{C}$, minimal quantifier rank: **PTIME** (Grohe '99)
- $k = 16$ and $\log(n)$ quantifiers suffice for ... (Pikhurko, Verbitsky '10)

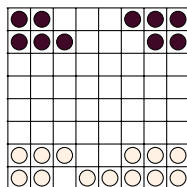
Extensions: TC^m and guarded formulas, greedy shortening, ...

computed formula: $\exists x (\mathbf{W}(x) \wedge \forall y \neg \mathbf{C}(x, y))$

Learning Winning Conditions



Positive Example \mathfrak{A}



Negative Example \mathfrak{B}

Find **minimal** φ such that $\mathfrak{A} \models \varphi$, $\mathfrak{B} \models \neg\varphi$

Which logic and minimality?

- Full FO, minimal quantifier rank: **PSPACE-complete** (Pezzoli '98)
- $\text{FO}^k + \text{C}$, minimal quantifier rank: **PTIME** (Grohe '99)
- $k = 16$ and $\log(n)$ quantifiers suffice for ... (Pikhurko, Verbitsky '10)

Extensions: TC^m and guarded formulas, greedy shortening, ...

computed formula: $\exists x (\mathbf{W}(x) \wedge \forall y \neg \mathbf{C}(x, y))$

Learning game rules from videos (K., AAI-12)

<http://toss.sf.net/learn.html>

Learning Reductions

Formula Outlines

Conjunction outline (conjunction with Boolean guards on all atoms)

$$\begin{aligned} X_1 \mathbf{E}(x_1, x_1) \quad \wedge \quad X_2 \mathbf{E}(x_1, x_2) \quad \wedge \quad X_3 \mathbf{E}(x_2, x_1) \quad \wedge \quad X_4 \mathbf{E}(x_2, x_2) \quad \wedge \\ X_5 \neg \mathbf{E}(x_1, x_1) \quad \wedge \quad X_6 \neg \mathbf{E}(x_1, x_2) \quad \wedge \quad X_7 \neg \mathbf{E}(x_2, x_1) \quad \wedge \quad X_8 \neg \mathbf{E}(x_2, x_2) \end{aligned}$$

Formula Outlines

Conjunction outline (conjunction with Boolean guards on all atoms)

$$\begin{aligned} X_1 \mathbf{E}(x_1, x_1) \quad \wedge \quad X_2 \mathbf{E}(x_1, x_2) \quad \wedge \quad X_3 \mathbf{E}(x_2, x_1) \quad \wedge \quad X_4 \mathbf{E}(x_2, x_2) \quad \wedge \\ X_5 \neg \mathbf{E}(x_1, x_1) \quad \wedge \quad X_6 \neg \mathbf{E}(x_1, x_2) \quad \wedge \quad X_7 \neg \mathbf{E}(x_2, x_1) \quad \wedge \quad X_8 \neg \mathbf{E}(x_2, x_2) \end{aligned}$$

/-DNF outline (all quantifier-free formulas)

$$C_1 \vee C_2 \vee \cdots \vee C_I$$

Formula Outlines

Conjunction outline (conjunction with Boolean guards on all atoms)

$$X_1\mathbf{E}(x_1, x_1) \quad \wedge \quad X_2\mathbf{E}(x_1, x_2) \quad \wedge \quad X_3\mathbf{E}(x_2, x_1) \quad \wedge \quad X_4\mathbf{E}(x_2, x_2) \quad \wedge \\ X_5\neg\mathbf{E}(x_1, x_1) \quad \wedge \quad X_6\neg\mathbf{E}(x_1, x_2) \quad \wedge \quad X_7\neg\mathbf{E}(x_2, x_1) \quad \wedge \quad X_8\neg\mathbf{E}(x_2, x_2)$$

