Oracle inequalities for tuning hyperparameters via validation sets with applications to penalized regression

Jean Feng*
Department of Biostatistics, University of Washington
and
Noah Simon
Department of Biostatistics, University of Washington

November 20, 2016

Abstract

In the regression setting, model-estimation procedures construct models given training data and some hyperparameters. The optimal hyperparameters that minimize the model error are unknown so they are often estimated using validation set approaches. Up to now, there is an open question regarding how the model error grows with the number of hyperparameters. To answer this question, we establish finite-sample oracle inequalities for training/validation split framework and cross-validation. If the model-estimation procedures are smoothly parameterized by the hyperparameters, the error incurred from tuning hyperparameters shrinks at roughly a parametric rate. Hence for semi- and non-parametric model-estimation procedures, this additional error is negligible and for parametric model-estimation procedures, adding a hyperparameter is roughly equivalent to adding a parameter to the model itself. Our main application is penalized regression problems with multiple penalty parameters. We establish the fitted models are Lipschitz in the penalty parameters, so a similar relationship holds between model error and the number of penalty parameters. This result encourages development of regularization methods with many penalty parameters.

Keywords: Hyperparameter selection, Cross-validation, Regularization, Regression

*Jean Feng was supported by NIH grants ???. Noah Simon was supported by NIH grant DP5OD019820. The content is solely the responsibility of the authors and does not necessarily represent the official views of the National Institutes of Health.

1 Introduction

Per the usual regression framework, we observe response $y \in \mathbb{R}$ and predictors $\boldsymbol{x} \in \mathbb{R}^p$. Suppose y is generated by a true model g^* plus random error ϵ with expectation zero, as follows

$$y = g^*(\boldsymbol{x}) + \epsilon \tag{1}$$

Our goal is to estimate g^*.

Many model-estimation procedures can be formulated as selecting a model from some function class $\mathcal{G}$ given training data T and J-dimensional hyperparameter vector $\boldsymbol{\lambda}$. For example, in penalized regression problems, the fitted model can be expressed as the minimizer of the penalized training criterion

$$\hat{g}(\boldsymbol{\lambda}|T) = \arg\min_{g \in \mathcal{G}} \sum_{(x_i, y_i) \in T} \left(y_i - g(x_i)\right)^2 + \sum_{j=1}^{J} \lambda_j P_j(g) \tag{2}$$

where P_j are penalty functions and λ_j are penalty parameters. As suggested by the notation in (2), the penalty parameters are the hyperparameters in this model-estimation procedure.

Given a set of possible hyperparameters Λ, there is some oracle hyperparameter $\tilde{\boldsymbol{\lambda}}$ in Λ that minimizes the difference between the fitted model and the true model. Usually $\tilde{\boldsymbol{\lambda}}$ is unknown so it is estimated using training/validation split or cross-validation. The basic idea is to fit models on a random partition of the observed data and evaluate their error on the remaining data. The final hyperparameters $\hat{\boldsymbol{\lambda}}$ are the minimizer of the error on this validation set. For a more complete review of cross-validation, refer to Arlot et al. (2010).

The performance of validation set procedures is typically characterized by an oracle inequality that bounds the generalization error of the expected model selected from the validation set procedure. For Λ that are finite, oracle inequalities have been established for a training/validation framework Györfi et al. (2006) and a general cross-validation framework (Van Der Laan & Dudoit 2003, van der Laan et al. 2004). To handle continuous Λ, one can use entropy-based approaches (Lecué et al. 2012).

The goal of this paper is to characterize the performance of models when the hyperparameters must be tuned by some validation set procedure. We are particularly interested in an open question raised in Bengio (2000) (and possibly by others): what is the "amount

of overfitting... when too many hyperparameters are optimized"? To do this, we establish finite-sample oracle inequalities of the form

$$\left\| g^* - \hat{g}\left(\hat{\boldsymbol{\lambda}}, T\right) \right\|^2 \leq (1+a) \underbrace{\inf_{\lambda \in \Lambda} \left\| g^* - \hat{g}\left(\boldsymbol{\lambda}, T\right) \right\|^2}_{\text{Oracle error}} + \text{error} \tag{3}$$

for some norm $\|\cdot\|$ and constant $a \geq 0$. Under the assumption that the model-estimation procedure is smoothly parameterized by the hyperparameters, we find that the error from tuning hyperparameters shrinks at roughly a parametric rate. For parametric model-estimation procedures, the additional error of adding a hyperparameter is roughly equivalent to adding a parameter to the model itself. For semi- and non-parametric model-estimation procedures, this error is generally dominated by the oracle error and the number of hyperparameters can actually grow without affecting the asymptotic convergence rate.

