Learning equivalence relations

joint work with Ted Slaman and Tomasz Steifer

Dino Rossegger
Technische Universität Wien

Online Logic Seminar

This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 101026834.

Learning from Informant

Gold (1961) considered the following application of inductive inference:

Given a fixed sequence of languages $l_1, l_2, \ldots$ an ***informant*** secretly fixes a language l_i and at every stage s, presents the ***learner*** with a new word $w_s \in l_i$. The learner makes a guess $l(w_0, \ldots, w_s)$ to identify the presented language. They ***learn*** the language if they correctly guess in the limit, i.e., $\lim_s l(w_0, \ldots, w_s) = i$.

Gold (1961) considered the following application of inductive inference:

Given a fixed sequence of languages $l_1, l_2, \dots$ an ***informant*** secretly fixes a language l_i and at every stage s, presents the ***learner*** with a new word $w_s \in l_i$. The learner makes a guess $l(w_0, \dots, w_s)$ to identify the presented language. They ***learn*** the language if they correctly guess in the limit, i.e., $\lim_s l(w_0, \dots, w_s) = i$.

Fokina-Kötzing-San Mauro (2019) looked at this in a computable structure theory setting:

InfEx learning: Given a fixed sequence of countable pairwise non-isomorphic structures $\mathcal{A}_0, \mathcal{A}_1 \dots$, the informant fixes an ***"example"*** $\mathcal{B} \cong \mathcal{A}_i$ and at stage s plays the substructure on the first s elements of $\mathcal{B}$. Again the learner makes a guess $l(\mathcal{B} \upharpoonright s)$ and learns the family $\mathcal{A}_0, \mathcal{A}_1, \dots$ if for any $\mathcal{A}_i$ and $\mathcal{B} \cong \mathcal{A}_i$, $\lim_s(\mathcal{B} \upharpoonright s) = i$.

For given vocabulary τ fix an enumeration $\varphi_i(x_0, \dots, x_i)$ of the atomic τ-formulas and let the atomic diagram of a τ-structure $\mathcal{A}$ with universe ω be

$$D(\mathcal{A})(i) = \begin{cases} 1 & \varphi_i[x_0 \dots x_i \mapsto 0 \dots i]^{\mathcal{A}} \\ 0 & \text{otherwise} \end{cases}$$

Allows us to identify structures with elements of 2^ω.

Definition
A pairwise non-isomorphic family of countable structures $\mathcal{A}_0, \dots$ is ***InfEx learnable*** if there is a function $l : 2^{<\omega} \to \omega$ such that for any i and $\mathcal{B} \cong \mathcal{A}_i$, $L(\mathcal{B}) = \lim_s l(\mathcal{B} \upharpoonright s) = i$.

An infinitary formula φ is Σ_2^{in} if it is of the form $\bigvee \exists \bar{x}_i \bigwedge \forall \bar{y}_{ij} \varphi_{ij}$ where φ_{ij} is quantifier-free.

Theorem (Bazhenov, Fokina, San Mauro 2020)
A countable sequence $\mathcal{A}_0, \dots$ is ***InfEx*** *learnable if and only if there are Σ_2^{in} formulas φ_i such that $\mathcal{A}_i \models \varphi_i$ and $\mathcal{A}_j \not\models \varphi_i$ for $i \neq j$.*

A binary relation E on a Polish space X is ***continuously (Borel) reducible*** to F on Y, $E \leq_{c(B)} F$ if there is a continuous (Borel) function $f : X \to Y$ s.t. for all $x, y \in X$, xEy iff $f(x) F f(y)$.

Theorem (Bazhenov, Cipriani, San Mauro 2023)
A countable sequence $\mathcal{A}_0, \dots$ is ***InfEx*** *learnable if and only if $\cong_{(\mathcal{A}_i)} \leq_c E_0$.*

1. The space of countable τ-structures $Mod(\tau)$ is a Polish space.
2. $Mod(\tau)/\cong$ is not countable in non-trivial cases
3. **InfEx** learnability is a property of $\cong$ on $\mathcal{A}_0, \ldots$

What about $\cong$ on the whole space or uncountable invariant subsets of $Mod(\tau)$?

Why restrict to $\cong$ and not look at other equivalence relations on Polish spaces?