/-DNF outline (all quantifier-free formulas)

$$C_1 \vee C_2 \vee \cdots \vee C_I$$

Extensions

- **k-Variable \exists /-DNF outline**

$$\exists x_1 \dots x_k (C_1 \vee C_2 \vee \cdots \vee C_I)$$

Formula Outlines

Conjunction outline (conjunction with Boolean guards on all atoms)

$$X_1\mathbf{E}(x_1, x_1) \quad \wedge \quad X_2\mathbf{E}(x_1, x_2) \quad \wedge \quad X_3\mathbf{E}(x_2, x_1) \quad \wedge \quad X_4\mathbf{E}(x_2, x_2) \quad \wedge \\ X_5\neg\mathbf{E}(x_1, x_1) \quad \wedge \quad X_6\neg\mathbf{E}(x_1, x_2) \quad \wedge \quad X_7\neg\mathbf{E}(x_2, x_1) \quad \wedge \quad X_8\neg\mathbf{E}(x_2, x_2)$$

/-DNF outline (all quantifier-free formulas)

$$C_1 \vee C_2 \vee \cdots \vee C_l$$

Extensions

- **k -Variable \exists /-DNF outline**

$$\exists x_1 \dots x_k (C_1 \vee C_2 \vee \cdots \vee C_l)$$

- **m -Predicate k -Variable \exists /-DNF outline**

$$P_i(x) = \exists x_1 \dots x_k (C_1 \vee C_2 \vee \cdots \vee C_l), \quad i = 1 \dots m$$

Formula Outlines

Conjunction outline (conjunction with Boolean guards on all atoms)

$$X_1\mathbf{E}(x_1, x_1) \quad \wedge \quad X_2\mathbf{E}(x_1, x_2) \quad \wedge \quad X_3\mathbf{E}(x_2, x_1) \quad \wedge \quad X_4\mathbf{E}(x_2, x_2) \quad \wedge \\ X_5\neg\mathbf{E}(x_1, x_1) \quad \wedge \quad X_6\neg\mathbf{E}(x_1, x_2) \quad \wedge \quad X_7\neg\mathbf{E}(x_2, x_1) \quad \wedge \quad X_8\neg\mathbf{E}(x_2, x_2)$$

/-DNF outline (all quantifier-free formulas)

$$C_1 \vee C_2 \vee \dots \vee C_l$$

Extensions

- **k -Variable \exists /-DNF outline**

$$\exists x_1 \dots x_k (C_1 \vee C_2 \vee \dots \vee C_l)$$

- **m -Predicate k -Variable \exists /-DNF outline**

$$P_i(x) = \exists x_1 \dots x_k (C_1 \vee C_2 \vee \dots \vee C_l), \quad i = 1 \dots m$$

- **n -Layer m -Predicate k -Variable \exists /-DNF outline**

$$P_i^1(x) = \dots (atoms); P_i^2(x) = \dots (P^1s); \dots P_i^n(x) = \dots (P^{n-1}s)$$

Automatic Reduction Finding

Representing reductions by k -dimensional quantifier-free queries

($k = 2$, $\varphi_U = \top$, $\psi_E(x_1, x_2, y_1, y_2) = \mathbf{E}(x_1, y_1) \wedge (x_2 = y_2 \vee y_2 = \mathbf{s})$)



Automatic Reduction Finding

Representing reductions by k -dimensional quantifier-free queries

($k = 2$, $\varphi_U = \top$, $\psi_E(x_1, x_2, y_1, y_2) = \mathbf{E}(x_1, y_1) \wedge (x_2 = y_2 \vee y_2 = \mathbf{s})$)



Finding reductions by CEGAR and SAT-solvers

- Find a I -DNF reduction θ_i good on counter-examples $\mathfrak{E}_0, \dots, \mathfrak{E}_i$
- Find a counter-example \mathfrak{E}_{i+1} to θ_i , iterate

(Jordan, K., SAT '13 improving on Crouch, Immerman, Moss '10)

Automatic Reduction Finding

Representing reductions by k -dimensional quantifier-free queries

($k = 2$, $\varphi_U = \top$, $\psi_E(x_1, x_2, y_1, y_2) = \mathbf{E}(x_1, y_1) \wedge (x_2 = y_2 \vee y_2 = s)$)



Finding reductions by **CEGAR** and **SAT-solvers**

- Find a l -DNF reduction θ_i good on counter-examples $\mathfrak{E}_0, \dots, \mathfrak{E}_i$
- Find a counter-example \mathfrak{E}_{i+1} to θ_i , iterate

