Approximate Vanishing Ideal Computations at Scale

Elias Wirth[1,2] Hiroshi Kera[3] Sebastian Pokutta[1,2]

[1]Technische Universität Berlin

[2]Zuse Institute Berlin

[3]Chiba University

September 19, 2022

Binary Classification

- Input space $\mathcal{X} \subseteq \mathbb{R}^n$ and output space $\mathcal{Y} = \{-1, +1\}$
- Training sample

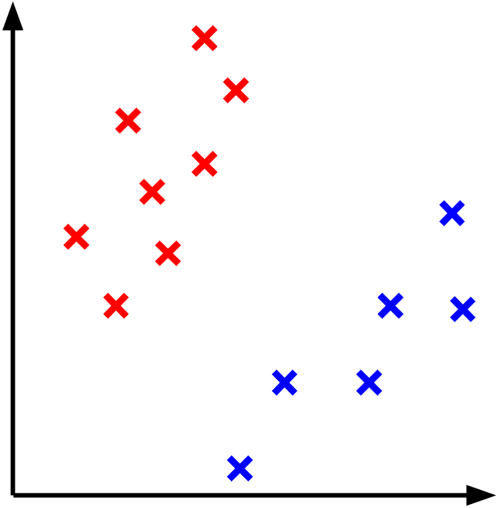
$$S = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_m, y_m)\} \in (\mathcal{X} \times \mathcal{Y})^m$$

  drawn *i.i.d.* from some unknown distribution $\mathcal{D}$

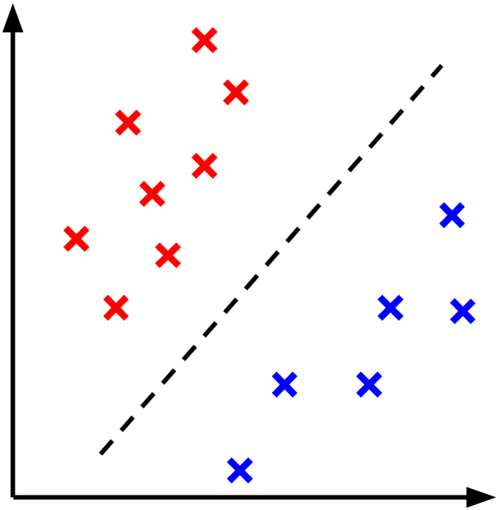
- Determine a *hypothesis* $h \colon \mathcal{X} \to \mathcal{Y}$ with small *generalization error*

$$\mathbb{P}_{(\mathbf{x},y)\backsim\mathcal{D}}[h(\mathbf{x}) \neq y]$$

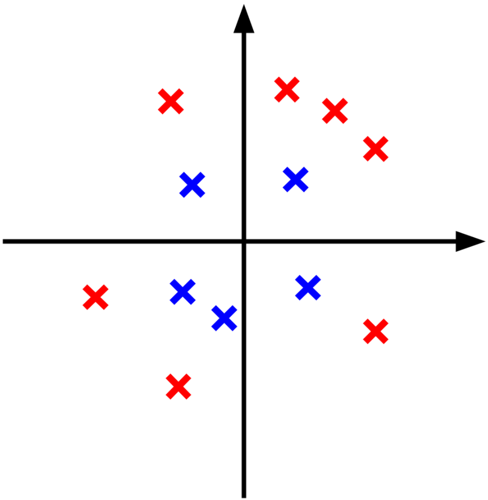
Linearly Separable Case



Linearly Separable Case



Non-Separable Case



Algebraic Set

- Let $\mathcal{P}$ denote the polynomial ring in n variables

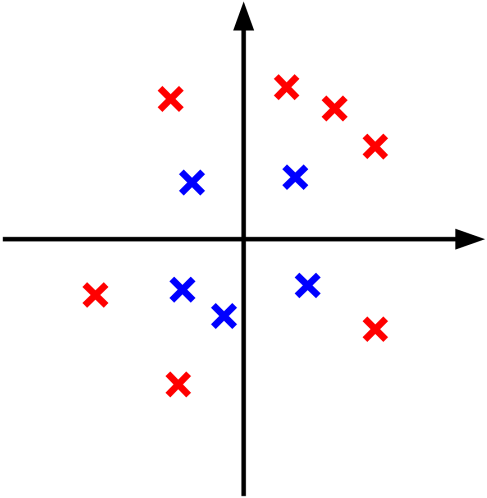
Definition 1 (Algebraic Set).

