Provably tuning the ElasticNet across instances

Maria-Florina Balcan **Mikhail Khodak** **Dravyansh Sharma** **Ameet Talwalkar**
Carnegie Mellon University*

Abstract

An important unresolved challenge in the theory of regularization is to set the regularization coefficients of popular techniques like the ElasticNet with general provable guarantees. We consider the problem of tuning the regularization parameters of Ridge regression, LASSO, and the ElasticNet across multiple problem instances, a setting that encompasses both cross-validation and multi-task hyperparameter optimization. We obtain a novel structural result for the ElasticNet which characterizes the loss as a function of the tuning parameters as a piecewise-rational function with algebraic boundaries. We use this to bound the structural complexity of the regularized loss functions and show generalization guarantees for tuning the ElasticNet regression coefficients in the statistical setting. We also consider the more challenging online learning setting, where we show vanishing average expected regret relative to the optimal parameter pair. We further extend our results to tuning classification algorithms obtained by thresholding regression fits regularized by Ridge, LASSO, or ElasticNet. Our results are the first general learning-theoretic guarantees for this important class of problems that avoid strong assumptions on the data distribution. Furthermore, our guarantees hold for both validation and popular information criterion objectives.

1 Introduction

Ridge regression [30, 43], LASSO [41], and their generalization the ElasticNet [28] are among the most popular algorithms in machine learning and statistics, with applications to linear classification, regression, data analysis, and feature selection [15, 46, 28, 20, 24]. Given a supervised dataset $(X, y) \in \mathbb{R}^{m\times p} \times \mathbb{R}^m$ with m datapoints and p features, these algorithms compute the linear predictor

$$\hat{\beta}^{(X,y)}_{\lambda_1,\lambda_2} = \operatorname*{arg\,min}_{\beta\in\mathbb{R}^p} \|y - X\beta\|_2^2 + \lambda_1\|\beta\|_1 + \lambda_2\|\beta\|_2^2 \tag{1}$$

Here $\lambda_1, \lambda_2 \geq 0$ are *regularization coefficients* constraining the ℓ_1 and ℓ_2 norms, respectively, of the model β. For general λ_1 and λ_2 the above algorithm is the ElasticNet, while setting $\lambda_1 = 0$ recovers Ridge and setting $\lambda_2 = 0$ recovers LASSO.

These coefficients play a crucial role across fields: in machine learning controlling the norm of β implies provable generalization guarantees and prevent over-fitting in practice [34], in data analysis their combined use yields parsimonious and interpretable models [28], and in Bayesian statistics they correspond to imposing specific priors on β [35, 33]. In practice, λ_2 regularizes β by uniformly shrinking all coefficients, while λ_1 encourages the model vector to be sparse. This means that while they do yield learning-theoretic and statistical benefits, setting them to be too high will cause models to under-fit the data. The question of how to set the regularization coefficients becomes even more unclear in the case of the ElasticNet, as one must juggle trade-offs between sparsity, feature correlation, and bias when setting both λ_1 and λ_2 simultaneously. As a result, there has been intense empirical and theoretical effort devoted to automatically tuning these parameters. Yet the

---

*Correspondence: `dravyans@cs.cmu.edu`. Author emails: `{ninamf,dravyans}@cs.cmu.edu`, `{khodak,talwalkar}@cmu.edu`

36th Conference on Neural Information Processing Systems (NeurIPS 2022).

state-of-the-art is quite unsatisfactory: proposed work consists of either heuristics without formal guarantees [26, 31], approaches that optimize over a finite grid or random set instead of the full continuous domain [17], or analyses that involve very strong theoretical assumptions [44].

In this work, we study a variant on the above well-established and intensely studied formulation. The key distinction is that instead of a single dataset (X, y), we consider a collection of datasets or instances of the same underlying regression problem $(X^{(i)}, y^{(i)})$ and would like to learn a pair (λ_1, λ_2) that selects a model in equation (1) that has low loss on a validation dataset. This can be useful to model practical settings, for example where new supervised data is obtained several times or where the set of features may change frequently [19]. We do not require all examples across datasets to be i.i.d. draws from the same data distribution, and can capture more general data generation scenarios like cross-validation and multi-task learning [45]. Despite these advantages, we remark that our problem formulation is quite different from the standard single dataset setting. Our formulation treats the selection of regularization coefficients as *data-driven algorithm design*, which is often used to study combinatorial problems [27, 3], and has connections to meta-learning [12].

Our main contribution is a new structural result for the ElasticNet Regression problem, which implies generalization guarantees for selecting ElasticNet Regression coefficients in the multiple-instance setting. In particular, Ridge and LASSO regressions are special cases. We extend our results to obtain low regret in the online learning setting, and to tuning related linear classification algorithms. In summary, we make the following key contributions:

- We formulate the problem of tuning the ElasticNet as a question of learning λ_1 and λ_2 simultaneously across multiple problem instances, either generated statistically or coming online. Our formulation captures relevant settings like cross-validation and multi-task learning.
- We provide a novel structural result (Theorem 2.2) that characterizes the loss of the ElasticNet fit. We show that the hyperparameter space can be partitioned by polynomial curves of bounded degrees into pieces where the loss is a bivariate rational function. The result holds for both the usual ElasticNet validation objective and when it is augmented with information criteria like the AIC or BIC.
- An important consequence of our structural result is a bound on the pseudo-dimension (Definition 5) for the loss function class, which yields strong generalization bounds for tuning λ_1 and λ_2 simultaneously in the statistical learning setting (Theorem 3.2). Informally, for ElasticNet regression problems with at most p parameters, for any problem distribution $\mathcal{D}$, we show that $O\left(\frac{1}{\epsilon^2}(p^2 \log \frac{1}{\epsilon} + \log \frac{1}{\delta})\right)$ problem instances (or datasets) are sufficient to learn an ϵ-approximation to the best (λ_1, λ_2), with probability at least $1 - \delta$.
- In the online setting, we show under very mild data assumptions—much weaker than prior work—that the problem satisfies a dispersion condition [6, 9]. As a result we can tune all parameters across a sequence of instances appearing online and obtain vanishing regret relative to the optimal parameter in hindsight over the sequence (Theorem 3.3) at the rate $\tilde{O}(1/\sqrt{T})$[2] wrt the length T of the sequence.
- We also give distributional and online learning results for regularized classifiers (Theorems 4.1, 4.2).

We include a couple of remarks to emphasize the generality and significance of our results. First, in our multiple-instance formulation the different problem instances need not have the same number of examples, or even the same set of features. This allows us to handle practical scenarios where the set of features changes across datasets, and we can learn parameters that work well on average across multiple different but related regression tasks. Second, by generating problem instances iid from a fixed (training + validation) dataset, we can obtain iterations (training/validation splits) of popular cross-validation techniques (including the popular leave-one-out and Monte Carlo CV) and our result implies that $\tilde{O}(p^2/\epsilon^2)$ iterations are enough to determine an ElasticNet parameter $\hat{\lambda}$ with loss within ϵ (w.h.p.) of the optimal parameter λ^* over the distribution induced by the cross-validation splits.

Key challenges and insights. A major challenge in learning the ElasticNet parameters is that the variation of the solution path as a function of λ_2 is hard to characterize. Indeed the original ElasticNet paper [47] suggests using the heuristic of grid search to learn a good λ_2, even though λ_1 may be exactly optimized by computing full solution paths (for each λ_2). We approach this indirectly by utilizing a

---

[2] The soft-O notation is used to emphasize dependence on T, and suppresses other factors as well as logarithmic terms.

characterization of the LASSO solution by [42], which is based on the KKT (Karush–Kuhn–Tucker) optimality conditions, to arrive at a precise piecewise structure for the problem. In more detail, we use these conditions to come up with a set of algebraic curves (polynomial equations in λ_1 and λ_2) of bounded degrees, such that the set of possible discontinuities is contained within the zero-set of these curves, and the loss function behaves well in the each piece of the partition of the parameter domain by these curves. This characterization is crucial in establishing a bound on the structural complexity needed to provide strong generalization guarantees. We further show additional structure on these algebraic curves that (roughly speaking) imply that the curves do not concentrate in any region of the domain, allowing us to use the powerful recipe of [8] for online learning.

Related work. Model selection for Ridge regression, LASSO or ElasticNet typically involves selecting the regularization parameter λ for given data, although some parameter-free techniques for variable selection have been recently proposed [32]. Choosing 'optimal' parameters for tuning the regularization has been a subject of extensive theoretical and applied research. Much of this effort is heuristic [26, 31] or focused on developing tuning objectives beyond validation accuracy like AIC or BIC [1, 39] without providing procedures for provably optimizing them. The standard approach given a tuning objective is to optimize it over a grid or random set of parameters, for which there are guarantees [17], but this does not ensure optimality over the entire continuous tuning domain, especially since objectives such as 0-1 validation error or information criteria can have many discontinuities. Selecting a grid that is too fine or too coarse can result in either very inefficient or highly inaccurate estimates (respectively) for good parameters. Other guarantees make strong assumptions on the data distribution such as sub-Gaussian noise [44, 16] or depend on unknown parameters that are hard to quantify in practice [23]. Recent work has shown asymptotic consistency of cross-validation for ridge regression, even in the limiting case $\lambda_2 \to 0$ which is particularly interesting for the overparameterized regime [29, 36]. A successful line of work has focused on efficiently obtaining models for different values of λ_1 using regularization paths [22], but the guarantees are computational rather than learning-theoretic or statistical. In contrast, we provide principled approaches that guarantee near-optimality of selected parameters with high confidence over the entire continuous domain of parameters.

Data-driven algorithm design has proved successful for tuning parameters for a variety of combinatorial problems like clustering, integer programming, auction design and graph-based learning [7, 11, 5, 4]. We provide an application of these techniques to parameter tuning in a problem that is not inherently combinatorial by revealing a novel discrete structure. We identify the underlying piecewise structure of the ElasticNet loss function which is extremely effective in establishing learning-theoretic guarantees [10]. To exploit this piecewise structure, we analyze the learning-theoretic complexity of rational algebraic function classes and infer generalization guarantees. We also employ and extend general tools and techniques for online data-driven learning from [8, 4] to rational functions in order to prove our online learning guarantees for regularization coefficient tuning.

2 Preliminaries and a Key Structural Result

Given data (X, y) with $X \in \mathbb{R}^{m\times p}$ and $y \in \mathbb{R}^m$, consisting of m labeled examples with p features, we seek estimators $\beta \in \mathbb{R}^p$ which minimize the regularized loss. Popular regularization methods like LASSO and ElasticNet can be expressed as computing the solution of an optimization problem

$$\hat{\beta}^{(X,y)}_{\lambda,f} \in \operatorname*{arg\,min}_{\beta\in\mathbb{R}^p} \|y - X\beta\|_2^2 + \langle \lambda, f(\beta)\rangle$$

where $f : \mathbb{R}^p \to \mathbb{R}^d_{\geq 0}$ gives the regularization penalty for estimator β, $\lambda \in \mathbb{R}^d_{\geq 0}$ is the regularization parameter, and d is the number of regularization parameters. $d = 1$ for Ridge and LASSO, and $d = 2$ for the ElasticNet. Setting $f = f_2$ with $f_2(\beta) = \|\beta\|_2^2$ yields Ridge regression, and setting $f(\beta) = f_1(\beta) := \|\beta\|_1$ corresponds to LASSO. Also using $f_{\text{EN}}(\beta) = (f_1(\beta), f_2(\beta))$ gives the ElasticNet with regularization parameter $\lambda = (\lambda_1, \lambda_2)$. Note that we use the same λ (with some notational overloading) to denote the regularization parameters for ridge, LASSO, or ElasticNet. We write $\hat{\beta}^{(X,y)}_{\lambda,f}$ as simply $\hat{\beta}_{\lambda,f}$ when the dataset (X, y) is clear from context. On any instance $x \in \mathbb{R}^p$ from the feature space, the prediction of the regularized estimator is given by the dot product $\langle x, \hat{\beta}_{\lambda,f}\rangle$. The average squared loss over a dataset (X', y') with $X' \in \mathbb{R}^{m'\times p}$ and $y' \in \mathbb{R}^{m'}$ is given by $l_r(\hat{\beta}_{\lambda,f},(X',y')) = \frac{1}{m'}\left\|y' - X'\hat{\beta}_{\lambda,f}\right\|_2^2$. By setting (X',y') to be the training data (X,y), we get the training loss $l_r(\hat{\beta}_{\lambda,f},(X,y))$. We use $(X_{\text{val}},y_{\text{val}})$ to denote a validation split.

Distributional and Online Settings. In the *distributional or statistical* setting, we receive a collection of n instances of the regression problem $P^{(i)} = (X^{(i)},y^{(i)},X^{(i)}_{\text{val}},y^{(i)}_{\text{val}}) \in \mathcal{R}_{m_i,p_i,m'_i} := \mathbb{R}^{m_i\times p_i}\times\mathbb{R}^{m_i}\times\mathbb{R}^{m'_i\times p_i}\times\mathbb{R}^{m'_i}$ for $i\in[n]$ generated i.i.d. from some problem distribution $\mathcal{D}$. The problems are in the problem space given by $\Pi_{m,p} = \bigcup_{m_1\geq 0,m_2\leq m,p_1\leq p}\mathcal{R}_{m_1,p_1,m_2}$ (note that the problem distribution $\mathcal{D}$ is over $\Pi_{m,p}$). On any given instance $P^{(i)}$ the loss is given by the squared loss on the validation set, $\ell_{\text{EN}}(\lambda,P^{(i)}) = l_r(\hat{\beta}^{(X^{(i)},y^{(i)})}_{\lambda,f_{\text{EN}}},(X^{(i)}_{\text{val}},y^{(i)}_{\text{val}}))$. On the other hand, in the *online setting*, we receive a sequence of T instances of the ElasticNet regression problem $P^{(i)} = (X^{(i)},y^{(i)},X^{(i)}_{\text{val}},y^{(i)}_{\text{val}}) \in \Pi_{m,p}$ for $i\in[T]$ online. On any given instance $P^{(i)}$, the online learner is required to select the regularization parameter $\lambda^{(i)}$ without observing $y^{(i)}_{\text{val}}$, and experiences loss given by $\ell(\lambda^{(i)},P^{(i)}) = l_c(\hat{\beta}^{(X^{(i)},y^{(i)})}_{\lambda^{(i)},f_{EN}},(X^{(i)}_{\text{val}},y^{(i)}_{\text{val}}))$. The goal is to minimize the regret w.r.t. choosing the best fixed parameter in hindsight for the same problem sequence, i.e. $R_T = \sum_{i=1}^T \ell(\lambda^{(i)},P^{(i)}) - \min_\lambda\sum_{i=1}^T\ell(\lambda,P^{(i)})$. We also define average regret as $\frac{1}{T}R_T$ and expected regret as $\mathbb{E}[R_T]$ where the expectation is over both the randomness of the loss functions and any random coins used by the online algorithm.