(**Jordan, K., SAT '13** improving on Crouch, Immerman, Moss '10)

Easy example: s-t reachability to strongly connected (both NL-complete)

$$\text{Reach} = [\text{tc}_{x,y} \mathbf{E}(x, y)](.s, .t) \quad \text{SC} := \forall x, y (\text{tc}_{x,y} \mathbf{E}(x, y))$$

Automatic Reduction Finding

Representing reductions by k -dimensional quantifier-free queries

($k = 2$, $\varphi_U = \top$, $\psi_E(x_1, x_2, y_1, y_2) = \mathbf{E}(x_1, y_1) \wedge (x_2 = y_2 \vee y_2 = s)$)



Finding reductions by **CEGAR** and **SAT-solvers**

- Find a \perp -DNF reduction θ_i good on counter-examples $\mathfrak{E}_0, \dots, \mathfrak{E}_i$
- Find a counter-example \mathfrak{E}_{i+1} to θ_i , iterate

(**Jordan, K., SAT '13** improving on Crouch, Immerman, Moss '10)

Easy example: s-t reachability to strongly connected (both NL-complete)

$\text{Reach} = [\text{tc}_{x,y} \mathbf{E}(x, y)](s, t)$ $\text{SC} := \forall x, y (\text{tc}_{x,y} \mathbf{E}(x, y))$

<http://toss.sf.net/reduct.html>

Automatic Reduction Finding

Representing reductions by k -dimensional quantifier-free queries

($k = 2$, $\varphi_U = \top$, $\psi_E(x_1, x_2, y_1, y_2) = \mathbf{E}(x_1, y_1) \wedge (x_2 = y_2 \vee y_2 = \mathbf{s})$)



Finding reductions by **CEGAR** and **SAT-solvers**

- Find a I -DNF reduction θ_i good on counter-examples $\mathfrak{E}_0, \dots, \mathfrak{E}_i$
- Find a counter-example \mathfrak{E}_{i+1} to θ_i , iterate

(**Jordan, K., SAT '13** improving on Crouch, Immerman, Moss '10)

Easy example: s-t reachability to strongly connected (both NL-complete)

$\text{Reach} = [\text{tc}_{x,y} \mathbf{E}(x, y)](.s, .t)$ $\text{SC} := \forall x, y (\text{tc}_{x,y} \mathbf{E}(x, y))$

($k = 1$, $\varphi_U = \top$, $\psi_E = x_1 = \mathbf{s} \vee x_2 = \mathbf{t} \vee \mathbf{E}(x_2, x_1)$)

Automatic Reduction Finding

Representing reductions by k -dimensional quantifier-free queries

($k = 2$, $\varphi_U = \top$, $\psi_E(x_1, x_2, y_1, y_2) = \mathbf{E}(x_1, y_1) \wedge (x_2 = y_2 \vee y_2 = \mathbf{s})$)



Finding reductions by **CEGAR** and **SAT-solvers**

- Find a I -DNF reduction θ_i good on counter-examples $\mathfrak{E}_0, \dots, \mathfrak{E}_i$
- Find a counter-example \mathfrak{E}_{i+1} to θ_i , iterate

(**Jordan, K., SAT '13** improving on Crouch, Immerman, Moss '10)

Easy example: s-t reachability to strongly connected (both NL-complete)

$\text{Reach} = [\text{tc}_{x,y} \mathbf{E}(x, y)](.s, .t)$ $\text{SC} := \forall x, y (\text{tc}_{x,y} \mathbf{E}(x, y))$

($k = 1$, $\varphi_U = \top$, $\psi_E = x_1 = \mathbf{s} \vee x_2 = \mathbf{t} \vee \mathbf{E}(x_2, x_1)$)

Other applications: game rule learning, program synthesis, ...