A set $X \subseteq \mathbb{R}^n$ is *algebraic* if there exists a finite set of polynomials $\mathcal{G} \subseteq \mathcal{P}$, such that X is the set of common roots of $\mathcal{G}$.

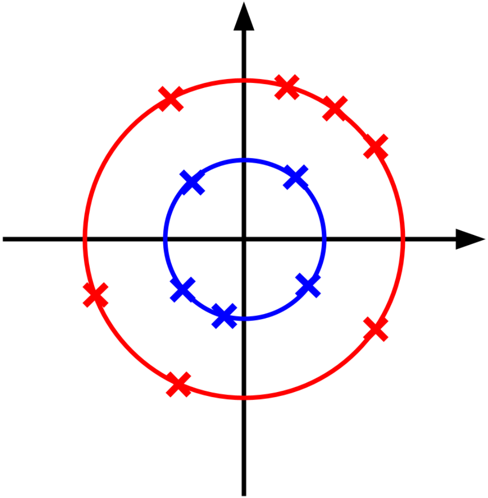
Example 2 (Ball of Radius 1).

$$X = \{\mathbf{x} \in \mathbb{R}^2 \mid x_1^2 + x_2^2 = 1\} \Rightarrow \mathcal{G} = \{x_1^2 + x_2^2 - 1\}$$

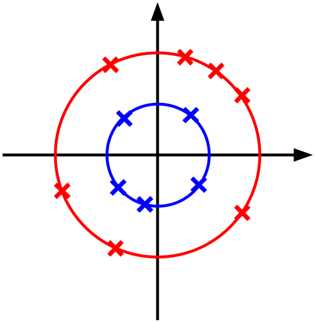
Non-Separable Case



Non-Separable Case

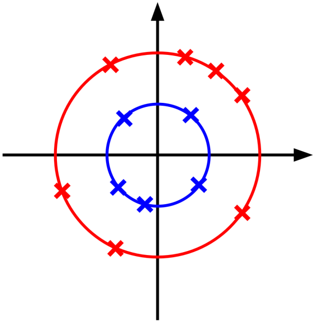


Non-Separable Case



$$X^{+1} \subseteq \{\mathbf{x} \in \mathbb{R}^2 \mid x_1^2 + x_2^2 = 1\}$$
$$X^{-1} \subseteq \{\mathbf{x} \in \mathbb{R}^2 \mid x_1^2 + x_2^2 = 2\}$$

Non-Separable Case

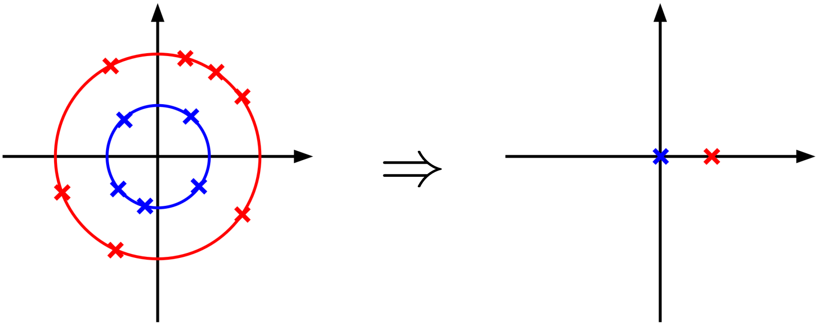


$$X^{+1} \subseteq \{\mathbf{x} \in \mathbb{R}^2 \mid x_1^2 + x_2^2 = 1\}$$
$$X^{-1} \subseteq \{\mathbf{x} \in \mathbb{R}^2 \mid x_1^2 + x_2^2 = 2\}$$

$$g(\mathbf{x}) = x_1^2 + x_2^2 - 1$$
$$|g(\mathbf{x}^{+1})| = 0$$
$$|g(\mathbf{x}^{-1})| = 1$$