Given a class of regularization algorithms $\mathcal{A}$ parameterized by regularization parameter λ over a set of problem instances $\mathcal{X}$, and given loss function $\ell:\mathcal{A}\times\mathcal{X}\to\mathbb{R}$ which measures the loss of any algorithm in $\mathcal{A}$ on any fixed problem instance, consider the set of functions $\mathcal{H}_{\mathcal{A}} = \{\ell(A,\cdot)\mid A\in\mathcal{A}\}$. For example, for the ElasticNet we have $\ell_{\text{EN}}(\lambda,P) = l_r(\hat{\beta}^{(X_P,y_P)}_{\lambda,f_{\text{EN}}},(X'_P,y'_P))$, where (X_P,y_P) and (X'_P,y'_P) are the training and validation sets associated with problem $P\in\mathcal{X}$ respectively. Bounding the pseudo-dimension of $\mathcal{H}_{\mathcal{A}}$ gives a bound on the sample complexity for uniform convergence guarantees, i.e. a bound on the sample size n for which the algorithm $\hat{A}_S\in\mathcal{A}$ which minimizes the average loss on any sample S of size n drawn i.i.d. from any problem distribution $\mathcal{D}$ is guaranteed to be near-optimal with high probability [21]. See Appendix A for the relevant classic definitions and results. Define the *dual class* $\mathcal{H}^*$ of a set of real-valued functions $\mathcal{H}\subseteq 2^{\mathcal{X}}$ as $\mathcal{H}^* = \{h^*_x:\mathcal{H}\to\mathbb{R}\mid x\in\mathcal{X}\}$ where $h^*_x(h) = h(x)$. In the context of regression problems $\mathcal{X}$, for each fixed problem instance $x\in\mathcal{X}$ there is a dual function h^*_x that computes the loss $\ell(A,x)$ for any (primal) function $h_A = \ell(A,\cdot)\in\mathcal{H}_{\mathcal{A}}$. For a function class $\mathcal{H}$, showing that dual class $\mathcal{H}^*$ is piecewise-structured in the sense of Definition 1 and bounding the complexity of the duals of boundary and piece functions of $\mathcal{H}^*$ are useful to understand the learnability of $\mathcal{H}$ [10].

Definition 1 (Piecewise structured functions, [10]). *A function class $H\subseteq\mathbb{R}^{\mathcal{X}}$ that maps a domain $\mathcal{X}$ to $\mathbb{R}$ is* (F,G,k)-piecewise decomposable *for a class $G\subseteq\{0,1\}^{\mathcal{X}}$ of boundary functions and a class $F\subseteq\mathbb{R}^{\mathcal{X}}$ of piece functions if the following holds: for every $h\in H$, there are k boundary functions $g_1,\dots,g_k\in G$ and a piece function $f_{\mathbf{b}}\in F$ for each bit vector $\mathbf{b}\in\{0,1\}^k$ such that for all $x\in\mathcal{X}$, $h(x) = f_{\mathbf{b}_x}(x)$ where $\mathbf{b}_x = (g_1(x),\dots,g_k(x))\in\{0,1\}^k$.*

Intuitively, a real-valued function is piecewise-structured if the domain can be divided into pieces by a finite number of boundary functions (say linear or polynomial thresholds) and the function value over each piece is easy to characterize (e.g. constant, linear, polynomial). To state and understand our structural insights into the ElasticNet problem we will also need the definition of equicorrelation sets, the subset of features with maximum absolute correlation for any fixed λ_1, useful for characterizing LASSO/ElasticNet solutions. For any subset $\mathcal{E}\subseteq[p]$ of the features, we define $X_{\mathcal{E}} = (\dots X_{*i}\dots)_{i\in\mathcal{E}}$ as the $m\times|\mathcal{E}|$ matrix of columns X_{*i} of X corresponding to indices $i\in\mathcal{E}$. Similarly $\beta_{\mathcal{E}}\in\mathbb{R}^{|\mathcal{E}|}$ is the subset of estimators in β corresponding to indices in $\mathcal{E}$. We will assume all the feature matrixes X (for training datasets) are in general position (Definition 6).

Definition 2 (Equicorrelation sets, [42]). *Let $\beta^*\in\arg\min_{\beta\in\mathbb{R}^p}\|y-X\beta\|_2^2+\lambda_1\|\beta\|_1$. The equicorrelation set corresponding to β^*, $\mathcal{E} = \{j\in[p]\mid|\boldsymbol{x}_j^T(y-X\beta^*)| = \lambda_1\}$, is simply the set of covariates with maximum absolute correlation. We also define the equicorrelation sign vector for β^* as $s = \mathsf{sign}(X_{\mathcal{E}}^T(y-X\beta^*))\in\{\pm 1\}$.*

Consider the class of algorithms consisting of ElasticNet regressors for different values of $\lambda = (\lambda_1,\lambda_2)\in(0,\infty)\times(0,\infty)$. We assume $\lambda_1>0$ for technical simplicity (cf. [42]). We seek to solve problems of the form $P = (X, y, X_{\text{val}}, y_{\text{val}}) \in \Pi_{m,p}$, where (X, y) is the training set, $(X_{\text{val}}, y_{\text{val}})$ is the validation set with the same set of features, and m, p are upper bounds on the number of examples and features respectively in any dataset. Let $\mathcal{H}_{\text{EN}} = \{\ell_{\text{EN}}(\lambda, \cdot) \mid \lambda \in (0, \infty) \times (0, \infty)\}$ denote the set of loss functions for the class of algorithms consisting of ElasticNet regressors for different values of $\lambda \in \mathbb{R}^+ \times \mathbb{R}^+$. Additionally, we will consider information criterion based loss functions, $\ell_{\text{EN}}^{\text{AIC}}(\lambda, P) = \ell_{\text{EN}}(\lambda, P) + 2||\hat{\beta}^{(X,y)}_{\lambda, f_{\text{EN}}}||_0$ and $\ell_{\text{EN}}^{\text{BIC}}(\lambda, P) = \ell_{\text{EN}}(\lambda, P) + 2||\hat{\beta}^{(X,y)}_{\lambda, f_{\text{EN}}}||_0 \log m$ [1, 39]. Let $\mathcal{H}_{\text{EN}}^{\text{AIC}}$ and $\mathcal{H}_{\text{EN}}^{\text{BIC}}$ denote the corresponding sets of loss functions. These criteria are popularly used to compute the squared loss on the training set, to give alternatives to cross-validation. We do not make any assumption on the relation between training and validation sets in our formulation, so our analysis can capture these settings as well.

We will now establish a piecewise structure of the dual class loss functions (Definition 1). A key observation is that if the signed equicorrelation set $(\mathcal{E}, s)$ (i.e. a subset of features $\mathcal{E} \subseteq [p]$ with the same maximum absolute correlation, assigned a fixed sign pattern $\{-1, +1\}^{|\mathcal{E}|}$, see Definition 2) is fixed, then the ElasticNet coefficients may be characterized (Lemma C.1) and the loss is a fixed rational polynomial piece function of the parameters λ_1, λ_2. We then show the existence of a set of boundary function curves $\mathcal{G}$, such that any region of the parameter space located on a fixed side of all the curves (more formally, for a fixed sign pattern in Definition 1) in $\mathcal{G}$ has the same signed equicorrelation set. The boundary functions are a collection of possible curves at which a covariate may enter or leave the set $\mathcal{E}$ and correspond to algebraic curves. We make repeated use of the following lemma which provides useful properties of the piece functions as well the the boundary functions of the dual class loss functions.

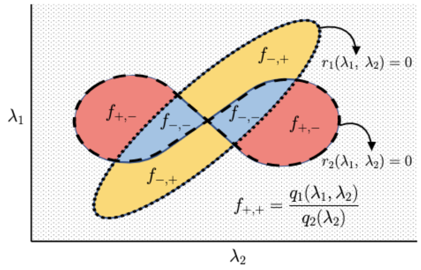
λ_1
λ_2
$f_{-,+}$
$f_{+,-}$
$f_{-,-}$
$f_{-,-}$
$f_{+,-}$
$f_{-,+}$
$r_1(\lambda_1, \lambda_2) = 0$
$r_2(\lambda_1, \lambda_2) = 0$
$f_{+,+} = \frac{q_1(\lambda_1, \lambda_2)}{q_2(\lambda_2)}$

Figure 1: An illustration of the piecewise structure of the ElasticNet loss, as a function of the regularization parameters, for a fixed problem instance. Pieces are regions where some bounded degree polynomials (r_1, r_2) have a fixed sign pattern (one of $\pm 1, \pm 1$), and in each piece the loss is a fixed (rational) function.

Lemma 2.1. *Let A be an $r \times s$ matrix. Consider the matrix $B(\lambda) = (A^T A + \lambda I_s)^{-1}$ and $\lambda > 0$.*

1. *Each entry of $B(\lambda)$ is a rational polynomial $P_{ij}(\lambda)/Q(\lambda)$ for $i, j \in [s]$ with each P_{ij} of degree at most $s - 1$, and Q of degree s.*
2. *Further, for $i = j$, P_{ij} has degree $s - 1$ and leading coefficient 1, and for $i \neq j$ P_{ij} has degree at most $s - 2$. Also, $Q(\lambda)$ has leading coefficient 1.*

The proof is straightforward (Appendix C). We will now formally state and prove our key structural result which is needed to establish our generalization and online regret guarantees in Section 3.

Theorem 2.2. *Let $\mathcal{L}$ be a set of functions $\{l_\lambda : \Pi_{m,p} \to \mathbb{R}_{\geq 0} \mid \lambda \in \mathbb{R}^+ \times \mathbb{R}_{\geq 0}\}$ that map a regression problem instance $P \in \Pi_{m,p}$ to the validation loss $\ell_{EN}(\lambda, P)$ of ElasticNet trained with regularization parameter $\lambda = (\lambda_1, \lambda_2)$. The dual class $\mathcal{L}^*$ is $(\mathcal{F}, \mathcal{G}, p3^p)$-piecewise decomposable, with $\mathcal{F} = \{f_q : \mathcal{L} \to \mathbb{R}\}$ consisting of rational polynomial functions $f_{q_1, q_2} : l_\lambda \mapsto \frac{q_1(\lambda_1, \lambda_2)}{q_2(\lambda_2)}$, where q_1, q_2 have degrees at most $2p$, and $\mathcal{G} = \{g_r : \mathcal{L} \to \{0, 1\}\}$ consisting of semi-algebraic sets[3] bounded by algebraic curves $g_r : u_\lambda \mapsto \mathbb{I}\{r(\lambda_1, \lambda_2) < 0\}$, where r is a polynomial of degree 1 in λ_1 and at most p in λ_2.*

Proof. Let $P = (X, y, X_{\text{val}}, y_{\text{val}}) \in \Pi_{m,p}$ be a regression problem instance. By using the standard reduction to LASSO [47] and well-known characterization of the LASSO solution in terms of equicorrelation sets, we can characterize the solution $\hat{\beta}_{\lambda, f_{EN}}$ of the Elastic Net as follows (Lemma C.1):

$$\hat{\beta}_{\lambda, f_{EN}} = (X_{\mathcal{E}}^T X_{\mathcal{E}} + \lambda_2 I_{|\mathcal{E}|})^{-1} X_{\mathcal{E}}^T y - \lambda_1 (X_{\mathcal{E}}^T X_{\mathcal{E}} + \lambda_2 I_{|\mathcal{E}|})^{-1} s$$

for some $\mathcal{E} \in [p]$ and $s \in \{-1, 1\}^p$. Thus for any $\lambda = (\lambda_1, \lambda_2)$, the prediction $\hat{y}$ on any validation example with features $\boldsymbol{x} \in \mathbb{R}^p$ satisfies (for some $\mathcal{E}, s \in 2^{[p]} \times \{-1, 1\}^p$)

$$\hat{y}_j = \boldsymbol{x} \hat{\beta}_{\lambda, f_{EN}} = \boldsymbol{x} (X_{\mathcal{E}}^T X_{\mathcal{E}} + \lambda_2 I_{|\mathcal{E}|})^{-1} X_{\mathcal{E}}^T y - \lambda_1 \boldsymbol{x} (X_{\mathcal{E}}^T X_{\mathcal{E}} + \lambda_2 I_{|\mathcal{E}|})^{-1} s$$

---

[3] See Definition 7 for definitions of standard terminology from algebraic geometry.

For any subset $R \subseteq \mathbb{R}^2$, if the signed equicorrelation set $(\mathcal{E}, s)$ is fixed over R, then the above observation, together with Lemma C.2 implies that the loss function $\ell_{\text{EN}}(\lambda, P)$ is a rational function of the form $\frac{q_1(\lambda_1,\lambda_2)}{q_2(\lambda_2)}$, where q_1 is a bivariate polynomial with degree at most $2|\mathcal{E}|$ and q_2 is univariate with degree $2|\mathcal{E}|$.