The investigation of these two questions is the goal of this project.

- Consider ω with the discrete topology and 2^ω with the product topology.
- The function $L : 2^\omega \to \omega$, $L(\mathcal{A}) = \lim_s l(\mathcal{A} \upharpoonright s)$ is not continuous.
- The function $l_s : 2^\omega \to \omega$ $l_s(\mathcal{A}) = l(\mathcal{A} \upharpoonright s)$ is continuous. So, $L(\mathcal{A}) = \lim_s l_s(\mathcal{A})$.

Definition
Let E be an equivalence relation on a Polish space X and assume ω is equipped with the discrete topology. E is ***uniformly learnable***, or just ***learnable***, if there are continuous functions $l_n : X^\omega \times X \to \omega$ such that for $x \in X$ and $\vec{x} = (x_i)_{i\in\omega} \in X^\omega$, if $x \mathrel{E} x_i$ for some $i \in \omega$, then $\lim l_n(\vec{x}, x)$ exists and $x \mathrel{E} x_{L(\vec{x},x)}$ where $L(\vec{x}, x) = \lim l_n(\vec{x}, x)$.

If $x \not\mathrel{E} \vec{x}$, then L might either diverge or converge to some i. In the latter case we say that L produces a ***false positive***.

Disclaimer: To avoid dealing with effective Polish spaces and make proofs easier we will assume that we are working on 2^ω. Unless stated otherwise, all proofs work for arbitrary Polish spaces, mutatis mutandis.

For $a \in 2^\omega$ we say that a learner L is a-computable if there is an a-recursive function $f : \omega \to \omega$ such that $l_s = \Phi^a_{f(s)}$ where $(\Phi_i)_{i\in\omega}$ is a canonical enumeration of Turing operators.

Theorem (RSS)
Fix $a \in 2^\omega$. An equivalence relation E on a Polish space X is learnable by an a-computable learner if and only if it is $\Sigma^0_2(a)$.

Proof. $(\Leftarrow)$. Say E is $\Sigma^0_2(a)$, then there exists an a-recursive predicate R such that

$$x\,E\,y \iff \exists n \forall m R(x, y, n, m).$$

Define l_s by

$$l_s(\vec{x}, x) = \begin{cases} \mu i < s[(\exists n < s)(\forall m < s)R(x, x_i, m)] & \text{if such } i < s \text{ exists} \\ s & \text{otherwise} \end{cases}$$

Note that l_s is recursive in a and that $\lim_s l_s(\vec{x}, x) = \mu j[x_j\ E\ x]$, if such j exists. Otherwise $\lim_s l_s(\vec{x}, x) \uparrow$. Hence, $L = \lim_s l_s$ does not produce false positives!

$(\Rightarrow)$. Say E is learnable by an a-computable learner L and consider arbitrary $x, y \in 2^\omega$. We will extract a $\Sigma^0_2(a)$ definition using forcing. The main idea is that if we take a sufficiently mutually (a, x, y)-generic sequence $\vec{g}$, then the behaviour of the learner on $L(x, y^\frown\vec{g})$ is forced by some condition $\vec{p}$, and similarly for $L(y, x^\frown\vec{g})$. Using this $\vec{p}$ we can extract a $\Sigma^0_2(a)$ formula that is independent of x, y and defines E.

Lemma

If $g, \vec{g}$ is a sequence of sufficiently mutually generics relative to L and $L(g, \vec{g}) = k$, then there are $p, \vec{p} \prec g, \vec{g}$, such that L does not produce false positives for any $h, \vec{h} \succ p, \vec{p}$.

Proof.

Take $g, \vec{g}$ sufficiently L-generic. We have that $L(\vec{g}, g) \downarrow = k$ iff $\exists n (\forall m > n) l_m(g, \vec{g}) = k$. Let n_0 be the least such n. Then the above statement must be forced by some $p, \vec{p}$, i.e.,

$$\vec{p}, p \Vdash (\forall m > n_0) l_m(\dot{\vec{g}}, \dot{g}) \downarrow = k$$
$$\iff (\forall m > n_0)(\forall \vec{q}, q \leq \vec{p}, p)(l_m(\vec{q}, q) \downarrow \implies l_m(\vec{q}, q) = k)$$