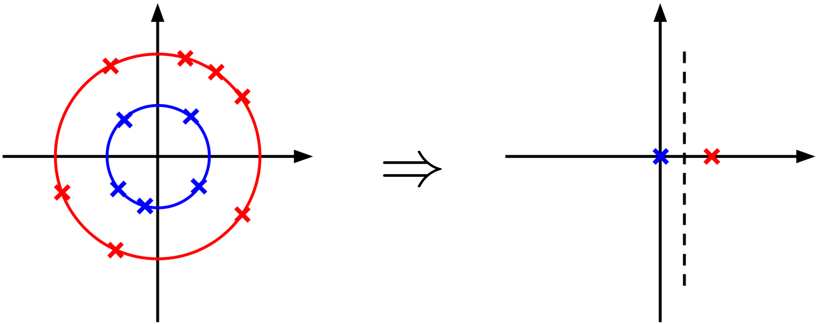
Non-Separable Case

$$g(\mathbf{x}) = x_1^2 + x_2^2 - 1$$



Non-Separable Case

$$g(\mathbf{x}) = x_1^2 + x_2^2 - 1$$



Vanishing Ideal

- Data set $X = \{\mathbf{x}_1, \ldots, \mathbf{x}_m\} \subseteq \mathbb{R}^n$
- Vanishing Ideal

$$\mathcal{I}_X = \{f \in \mathcal{P} \mid f(\mathbf{x}) = 0 \text{ for all } \mathbf{x} \in X\}$$

- Finite set of generators $\mathcal{G} = \{g_1, \ldots, g_k\} \subseteq \mathcal{I}_X$ such that for all $f \in \mathcal{I}_X$, there exist $h_1, \ldots, h_k \in \mathcal{P}$ with

$$f = \sum_{i=1}^{k} g_i h_i.$$

[Cox et al., 2013]

Approximately Vanishing Polynomial

Definition 3 (Approximately Vanishing Polynomial).

Let $X = \{\mathbf{x}_1, \ldots, \mathbf{x}_m\} \subseteq \mathbb{R}^n$. A polynomial $g = \sum_{i=1}^{k} c_i t_i \in \mathcal{P}$ is called $(\psi, 1, \tau)$-*approximately vanishing* (over X) if

1. $\mathrm{MSE}(g, X) := \frac{1}{m} \sum_{i=1}^{m} g(\mathbf{x}_i)^2 \leq \psi$,
2. $\mathrm{LTC}(g) = c_k = 1$,
3. $\|g\|_1 := \|\mathbf{c}\|_1 \leq \tau$.

Definition 4 (Approximate Vanishing Ideal).

Let $X = \{\mathbf{x}_1, \ldots, \mathbf{x}_m\} \subseteq \mathbb{R}^n$. The (ψ, τ)-approximate vanishing ideal is the ideal generated by all $(\psi, 1, \tau)$-approximately vanishing polynomials.

Classification Pipeline

Algorithm 1: Pipeline

Input : Training sample $S = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_m, y_m)\}$ with $X = \{\mathbf{x}_1, \ldots, \mathbf{x}_m\} \subseteq \mathbb{R}^n$ and $y_1, \ldots, y_m \in \{-1, +1\}$.

$X^{\pm 1} \leftarrow \{\mathbf{x}_j \in X \mid y_j = \pm 1\} \subseteq X$

$\mathcal{G}^{\pm 1} \leftarrow$ generating set of the approximate vanishing ideal $\mathcal{I}^{\psi}_{X^{\pm 1}}$

$\mathcal{G} = \{g_1, \ldots, g_{|\mathcal{G}|}\} \leftarrow \mathcal{G}^{+1} \cup \mathcal{G}^{-1}$

for $j = 1, \ldots, m$ **do**

  $\tilde{\mathbf{x}}_j \leftarrow \left(|g_1(\mathbf{x}_j)|, \ldots, |g_{|\mathcal{G}|}(\mathbf{x}_j)|\right)^\intercal \in \mathbb{R}^{|\mathcal{G}|}$

end

train linear classifier on $\tilde{X} = \{\tilde{\mathbf{x}} \mid \mathbf{x} \in X\}$

Algorithm

Algorithm 2: Oracle Approximate Vanishing Ideal Algorithm (`OAVI`)

Input : $X = \{\mathbf{x}_1, \ldots, \mathbf{x}_m\} \subseteq \mathbb{R}^n$, $\psi \geq 0$, and $\tau \geq 2$.
Output: $\mathcal{G} \subseteq \mathcal{P}$ and $\mathcal{O} \subseteq \mathcal{T}$.