To show the piecewise structure, we need to demonstrate a set boundary functions $\mathcal{G} = \{g_1, \dots, g_k\}$ such that for any sign pattern $\mathbf{b} \in \{0,1\}^k$, the signed equicorrelation set $(\mathcal{E}, s)$ for the region with sign pattern $\mathbf{b}$ is fixed. To this end, based on the observation above, we will consider the conditions (on λ) under which a covariate may enter or leave the equicorrelation set. We will show that this can happen only at one of a finite number of algebraic curves (with bounded degrees).

Condition for joining $\mathcal{E}$. Fix $\mathcal{E}, s$. Also fix $j \notin \mathcal{E}$. If covariate j enters the equicorrelation set, the KKT conditions (Lemma B.1) applied to the LASSO problem corresponding to the ElasticNet (Lemma C.1) imply

$$(\boldsymbol{x}_j^*)^T(y^* - X_{\mathcal{E}}^*(c_1 - c_2\lambda_1^*)) = \pm\lambda_1^*,$$

where $c_1 = ({X_{\mathcal{E}}^*}^T X_{\mathcal{E}}^*)^{-1} {X_{\mathcal{E}}^*}^T y^*$, $c_2 = ({X_{\mathcal{E}}^*}^T X_{\mathcal{E}}^*)^{-1} s$, $X^* = \frac{1}{\sqrt{1+\lambda_2}} \begin{pmatrix} X \\ \sqrt{\lambda_2} I_p \end{pmatrix}$, $y^* = \begin{pmatrix} y \\ 0 \end{pmatrix}$, and $\lambda_1^* = \frac{\lambda_1}{\sqrt{1+\lambda_2}}$. Rearranging, and simplifying, we get

$$\lambda_1^* = \frac{(\boldsymbol{x}_j^*)^T X_{\mathcal{E}}^* ({X_{\mathcal{E}}^*}^T X_{\mathcal{E}}^*)^{-1} (X_{\mathcal{E}}^*)^T y^* - (\boldsymbol{x}_j^*)^T y^*}{(\boldsymbol{x}_j^*)^T X_{\mathcal{E}}^* ({X_{\mathcal{E}}^*}^T X_{\mathcal{E}}^*)^{-1} s \pm 1}, \textit{or}$$

$$\lambda_1 = \frac{\boldsymbol{x}_j^T X_{\mathcal{E}} ({X_{\mathcal{E}}}^T X_{\mathcal{E}} + \lambda_2 I_{|\mathcal{E}|})^{-1} {X_{\mathcal{E}}}^T y - \boldsymbol{x}_j^T y}{\boldsymbol{x}_j^T X_{\mathcal{E}} ({X_{\mathcal{E}}}^T X_{\mathcal{E}} + \lambda_2 I_{|\mathcal{E}|})^{-1} s \pm 1}.$$

Note that the terms $(\boldsymbol{x}_j^*)^T X_{\mathcal{E}}^* = \boldsymbol{x}_j^T X_{\mathcal{E}}$, $(X_{\mathcal{E}}^*)^T y^* = X_{\mathcal{E}}^T y$, and $(\boldsymbol{x}_j^*)^T y^* = \boldsymbol{x}_j^T y$ do not depend on λ_1 or λ_2 (the λ_2 terms are zeroed out since $j \notin \mathcal{E}$). Moreover, $({X_{\mathcal{E}}^*}^T X_{\mathcal{E}}^*)^{-1} = ({X_{\mathcal{E}}}^T X_{\mathcal{E}} + \lambda_2 I_{|\mathcal{E}|})^{-1}$. Using Lemma C.2, we get an algebraic curve $r_{j,\mathcal{E},s}(\lambda_1, \lambda_2) = 0$ with degree 1 in λ_1 and $|\mathcal{E}|$ in λ_2 corresponding to addition of $j \notin \mathcal{E}$ given $\mathcal{E}, s$.

Condition for leaving $\mathcal{E}$. Now consider a fixed $j' \in \mathcal{E}$, given fixed $\mathcal{E}, s$. The coefficient of j' will be zero for $\lambda_1^* = \frac{(c_1)_{j'}}{(c_2)_{j'}}$, which simplifies to $\lambda_1(({X_{\mathcal{E}}}^T X_{\mathcal{E}} + \lambda_2 I_{|\mathcal{E}|})^{-1} s)_{j'} = (({X_{\mathcal{E}}}^T X_{\mathcal{E}} + \lambda_2 I_{|\mathcal{E}|})^{-1} {X_{\mathcal{E}}}^T y)_{j'}$. Again by Lemma C.2, we get an algebraic curve $r_{j',\mathcal{E},s}(\lambda_1, \lambda_2) = 0$ with degree 1 in λ_1 and at most $|\mathcal{E}|$ in λ_2 corresponding to removal of $j' \in \mathcal{E}$ given $\mathcal{E}, s$.

Putting the two together, we get $\sum_{i=0}^{p} 2^i \binom{p}{i} ((p-i)+i) = p3^p$ algebraic curves of degree 1 in λ_1 and at most p in λ_2, across which the signed equicorrelation set may change. These curves characterize the complete set of points (λ_1, λ_2) at which $(\mathcal{E}, s)$ may possibly change. Thus by setting these $p3^p$ curves as the set of boundary functions $\mathcal{G}$, $\mathcal{E}, s$ is guaranteed to be fixed for each sign pattern, and the corresponding loss takes the rational function form shown above. ☐

The exact same piecewise structure can be established for the dual function classes for loss functions $\ell_{\text{EN}}^{\text{AIC}}(\lambda, \cdot)$ and $\ell_{\text{EN}}^{\text{BIC}}(\lambda, \cdot)$. This is evident from the proof of Theorem 2.2, since any dual piece has a fixed equicorrelation set, and therefore $||\beta||_0$ is fixed. Given this piecewise structure, a challenge to learning values of λ that minimize the loss function is that the function may not be differentiable (or may even be discontinuous, for the information criteria based losses) at the piece boundaries, making well-known gradient-based (local) optimization techniques inapplicable here. In the following (specifically Algorithm 1) we will show that techniques from data-driven design may be used to overcome this optimization challenge.

3 Learning to Regularize the ElasticNet

We will consider the problem of learning provably good ElasticNet parameters for a given problem domain, from multiple datasets (problem instances) either available as a collection (Section 3.1), or arriving online (Section 3.2). Our parameter tuning techniques also apply to simpler regression techniques typically used for variable selection, like LARS and LASSO, which are reasonable choices if the features are not multicollinear. Proof details for this section are located in Appendix C.

3.1 Distributional Setting

Our main result in this section is the following upper bound on the pseudo-dimension of the classes of loss functions for the ElasticNet, which implies that in our distributional setting it is possible to learn near-optimal values of λ with polynomially many problem instances.

Theorem 3.1. $\textsc{Pdim}(\mathcal{H}_{EN}) = O(p^2)$. *Further,* $\textsc{Pdim}(\mathcal{H}_{EN}^{\mathsf{AIC}}) = O(p^2)$ *and* $\textsc{Pdim}(\mathcal{H}_{EN}^{\mathsf{BIC}}) = O(p^2)$.

Proof Sketch. We use the $(\mathcal{F}, \mathcal{G}, p3^p)$-piecewise decomposable structure for the dual class function $\mathcal{H}_{\text{EN}}^*$ established in Theorem 2.2. We can bound the pseudo-dimension of the dual class of piece functions $\mathcal{F}^*$ (a class of bivariate rational functions) by $O(\log p)$ by giving an upper bound (of $O(k^3 d^3)$) on the number of sign patterns over $\mathbb{R}^2$ induced by k algebraic curves of degree at most d. We can also bound the VC dimension of the dual class of boundary functions $\mathcal{G}^*$ (semi-algebraic sets in two variates) by $O(p)$ using a standard linearization argument. Finally, a powerful result from [10] (Theorem C.3) allows us to bound the pseudodimension of $\mathcal{H}$ by combining the above results. $\square$

A key challenge to establish the theorem is providing new bounds on the pseudo-dimension of rational functions of bounded degrees (Lemma C.5). The upper bound above implies a guarantee on the sample complexity of learning the ElasticNet tuning parameter, using standard learning-theoretic results [2]. In our setting of learning from multiple problem instances, each sample is a dataset instance, so the sample complexity is simply the number of regression problem instances needed to learn the tuning parameters to any given approximation and confidence level.

Theorem 3.2 (Sample complexity of tuning the ElasticNet)**.** *Let $\mathcal{D}$ be an arbitary distribution over the problem space $\Pi_{m,p}$. There is an algorithm which given $n = O\left(\frac{1}{\epsilon^2}(p^2 \log \frac{1}{\epsilon} + \log \frac{1}{\delta})\right)$ problem samples drawn from $\mathcal{D}$, for any $\epsilon > 0$ and $\delta \in (0,1)$, outputs a regularization parameter $\hat{\lambda}$ for the ElasticNet such that with probability at least $1-\delta$ over the draw of the problem samples, we have that*

$$\left|\mathbb{E}_{P\sim\mathcal{D}}[\ell_{EN}(\hat{\lambda}, P)] - \min_{\lambda} \mathbb{E}_{P\sim\mathcal{D}}[\ell_{EN}(\lambda, P)]\right| \leq \epsilon$$

Proof. This follows from substituting our result in Theorem 3.1 into well-known generalization guarantee for function classes with bounded pseudo-dimensions (Theorem A.1). $\square$

Discussion and applications. Computing the parameters which minimize the loss on the problem samples (aka Empirical Risk Minimization, or ERM) achieves the sample complexity bound in Theorem 3.2. Even though we only need polynomially many samples to guarantee the selection of nearly-optimal parameters, it is not clear how to implement the ERM efficiently. Note that we do not assume the set of features is the same across problem instances, so our approach can handle feature reset i.e. different problem instances can differ in not only the number of examples but also the number of features. Moreover, as a special case application, we consider the commonly used techniques of leave-one-out cross validation (LOOCV) and Monte Carlo cross validation (repeated random test-validation splits, typically independent and in a fixed proportion). Given a dataset of size m_{tr}, LOOCV would require m_{tr} regression fits which can be inefficient for large dataset size. Alternately, we can consider draws from a distribution $\mathcal{D}_{LOO}$ which generates problem instances P from a fixed dataset $(X, y) \in \mathbb{R}^{m+1\times p} \times \mathbb{R}^{m+1}$ by uniformly selecting $j \in [m+1]$ and setting $P = (X_{-j*}, y_{-j}, X_{j*}, y_j)$. Theorem 3.2 now implies that $\tilde{O}(p^2/\epsilon^2)$ iterations are enough to determine an ElasticNet parameter $\hat{\lambda}$ with loss within ϵ (w.h.p.) of the parameter λ^* obtained from running the full LOOCV. Similarly, we can define a distribution $\mathcal{D}_{MC}$ to capture the Monte Carlo cross validation procedure and determine the number of iterations sufficient to get an ϵ-approximation of the loss corresponding parameter selection with arbitrarily large number of runs of the procedure. Thus, in a very precise sense, our results answer the question of how much cross-validation is enough to effectively implement the above techniques.

Remark 1. *While our result implies polynomial sample complexity, the question of learning the provably near-optimal parameter efficiently (even in output polynomial time) is left open. For the special cases of LASSO ($\lambda_2 = 0$) and Ridge ($\lambda_1 = 0$), the piece boundaries of the piecewise polynomial dual class (loss) function may be computed efficiently (using the LARS-LASSO algorithm of [22] for LASSO, and solving linear systems and locating roots of polynomials for Ridge). This applies to online and classification settings in the following sections as well.*

3.2 Online Learning

We will now extend our results to learning the regularization coefficients given an online sequence of regression problems, such as when one needs to solve a new regression problem each day. Unlike the distributional setting above, we will not assume any problem distribution and our results will hold for an adversarial sequence of problem instances. We will need very mild assumptions on the data, namely boundedness of feature and prediction values and 'smoothness' of predictions (formally stated as Assumptions 1 and 2), while our distributional results above hold for worst-case problem datasets. Our first assumption is that all feature values and predictions are bounded, for training as well as validation examples.

Assumption 1 (Boundedness)**.** *The predicted variable and all feature values are bounded by an absolute constant R, i.e.* $\max\{||X^{(i)}||_{\infty,\infty}, ||y^{(i)}||_{\infty}, ||X^{(i)}_{val}||_{\infty,\infty}, ||y^{(i)}_{val}||_{\infty}\} \leq R$*.*

We will need the following definition of distribution smoothness to state our second assumption.

Definition 3. *A continuous probability distribution is said to be κ-bounded if the probability density function $p(x)$ satisfies $p(x) \leq \kappa$ for any x in the sample space.*

For example, the *normal* distribution $\mathcal{N}(\mu, \sigma^2)$ with mean μ and standard deviation σ is $\frac{1}{\sigma\sqrt{2\pi}}$-bounded. We assume that the predicted variable y in the training set comes from a κ-bounded (i.e. smooth) distribution, which does not require the strong tail decay of sub-Gaussian distributions [44, 13]. Moreover, the online adversary is allowed to change the distribution as long as it is κ-bounded. Note that our assumption also captures common data preprocessing steps, for example the `jitter` parameter in the popular Python library scikit-learn [37] adds a uniform noise to the y values to help model stability. The assumption is formally stated as follows:

Assumption 2 (Smooth predictions)**.** *The predicted variables $y^{(i)}$ in the training set are drawn from a joint κ-bounded distribution, i.e. for each i, the variables $y^{(i)}$ have a joint distribution with probability density bounded by κ.*

Under these assumptions, we can show that it is possible to learn the ElasticNet parameters with sublinear expected regret when the problem instances arrive online. The learning algorithm (Algorithm 1) that achieves this regret is a continuous variant of the classic Exponential Weights algorithm [14, 6]. It samples points in the domain with probability inversely propotional to the exponentiated loss. To formally state our result, we will need the following definition of *dispersed* loss functions. Informally speaking, it captures how amenable a set of non-Lipschitz functions is to online learning by measuring the worst rate of occurrence of non-Lipschitzness (or discontinuities) between any pair of points in the domain. [6, 9, 8] show that dispersion is necessary and sufficient for learning piecewise Lipschitz functions.