Looking Forward

Machine Learning Motivation



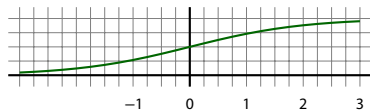
- Handwriting recognition MNIST (many), Arabic HWX (IDSIA)
- OCR in the Wild [2011]: StreetView House Numbers (NYU and others)
- Traffic sign recognition [2011] GTSRB competition (IDSIA, NYU)
- Pedestrian Detection [2013]: INRIA datasets and others (NYU)
- Volumetric brain image segmentation [2009] connectomics (IDSIA, MIT)
- Human Action Recognition [2011] Hollywood II dataset (Stanford)
- Object Recognition [2012] ImageNet competition
- Scene Parsing [2012] Stanford bgd, SiftFlow, Barcelona (NYU)
- Scene parsing from depth images [2013] NYU RGB-D dataset (NYU)
- Speech Recognition [2012] Acoustic modeling (IBM and Google)
- Breast cancer cell mitosis detection [2011] MITOS (IDSIA)
- The list of perceptual tasks for which ConvNets hold the record is growing.
- Most of these tasks (but not all) use purely supervised convnets.

It's hard to prove anything about deep learning systems

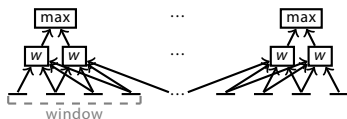
Y. LeCun, COLT '13

What are Convolutional Networks?

Neuron: $\sigma(\text{weighted sum})$

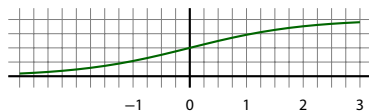


Network with **shared weights**



What are Convolutional Networks?

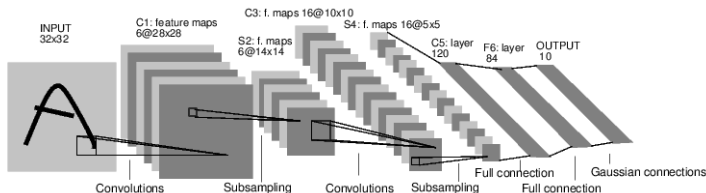
Neuron: $\sigma(\text{weighted sum})$



Network with **shared weights**

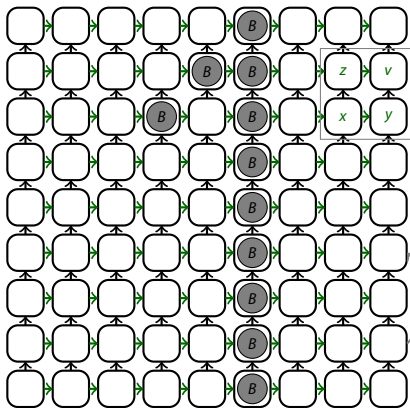


Example (LeNet-5, 0.95% MNIST error rate)



(credit: EBLearn (eblearn.cs.nyu.edu))

Threshold Convolutional Formulas



Convolution

$$P^1(x) = \exists y, z, v. (R(x, y) \wedge C(x, z) \wedge R(z, v)) \varphi$$

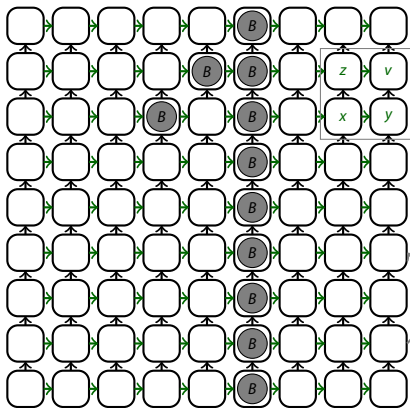
$$\text{with } \varphi = w_1 \chi[B(x)] + \dots + w_4 \chi[B(v)] \geq t$$

Subsampling (max-pooling)

$$P^2(x) = \exists y, z, v. (R_2(x, y) \wedge C_2(x, z) \wedge \dots) \psi$$

$$\text{where } C_2(x, y) = \exists z (C(x, z) \wedge C(z, y))$$
$$\text{and } \psi = w_1 \chi[P^1_1(x)] + \dots + w_4 \chi[P^1_m(v)] \geq t'$$