$d \leftarrow 1$, $\mathcal{O} \leftarrow \{\mathbb{1}\}$, $\mathcal{G} \leftarrow \emptyset$
while $\partial_d \mathcal{O} = \{u_1, \ldots, u_k\} \neq \emptyset$ **do**
    for $i = 1, \ldots, k$ **do**
        $g \leftarrow$ construct candidate polynomial
        if g *vanishes approximately* **then**
            $\mathcal{G} \leftarrow \mathcal{G} \cup \{g\}$
        else
            $\mathcal{O} = \{t_1, \ldots, t_\ell\} \leftarrow \mathcal{O} \cup \{u_i\}$
        end
    end
    $d \leftarrow d + 1$
end

Algorithm

Algorithm 2: Oracle Approximate Vanishing Ideal Algorithm (`OAVI`)

Input : $X = \{\mathbf{x}_1, \ldots, \mathbf{x}_m\} \subseteq \mathbb{R}^n$, $\psi \geq 0$, and $\tau \geq 2$.
Output: $\mathcal{G} \subseteq \mathcal{P}$ and $\mathcal{O} \subseteq \mathcal{T}$.

$d \leftarrow 1$, $\mathcal{O} \leftarrow \{\mathbb{1}\}$, $\mathcal{G} \leftarrow \emptyset$
while $\partial_d \mathcal{O} = \{u_1, \ldots, u_k\} \neq \emptyset$ **do**
  for $i = 1, \ldots, k$ **do**
    $g \leftarrow$ construct candidate polynomial
    if g *vanishes approximately* **then**
      $\mathcal{G} \leftarrow \mathcal{G} \cup \{g\}$
    else
      $\mathcal{O} = \{t_1, \ldots, t_\ell\} \leftarrow \mathcal{O} \cup \{u_i\}$
    end
  end
  $d \leftarrow d + 1$
end

Algorithm

Algorithm 2: Oracle Approximate Vanishing Ideal Algorithm (`OAVI`)

Input : $X = \{\mathbf{x}_1, \ldots, \mathbf{x}_m\} \subseteq \mathbb{R}^n$, $\psi \geq 0$, and $\tau \geq 2$.
Output: $\mathcal{G} \subseteq \mathcal{P}$ and $\mathcal{O} \subseteq \mathcal{T}$.

$d \leftarrow 1$, $\mathcal{O} \leftarrow \{\mathbb{1}\}$, $\mathcal{G} \leftarrow \emptyset$
while $\partial_d \mathcal{O} = \{u_1, \ldots, u_k\} \neq \emptyset$ **do**
  for $i = 1, \ldots, k$ **do**
    $g \leftarrow$ constructed candidate polynomial
    if g *vanishes approximately* **then**
      $\mathcal{G} \leftarrow \mathcal{G} \cup \{g\}$
    else
      $\mathcal{O} = \{t_1, \ldots, t_\ell\} \leftarrow \mathcal{O} \cup \{u_i\}$
    end
  end
  $d \leftarrow d + 1$
end

Algorithm: Border

- Let $\mathcal{T}$ denote the set of monomials in n variables

Definition 5 (Border).

Let $\mathcal{O} \subseteq \mathcal{T}$. The *(degree-d) border* of $\mathcal{O}$ is defined as
$\partial_d \mathcal{O} = \{u \in \mathcal{T}_d : t \in \mathcal{O}_{\leq d-1}$ for all $t \in \mathcal{T}_{\leq d-1}$ such that $t \mid u\}$.

Example 6 (Simple Border Example).