But then in particular $L(h, h_0 \ldots h_i h h_{i+2} \ldots) = k$ where $i \neq k - 1$ and $i > |\vec{p}|$ and $h \mathrel{E} h_k$ as L cannot give false positives since $h \mathrel{E} h$. □

($\Leftarrow$). Say E is learnable by an a-computable learner L and $x \mathrel{E} y$. Take $\vec{g}$ sufficiently mutually (x, y, a)-generic and look at $L(x, y^\frown\vec{g})$, $L(y, x^\frown\vec{g})$. Say $L(x, y^\frown\vec{g}) = 0$, then by genericity x, y satisfy the following formula:

$$\exists n_0 \exists \vec{p} (\forall \vec{q} \leq \vec{p})(\forall n > n_0) \left(l_n(x, y^\frown\vec{q}) \downarrow \implies l_n(x, y^\frown\vec{q}) = 0 \right) \tag{*}$$

$(\Leftarrow)$. Say E is learnable by an a-computable learner L and $x \mathrel{E} y$. Take $\vec{g}$ sufficiently mutually (x, y, a)-generic and look at $L(x, y^\frown \vec{g})$, $L(y, x^\frown \vec{g})$. Say $L(x, y^\frown \vec{g}) = 0$, then by genericity x, y satisfy the following formula:

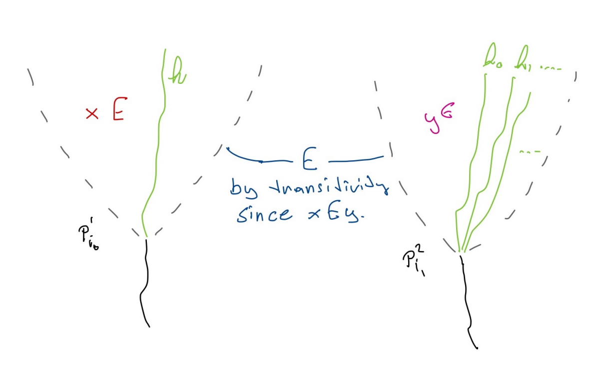
$$\exists n_0 \exists \vec{p} (\forall \vec{q} \leq \vec{p})(\forall n > n_0) \left(l_n(x, y^\frown \vec{q}) \downarrow \implies l_n(x, y^\frown \vec{q}) = 0 \right) \qquad (*)$$

Likewise if $L(y, x^\frown \vec{g}) = 0$ then

$$\exists n_0 \exists \vec{p} (\forall \vec{q} \leq \vec{p})(\forall n > n_0) \left(l_n(y, x^\frown \vec{q}) \downarrow \implies l_n(y, x^\frown \vec{q}) = 0 \right) \qquad (**)$$

If $L(x, y^\frown \vec{g}) = i_0 \neq 0$ and $L(y, x^\frown \vec{g}) = i_1 \neq 0$, then $x \mathrel{E} g_{i_0}$, $y \mathrel{E} g_{i_1}$, and this is forced by some $\vec{p}^1$ and $\vec{p}^2$ and for any $\vec{h} \succ \vec{p}^1$, $L(x, y^\frown \vec{h}) = i_0$ (same for $\vec{h} \succ \vec{p}^2$) and $x \mathrel{E} h_{i_0}$, $y \mathrel{E} h_{i_1}$.

Look at $h \succ \vec{p}^{1}_{i_0}, \vec{h} \succ \vec{(}p^{2}_{i_1})^{\infty}$. By transitivity of E, $L(h, \vec{h}) = k$ for some k and this is again forced.



$$\begin{aligned}\exists n_0 \exists \vec{p}_0, \vec{p}_1 \exists i_0, i_1 (\forall n > n_0)(\forall \vec{q} \le \vec{p}_0)\,(L(x, y^\frown \vec{q}, n)\downarrow \implies L(x, y^\frown \vec{q}, n) = i_0) \\ \wedge(\forall \vec{q} \le \vec{p}_1)\,(L(y, x^\frown \vec{q}, n)\downarrow \implies L(y, x^\frown \vec{q}, n) = i_1) \\ \wedge \exists k (\exists r \le \vec{p}^{0}_{i_0})(\exists \vec{r} \le p^{\vec{1}}_{i_1}{}^{\infty})(\forall n > n_0)(\forall q \le r)(\forall \vec{q} \le \vec{r}) \\ (L(q, \vec{q}, n)\downarrow \implies L(q, \vec{q}, n) = k)\end{aligned} \tag{***}$$

If $x \mathrel{E} y$ then they satisfy $(*)$, $(**)$ or $(***)$. If $x \not\mathrel{E} y$, they might still satisfy $(***)$ if $L(h, \vec{h})$ gives false positives. But that cannot happen by our lemma. $(*) \vee (**) \vee (***)$ define E.