The main application in this paper is penalized regression models of the form (2). We show that the fitted model is indeed smoothly parameterized by the penalty parameters so our oracle inequalities apply. Again, we find that additional penalty parameters only add a near-parametric error term, which has a negligible effect on the model error in semi- and non-parametric settings. This result suggests that the recent interest in combining penalty functions (e.g. elastic net and sparse group lasso (Zou & Hastie 2003, Simon et al. 2013)) may have artificially restricted themselves to two-component combinations. Adding more penalties may lead to better models.

We did not find any theoretical results addressing the relationship betwen the number of hyperparameters and model error. Most oracle inequalities only consider one-dimensional hyperparameters (Van Der Laan & Dudoit 2003, van der Laan et al. 2004, Györfi et al. 2006). Within the context of penalized regression problems, the oracle inequalities are also only for the case of a single penalty parameter (Golub et al. 1979, Chetverikov & Liao 2016, Chatterjee & Jafarov 2015). Only Lecué et al. (2012) has a result that is relevant to answering our question of interest. We will use his framework to address the question of cross-validation for multiple hyperparameters. A potential reason for this dearth of literature is that, historically, tuning multiple hyperparameters was computationally difficult. However, there have been many proposals recently for overcoming this computational hurdle (Bengio 2000, Foo et al. 2008, Snoek et al. 2012).

Section 2 presents oracle inequalities for model-estimation procedures that are smoothly parameterized by the hyperparameters. These results answer our question regarding how the number of hyperparameters affects the model error. Section 3 applies these results to penalized regression models. Section 4 provides a simulation study to support our theoretical results. Section 5 discusses our findings and potential future work. Section 6 presents oracle inequalities for general model-estimation procedures and proofs for all the results.

2 Main Result

In this section, we establish oracle inequalities for the model error from tuning hyperparameters by training/validation split and cross-validation. We first introduce some notation and formalize the model-estimation procedure.

Let $D^{(n)}$ denote a dataset with n samples from the model (1) where ϵ are independent random variables with expectation zero. Suppose $\epsilon_1, ..., \epsilon_n$ are uniformly sub-Gaussian with parameter $b > 0$; i.e. $\max_{i=1,...,n} \mathbb{E}e^{t\epsilon_i} \leq e^{b^2t^2/2}$ for all $t \in \mathbb{R}$.

The model-estimation procedure accepts some hyperparameter of dimension J and training data of size m to output a fitted model from some model class $\mathcal{G}$. This can be formulated as an operator $\hat{g}^{(m)}(\cdot|D^{(m)})$ that maps a hyperparameter vector $\boldsymbol{\lambda}$ from some set $\Lambda \subseteq \mathbb{R}^J$ to a function in $\mathcal{G}$.

In this section, we focus on model-estimation procedures that are Lipschitz.

Definition 1. *Let $\mathcal{F}$ be a function class. Let $\Lambda \subseteq \mathbb{R}^J$. The operator $\hat{f} : \Lambda \mapsto \mathcal{F}$ is C-Lipschitz in $\boldsymbol{\lambda}$ with respect to norm $\|\cdot\|$ over Λ if*

$$\left\| \hat{f}(\boldsymbol{\lambda}) - \hat{f}(\boldsymbol{\lambda}') \right\| \leq C\|\boldsymbol{\lambda} - \boldsymbol{\lambda}'\|_2 \quad \forall \boldsymbol{\lambda}, \boldsymbol{\lambda}' \in \Lambda \tag{4}$$

We hypothesize that many model-estimation procedures satisfy this Lipschitz assumption since it ensures that the procedure is well-behaved. Section 3 shows that penalized regression models indeed satisfy this assumption. The following results show that the additional error from tuning multiple hyperparameters for such procedures shrinks at roughly a parametric rate. Hence for semi- or non-parametric model-estimation procedures, the error from tuning multiple hyperparameters is very small.