Threshold Convolutional Formulas



Convolution

$$P^1(x) = \exists y, z, v. (R(x, y) \wedge C(x, z) \wedge R(z, v)) \varphi$$

$$\text{with } \varphi = w_1 \chi[B(x)] + \dots + w_4 \chi[B(v)] \geq t$$

Subsampling (max-pooling)

$$P^2(x) = \exists y, z, v. (R_2(x, y) \wedge C_2(x, z) \wedge \dots) \psi$$

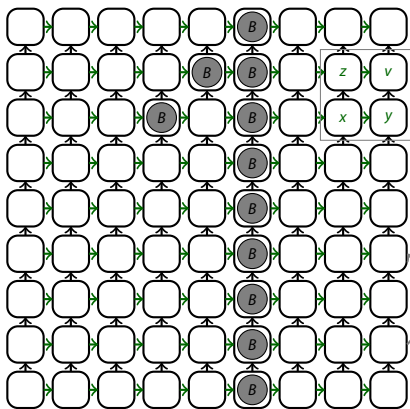
$$\text{where } C_2(x, y) = \exists z (C(x, z) \wedge C(z, y))$$

$$\text{and } \psi = w_1 \chi[P^1_1(x)] + \dots + w_4 \chi[P^1_m(v)] \geq t'$$

= n -Layer m -Predicate $(k \times k)$ -Variable \exists guarded threshold outline

$$P^1_i(x) = \dots (\text{atoms}); P^2_i(x) = \dots (P^1_s); \dots P^n_i(x) = \dots (P^{n-1}_s)$$

Threshold Convolutional Formulas



Convolution

$$P^1(x) = \exists y, z, v. (R(x, y) \wedge C(x, z) \wedge R(z, v)) \varphi$$

$$\text{with } \varphi = w_1 \chi[B(x)] + \dots + w_4 \chi[B(v)] \geq t$$

Subsampling (max-pooling)

$$P^2(x) = \exists y, z, v. (R_2(x, y) \wedge C_2(x, z) \wedge \dots) \psi$$

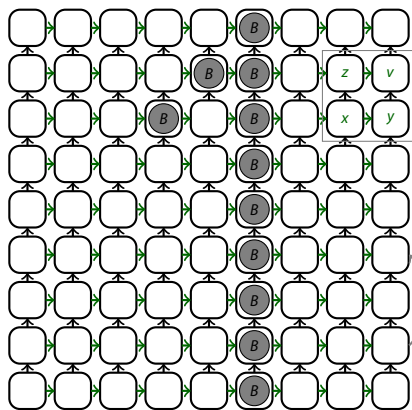
$$\text{where } C_2(x, y) = \exists z (C(x, z) \wedge C(z, y))$$
$$\text{and } \psi = w_1 \chi[P^1_1(x)] + \dots + w_4 \chi[P^1_m(v)] \geq t'$$

= n -Layer m -Predicate ($k \times k$)-Variable \exists guarded threshold outline

$$P^1_i(x) = \dots (atoms); P^2_i(x) = \dots (P^1_s); \dots P^n_i(x) = \dots (P^{n-1}_s)$$

Inspiring goal: uniform learning platform and theory

Threshold Convolutional Formulas



Convolution

$$P^1(x) = \exists y, z, v. (R(x, y) \wedge C(x, z) \wedge R(z, v)) \varphi$$

$$\text{with } \varphi = w_1 \chi[B(x)] + \dots + w_4 \chi[B(v)] \geq t$$

Subsampling (max-pooling)

$$P^2(x) = \exists y, z, v. (R_2(x, y) \wedge C_2(x, z) \wedge \dots) \psi$$

$$\text{where } C_2(x, y) = \exists z (C(x, z) \wedge C(z, y))$$

$$\text{and } \psi = w_1 \chi[P^1_1(x)] + \dots + w_4 \chi[P^1_m(v)] \geq t'$$

= n -Layer m -Predicate $(k \times k)$ -Variable \exists guarded threshold outline

$$P^1_i(x) = \dots (atoms); P^2_i(x) = \dots (P^1_s); \dots P^n_i(x) = \dots (P^{n-1}_s)$$

Inspiring goal: uniform learning platform and theory Thank You!