- Eventual equality on 2^ω: $x \; E_0 \; y \iff \exists m (\forall n > m) x(n) = y(n)$
- Eventual equality on $2^{\omega\omega}$: $(x_i) \; E_1 \; (y_i) \iff \exists m (\forall n > m) x_n = y_n$
- The shift action of F_2 on 2^{F_2}, $E(F_2, 2)$.

Theorem (Bazhenov, Cipriani, San Mauro)
*A countable sequence $\mathcal{A}_0, \ldots$ is **InfEx** learnable if and only if $\cong_{(\mathcal{A}_i)} \leq_c E_0$.*

- This characterization fails in our case, even for Borel reducibility, neither E_1, nor $E(F_2, 2)$ are Borel reducible to E_0.
- It follows from the Feldman-Moore theorem that for any countable Borel equivalence relation E, there is a topology such that E is $\mathbf{\Sigma^0_2}$.

- Eventual equality on 2^ω: $x \; E_0 \; y \iff \exists m (\forall n > m) x(n) = y(n)$
- Eventual equality on $2^{\omega\omega}$: $(x_i) \; E_1 \; (y_i) \iff \exists m (\forall n > m) x_n = y_n$
- The shift action of F_2 on 2^{F_2}, $E(F_2, 2)$.

Theorem (Bazhenov, Cipriani, San Mauro)
A countable sequence $\mathcal{A}_0, \ldots$ is ***InfEx*** *learnable if and only if $\cong_{(\mathcal{A}_i)} \leq_c E_0$.*

- This characterization fails in our case, even for Borel reducibility, neither E_1, nor $E(F_2, 2)$ are Borel reducible to E_0.
- It follows from the Feldman-Moore theorem that for any countable Borel equivalence relation E, there is a topology such that E is $\mathbf{\Sigma^0_2}$.

Question: Is there a universal learnable equivalence relation for continuous or Borel reducibility?

This question (with $\mathbf{\Sigma^0_2}$) seems to be open for many years.

By a result of Arnie Miller ('83), no structure can have a Σ_2^{in} Scott sentence, i.e., for no $\mathcal{A}$, $[\mathcal{A}]_{\cong}$ is $\mathbf{\Sigma}_2^0$. This implies that $\cong$ in a vocabulary τ cannot be $\mathbf{\Sigma}_2^0$.

Proposition

Let τ be a countable vocabulary. Then $Mod(\tau)$ is not learnable.

If we restrict to structures satisfying a fixed $L_{\omega_1\omega}$ sentence φ, we can find examples of learnable structures.

Example: Let φ axiomatize torsion free Abelian groups of rank 1, then $\cong_\varphi$ is learnable.

- Represent $\mathbf{\Sigma^0_2}$ relations using Borel codes. (Well-founded infinitely branching trees labeled with $\cup$, $\cap$, and codes for finite intersections of basic open sets)
- These codes can be coded by elements of 2^ω.
- How complicated is the set of codes of learnable Borel equivalence relations?

- Represent $\boldsymbol{\Sigma}_2^0$ relations using Borel codes. (Well-founded infinitely branching trees labeled with $\cup$, $\cap$, and codes for finite intersections of basic open sets)
- These codes can be coded by elements of 2^ω.
- How complicated is the set of codes of learnable Borel equivalence relations?

Theorem (Louveau 1980)
If X is a recursive Polish space, $A_0, A_1 \in \Sigma_1^1$, $A_0 \cap A_1 = \emptyset$ s.t. there is $B \in \boldsymbol{\Sigma}_\alpha^0$ with $A_0 \subseteq B$ and $A_1 \cap B = \emptyset$, then B can be taken in $\Sigma_\alpha^0(HYP)$.