Let $n = 2$ and $\mathcal{O} = \{1, x, y, xy, y^2\}$. Then, $\partial_3 \mathcal{O} = \{xy^2, y^3\}$.

y^3	xy^3	x^2y^3	x^3y^3
y^2	xy^2	x^2y^2	x^3y^2
y	xy	x^2y	x^3y
1	x	x^2	x^3

Algorithm

Algorithm 2: Oracle Approximate Vanishing Ideal Algorithm (`OAVI`)

Input : $X = \{\mathbf{x}_1, \ldots, \mathbf{x}_m\} \subseteq \mathbb{R}^n$, $\psi \geq 0$, and $\tau \geq 2$.
Output: $\mathcal{G} \subseteq \mathcal{P}$ and $\mathcal{O} \subseteq \mathcal{T}$.

$d \leftarrow 1$, $\mathcal{O} \leftarrow \{\mathbb{1}\}$, $\mathcal{G} \leftarrow \emptyset$
while $\partial_d \mathcal{O} = \{u_1, \ldots, u_k\} \neq \emptyset$ **do**
  for $i = 1, \ldots, k$ **do**
    $g \leftarrow$ constructed candidate polynomial
    if g *vanishes approximately* **then**
      $\mathcal{G} \leftarrow \mathcal{G} \cup \{g\}$
    else
      $\mathcal{O} = \{t_1, \ldots, t_\ell\} \leftarrow \mathcal{O} \cup \{u_i\}$
    end
  end
  $d \leftarrow d + 1$
end

Algorithm: Candidate Polynomial

- $X = \{\mathbf{x}_1, \ldots, \mathbf{x}_m\} \subseteq \mathbb{R}^n$
- $\mathcal{O} = \{t_1, \ldots, t_\ell\} \subseteq \mathcal{T}$ and let $A := \mathcal{O}(X) \in \mathbb{R}^{m \times \ell}$
- $u \in \partial_d \mathcal{O} \subseteq \mathcal{T}$ and let $\mathbf{b} := u(X) \in \mathbb{R}^m$
- Solve $\mathbf{c}^* \in \operatorname{argmin}_{\|\mathbf{c}\|_1 \leq \tau} \frac{1}{m}\|\mathbf{b} + A\mathbf{c}\|_2^2$
- $g \leftarrow u + \sum_{i=1}^{\ell} c_i^* t_i$

Theorem 7 (Wirth and Pokutta, 2022).

If there exists a $(\psi, 1, \tau)$-approximately vanishing polynomial, then g is one of them.

Algorithm

Algorithm 2: Oracle Approximate Vanishing Ideal Algorithm (`OAVI`)

Input : $X = \{\mathbf{x}_1, \ldots, \mathbf{x}_m\} \subseteq \mathbb{R}^n$, $\psi \geq 0$, and $\tau \geq 2$.
Output: $\mathcal{G} \subseteq \mathcal{P}$ and $\mathcal{O} \subseteq \mathcal{T}$.

```
d ← 1, O ← {𝟙}, G ← ∅
while ∂_d O = {u_1, ..., u_k} ≠ ∅ do
    for i = 1, ..., k do
        g ← constructed candidate polynomial
        if g vanishes approximately then
            G ← G ∪ {g}
        else
            O = {t_1, ..., t_ℓ} ← O ∪ {u_i}
        end
    end
    d ← d + 1
end
```

Computational Complexity: Theory

Theorem 8 (Complexity [Wirth and Pokutta, 2022]).

Let $X = \{\mathbf{x}_1, \ldots, \mathbf{x}_m\} \subseteq \mathbb{R}^n$, $\psi \in [0, 1[$, $\tau \geq 2$, *and* $(\mathcal{G}, \mathcal{O}) = \texttt{OAVI}(X, \psi, \tau)$.

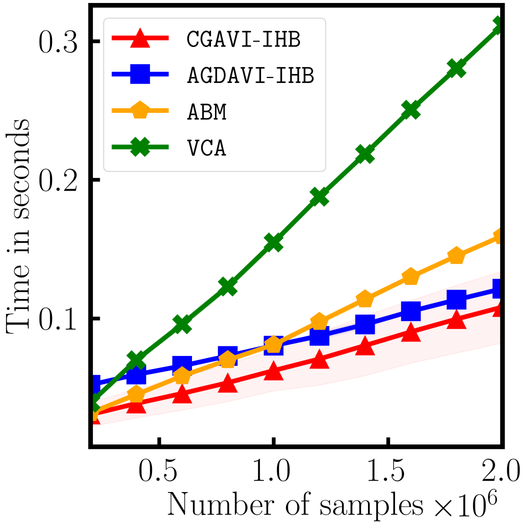
- *Time:* $O((|\mathcal{G}| + |\mathcal{O}|)^2 + (|\mathcal{G}| + |\mathcal{O}|) T_{\texttt{ORACLE}})$.
- *Space:* $O((|\mathcal{G}| + |\mathcal{O}|)m + S_{\texttt{ORACLE}})$.