2.1 Training/Validation Split

In the training/validation split framework, the dataset $D^{(n)}$ is randomly partitioned into a training set $T = (X_T, Y_T)$ and validation set $V = (X_V, Y_V)$ with n_T and n_V observations, respectively. The selected hyperparameter $\hat{\boldsymbol{\lambda}}$ is the minimizer of the validation loss

$$\hat{\boldsymbol{\lambda}} = \arg\min_{\boldsymbol{\lambda} \in \Lambda} \frac{1}{2} \left\| y - \hat{g}^{(n_T)}(\boldsymbol{\lambda} | D_T^{(n_T)}) \right\|_V^2 \tag{5}$$

where $\|h\|_V = \frac{1}{n_V} \sum_{i \in V} h^2(x_i)$ for any function h.

We now present a finite-sample oracle inequality for the training/validation split framework assuming the model-estimation procedure is Lipschitz. The oracle inequality is sharp, i.e. $a = 0$ in (3), unlike most other work (Györfi et al. 2006, Lecué et al. 2012, Van Der Laan & Dudoit 2003). The reason for this difference is that the model error is taken with respect to the norm $\|\cdot\|_V$. Note that the result below is a special case of Theorem 3, which applies to general model-estimation procedures.

Theorem 1. *Let $\Lambda = [\lambda_{\min}, \lambda_{\max}]^J$ where $0 < \lambda_{\min} < \lambda_{\max}$. Suppose independent random variables $\epsilon_1, ...\epsilon_n$ have expectation zero and are uniformly sub-Gaussian with parameter b. Suppose there are constants $\sigma > 0$ and $C_\Lambda \geq 5/(n\lambda_{\max})$ such that for any dataset $D^{(n_T)}$ with $\|\boldsymbol{\epsilon}\|_{D^{(n_T)}} \leq \sigma$, $\hat{g}^{(n_T)}(\boldsymbol{\lambda}|D^{(n_T)})$ is C_Λ-Lipschitz with respect to $\|\cdot\|_V$ over Λ.*

Let

$$\tilde{\boldsymbol{\lambda}} = \arg\min_{\lambda \in \Lambda} \left\| g^* - \hat{g}^{(n_T)}(\boldsymbol{\lambda}|T) \right\|_V^2 \tag{6}$$

Then there is a constant $c > 0$ only depending on b such that for all δ satisfying

$$\delta^2 \geq c \left(\frac{J \log(nC_\Lambda \lambda_{\max})}{n_V} \vee \sqrt{\frac{J \log(nC_\Lambda \lambda_{\max})}{n_V} \left\| g^* - \hat{g}^{(n_T)}(\tilde{\boldsymbol{\lambda}}|T) \right\|_V^2} \right) \tag{7}$$

we have

$$\begin{aligned} Pr\left(\left\| g^* - \hat{g}^{(n_T)}(\hat{\boldsymbol{\lambda}}|T) \right\|_V^2 - \left\| g^* - \hat{g}^{(n_T)}(\tilde{\boldsymbol{\lambda}}|T) \right\|_V^2 \geq \delta^2 \middle| T, X_V \right) &\leq c \exp\left(-\frac{n_V \delta^4}{c^2 \left\| g^* - \hat{g}^{(n_T)}(\tilde{\boldsymbol{\lambda}}|T) \right\|_V^2} \right) \\ &\quad + c \exp\left(-\frac{n_V \delta^2}{c^2} \right) \end{aligned}$$

Theorem 1 states that as the number of validation samples grows, the difference between the selected model error and the oracle model error shrinks at the rate of δ^2 with high probability. δ^2 can be thought of as the error incurred during the hyperparameter selection process. As seen in (7), it is the maximum of two terms: a near-parametric term and a geometric mean of the near-parametric term and the oracle error. To see this more clearly, we express Theorem 1 using asymptotic notation.

