# Conditional Gradients for the Approximately Vanishing Ideal

Elias Wirth [1,2] Sebastian Pokutta [1,2]

[1]Technische Universität Berlin, [2]Zuse Institute Berlin

## Abstract

The vanishing ideal of a set of points $X \subseteq \mathbb{R}^n$ is the set of polynomials that evaluate to 0 over all points $\mathbf{x} \in X$ and admits an efficient representation by a finite set of polynomials called generators. To accommodate the noise in the data set, we introduce the *Conditional Gradients Approximately Vanishing Ideal algorithm* (CGAVI) for the construction of the set of generators of the approximately vanishing ideal. The constructed set of generators captures polynomial structures in data and gives rise to a feature map that can, for example, be used in combination with a linear classifier for supervised learning. In CGAVI, we construct the set of generators by solving specific instances of (constrained) convex optimization problems with the *Pairwise Frank-Wolfe algorithm* (PFW). Among other things, the constructed generators inherit the LASSO generalization bound and not only vanish on the training but also on out-sample data. Moreover, CGAVI admits a compact representation of the approximately vanishing ideal by constructing few generators with sparse coefficient vectors.

## Motivation

- Classifier accuracy depends on feature quality
- We study feature transformations for linear kernel *Support Vector Machines* (SVMs) [5]
- High accuracy requires linear separability
- ⇒ **achievable via the vanishing ideal**

## Noisy data

For ease of exposition, we consider the vanishing ideal. In practice, however, data is noisy, and instead of constructing generators of the vanishing ideal, we construct generators of the approximately vanishing ideal.

## Vanishing Ideal

Given data set $X = \{\mathbf{x}_1, \ldots, \mathbf{x}_m\} \subseteq \mathbb{R}^n$,

$$\mathcal{I}_X = \{f \in \mathbb{R}[x_1, \ldots, x_n] \mid f(\mathbf{x}) = 0 \; \forall \mathbf{x} \in X\},$$

the *vanishing ideal*, succinctly characterizes $X$. By Hilbert's basis theorem [1], there exists a finite number of *generators* $g_1, \ldots, g_k \in \mathcal{I}_X$, with $k \in \mathbb{N}$, such that for any $f \in \mathcal{I}_X$, there exist $h_1, \ldots, h_k \in \mathbb{R}[x_1, \ldots, x_n]$ such that

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

## Feature transformations with generators

**Setting:**

- Input space $\mathcal{X} \subseteq [-1, 1]^n$
- Output space $\mathcal{Y} = [k]$
- 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 $\mathcal{D}$

**Goal:**

- Determine a *hypothesis* $h \colon \mathcal{X} \to \mathcal{Y}$ with small *generalization error* $\mathbb{P}_{(\mathbf{x},y)\sim\mathcal{D}}[h(\mathbf{x}) \neq y]$

**Pipeline:**

- Let $X = \{\mathbf{x}_1, \ldots, \mathbf{x}_m\}$
- For all $i \in [k]$, let $X^i \subseteq X$ denote the set of feature vectors corresponding to class $i$
- For all $i \in [k]$, construct a set of generators $\mathcal{G}^i = \{g_j^{(i)}\}_{j=1}^{|\mathcal{G}^i|}$ for the vanishing ideal $\mathcal{I}_{X^i}$
- Transform samples $\mathbf{x} \in X$ via the feature transformation
$$\mathbf{x} \mapsto \tilde{\mathbf{x}} = \left(\ldots, |g_1^{(i)}(\mathbf{x})|, \ldots, |g_{|\mathcal{G}^i|}^{(i)}(\mathbf{x})|, \ldots\right)^\intercal$$
- $\tilde{S} = \{(\tilde{\mathbf{x}}, y) \mid (\mathbf{x}, y) \in S\}$ is linearly separable
- Train a linear kernel SVM on $\tilde{S}$

**Open question:**

- How to construct the sets of generators $\mathcal{G}^i$?