- $|\mathcal{G}| + |\mathcal{O}| = O(mn)$ [Limbeck, 2013, Livni et al., 2013, Wirth and Pokutta, 2022]

Corollary 9.

- *Time:* $O(m^3)$
- *Space:* $O(m^2)$

Computational Complexity: Experiment



Computational Complexity: Theory II

- Time: $O((|\mathcal{G}| + |\mathcal{O}|)^2 + (|\mathcal{G}| + |\mathcal{O}|)\, T_{\texttt{ORACLE}})$.
- Space: $O((|\mathcal{G}| + |\mathcal{O}|)m + S_{\texttt{ORACLE}})$.

Theorem 10 ([Wirth et al., 2022]).

Let $X = \{\mathbf{x}_1, \ldots, \mathbf{x}_m\} \subseteq [0,1]^n$, $\psi \in]0,1[$, $D = \lceil -\log(\psi)/\log(4) \rceil$, $\tau \geq (3/2)^D$, *and* $(\mathcal{G}, \mathcal{O}) = \texttt{OAVI}(X, \psi, \tau)$. *Then,* $|\mathcal{G}| + |\mathcal{O}| \leq \binom{D+n}{D}$.

Corollary 11.

- *Time:* $O(m^3) \Rightarrow O(m)$
- *Space:* $O(m^2) \Rightarrow O(m)$

Learning Guarantees

Theorem 12 ([Wirth and Pokutta, 2022, Wirth et al., 2022]).

Let $\mathcal{X} \subseteq [-1,1]^n$, *let* $\psi \in]0,1[$, *let* $D = \lceil -\log(\psi)/\log(4) \rceil$, *let* $\tau \geq (3/2)^D$, *let* $k = \binom{D+n}{D} \leq \left(\frac{e(D+n)}{D}\right)^D$, *and let* $X = \{\mathbf{x}_1, \ldots, \mathbf{x}_m\} \in \mathcal{X}^m$ *be drawn i.i.d. according to a distribution* $\mathcal{D}$. *Let* $(\mathcal{G}, \mathcal{O}) = \texttt{OAVI}(X, \psi, \tau)$. *Then, for any* $\delta > 0$, *with probability at least* $1 - \delta$, *the following inequality holds for all* $g \in \mathcal{G}$:

$$\mathbb{E}_{\mathbf{x} \sim \mathcal{D}} [\mathrm{MSE}(g, \{\mathbf{x}\})] \leq \mathrm{MSE}(g, X) + 4\tau^2 \sqrt{\frac{2k \log(2(n+1)k)}{m}} + 12\tau^2 \sqrt{\frac{\log(2\delta^{-1})}{2m}}.$$

Conclusion

Contributions

- Learning Guarantees [Wirth and Pokutta, 2022]
- Computational complexity depends linearly on the number of samples [Wirth et al., 2022]

Open Problems

- Learning guarantees for related methods
- A "better" notion than approximate vanishing ideal

References I

D. Cox, J. Little, and D. O'Shea. *Ideals, varieties, and algorithms: an introduction to computational algebraic geometry and commutative algebra*. Springer Science & Business Media, 2013.

J. Limbeck. Computation of approximate border bases and applications. 2013.

R. Livni, D. Lehavi, S. Schein, H. Nachliely, S. Shalev-Shwartz, and A. Globerson. Vanishing component analysis. In *Proceedings of the International Conference on Machine Learning*, pages 597–605. PMLR, 2013.

E. Wirth and S. Pokutta. Conditional gradients for the approximately vanishing ideal. *arXiv preprint arXiv:2202.03349*, 2022.

E. Wirth, H. Kera, and S. Pokutta. Approximate vanishing ideal computations at scale. *arXiv preprint arXiv:2207.01236*, 2022.