Definition 4. *Dispersion [8]. The sequence of random loss functions $l_1, \dots, l_T$ is β-dispersed for the Lipschitz constant L if, for all T and for all $\epsilon \geq T^{-\beta}$, we have that, in expectation, at most $\tilde{O}(\epsilon T)$ functions (the soft-O notation suppresses dependence on quantities beside ϵ, T and β, as well as logarithmic terms) are not L-Lipschitz for any pair of points at distance ϵ in the domain $\mathcal{C}$. That is, for all T and for all $\epsilon \geq T^{-\beta}$,* $\mathbb{E}\Big[\max_{\substack{\rho,\rho'\in\mathcal{C}\\ \left\|\rho-\rho'\right\|_2\leq\epsilon}} \big|\{t \in [T] \mid l_t(\rho) - l_t(\rho') > L\left\|\rho - \rho'\right\|_2\}\big|\Big] \leq \tilde{O}(\epsilon T)$*.*

Our key contribution is to show that the loss sequence is dispersed (Definition 4) under the above assumptions. This involves establishing additional structure for the problem, specifically about the location of boundary functions in the piecewise structure from Theorem 2.2. This stronger characterization coupled with results from [8] on dispersion of algebraic discontinuities completes the proof.

Theorem 3.3. *Suppose Assumptions 1 and 2 hold. Let $l_1, \dots, l_T : (0, \lambda_{\max})^2 \to \mathbb{R}_{\geq 0}$ denote an independent sequence of losses (e.g. fresh randomness is used to generate the validation set features in each round) as a function of the ElasticNet regularization parameter $\lambda = (\lambda_1, \lambda_2)$, $l_i(\lambda) = l_r(\hat{\beta}^{(X^{(i)},y^{(i)})}_{\lambda,f_{EN}}, (X^{(i)}_{val}, y^{(i)}_{val}))$. The sequence of functions is $\frac{1}{2}$-dispersed, and there is an online algorithm with $\tilde{O}(\sqrt{T})$[4] expected regret. The result also holds for loss functions adjusted by information criteria AIC and BIC.*

Proof Sketch. We start with the $(\mathcal{F}, \mathcal{G}, p3^p)$-piecewise decomposable structure for the dual class function $\mathcal{H}^*_{\text{EN}}$ from Theorem 2.2. Observe that the rational piece functions in $\mathcal{F}$ do not introduce

---

[4] The $\tilde{O}(\cdot)$ notation hides dependence on logarithmic terms, as well as on quantities other than T.

Algorithm 1 Data-driven Regularization (ζ)

```
Input: Problems $(X^{(i)}, y^{(i)})$ and regularization penalty function $f$.
Hyperparameter: step size parameter $\zeta \in (0, 1]$.
Output: Parameters $(\lambda_i)_{i\in[T]} \in C$, $C \subset \mathbb{R}^+$ (LASSO/Ridge) or $C \subset {\mathbb{R}^+}^2$ (ElasticNet).
Set $w_1(\lambda) = 1$ for all $\lambda \in C$.
for $i = 1, 2, \ldots, T$ do
    $W_i := \int_C w_i(\lambda) d\lambda$.
    Sample $\lambda$ with probability $p_t(\lambda) = \frac{w_i(\lambda)}{W_i}$, output as $\lambda_i$.
    Compute average loss function $l_i(\lambda) = \frac{1}{|y^{(i)}|} l(\hat{\beta}_{\lambda,f}, (X^{(i)}, y^{(i)}))$.
    For each $\lambda \in C,$ update weights $w_{i+1}(\lambda) = e^{\zeta(1-l_i(\lambda))} w_i(\lambda)$.
```

any new discontinuities since the denominator polynomials do not have positive roots. For each of two types of boundary functions in $\mathcal{G}$ (corresponding to leaving/entering the equicorrelation set) we show that the discontinuities between any pair of points λ, λ' lie along the roots of polynomials with non-leading coefficients bounded and smoothly distributed (bounded joint density). This allows us to use results from [8] to establish dispersion, and therefore online learnability. ☐

We remark that the above result holds for arbitrary training features and validation sets in the problem sequence that satisfy our assumptions, in particular the loss functions are only assumed to be independent but not identically distributed. In contrast, the results in the previous section needed them to be drawn from the same distribution. Also the parameters need to be selected online, and cannot be changed for already seen instances. This setting captures interesting practical settings where the set of features (including feature dimensions) and the relevant training set (including training set size) may change over the online sequence. It is not clear how usual model selection techniques like cross-validation may be adapted to these challenging settings.

4 Extension to Regularized Least Squares Classification

Regression techniques can also be used to train binary classifiers by using an appropriate threshold on top of the regression estimate. Intuitively, regression learns a linear mapping which projects the datapoints onto a one-dimensional space, i.e. a real number, after which a threshold may be applied to classify the points. The use of thresholds to make discrete classifications adds discontinuities to the empirical loss function. Thus, in general, the classification setting is more challenging as it already includes the piecewise structure in the regression loss. We provide statistical and online learning guarantees for Ridge and LASSO. For the ElasticNet we present the extensions needed to the arguments from the previous sections to obtain results in the classification setting.

More formally, we will restrict y to $\{0, 1\}^m$. The estimator $\hat{\beta}_{\lambda,f}$ is obtained as before, and the prediction on a test instance x may be obtained by taking the sign of a thresholded regression estimate, $\mathsf{sign}(\langle x, \hat{\beta}_{\lambda,f}\rangle - \tau)$, where $\mathsf{sign} : \mathbb{R} \to \{0, 1\}$ maps $x \in \mathbb{R}$ to $\mathbb{I}\{x \geq 0\}$ and $\tau \in \mathbb{R}$ is the *threshold*. The threshold τ corresponds to the intercept or bias of the learned linear classifier, here we will treat it as a tunable hyperparameter (in addition to λ_1, λ_2)[5]. The average 0-1 loss over the dataset (X, y) is given by $l_c(\hat{\beta}_{\lambda,f}, (X, y), \tau) = \frac{1}{m}\sum_{i=1}^{m} |y_i - \mathsf{sign}(\langle X_i, \hat{\beta}_{\lambda,f}\rangle - \tau)|$[6]. Proofs from this section are in Appendix D.

4.1 Distributional setting

The problem setting is the same as in Section 3.1, except that the labels y are binary and we use threshold for prediction. We bound the pseudo-dimension for classification loss on these problem instances, which as before (c.f. Theorems 3.1 and 3.2) imply that polynomially many problem samples are sufficient to generalize well over the problem distribution $\mathcal{D}$. For Ridge and LASSO we

---

[5] We can still have a problem instance specific bias in β using the standard trick of adding a unit feature to X, thus we generalize the common practice of using a fixed threshold. For example, the RidgeClassifier implementation in Python library scikit-learn 1.1.1 [37] assumes $y \in \{-1, +1\}^m$ and sets $\tau = 0$.

[6] Squared loss and 0-1 loss are identical in this setting.

upper bound the number of discontinuities of the piecewise constant classification loss by determining the values of λ where any prediction changes.

Theorem 4.1. *Let $\mathcal{H}^c_{\mathit{Ridge}}$, $\mathcal{H}^c_{\mathit{LASSO}}$ and $\mathcal{H}^c_{\mathit{EN}}$ denote the set of loss functions for classification problems with at most m examples and p features, for linear classifiers regularized using Ridge, LASSO and ElasticNet regression respectively.*

(i) $\text{PDIM}(\mathcal{H}^c_{\mathit{Ridge}}) = O(\log mp)$

(ii) $\text{PDIM}(\mathcal{H}^c_{\mathit{LASSO}}) = O(p \log m)$. *Further, in the overparameterized regime ($p \gg m$), we have that* $\text{PDIM}(\mathcal{H}^c_{\mathit{LASSO}}) = O(m \log \frac{p}{m})$.

(iii) $\text{PDIM}(\mathcal{H}^c_{\mathit{EN}}) = O(p^2 + p \log m)$.

The key difference with the bound for the regression loss in Theorem 3.1 is the additional $O(p \log m)$ term which corresponds to discontinuities induced by the thresholding in the regression based classifiers. We can establish a structure similar to Theorem 2.2 in this case (Lemma D.1).

4.2 Online setting

As in Section 3.2, we can define an online learning setting for classification. Note that the smoothness of the predicted variable is not meaningful here, since y is a binary vector. Instead we will assume that the validation examples have smooth feature values. Intuitively this means that small perturbations to the feature values does not meaningfully change the problem.

Assumption 3 (Smooth validation features)**.** *The feature values $(X^{(i)}_{\mathit{val}})_{jk}$ in the validation examples are drawn from a joint κ-bounded distribution.*

Under the assumption, we show that we can learn the regularization parameters online, for each of Ridge, LASSO and ElasticNet estimators. The proofs are straightforward extensions of the structural results developed in the previous sections, with minor technical changes to use the above validation set feature smoothness instead of Assumption 2, and are deferred to the appendix.

Theorem 4.2. *Suppose Assumptions 1 and 3 hold. Let $l_1, \ldots, l_T : (0, H]^d \times [-H, H] \to \mathbb{R}$ denote an independent sequence of losses as a function of the regularization parameter λ, $l_i(\lambda, \tau) = l_c(\hat{\beta}_{\lambda,f}, (X^{(i)}, y^{(i)}), \tau)$. If f is given by f_1 (LASSO), f_2 (Ridge), or f_{EN} (ElasticNet) then the sequence of functions is $\frac{1}{2}$-dispersed and there is an online algorithm with $\tilde{O}(\sqrt{T})$ expected regret.*

5 Conclusions and Future Work

We obtain a novel structural result for the ElasticNet loss as a function of the tuning parameters. Our characterization gives polynomial upper bounds for the sample complexity of learning the parameters from multiple instances coming from the same problem domain. For the ElasticNet we show generalization and online regret guarantees, but efficient implementation of the algorithms is an interesting question for further work. Also we show general learning-theoretic guarantees, i.e. without any significant restrictions on the data-generating distribution, in learning from multiple problems. The problems may be drawn i.i.d. from an arbitrary *problem* distribution, or even arrive in an online sequence but with some smoothness properties. It is unclear if such guarantees may be given for tuning parameters for the more standard setting of tuning a single training set. In this work we only give upper bounds on the sample complexity by bounding the pseudodimension, showing lower bounds is an interesting direction for future work.

Acknowledgments

We thank Ryan Tibshirani for helpful discussions. This material is based on work supported by the National Science Foundation under grants CCF-1910321, IIS-1705121, IIS-1838017, IIS-1901403, IIS-2046613, IIS-2112471, and SES-1919453; the Defense Advanced Research Projects Agency under cooperative agreement HR00112020003; a Simons Investigator Award; an AWS Machine Learning Research Award; an Amazon Research Award; a Bloomberg Research Grant; a Microsoft Research Faculty Fellowship; funding from Meta, Morgan Stanley, and Amazon; and a Facebook PhD Fellowship. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of any of these funding agencies.

References

[1] Hirotugu Akaike. A new look at the statistical model identification. *IEEE transactions on automatic control*, 19(6):716–723, 1974.

[2] Martin Anthony and Peter L Bartlett. *Neural network learning: Theoretical foundations*, volume 9. cambridge university press Cambridge, 1999.

[3] Maria-Florina Balcan. Book chapter Data-Driven Algorithm Design. In *Beyond Worst Case Analysis of Algorithms, T. Roughgarden (Ed)*. Cambridge University Press, 2020.

[4] Maria-Florina Balcan and Dravyansh Sharma. Data driven semi-supervised learning. *Advances in Neural Information Processing Systems*, 34, 2021.

[5] Maria-Florina Balcan, Tuomas Sandholm, and Ellen Vitercik. Sample complexity of automated mechanism design. *Advances in Neural Information Processing Systems*, 29, 2016.

[6] Maria-Florina Balcan, Travis Dick, and Ellen Vitercik. Dispersion for data-driven algorithm design, online learning, and private optimization. In *2018 IEEE 59th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 603–614. IEEE, 2018.

[7] Maria-Florina Balcan, Travis Dick, and Manuel Lang. Learning to link. In *International Conference on Learning Representations*, 2019.

[8] Maria-Florina Balcan, Travis Dick, and Wesley Pegden. Semi-bandit optimization in the dispersed setting. In *Conference on Uncertainty in Artificial Intelligence*, pages 909–918. PMLR, 2020.

[9] Maria-Florina Balcan, Travis Dick, and Dravyansh Sharma. Learning piecewise Lipschitz functions in changing environments. In *International Conference on Artificial Intelligence and Statistics*, pages 3567–3577. PMLR, 2020.

[10] Maria-Florina Balcan, Dan DeBlasio, Travis Dick, Carl Kingsford, Tuomas Sandholm, and Ellen Vitercik. How much data is sufficient to learn high-performing algorithms? generalization guarantees for data-driven algorithm design. In *Proceedings of the 53rd Annual ACM SIGACT Symposium on Theory of Computing*, pages 919–932, 2021.

[11] Maria-Florina Balcan, Siddharth Prasad, Tuomas Sandholm, and Ellen Vitercik. Sample complexity of tree search configuration: Cutting planes and beyond. *Advances in Neural Information Processing Systems*, 34, 2021.

[12] Maria-Florina F Balcan, Mikhail Khodak, Dravyansh Sharma, and Ameet Talwalkar. Learning-to-learn non-convex piecewise-lipschitz functions. *Advances in Neural Information Processing Systems*, 34:15056–15069, 2021.

[13] Emmanuel J Candès and Yaniv Plan. Near-ideal model selection by l1 minimization. *The Annals of Statistics*, 37(5A):2145–2177, 2009.