## Oracle Approximately Vanishing Ideal algorithm (OAVI)

```
Algorithm 1 OAVI
Input: $X = \{\mathbf{x}_1, \ldots, \mathbf{x}_m\} \subseteq \mathbb{R}^n$ and $\psi \geq \varepsilon \geq 0$.
Output: $\mathcal{G}, \mathcal{O} \subseteq \mathbb{R}[x_1, \ldots, x_n]$.
$d \leftarrow 1$
$\mathcal{O} = \{t_1\}_\sigma \leftarrow \{1\}_\sigma$
$\mathcal{G} \leftarrow \emptyset$
while $\partial_d\mathcal{O} = \{u_1, \ldots, u_k\}_\sigma \neq \emptyset$ do
    for $i = 1, \ldots, k$ do
        $g \leftarrow$ ORACLE$(X, \mathcal{O}, u_i, \varepsilon)$
        if $\mathrm{MSE}(g, X) \leq \psi$ then
            $\mathcal{G} \leftarrow \mathcal{G} \cup \{g\}$
        else
            $\mathcal{O} \leftarrow (\mathcal{O} \cup \{u_i\})_\sigma$
    $d \leftarrow d + 1$
```

**Explanation:**

- $\mathcal{O}$: set of non-leading terms of generators
- $\mathcal{G}$: set of generators
- $\partial_d\mathcal{O}$: terms of degree $d$ for which OAVI checks whether they are leading terms of generators
- ORACLE: constructs a polynomial by solving a constrained convex optimization problem

**Properties:** When ORACLE is implemented with PFW [3], OAVI is called CGAVI and

- $\mathcal{G}$ contains few and sparse generators
- Generators in $\mathcal{G}$ vanish on out-sample data
- CGAVI + linear kernel SVM ⇒ margin bound

## Numerical Experiments

We compare CGAVI to related methods such as the *Approximate Vanishing Ideal algorithm* (AVI) [2] and *Vanishing Component Analysis* (VCA) [4] as feature transformation techniques for a linear kernel SVM. We also compare the methods to a polynomial kernel SVM.

| Algorithms | | Data sets | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | bank | cancer | htru2 | iris | seeds | sonar | spam | voice | wine |
| SPAR | CGAVI | **0.65** | **0.54** | **0.53** | **0.15** | **0.13** | **0.82** | **0.37** | **0.36** | **0.51** |
| | AVI | 0.01 | 0.03 | 0.02 | 0.00 | 0.05 | 0.02 | 0.04 | 0.06 | 0.03 |
| | VCA | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| Error | CGAVI | 0.09 | 3.42 | **2.05** | 4.33 | 5.95 | **20.95** | **5.90** | 19.80 | **2.08** |
| | AVI | **0.00** | 3.46 | 2.11 | 4.00 | **4.76** | 26.43 | 6.64 | 23.14 | 3.33 |
| | VCA | **0.00** | 5.44 | 2.15 | 4.17 | 5.71 | 31.90 | 7.13 | 29.02 | 3.06 |
| | SVM | **0.00** | **2.72** | **2.05** | **3.17** | 6.79 | 21.07 | 7.22 | **18.43** | 3.19 |

Table 1: We compare the *sparsity* of the feature transformation, SPAR , larger SPAR indicating sparser generators, and the classification error on the test set in %, Error. The results are averaged over ten random 60%/40% train/test partitions and the best results in each category are in bold.

## References

[1] David Cox, John Little, and Donal OShea. *Ideals, varieties, and algorithms: an introduction to computational algebraic geometry and commutative algebra*. Springer Science & Business Media, 2013.

[2] Daniel Heldt, Martin Kreuzer, Sebastian Pokutta, and Hennie Poulisse. Approximate computation of zero-dimensional polynomial ideals. *Journal of Symbolic Computation*, 44(11):1566–1591, 2009.

[3] Simon Lacoste-Julien and Martin Jaggi. On the global linear convergence of Frank-Wolfe optimization variants. In *Advances in neural information processing systems*, pages 496–504, 2015.

[4] Roi Livni, David Lehavi, Sagi Schein, Hila Nachliely, Shai Shalev-Shwartz, and Amir Globerson. Vanishing component analysis. In *International Conference on Machine Learning*, pages 597–605, 2013.

[5] Johan AK Suykens and Joos Vandewalle. Least squares support vector machine classifiers. *Neural processing letters*, 9(3):293–300, 1999.