Lemma
If E is Δ_1^1 and learnable, then it is learnable by a hyperarithmetical learner.

Proof sketch.
By Louveau's separation theorem, if E is $\boldsymbol{\Sigma}_2^0$ and Δ_1^1, then it is $\Sigma_2^0(HYP)$. Thus by our theorem, it is learnable by a hyperarithmetical learner. □

Lemma
The set of codes of learnable Borel equivalence is Π^1_1.

Proof sketch.
T codes a learnable equivalence relation iff (1) T is well-founded, (2) its labeling is correct, and (3) there is a learner learning E_T. The first statement is Π^1_1, the second arithmetical and the third can be replaced by (3') there is a learner hyperarithmetical in T by the above Lemma. By the Spector-Gandy theorem, (3') is Π^1_1. □

Lemma
The set of codes of learnable Borel equivalence is Π^1_1.

Proof sketch.
T codes a learnable equivalence relation iff (1) T is well-founded, (2) its labeling is correct, and (3) there is a learner learning E_T. The first statement is Π^1_1, the second arithmetical and the third can be replaced by (3') there is a learner hyperarithmetical in T by the above Lemma. By the Spector-Gandy theorem, (3') is Π^1_1. □

Theorem (RSS)
The set of learnable $\mathbf{\Pi^0_2}$ *equivalence relations on* 2^ω *is* $\mathbf{\Pi^1_1}$ *complete in the codes.*

Proof sketch.
Use the fact that the set of well-founded trees in $\omega^{<\omega}$ is $\mathbf{\Pi}^1_1$ complete and that $Inf = \{x \in 2^\omega : |dom(x)| = \infty\}$ is $\mathbf{\Pi}^0_2$ complete. For $x \in Inf$ let p_x be the principal function of x and define E_T as

$$xE_Ty \iff x = y \vee (p_{x_1}, p_{y_1} \in [T] \wedge x_2, y_2 \in Inf)$$

If T is well-founded, $E_T = id$. Otherwise fix $x \in [T]$, E_T can't be learnable since then

Definition

Let E be an equivalence relation on a Polish space X and assume ω is equipped with the discrete topology. E is ***uniformly Borel learnable***, or just ***Borel learnable***, if there are Borel functions $l_n : X^\omega \times X \to \omega$ such that for $x \in X$ and $\vec{x} = (x_i)_{i \in \omega} \in X^\omega$, if $x \mathrel{E} x_i$ for some $i \in \omega$, then $\lim l_n(\vec{x}, x)$ exists and $x \mathrel{E} x_{L(\vec{x},x)}$ where $L(\vec{x}, x) = \lim l_n(\vec{x}, x)$.

Borel learnability is connected to uniform learnability via the following classic fact:

Theorem

For Polish spaces $(X, \sigma), Y$ and a Borel function $f : X \to Y$, there exists a topology $\tau \supseteq \sigma$ of X such that $B((X, \tau)) = B((X, \sigma))$ and $f : X \to Y$ is τ-continuous.

Proposition

An equivalence relation E is on X is Borel learnable if and only if there exists a refinement of the topology on X such that E is uniformly learnable.

Definition
Let E be an equivalence relation on a Polish space X and assume ω is equipped with the discrete topology. We say that E is ***non-uniformly learnable***, if for every $\vec{x} = (x_i)_{i\in\omega} \in X^\omega$ there are continuous functions $l_n : X^\omega \times X \to \omega$ such that for $x \in X$, if $x \mathrel{E} x_i$ for some $i \in \omega$, then $\lim l_n(\vec{x}, x)$ exists and $x \mathrel{E} x_{L(\vec{x},x)}$ where $L(\vec{x}, x) = \lim l_n(\vec{x}, x)$.

Theorem
An equivalence relation E is non-uniformly learnable without false positives if and only if every equivalence class is $\mathbf{\Sigma^0_2}$.

Theorem
An equivalence relation E is non-uniformly learnable with false positives if and only if for every countable sequence $\vec{x}$ there is a sequence of $\mathbf{\Sigma^0_2}$ sets $(S_i)_{i\in\omega}$ such that $[x_i] \subseteq S_i$ and $[x_j] \cap S_i = \emptyset$ for $i \neq j$.