[14] Nicolo Cesa-Bianchi and Gábor Lugosi. *Prediction, learning, and games*. Cambridge university press, 2006.

[15] JM Chambers and TJ Hastie. Linear models. Chapter 4 of statistical models in S. *Wadsworth & Brooks/Cole*, 1992, 1992.

[16] Denis Chetverikov, Zhipeng Liao, and Victor Chernozhukov. On cross-validated lasso in high dimensions. *The Annals of Statistics*, 49(3):1300–1317, 2021.

[17] Michael Chichignoud, Johannes Lederer, and Martin J Wainwright. A practical scheme and fast algorithm to tune the lasso with optimality guarantees. *The Journal of Machine Learning Research*, 17(1):8162–8181, 2016.

[18] Thomas M Cover. Geometrical and statistical properties of systems of linear inequalities with applications in pattern recognition. *IEEE transactions on electronic computers*, (3):326–334, 1965.

[19] Amit Dhurandhar and Marek Petrik. Efficient and accurate methods for updating generalized linear models with multiple feature additions. *The Journal of Machine Learning Research*, 15 (1):2607–2627, 2014.

[20] Edgar Dobriban and Stefan Wager. High-dimensional asymptotics of prediction: Ridge regression and classification. *The Annals of Statistics*, 46(1):247–279, 2018.

[21] Richard M Dudley. The sizes of compact subsets of Hilbert space and continuity of Gaussian processes. *Journal of Functional Analysis*, 1(3):290–330, 1967.

[22] Bradley Efron, Trevor Hastie, Iain Johnstone, and Robert Tibshirani. Least angle regression. *The Annals of statistics*, 32(2):407–499, 2004.

[23] Jianqing Fan and Jinchi Lv. A selective overview of variable selection in high dimensional feature space. *Statistica Sinica*, 20(1):101, 2010.

[24] Manuel Fernández-Delgado, Manisha Sanjay Sirsat, Eva Cernadas, Sadi Alawadi, Senén Barro, and Manuel Febrero-Bande. An extensive experimental survey of regression methods. *Neural Networks*, 111:11–34, 2019.

[25] Jean-Jacques Fuchs. Recovery of exact sparse representations in the presence of bounded noise. *IEEE Transactions on Information Theory*, 51(10):3601–3608, 2005.

[26] Diane Galarneau Gibbons. A simulation study of some ridge estimators. *Journal of the American Statistical Association*, 76(373):131–139, 1981.

[27] Rishi Gupta and Tim Roughgarden. A pac approach to application-specific algorithm selection. *SIAM Journal on Computing*, 46(3):992–1017, 2017.

[28] Trevor Hastie, Robert Tibshirani, Jerome H Friedman, and Jerome H Friedman. *The elements of statistical learning: data mining, inference, and prediction*, volume 2. Springer, 2009.

[29] Trevor Hastie, Andrea Montanari, Saharon Rosset, and Ryan J Tibshirani. Surprises in high-dimensional ridgeless least squares interpolation. *The Annals of Statistics*, 50(2):949–986, 2022.

[30] Arthur E Hoerl and Robert W Kennard. Ridge regression: applications to nonorthogonal problems. *Technometrics*, 12(1):69–82, 1970.

[31] Lisa-Ann Kirkland, Frans Kanfer, and Sollie Millard. LASSO tuning parameter selection. In *Annual Proceedings of the South African Statistical Association Conference*, volume 2015, pages 49–56. South African Statistical Association (SASA), 2015.

[32] Johannes Lederer and Christian Müller. Don't fall for tuning parameters: tuning-free variable selection in high dimensions with the TREX. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 29, 2015.

[33] Qing Li and Nan Lin. The bayesian elastic net. *Bayesian analysis*, 5(1):151–170, 2010.

[34] Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar. *Foundations of Machine Learning*. MIT Press, 2012.

[35] Kevin P Murphy. *Machine learning: a probabilistic perspective*. MIT press, 2012.

[36] Pratik Patil, Yuting Wei, Alessandro Rinaldo, and Ryan Tibshirani. Uniform consistency of cross-validation estimators for high-dimensional ridge regression. In *International Conference on Artificial Intelligence and Statistics*, pages 3178–3186. PMLR, 2021.

[37] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[38] David Pollard. *Convergence of stochastic processes*. Springer Science & Business Media, 2012.

[39] Gideon Schwarz. Estimating the dimension of a model. *The annals of statistics*, pages 461–464, 1978.

[40] Igor Rostislavich Shafarevich. *Basic Algebraic Geometry:(by) IR Shafarevich Transl. from the Russian by KA Hirsch*. Springer-Verlag, 1977.

[41] Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58(1):267–288, 1996.

[42] Ryan J Tibshirani. The lasso problem and uniqueness. *Electronic Journal of statistics*, 7: 1456–1490, 2013.

[43] Andrey N Tikonov and Vasily Y Arsenin. Solutions of ill-posed problems. *New York: Winston*, 1977.

[44] Tong Zhang. Some sharp performance bounds for least squares regression with l1 regularization. *The Annals of Statistics*, 37(5A):2109–2144, 2009.

[45] Yu Zhang and Qiang Yang. A survey on multi-task learning. *IEEE Transactions on Knowledge and Data Engineering*, 2021.

[46] Peng Zhao and Bin Yu. Stagewise lasso. *The Journal of Machine Learning Research*, 8: 2701–2726, 2007.

[47] Hui Zou and Trevor Hastie. Regularization and variable selection via the elastic net. *Journal of the royal statistical society: series B (statistical methodology)*, 67(2):301–320, 2005.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes] We emphasize we are in a multiple instance setting instead of the standard setting.
   (c) Did you discuss any potential negative societal impacts of your work? [N/A]
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes]
   (b) Did you include complete proofs of all theoretical results? [Yes]
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [N/A]
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [N/A]
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [N/A]
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [N/A]
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [N/A]
   (b) Did you mention the license of the assets? [N/A]
   (c) Did you include any new assets either in the supplemental material or as a URL? [N/A]

   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]

    (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]

5. If you used crowdsourcing or conducted research with human subjects...
    (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
    (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
    (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]

A A classic Generalization Bound

The pseudo-dimension (also known as the *Pollard dimension*) is a generalization of the VC-dimension to real-valued functions, and may be defined as follows.

Definition 5 (Pseudo-dimension [38])**.** *Let $\mathcal{H}$ be a set of real valued functions from input space $\mathcal{X}$. We say that $C = (x_1, \dots, x_n) \in \mathcal{X}^n$ is pseudo-shattered by $\mathcal{H}$ if there exists a vector $r = (r_1, \dots, r_n) \in \mathbb{R}^n$ (called "witness") such that for all $b = (b_1, \dots, b_n) \in \{\pm 1\}^n$ there exists $h_b \in \mathcal{H}$ such that $sign(h_b(x_i) - r_i) = b_i$. Pseudo-dimension of $\mathcal{H}$, denoted by $\textsc{Pdim}(\mathcal{H})$, is the cardinality of the largest set pseudo-shattered by $\mathcal{H}$.*

The following theorem connects the sample complexity of uniform learning over a class of real-valued functions to the pseudo-dimension of the class. Let $h^* : \mathcal{X} \to \{0, 1\}$ denote the target concept. We say $\mathcal{H}$ is (ϵ, δ)-uniformly learnable[7] with sample complexity n if, for every distribution $\mathcal{D}$, given a sample S of size n, with probability $1 - \delta$, $\left|\frac{1}{n}\sum_{s \in S} |h(s) - h^*(s)| - \mathbb{E}_{s\sim\mathcal{D}}[|h(s) - h^*(s)|]\right| < \epsilon$ for every $h \in \mathcal{H}$.

Theorem A.1 ([2])**.** *Suppose $\mathcal{H}$ is a class of real-valued functions with range in $[0, H]$ and pseudo-dimension $\textsc{Pdim}(\mathcal{H})$. For every $\epsilon > 0, \delta \in (0, 1)$, the sample complexity of (ϵ, δ)-uniformly learning the class $\mathcal{H}$ is $O\left(\left(\frac{H}{\epsilon}\right)^2 \left(\textsc{Pdim}(\mathcal{H}) \ln \frac{H}{\epsilon} + \ln \frac{1}{\delta}\right)\right)$.*

B Known characterization of LASSO solutions

We will review some properties of LASSO solutions from prior work that are useful in proving our results. Let (X, y) with $X = [\boldsymbol{x}_1, \dots, \boldsymbol{x}_p] \in \mathbb{R}^{m\times p}$ and $y \in \mathbb{R}^m$ denote a (training) dataset consisting of m labeled examples with p features. As noted in Section 2, LASSO regularization may be formulated as the following optimization problem.

$$\min_{\beta\in\mathbb{R}^p} \|y - X\beta\|_2^2 + \lambda_1||\beta||_1,$$

where $\lambda_1 \in \mathbb{R}^+$ is the regularization parameter. Dealing with the case $\lambda_1 = 0$ (i.e. Ordinary Least Squares) is not difficult, but is omitted here to keep the statements of the definitions and results simple. We will use the following well-known facts about the solution of the LASSO optimization problem [25, 42]. Applying the Karush-Kuhn-Tucker (KKT) optimality conditions to the problem gives,

Lemma B.1 (KKT Optimality Conditions for LASSO)**.** *$\beta^* \in \arg\min_{\beta\in\mathbb{R}^p} \|y - X\beta\|_2^2 + \lambda_1||\beta||_1$ iff for all $j \in [p]$,*

$$\begin{aligned} \boldsymbol{x}_j^T(y - X\beta^*) &= \lambda_1 \mathsf{sign}(\beta^*), \ \textit{if } \beta_j^* \neq 0, \\ |\boldsymbol{x}_j^T(y - X\beta^*)| &\leq \lambda_1, \ \textit{otherwise.} \end{aligned}$$

Here $\boldsymbol{x}_j^T(y - X\beta^*)$ is simply the correlation of the the j-th covariate with the residual $y - X\beta^*$ (when y, X have been standardized). This motivates the definition of *equicorrelation sets* of covariates (Definition 2).

In terms of the equicorrelation set and the equicorrelation sign vector, the characterization of the LASSO solution in Lemma B.1 implies

$$X_{\mathcal{E}}^T(y - X_{\mathcal{E}}\beta_{\mathcal{E}}^*) = \lambda_1 s.$$

This implies a necessary and sufficient condition for the uniqueness of the LASSO solution, namely that $X_{\mathcal{E}}$ is full rank for all equicorrelation sets $\mathcal{E}$ [42]. Our results will hold if the dataset X satisfies this condition, but for simplicity we will use the a simpler (and possibly more natural) sufficient condition involving the *general position*.

Definition 6. *A matrix $X \in \mathbb{R}^{m\times p}$ is said to have its columns in the general position if the affine span of any $k \leq m$ points $(\sigma_i\boldsymbol{x}_{j_i})_{i\in[k],\{j_i\}_i=J\subseteq[p]}$ for arbitrary signs $\sigma_{[k]} \in \{-1, 1\}^k$ and subset J of the columns of size k, does not contain any element of $\{\boldsymbol{x}_i \mid i \notin J\}$.*

---

[7] (ϵ, δ)-uniform learnability with n samples implies (ϵ, δ)-PAC learnability with n samples.

Finally, we state the following useful characterization of the LASSO solutions in terms of the equicorrelation sets and sign vectors.

Lemma B.2 ([42], Lemma 3). *If the columns of X are in general position, then for any y and $\lambda_1 > 0$, the LASSO solution is unique and is given by*

$$\beta^*_{\mathcal{E}} = (X_{\mathcal{E}}^T X_{\mathcal{E}})^{-1}(X_{\mathcal{E}}^T y - \lambda_1 s), \beta^*_{[p]\setminus\mathcal{E}} = 0.$$

We remark that Lemma B.2 does not give a way to compute β^* for a given value of λ_1, since $\mathcal{E}$ and s depend on β^*, but still gives a property of β^* that is convenient to use. In particular, since we have at most 3^p possible choices for $(\mathcal{E}, s)$, this implies that the LASSO solution $\beta^*(\lambda_1)$ is a piecewise linear function of λ_1, with at most 3^p pieces (for $\lambda_1 > 0$). Following popular terminology, we will refer to this function as a *solution path* of LASSO for the given dataset (X, y). LARS-LASSO of [47] is an efficient algorithm for computing the *solution path* of LASSO.

Corollary B.3. *Let X be a matrix with columns in the general position. If the unique LASSO solution for the dataset (X, y) is given by the function $\beta^* : \mathbb{R}^+ \to \mathbb{R}^p$, then β^* is piecewise linear with at most 3^p pieces given by Lemma B.2.*

C Lemmas and proof details for Section 3

We start with a helper lemma that characterizes the solution of the ElasticNet in terms of equicorrelation sets and sign vectors.

Lemma C.1. *Let X be a matrix with columns in the general position, and $\lambda = (\lambda_1, \lambda_2) \in (0, \infty) \times [0, \infty)$. Then the ElasticNet solution $\hat{\beta}_{\lambda, f_{EN}} \in \arg\min_{\beta \in \mathbb{R}^p} \|y - X\beta\|_2^2 + \langle \lambda, f_{EN}(\beta) \rangle$ is unique for any dataset (X, y) and satisfies*

$$\hat{\beta}_{\lambda, f_{EN}} = (X_{\mathcal{E}}^T X_{\mathcal{E}} + \lambda_2 I_{|\mathcal{E}|})^{-1} X_{\mathcal{E}}^T y - \lambda_1 (X_{\mathcal{E}}^T X_{\mathcal{E}} + \lambda_2 I_{|\mathcal{E}|})^{-1} s$$

for some $\mathcal{E} \in [p]$ and $s \in \{-1, 1\}^p$.