Corollary 1. *Under the assumptions given in Theorem 1, we have*

$$\left\|g^* - \hat{g}^{(n_T)}(\hat{\boldsymbol{\lambda}}|T)\right\|_V^2 \leq \left\|g^* - \hat{g}^{(n_T)}(\tilde{\boldsymbol{\lambda}}|T)\right\|_V^2 \tag{8}$$

$$+O_p\left(\frac{J\log(nC_\Lambda\lambda_{\max})}{n_V}\right) \tag{9}$$

$$+O_p\left(\sqrt{\frac{J\log(nC_\Lambda\lambda_{\max})}{n_V}\left\|g^* - \hat{g}^{(n_T)}(\tilde{\boldsymbol{\lambda}}|T)\right\|_V^2}\right) \tag{10}$$

Hence the error of the selected model is bounded by the error of the oracle model, the near-parameteric term (9), and the geometric mean of the two values (10). We refer to (9) as near-parametric because the error term in parametric regression models are usually $O_p(J/n)$, where J is the parameter dimension and n is the number of training samples. Analogously, (9) is roughly $O_p(J/n_V)$ modulo a $\log n$ term in the numerator. ($\log n$ grows at a sub-polynomial rate, so it changes the convergence rate by a very small amount.)

In the semi- and non-parametric regression setting, the oracle error usually shrinks at a rate of n^ω where $\omega \in (-1, 0)$, which means that for large n, the oracle error will tend to dominate both the error terms. Therefore increasing the number of hyperparameters for such problems only results in small increases in the model error/degree of overfitting. In fact, if the oracle error rate is $O_p(n^\omega)$, the number of hyperparameters J can grow at the rate

$$\frac{n_V n^\omega}{\log(nC_\Lambda\lambda_{\max})} \tag{11}$$

without affecting the asymptotic convergence rate. Note that for parametric regression problems, this will not be the case. Adding hyperparameters incurs a similar cost as adding parameters to the model itself.

The appearance of the parametric term (9) suggests that we can interpret the problem of tuning hyperparameters as a parametric regression problem over a J-dimensional parameter

space where the validation data is the training data. However, this interpretation is an oversimplification due to model misspecification. Recall that we perform training/validation split over the model class

$$\mathcal{G}(T) = \left\{\hat{g}^{(n_T)}(\boldsymbol{\lambda}|T) : \boldsymbol{\lambda} \in \Lambda\right\} \tag{12}$$

$\mathcal{G}(T)$ is unlikely to contain the true model g^* and is biased by

$$\min_{\lambda \in \Lambda} \left\| g^* - \hat{g}^{(n_T)}(\boldsymbol{\lambda}|T) \right\|_V^2 \tag{13}$$

This bias term contributes to the convergence rate in the geometric mean (10).

2.2 Cross-Validation

In this section, we give an oracle inequality for K-fold cross-validation. Previously, the oracle inequality was with respect to the L2 norm over the validation covariates. We are now interested in the generalization error

$$\|g - g^*\|^2 = \int |g(x) - g^*(x)|^2 \, dx \tag{14}$$

We will follow the framework in Lecué et al. (2012).

The problem setup for K-fold cross-validation is as follows. Let dataset $D^{(n)}$ be randomly partitioned into K sets, which we assume to have equal size for simplicity. Partition k will be denoted $D_k^{(n_V)}$ and its complement will be denoted $D_{-k}^{(n_T)} = D \setminus D_k^{(n_V)}$. We perform our model-selection procedure over $D_{-k}^{(n_T)}$ for $k = 1, ..., K$ and select the hyperparameter that minimizes the average validation loss

$$\hat{\boldsymbol{\lambda}} \;=\; \arg\min_{\boldsymbol{\lambda} \in \Lambda} \frac{1}{2K} \sum_{k=1}^{K} \left\| y - \hat{g}^{(n_T)}(\boldsymbol{\lambda}|D_{-k}^{(n_T)}) \right\|_{D_k^{(n_V)}}^2 \tag{15}$$

In traditional cross-validation, the final model is retrained on all the data with $\hat{\boldsymbol{\lambda}}$. However, bounding the generalization error of the retrained model requires additional regularity assumptions (Lecué et al. 2012). We consider the "averaged version of cross-validation" instead

$$\bar{g}\left(\hat{\boldsymbol{\lambda}} \middle| D^{(n)}\right) = \frac{1}{K} \sum_{k=1}^{K} \hat{g}^{(n_T)}\left(\hat{\boldsymbol{\lambda}} \middle| D_{-k}^{(n_T)}\right) \tag{16}$$

The following theorem bounds the generalization error of (16). It is an application of Theorem 3.5 in Lecué et al. (2012), which is reproduced in Theorem 4 for convenience.