Proof. We start with the well-known characterization of the ElasticNet solution as the solution of a LASSO problem on a transformed dataset, obtained using simple algebra [47]. Given any dataset (X, y), the ElasticNet coefficients $\hat{\beta}_{\lambda, f_{EN}}$ are given by $\hat{\beta}_{\lambda, f_{EN}} = \frac{1}{\sqrt{1+\lambda_2}} \hat{\beta}^*_\lambda$[8] where $\hat{\beta}^*_\lambda$ is the solution for a LASSO problem on a modified dataset (X^*, y^*)

$$\hat{\beta}^*_\lambda = \arg\min_\beta \|y^* - X^*\beta\|_2^2 + \lambda_1^* f_1(\beta)$$

with $X^* = \frac{1}{\sqrt{1+\lambda_2}} \begin{pmatrix} X \\ \sqrt{\lambda_2} I_p \end{pmatrix}$, $y^* = \begin{pmatrix} y \\ 0 \end{pmatrix}$, and $\lambda_1^* = \frac{\lambda_1}{\sqrt{1+\lambda_2}}$.

If the columns of X are in general position (Definition 6), then the same is true of X^*. For $\mathcal{E} \subseteq [p]$, note that $X^{*T}_{\mathcal{E}} X^*_{\mathcal{E}} = \frac{1}{1+\lambda_2}(X_{\mathcal{E}}^T X_{\mathcal{E}} + \lambda_2 I_{|\mathcal{E}|})$ and $X^{*T}_{\mathcal{E}} y^* = \frac{1}{\sqrt{1+\lambda_2}} X_{\mathcal{E}}^T y$. By Lemma B.2, if $\mathcal{E}$ denotes the equicorrelation set of covariates and $s \in \{-1, 1\}^{|\mathcal{E}|}$ the equicorrelation sign vector for the LASSO problem, then the ElasticNet solution is given by

$$\hat{\beta}_{\lambda, f_{EN}} = c_1 - c_2 \lambda_1$$

where $c_1 = \frac{1}{\sqrt{1+\lambda_2}} (X^{*T}_{\mathcal{E}} X^*_{\mathcal{E}})^{-1} X^{*T}_{\mathcal{E}} y^* = (X_{\mathcal{E}}^T X_{\mathcal{E}} + \lambda_2 I_{|\mathcal{E}|})^{-1} X_{\mathcal{E}}^T y$, and
$c_2 = \frac{1}{1+\lambda_2} (X^{*T}_{\mathcal{E}} X^*_{\mathcal{E}})^{-1} s = (X_{\mathcal{E}}^T X_{\mathcal{E}} + \lambda_2 I_{|\mathcal{E}|})^{-1} s$. □

The following lemma helps determine the dependence of ElasticNet solutions on λ_2.

Lemma C.2. *Let A be an $r \times s$ matrix. Consider the matrix $B(\lambda) = (A^T A + \lambda I_s)^{-1}$ for $\lambda > 0$.*

---

[8] This corresponds to the "naive" ElasticNet solution in the terminology of [47]. They also define an ElasticNet 'estimate' given by $\sqrt{1 + \lambda_2} \hat{\beta}^*_\lambda$ with nice properties, to which our analysis is easily adapted.

1. *Each entry of $B(\lambda)$ is a rational polynomial $P_{ij}(\lambda)/Q(\lambda)$ for $i,j \in [s]$ with each P_{ij} of degree at most $s-1$, and Q of degree s.*
2. *Further, for $i=j$, P_{ij} has degree $s-1$ and leading coefficient 1, and for $i \neq j$ P_{ij} has degree at most $s-2$. Also, $Q(\lambda)$ has leading coefficient 1.*

Proof. Let $G = A^TA$ be the Gramian matrix. G is symmetric and therefore diagonalizable, and the diagonalization gives the eigendecomposition $G = E\Lambda E^{-1}$. Thus we have

$$(A^TA + \lambda I_s)^{-1} = (E\Lambda E^{-1} + \lambda EE^{-1})^{-1} = E(\Lambda + \lambda I_s)^{-1}E^{-1}$$

But Λ is the diagonal matrix $\mathrm{diag}(\Lambda_{11}, \dots, \Lambda_{ss})$, and therefore $(\Lambda + \lambda I_s)^{-1} = \mathrm{diag}((\Lambda_{11} + \lambda)^{-1}, \dots, (\Lambda_{ss} + \lambda)^{-1})$. This implies the desired characterization, with $Q(\lambda) = \Pi_{i\in[s]}(\Lambda_{ii} + \lambda)$ and

$$P_{ij}(\lambda) = Q(\lambda)\sum_{k=1}^{s} \frac{E_{ik}(E^{-1})_{kj}}{\Lambda_{kk} + \lambda} = \sum_{k=1}^{s} \left(E_{ik}(E^{-1})_{kj}\Pi_{i\in[s]\setminus k}(\Lambda_{ii} + \lambda)\right),$$

with coefficient of λ^{s-1} in $P_{ij}(\lambda)$ equal to $\sum_{k=1}^{s} E_{ik}(E^{-1})_{kj} = \mathbb{I}\{i = j\}$. □

C.1 Tuning the ElasticNet – Distributional setting

We first present some useful terminology from algebraic geometry.

Definition 7 (Semialgebraic sets, Algebraic curves.)**.** *A semialgebraic subset of $\mathbb{R}^n$ is a finite union of sets of the form $\{x \in \mathbb{R}^n \mid p_i(x) \geq 0$ for each $i \in [m]\}$, where $p_1, \dots, p_m$ are polynomials. An algebraic curve is the zero set of a polynomial in two dimensions.*

The result of Lemma 2.2 motivates the following results for bounding the complexity of dual piece functions and dual boundary functions, which can be used to bound the pseudodimension of $\mathcal{H}_{\text{EN}}$ (Theorem 3.1) using the following remarkable result from [10].

Theorem C.3 ([10])**.** *If the dual function class $\mathcal{H}^*$ is $(\mathcal{F}, \mathcal{G}, k)$-piecewise decomposable, then the pseudo-dimension of $\mathcal{H}$ may be bounded as*

$$\text{Pdim}(\mathcal{H}) = O((\text{Pdim}(\mathcal{F}^*) + d_{\mathcal{G}^*})\log(\text{Pdim}(\mathcal{F}^*) + d_{\mathcal{G}^*}) + d_{\mathcal{G}^*}\log k),$$

where $d_{\mathcal{G}^*}$ denotes the VC dimension of dual class boundary function $\mathcal{G}^*$.

We will first prove a useful lemma that bounds the number of pieces into which a finite set of algebraic curves with bounded degrees may partition $\mathbb{R}^2$.

Lemma C.4. *Let $\mathcal{H}$ be a collection of k functions $h_i : \mathbb{R}^2 \to \mathbb{R}$ that map $(x,y) \mapsto q_i(x,y)$ where q_i is a bivariate polynomial of degree at most d, for $i \in [k]$, then $\mathbb{R}^2 \setminus \{(x,y) \mid q_i(x,y) = 0$ for some $i \in [k]\}$ may be partitioned into at most $(kd+1)\left(d^2\binom{k}{2} + 2kd(d-1) + 1\right) = O(d^3k^3)$ disjoint sets such that the sign pattern $(\mathbb{I}\{q_i(x,y) > 0\})_{i\in[k]}$ is fixed over any set in the partition.*

Proof. Assume WLOG that the curves are in the general position. Simple applications of Bezout's theorem (which states that, in general, two algebraic curves of degrees d_1 and d_2 intersect in at most d_1d_2 points [40]) imply that there are at most $d^2\binom{k}{2}$ points where any pair of curves from the set $\{q_i(x,y)\}_{i\in[k]}$ may intersect, and at most $2kd(d-1)$ points of extrema (i.e. points $p_0 = (x_0, y_0)$ on the curve f such that there is an open neighborhood N around p_0 in which $x_0 \in \arg\min_{(x,y)\in N\cap f} x$, or $x_0 \in \arg\max_{(x,y)\in N\cap f} x$, or $y_0 \in \arg\min_{(x,y)\in N\cap f} y$, or $y_0 \in \arg\max_{(x,y)\in N\cap f} y$) for the k algebraic curves. Let $\mathcal{P}$ denote the set of these $\leq d^2\binom{k}{2} + 2kd(d-1)$ points.

Now a horizontal line $y = c$ will have the exact same set of intersections as line $y = c'$ with all the curves in $\mathcal{H}$, and in the same order (including multiplicities), if none of the points in $\mathcal{P}$ lie between these lines. There are thus at most $|\mathcal{P}| + 1$ distinct sequences of the k curves that may correspond to the intersection sequence of any horizontal line. Moreover, any such horizontal line may intersect any curve in the set at most d times (since a polynomial in degree d has at most d zeros), or at most kd intersections with all the curves. Summing up over the distinct intersection sequences, we have at most $(kd+1)(|\mathcal{P}|+1)$ distinct sign patterns induced by the set of curves. □

We will now use Lemma C.4 to bound the pseudo-dimension of the relevant function classes (Lemma 2.2).

Lemma C.5. *Let $\mathcal{F}^* = \{f^*_{q_1,q_2} : \mathbb{R}^2 \to \mathbb{R}\}$ be a function class consisting of rational polynomial functions $f^*_{q_1,q_2} : (\lambda_1, \lambda_2) \mapsto \frac{q_1(\lambda_1,\lambda_2)}{q_2(\lambda_1,\lambda_2)}$, where q_1, q_2 have degrees at most d. Then $\textsc{Pdim}(\mathcal{F}^*) = O(\log d)$.*

Proof. Suppose that $\textsc{Pdim}(\mathcal{F}^*) = N$. Then there exist functions $f^*_1, \ldots, f^*_N \in \mathcal{F}^*$ and witnesses $(r_1, \ldots, r_N) \in \mathbb{R}^N$ such that for every subset $T \subseteq [N]$, there exists a parameter setting $\lambda_T = (\lambda_1, \lambda_2) \in \mathbb{R}^2$ such that $f^*_i(\lambda_T) \geq r_i$ if and only if $i \in T$. In other words, we have a set of 2^N parameters (indexed by T) that induce all possible labelings of the binary vector $(\mathbb{I}\{f^*_i(\lambda_T) \geq r_i\})_{i\in[N]}$.

But $f^*_i(\lambda) \geq r_i$ are semi-algebraic sets bounded by N algebraic curves of degree at most d. By Lemma C.4, there are at most $O(d^3N^3)$ different sign-patterns induced by N algebraic curves over all possible values of $\lambda \in \mathbb{R}^2$. In particular, the distinct sign patterns over $\lambda \in \{\lambda_T\}_{T\subseteq[N]}$ is also $O(d^3N^3)$. Thus, we conclude $2^N = O(d^3N^3)$, or $N = O(\log d)$. □

Lemma C.6. *Let $R[x_1, x_2, \ldots, x_d]_D$ denote the set of all real polynomials in d variables of degree at most D in x_1, and degree at most 1 in $x_2, \ldots, x_d$. Further, let $P_{d,D} = \{\{x \in R^d : p(x) \geq 0\} \mid p \in R[x_1, x_2, \ldots, x_d]_D\}$. The VC-dimension of the set system $(\mathbb{R}^d, P_{d,D})$ is $O(dD)$.*

Proof. We will employ a standard *linearization* argument [18] that reduces the problem to bounding the VC dimension of halfspaces in higher dimensions. Let M be the set of all possible non-constant monomials of degree at most D in x_1, and at most one in $x_2, \ldots, x_d$. For example, when $d = 3$ and $D = 2$, we have $M = \{x_1, x_2, x_3, x_1x_2, x_1x_3, x_1^2, x_1^2x_2, x_1^2x_3\}$. Note that $|M| = (D + 1)d - 1$. Indeed for each x_1^i for $i = 0, \ldots, D$ we obtain a monomial by multiplying with each of $\{1, x_2, \ldots, x_d\}$. Excluding the constant monomial gives the result. The linearization we use is a map $\phi : \mathbb{R}^d \to \mathbb{R}^{|M|}$ which indexes the coordinates by monomials in M. For example when $d = 3$ and $D = 2$, $\phi(x_1, x_2, x_3) = (x_1, x_2, x_3, x_1x_2, x_1x_3, x_1^2, x_1^2x_2, x_1^2x_3)$.

Now, if $S \in \mathbb{R}^d$ is shattered by $P_{d,D}$, then $\phi(S)$ is shattered by half-spaces in $\mathbb{R}^{|M|}$. Indeed, suppose $p = p_0 + \langle \mathbf{p}, \phi(x_1, \ldots, x_d)\rangle \in P_{d,D}$ (for $\mathbf{p} \in \mathbb{R}^{|M|}$) is a polynomial that is positive over some $T \subseteq S$ and negative over $S \setminus T$. Define halfspace $h_p \in \mathbb{R}^{|M|}$ as $\{y \in \mathbb{R}^{|M|} \mid p_0 + \langle \mathbf{p}, y\rangle \geq 0\}$. Clearly $h_p \cap \phi(S) = \phi(T)$, and in general S is shattered by halfspaces in $\mathbb{R}^{|M|}$. Using the well-known result for the VC-dimension of halfspaces we have that the VC-dimension of $P_{d,D}$ over $\mathbb{R}^d$ is $(D+1)d$. □

Theorem 3.1 (restated). $\textsc{Pdim}(\mathcal{H}_{EN}) = O(p^2)$. *Further,* $\textsc{Pdim}(\mathcal{H}^{AIC}_{EN}) = O(p^2)$ *and* $\textsc{Pdim}(\mathcal{H}^{BIC}_{EN}) = O(p^2)$.

Proof. By Lemma 2.2, the dual class $\mathcal{H}^*_{\text{EN}}$ of $\mathcal{H}_{\text{EN}}$ is $(\mathcal{F}, \mathcal{G}, p3^p)$-piecewise decomposable, with $\mathcal{F} = \{f_{q_1,q_2} : \mathcal{L} \to \mathbb{R}\}$ consisting of rational polynomial functions $f_{q_1,q_2} : l_\lambda \mapsto \frac{q_1(\lambda_1,\lambda_2)}{q_2(\lambda_2)}$, where q_1, q_2 have degrees at most $2p$, and $\mathcal{G} = \{g_r : \mathcal{L} \to \{0, 1\}\}$ consisting of semi-algebraic sets bounded by algebraic curves $g_r : u_\lambda \mapsto \mathbb{I}\{r(\lambda_1, \lambda_2) < 0\}$, where r is a polynomial of degree 1 in λ_1 and at most p in λ_2.