Theorem 2. *Suppose the dataset can be partitioned into K equal-sized sets, where $K \geq 2$. Let $\Lambda = [\lambda_{\min}, \lambda_{\max}]^J$. Suppose independent random variables $\epsilon_1, ...\epsilon_n$ have expectation zero and are uniformly sub-Gaussian with parameter b. Suppose there is a $G \geq 2$ such that $\sup_{g \in \mathcal{G}} \|g\|_\infty \leq G$.*

Suppose there is a constant $C_\Lambda > 0$ such that for any dataset $D^{(n_T)}$ with $\|\boldsymbol{\epsilon}\|_{D^{(n_T)}} \leq \sigma$, $\hat{g}(\boldsymbol{\lambda}|D^{(n_T)})$ is C_Λ-Lipschitz with respect to $\|\cdot\|_\infty$ over Λ.

Then there is are absolute constants $c_1, c_2 > 0$ such that for all $a > 0$,

$$E_{D^{(n)}} \left\| \bar{g}(\hat{\boldsymbol{\lambda}}|D^{(n)}) - g^* \right\|^2 \leq (1+a) \min_{\boldsymbol{\lambda} \in \Lambda} E_{D^{(n_T)}} \left\| \hat{g}^{(n_T)}(\boldsymbol{\lambda}|D^{(n_T)}) - g^* \right\|^2 \tag{17}$$

$$+ c_1 \frac{(1+a)^2}{a} \frac{J}{n_V} \left(G \log(G C_\Lambda \lambda_{max}) \log n + c_2 \right) \tag{18}$$

As we can see, Theorems 1 and 2 are quite similar. The upper bounds in both theorems depend on the oracle error and a near-parametric term. For parametric model-estimation procedures, tuning hyperparameters incurs a similar cost as the model-estimation procedure itself. In semi- and non-parametric regression settings, tuning hyperparameters is a relatively "cheap" and incurs an error that is negligible asymptotically.

There are also some notable differences between Theorems 1 and 2. The Lipschitz condition in Theorem 2 is required to hold with respect to $\|\cdot\|_\infty$, which is stricter than that in Theorem 1. Also, we no longer have a sharp oracle inequality since the oracle error is scaled by $1 + a$ where $a > 0$. These differences occur when we are interested in characterizing the generalization error instead.

Finally, since the theorems in this section are finite-sample results, one could try to minimize the upper bound by increasing the number of hyperparameters or changing the ratio between the training and validation set sizes. Unfortunately, optimizing the upper bound in these oracle inequalities require knowing characteristics about the error variables. Instead one may need to rely on heuristic approaches (or even another layer of cross-validation).

3 Penalized regression models

Theorems 1 and 2 require the fitted functions $\hat{g}(\cdot|\boldsymbol{\lambda})$ to be Lipschitz when the norm of the error terms is bounded. As an example, we show that additive models are C-Lipschitz in the

penalty parameters. We will start from the simple example of parametric models fitted with smooth penalty functions, then consider nonsmooth penalty functions, and finally generalize the results to nonparametric additive models.

Recall that in many cases, we will want the range of Λ to grow at some polynomial rate in n. The convergence rates given in Lemmas **??** and **??** hold if the Lipschitz constant is polynomial in n. The following results indeed show that the fitted models are Cn^{κ}-Lipschitz for some $\kappa > 0$.

Finally, we note that additive models are not the only problems where the estimators are smoothly parameterized by the penalty functions. In the Appendix, we show that regression problems where we fit a single model $g(\cdot|\boldsymbol{\theta})$ with multiple, individually-scaled penalties $P_j(\boldsymbol{\theta})$ satisfies (**??**).