Now by Lemma C.5, we have $\textsc{Pdim}(\mathcal{F}^*) = O(\log p)$, and by Lemma C.6 the VC dimension of the dual boundary class is $d_{\mathcal{G}^*} = O(p)$. A straightforward application of Theorem C.3 yields

$$\textsc{Pdim}(\mathcal{H}) = O(p \log p + p \log(p3^p)) = O(p^2).$$

The dual classes $(\mathcal{H}^{\text{AIC}}_{\text{EN}})^*$ and $(\mathcal{H}^{\text{BIC}}_{\text{EN}})^*$ also follow the same piecewise decomposable structure given by Lemma 2.2. This is because in each piece the equicorrelation set $\mathcal{E}$, and therefore $||\beta||_0 = |\mathcal{E}|$ (by Lemma B.2) is fixed. Thus we can keep the same boundary functions, and the function value in each piece only changes by a constant (in λ) and is therefore also a rational function with the same degrees. The above argument then implies an identical upper bound on the pseudo-dimensions. □

C.2 Tuning the ElasticNet – Online learning

At a high level, the plan is to show dispersion (Definition 4) using the general recipe developed in [8]. The recipe may be summarized at a high level as follows.

- S1. Bound the probability density of the random set of discontinuities of the loss functions. Intuitively this corresponds to computing the average number of loss functions that may be discontinuous along a path connecting any two points within distance ϵ in the domain.
- S2. Use a VC-dimension based uniform convergence argument to transform this into a bound on the dispersion of the loss functions.

Formally, we have the following theorems from [8], which show how to use this technique when the discontinuities are roots of a random polynomial with bounded coefficients. The theorems implement steps S1 and S2 of the above recipe respectively.

Theorem C.7 ([8])**.** *Consider a random degree d polynomial ϕ with leading coefficient 1 and subsequent coefficients which are real of absolute value at most R, whose joint density is at most κ. There is an absolute constant K_0 depending only on d and R such that every interval I of length $\leq \epsilon$ satisfies Pr(ϕ has a root in I) $\leq \kappa\epsilon/K_0$.*

Theorem C.8 ([8])**.** *Let $l_1, \ldots, l_T : \mathbb{R} \to \mathbb{R}$ be independent piecewise L-Lipschitz functions, each having at most K discontinuities. Let $D(T, \epsilon, \rho) = |\{1 \leq t \leq T \mid l_t$ is not L-Lipschitz on $[\rho - \epsilon, \rho + \epsilon]\}|$ be the number of functions that are not L-Lipschitz on the ball $[\rho - \epsilon, \rho + \epsilon]$. Then we have $E[\max_{\rho\in\mathbb{R}} D(T, \epsilon, \rho)] \leq \max_{\rho\in\mathbb{R}} E[D(T, \epsilon, \rho)] + O(\sqrt{T \log(TK)})$.*

Lemma C.9. *Let A be an $r \times s$ matrix with R-bounded max-norm, i.e. $||A||_{\infty,\infty} = \max_{i,j} |A_{ij}| \leq R$. Then each entry of the matrix $(A^T A + \lambda I_s)^{-1}$ is a rational polynomial $P_{ij}(\lambda)/Q(\lambda)$ for $i, j \in [s]$ with each P_{ij} of degree at most $s - 1$, Q of degree s, and all the coefficients have absolute value at most $r^s(Rs)^{2s}$.*

Proof. Let $G = A^T A$ be the Gram matrix. $|G_{ij}| = |\sum_k A_{ki}A_{kj}| \leq \sum_k |A_{ki}A_{kj}| \leq rR^2$, by the triangle inequality and using $\max_{i,j} |A_{ij}| \leq R$. The determinant $\text{DET}(A^T A + \lambda I_s)$ is a sum of $s! \leq s^s$ signed terms, each a product of s elements of the form G_{ij} or $G_{ii} + \lambda$. Thus, in each of the $s!$ terms, the coefficient of λ^k is a sum of at most $\binom{s}{s-k} \leq s^k \leq s^s$ expressions of the form $\Pi_{(i,j)\in S} G_{ij}$ with $|S| \leq s - k$. Now $|\Pi_{(i,j)\in S} G_{ij}| \leq (rR^2)^{|S|} \leq (rR^2)^s$, and by triangle inequality the coefficient of λ^k is upper bounded by $(rR^2)^s \cdot s^k \cdot s^s$ for any k. This establishes the bound on the coefficients of $Q(\lambda)$. A similar argument implies the upper bound for each $P_{ij}(\lambda)$. □

We will also need the following result, which is a simple extension of Lemma 24 from [4].

Lemma C.10. *Suppose X and Y are real-valued random variables taking values in $[m, m + M]$ for some $m, M \in \mathbb{R}^+$ and suppose that their joint distribution is κ-bounded. Let c be an absolute constant. Then,*

(i) $Z = X + Y$ is drawn from a $K_1\kappa$-bounded distribution, where $K_1 \leq M$.

(ii) $Z = XY$ is drawn from a $K_2\kappa$-bounded distribution, where $K_2 \leq M/m$.

(iii) $Z = X - Y$ is drawn from a $K_1\kappa$-bounded distribution, where $K_1 \leq M$.

(iv) $Z = X + c$ has a κ-bounded distribution, and $Z = cX$ has a $\frac{\kappa}{|c|}$-bounded distribution.

Proof. Let $f_{X,Y}(x, y)$ denote the joint density of X, Y. (i) and (ii) are immediate from Lemma 24 from [4], (iii) is a simple extension. Indeed, the cumulative density function for Z is given by

$$\begin{aligned} F_Z(z) = \Pr(Z \leq z) &= \Pr(X - Y \leq z) = \Pr(X \leq z + Y) \\ &= \int_m^{m+M} \int_m^{z+y} f_{X,Y}(x, y)dxdy. \end{aligned}$$

The density function for Z can be obtained using Leibniz's rule as

$$\begin{aligned}
f_Z(z) = \frac{d}{dz}F_Z(z) &= \frac{d}{dz}\int_m^{m+M}\int_m^{z+y} f_{X,Y}(x,y)dxdy \\
&= \int_m^{m+M}\left(\frac{d}{dz}\int_m^{y} f_{X,Y}(x,y)dx + \frac{d}{dz}\int_0^{z} f_{X,Y}(t+y,y)dt\right)dy \\
&= \int_m^{m+M} f_{X,Y}(z+y,y)dy \\
&\leq \int_m^{m+M} \kappa dy \\
&= M\kappa.
\end{aligned}$$

Finally, (iv) follows from simple change of variable manipulations (e.g. Theorem 22 of [8]). $\square$

Theorem 3.3 (restated). *Assume that the predicted variable and all feature values are bounded by an absolute constant R, i.e.* $\max\{||X^{(i)}||_{\infty,\infty}, ||y^{(i)}||_\infty, ||X^{(i)}_{val}||_{\infty,\infty}, ||y^{(i)}_{val}||_\infty\} \leq R$. *Suppose the predicted variables* $y^{(i)}$ *in the training set are drawn from a joint* κ*-bounded distribution. Let* $l_1, \ldots, l_T : (0, \lambda_{\max})^2 \to \mathbb{R}_{\geq 0}$ *denote an independent sequence of losses (e.g. fresh randomness is used to generate the validation set features in each round) as a function of the ElasticNet regularization parameter* $\lambda = (\lambda_1, \lambda_2)$, $l_i(\lambda) = l_r(\hat{\beta}^{(X^{(i)},y^{(i)})}_{\lambda,f_{EN}}, (X^{(i)}_{val}, y^{(i)}_{val}))$. *The sequence of functions is* $\frac{1}{2}$*-dispersed, and there is an online algorithm with* $\tilde{O}(\sqrt{T})$ *expected regret. The result also holds for loss functions adjusted by information criteria AIC and BIC.*

Proof. We start with the piecewise-decomposable characterization of the dual class function in Theorem 2.2. On any fixed problem instance $P \in \Pi_{m,n}$, as the parameter λ is varied in the loss function $\ell_{\text{EN}}(\cdot, P)$ of ElasticNet trained with regularization parameter $\lambda = (\lambda_1, \lambda_2)$, we have the following piecewise structure. There are $k = p3^p$ boundary functions $g_1, \ldots, g_k$ for which the transition boundaries are algebraic curves $r_i(\lambda_1, \lambda_2)$, where r_i is a polynomial with degree 1 in λ_1 and at most p in λ_2. Also the piece function $f_{\mathbf{b}}$ for each sign pattern $\mathbf{b} \in \{0,1\}^k$ is a rational polynomial function $\frac{q_1^b(\lambda_1,\lambda_2)}{q_2^b(\lambda_2)}$, where $q_1^{\mathbf{b}}, q_2^{\mathbf{b}}$ have degrees at most $2p$, and corresponds to a fixed signed equicorrelation set $(\mathcal{E}, s)$. To show online learnability, we will examine this piecewise structure more closely – in particular analyse how the structure varies when the predicted variable is drawn from a smooth distribution.

In order to show dispersion for the loss functions $\{l_i(\lambda)\}$, we will use the recipe of [8] and bound the worst rate of discontinuities between any pair of points $\lambda = (\lambda_1, \lambda_2)$ and $\lambda' = (\lambda'_1, \lambda'_2)$ with $||\lambda - \lambda'||_2 \leq \epsilon$ along the axis-aligned path $\lambda \to (\lambda'_1, \lambda_2) \to \lambda'$. First observe that the only possible points at which $l_i(\lambda)$ may be discontinuous are

(a) (λ_1, λ_2) such that $r_i(\lambda_1, \lambda_2) = 0$ corresponding to some boundary function g_i.

(b) (λ_1, λ_2) such that $q_2^{\mathbf{b}}(\lambda_2) = 0$ corresponding to some piece function $f_{\mathbf{b}}$.

Fortunately the discontinuity of type (b) does not occur for $\lambda_2 > 0$. From the ElasticNet characterization in Lemma C.1, and using Lemma C.2, we know that $q_2(\lambda_2) = \Pi_{j\in[|\mathcal{E}|]}(\Lambda_j + \lambda_2)$, where $(\Lambda_j)_{j\in[|\mathcal{E}|]}$ are non-negative eigenvalues of the positive semi-definite matrix ${X^{(i)}_{\mathcal{E}}}^T X^{(i)}_{\mathcal{E}}$. It follows that $q_2^{\mathbf{b}}$ does not have positive zeros (for any sign vector $\mathbf{b}$).

Therefore it suffices to locate boundaries of type (a). To this end, we have two subtypes corresponding to a variable entering or leaving the equicorrelation set.

Addition of $j \notin \mathcal{E}$. As observed in the proof of Theorem 2.2, a variate $j \notin \mathcal{E}$ can enter the equicorrelation set $\mathcal{E}$ only for (λ_1, λ_2) satisfying $\lambda_1 = K_0\left(\boldsymbol{x}_j^T(X_{\mathcal{E}}({X_{\mathcal{E}}}^T X_{\mathcal{E}} + \lambda_2 I_{|\mathcal{E}|})^{-1}{X_{\mathcal{E}}}^T - \boldsymbol{x}_j^T\right) y$ (K_0 does not depend on λ_1, λ_2 or y). For fixed λ_2, the distribution of λ_1 at which the discontinuity occurs for insertion of j is $K_1\kappa$-bounded (by Lemma C.10) for some constant K_1 that only depends on

R, m, p and $\lambda_{\max}$. This implies an upper bound of $K_1\kappa\epsilon$ on the expected number of discontinuities corresponding to j along the segment $\lambda \to (\lambda_1', \lambda_2)$ for any $j, \mathcal{E}$.

For constant λ_1, we can use Lemma C.2 and a standard change of variable argument (e.g. Theorem 22 of [8]) to conclude that the discontinuities lie at the roots of a random polynomial in λ_2 of degree $|\mathcal{E}|$, leading coefficient 1, and bounded random coefficients with $K_2\kappa$-bounded density for some constant K_2 (that only depends on R, m, p and $\lambda_{\max}$). By Theorem C.7, the expected number of discontinuities along the segment $(\lambda_1', \lambda_2) \to \lambda'$ is upper bounded by $K_2K_p\kappa\epsilon$ (K_p only depends on p). This implies that the expected number of Lipschitz violations between λ and λ' along the axis aligned path is $\tilde{O}(\kappa\epsilon)$ and completes the first step of the recipe in this case ($\tilde{O}$ notation suppresses terms in R, m, p and $\lambda_{\max}$ as constants).

Removal of $j' \in \mathcal{E}$. The second case, when a variate $j' \in \mathcal{E}$ leaves the equicorrelation set $\mathcal{E}$ for (λ_1, λ_2) satisfying $\lambda_1(({X_\mathcal{E}}^T X_\mathcal{E} + \lambda_2 I_{|\mathcal{E}|})^{-1}s)_{j'} = (({X_\mathcal{E}}^T X_\mathcal{E} + \lambda_2 I_{|\mathcal{E}|})^{-1}{X_\mathcal{E}}^T y)_{j'}$, also yields the same bound using the above arguments. Putting together, and noting that we have at most $p3^p$ distinct curves each with $\tilde{O}(\kappa\epsilon)$ expected number of intersections with the axis aligned path $\lambda \to \lambda'$, the total expected number of discontinuities is also $\tilde{O}(\kappa\epsilon)$. This completes the first step (S1) of the above recipe.

We use Theorem 9 of [8] to complete the second step of the recipe, which employs a VC-dimension argument for K' algebraic curves of bounded degrees (here degree is at most $p+1$) to conclude that the expected worst number of discontinuities along any axis-aligned path between any pair of points $\leq \epsilon$ apart is at most $\tilde{O}(\epsilon T) + O(\sqrt{T \log K'T})$. $K' \leq p3^p$ as shown above. This implies that the sequence of loss functions is $\frac{1}{2}$-dispersed, and further there is an algorithm (Algorithm 4 of [6]) that achieves $\tilde{O}(\sqrt{T})$ expected regret.

Finally note that loss functions with AIC and BIC have the same dual class piecewise structure, and therefore the above analysis applies. The only difference is that the value of the piece functions $f_\mathbf{b}$ are changed by a constant (in λ), $K_{m,p} \leq p \log m$. The piece boundaries are the same, and are therefore $\frac{1}{2}$-dispersed as above. The range of the loss functions is now $[0, K_{m,p} + 1]$, so the same algorithm (Algorithm 4 of [6]) again achieves $\tilde{O}(\sqrt{T})$ expected regret. □

D Lemmas and proof details for Section 4

We will first extend the structure for the ElasticNet regression loss functions shown in Lemma 2.2 to the classification setting. The main new challenge is that there are additional discontinuities due to thresholding the loss function needed for binary classification, which intuitively makes the loss more jumpy and discontinuous as a function of the regularization parameters.

Lemma D.1. *Let $\mathcal{L}$ be a set of functions $\{l_\lambda : \Pi_{m,p} \to \mathbb{R}_{\geq 0} \mid \lambda \in \mathbb{R}^+ \times \mathbb{R}_{\geq 0}\}$ that map a regression problem instance $P \in \Pi_{m,p}$ to the validation classification loss $\ell^c_{EN}(\lambda, P)$ of ElasticNet trained with regularization parameter $\lambda = (\lambda_1, \lambda_2)$. The dual class $\mathcal{L}^*$ is $(\mathcal{F}, \mathcal{G}, (m+p)3^p)$-piecewise decomposable, with $\mathcal{F} = \{f_{q_1,q_2} : \mathcal{L} \to \mathbb{R}\}$ consisting of rational polynomial functions $f_{q_1,q_2} : l_\lambda \mapsto \frac{q_1(\lambda_1,\lambda_2)}{q_2(\lambda_2)}$, where q_1, q_2 have degrees at most $2p$, and $\mathcal{G} = \{g_r : \mathcal{L} \to \{0,1\}\}$ consisting of semi-algebraic sets bounded by algebraic curves $g_r : l_\lambda \mapsto \mathbb{I}\{r(\lambda_1, \lambda_2) < 0\}$, where r is a polynomial of degree 1 in λ_1 and at most p in λ_2.*

Proof. By Lemma C.1, the EN coefficients $\hat{\beta}_{EN}$ are fixed given the signed equicorrelation set $\mathcal{E}, s$. As in Lemma 2.2, we have $\leq p3^p$ boundaries corresponding to a change in the equicorrelation set, but the value of the loss also changes when a prediction vector coefficient $\mu_j = (X_{\text{val}})_j \hat{\beta}_{EN}$ cross the threshold $\frac{1}{2}$. This is given by $(c_1 - c_2\lambda_1)_j = \frac{1}{2}$ where $c_1 = (X_{\text{val}})_\mathcal{E}(X_\mathcal{E}^T X_\mathcal{E} + \lambda_2 I_{|\mathcal{E}|})^{-1} X_\mathcal{E}^T y$, and $c_2 = (X_{\text{val}})_\mathcal{E}(X_\mathcal{E}^T X_\mathcal{E} + \lambda_2 I_{|\mathcal{E}|})^{-1} s$. Therefore, $\hat{\mu}_j = 0$ corresponds to the 0-set of $(c_1 - c_2\lambda_1)_j - \frac{1}{2}$. By an application of Lemma C.2, this is an algebraic curve with degree at most $|\mathcal{E}|$ in λ_2 and degree 1 in λ_1. There are at most $m3^p$ such curves, corresponding to all possibilities of $j, \mathcal{E}, s$. Together with Lemma 2.2, we have the claimed stucture for the ElasticNet classification (dual class) loss function. □

Above piecewise decomposable structure is helpful in bounding the pseudodimension for the ElasticNet based classifier. For the special cases of Ridge and LASSO, we obtain the pseudodimension bounds from first principles.

Theorem 4.1 (restated). *Let $\mathcal{H}^c_{Ridge}$, $\mathcal{H}^c_{LASSO}$ and $\mathcal{H}^c_{EN}$ denote the set of loss functions for classification problems with at most m examples and p features, with Ridge, LASSO and ElasticNet regularization respectively.*

(i) $\textsc{Pdim}(\mathcal{H}^c_{Ridge}) = O(\log mp)$

(ii) $\textsc{Pdim}(\mathcal{H}^c_{LASSO}) = O(p \log m)$. *Further, in the overparameterized regime ($p \gg m$), we have* $\textsc{Pdim}(\mathcal{H}^c_{LASSO}) = O(m \log \frac{p}{m})$.

(iii) $\textsc{Pdim}(\mathcal{H}^c_{EN}) = O(p^2 + p^2 \log m)$.

Proof.

(i) For ridge regression, the estimator $\hat{\beta}_{\lambda,f_2}$ on the dataset $(X^{(i)}, y^{(i)})$ is given by the following closed form

$$\hat{\beta}_{\lambda,f_2} = ({X^{(i)}}^T X^{(i)} + \lambda I_{p_i})^{-1} {X^{(i)}}^T y^{(i)},$$

where I_{p_i} is the $p_i \times p_i$ identity matrix. By Lemma C.2 each coefficient $(\hat{\beta}_{\lambda,f_2})_k$ of the estimator $\hat{\beta}_{\lambda,f_2}$ is a rational polynomial in λ of the form $P_k(\lambda)/Q(\lambda)$, where P_k, Q are polynomials of degrees at most $p_i - 1$ and p_i respectively. Thus the prediction on any example $(X^{(i)}_{\text{val}})_j$ in the validation set $(X^{(i)}_{\text{val}}, y^{(i)}_{\text{val}})$ of any problem instance $P^{(i)}$ can change at most $p_i \le p$ times as λ is varied. Recall there are $m'_i \le m$ examples in any validation set. This implies we have at most mnp distinct values of the loss function over the n problem instances. The pseudo-dimension n therefore satisfies $2^n \le mnp$, or $n = O(\log mp)$.

(ii) Prior work [22] shows that the optimal vector $\hat{\beta} \in \mathbb{R}^p$ evolves piecewise linearly with λ, i.e. $\exists \lambda^{(0)} = 0 < \lambda^{(1)} < \cdots < \lambda^{(q)} = \infty$ and $\gamma_0, \gamma_1, \ldots, \gamma_{q-1} \in \mathbb{R}^p$ such that $\hat{\beta}_{\lambda,f_1} = \hat{\beta}_{\lambda^{(k)},f_1} + (\lambda - \lambda^{(k)})\gamma_k$ for $\lambda^{(k)} \le \lambda \le \lambda^{(k+1)}$. Each piece corresponds to the addition or removal of at least one of p coordinates to the *active set* of covariates with maximum correlation. For any data point x_j, $1 \le j \le m$, and any piece $[\lambda^{(k)}, \lambda^{(k+1)})$ we have that $x_j\hat{\beta}$ is monotonic since $\hat{\beta}$ varies along a fixed vector γ_k, and therefore can have at most one value of λ where the predicted label $\hat{y}$ changes. This gives an upper bound of mq on the total number of discontinuities on any single problem instance $(X^{(i)}, y^{(i)}, X^{(i)}_{\text{val}}, y^{(i)}_{\text{val}})$, where q is the number of pieces in the solution path. By Lemma 6 of [42], we have the number pieces in the solution path $q \le 3^p$. Also for the overparameterized regime $p \gg m$, we have the property that there are at most $m - 1$ variables in the active set for the entire sequence of solution paths (Section 7, [22]). Thus, we have that $q \le m\binom{p}{m-1} \le (\frac{ep}{m})^m$ in this case.

Over n problem instances, the pseudo-dimension satisfies $2^n \le mqn$, or $n = O(\log mq)$. Substituting the above inequalities for q completes the proof.

(iii) The proof of Theorem follows the same arguments as Theorem 3.1, using Lemma D.1 instead of Lemma 2.2 (and is omitted for brevity).

□

We will now restate and prove Theorem 4.2. This implies that under smoothness assumptions on the data distribution we can learn the data-dependent optimal regularization parameter in the online setting.

Theorem 4.2 (restated). *Suppose Assumptions 1 and 3 hold. Let $l_1, \ldots, l_T : \mathbb{R} \to \mathbb{R}$ denote an independent sequence of losses as a function of the Tikhonov (ridge) regularization parameter λ, $l_i(\lambda) = l_c(\hat{\beta}_{\lambda,f}, (X^{(i)}, y^{(i)}))$. The sequence of functions is $\frac{1}{2}$-dispersed, and there is an online algorithm with $\tilde{O}(\sqrt{T})$ expected regret, if f is given by*

(i) $f = f_1$,

(ii) $f = f_2$, *or*

(iii) $f = f_{EN}$ *(The result also holds for loss functions adjusted by information criteria AIC and BIC).*

Proof.

(i) On any dataset (X, y), the predictions are given by the coefficients of the prediction vector $\hat{\mu} = X(X^TX + \lambda I_p)^{-1}X^Ty$. Note that by Lemma C.2 $(X^TX + \lambda I_p)^{-1}$, and therefore $X_{\text{val}}(X^TX + \lambda I_p)^{-1}X^Ty$, has each element of the form $P_j(\lambda)/Q(\lambda)$ with degree of each P_j at most $p-1$ and degree of Q at most p. Further by also using Lemma C.9, $\hat{\mu}_j = \frac{1}{2}$ is polynomial equation in λ with degree p with bounded coefficients that have $K\kappa$-bounded density for some constant K (that depends on bounds on M, M', n and p, but not on κ) and leading coefficient 1. This completes step S1 of the recipe from [8], and Theorem C.7 gives a bound on the expected number of discontinuities in any interval I (over λ).

To complete step S2, note that the loss function l_i on any instance has at most pm discontinuities, since each coefficient $\hat{\mu}_j$ of the prediction vector can change sign at most p times as λ is varied. This implies the VC-dimension argument (Theorem C.8) applies and the expected maximum number of discontinuities in any interval of width ϵ is $O(\epsilon T) + O(\sqrt{T \log(mpT)})$, which is $\tilde{O}(\epsilon T)$ for $\epsilon \geq 1/\sqrt{T}$. Thus, using the recipe from [8], we have shown that the sequence of loss functions is $\frac{1}{2}$-dispersed. This further implies that Algorithm 1, which implements the Continuous Exp-Weights algorithm of [6] for setting the regularization parameter, achieves $\tilde{O}(\sqrt{T})$ expected regret ([6], Theorem 1).

(ii) Since the data-distribution is in particular assumed to be continuous, by Lemma 4 of [42] we know that the LASSO solutions are unique with probability 1. Moreover if $\mathcal{E} \subseteq [p]$ denotes the equicorrelation set of variables (i.e. covariates with the maximum absolute value of correlation), and $s \in \{-1, 1\}^{|\mathcal{E}|}$ the sign vector (i.e. the sign of the correlations of the covariates in $\mathcal{E}$), then the LASSO prediction vector $\hat{\mu} = X\hat{\beta}$ is a linear function of regularization parameter λ given by

$$\hat{\mu} = c_1 - c_2\lambda$$

where $c_1 = X_{\mathcal{E}}(X_{\mathcal{E}}^TX_{\mathcal{E}})^{-1}X_{\mathcal{E}}^Ty$ and $c_2 = X_{\mathcal{E}}(X_{\mathcal{E}}^TX_{\mathcal{E}})^{-1}s$. Thus for any fixed $\mathcal{E}, s$ (corresponding to a unique piece in the solution path for LARS-LASSO), we have at most one discontinuity corresponding to $\hat{\mu}_j = \frac{1}{2}$, the location of this discontinuity has a $K\kappa$-bounded distribution (for constant K independent of κ) by an application of Lemma C.10. Thus, the probability that this discontinuity is located within any given interval I of width ϵ is at most $K\kappa\epsilon$. A union bound over $j \in [m]$, and over 3^p choices of $\mathcal{E}, s$ (for example, Lemma 6 in [42]) gives the probability of a discontinuity in I is at most $m3^pK\kappa\epsilon$. This completes step S1 of the recipe above.

Now each loss function l_i has at most $m3^p$ discontinuities, and therefore by the VC-dimension argument of Theorem C.8, the expected maximum number of discontinuities in any interval of width ϵ is $O(\epsilon T) + O(\sqrt{T(p + \log(mT))})$, which is $\tilde{O}(\epsilon T)$ for $\epsilon \geq 1/\sqrt{T}$. This completes step S2 of the recipe from [8], and we have shown that the sequence of loss functions is $\frac{1}{2}$-dispersed. As in Theorem 4.2, this implies that Algorithm 1 achieves $\tilde{O}(\sqrt{T})$ expected regret ([6], Theorem 1).

While we use the worst case bound on the number of solution paths here, algorithmically we can use LARS-LASSO on the given dataset, which is much faster in practice and the running time scales linearly with the actual number of solution paths q (typically $q \ll 3^p$).

(iii) Note that both ridge and LASSO have a single parameter, and the loss function is piecewise constant i.e. discontinuities are points in the parameter space. But the parameter space is two-dimensional for the ElasticNet, and the discontinuities can lie along curves in the 2D space.

The proof uses the piecewise decomposable structure proved in Lemma D.1, and establishes dispersion using joint smoothness of $X_{\text{val}}^{(i)}$ instead of $y^{(i)}$ (similar to the proof of Theorem 4.2 (i)). It is otherwise identical to the proof of Theorem 3.3, and is omitted for brevity.

$\square$