3.1 Parametric additive models

Here we consider parametric additive models of the form

$$g(\cdot|\boldsymbol{\theta}^{(1)}, ..., \boldsymbol{\theta}^{(J)}) = \sum_{j=1}^{J} g_j(\cdot|\boldsymbol{\theta}^{(j)}) \tag{19}$$

where $\boldsymbol{\theta}^{(j)} \in \mathbb{R}^{p_j}$ and $p = \sum_{j=1}^{J} p_j$. For simplicity, let $\boldsymbol{\theta} = \left(\boldsymbol{\theta}^{(1)}, ..., \boldsymbol{\theta}^{(J)}\right)^{\top}$. Let $\boldsymbol{\theta}^*$ be the true model parameter. The number of dimensions p_j is allowed to grow with n, as commonly done in sieve estimation. We will suppose that the functions g_j are Lipschitz in $\boldsymbol{\theta}$ with respect to $\|\cdot\|_{\infty}$.

We consider training criteria of the form

$$L_T\left(y, \boldsymbol{\theta}|\boldsymbol{\lambda}\right) \coloneqq \frac{1}{2}\left\|y - g(X|\boldsymbol{\theta})\right\|_T^2 + \sum_{j=1}^{J} \lambda_j P_j(\boldsymbol{\theta}^{(j)}) \tag{20}$$

We show that the fitted models are indeed Lipschitz in the penalty parameters with respect to $\|\cdot\|_{\infty}$, which satisfies the condition in both Theorems 1 and 2.

3.1.1 Parametric regression with smooth penalties

We first suppose the penalty functions are all smooth. In the following section, we will generalize the results to include certain nonsmooth penalty functions. The following lemma states that the fitted models are Lipschitz in the penalty parameter vector.

Lemma 1. *Let*

$$\hat{\boldsymbol{\theta}}^{(j)}(\boldsymbol{\lambda}) = \underset{\boldsymbol{\theta}\in\mathbb{R}^p}{\arg\min}\, L_T(y, \boldsymbol{\theta}|\boldsymbol{\lambda}) \tag{21}$$

where L_T is defined in (20)

Suppose that $g_j(\cdot|\boldsymbol{\theta}^{(j)})$ are L-Lipschitz in $\boldsymbol{\theta}^{(j)}$ with respect to $\|\cdot\|_\infty$ for all $j = 1, .., J$.

Suppose $P_j(\boldsymbol{\theta})$ and $g_j(\cdot|\boldsymbol{\theta})$ are twice-differentiable and convex with respect to $\boldsymbol{\theta}^{(j)}$ for all $j = 1, .., J$. Suppose $L_T(y, \boldsymbol{\theta}|\boldsymbol{\lambda})$ is twice-differentiable and convex with respect to $\boldsymbol{\theta}$.

Suppose there is a $m > 0$ such that the Hessian of the penalized training criterion at the minimizer satisfies

$$\nabla^2_\theta L_T(y, \boldsymbol{\theta}|\boldsymbol{\lambda})\big|_{\theta=\hat{\theta}(\boldsymbol{\lambda})} \succeq mI \tag{22}$$

Let $\lambda_{\max} > \lambda_{\min} > 0$. Let

$$C_{\theta^*,\Lambda} = \frac{1}{2}\|y - g(\cdot|\boldsymbol{\theta}^*)\|^2_T + \lambda_{max}\sum_{j=1}^{J} P_j(\boldsymbol{\theta}^{(j),*}) \tag{23}$$

For any $\boldsymbol{\lambda}^{(1)}, \boldsymbol{\lambda}^{(2)} \in \Lambda \coloneqq [\lambda_{\min}, \lambda_{\max}]^J$, we have

$$\left\| g\left(\cdot|\hat{\boldsymbol{\theta}}(\boldsymbol{\lambda^{(1)}})\right) - g\left(\cdot|\hat{\boldsymbol{\theta}}(\boldsymbol{\lambda^{(2)}})\right)\right\|_\infty \leq \frac{L^2J^2\sqrt{2C_{\theta^*,\Lambda}}}{m\lambda_{min}}\left\|\boldsymbol{\lambda}^{(1)} - \boldsymbol{\lambda}^{(2)}\right\| \tag{24}$$

Notice that the result requires that the training criterion is strongly convex at its minimizer. If this is not true, one can add augment the penalty function $P_j(\boldsymbol{\theta}^{(j)})$ with a ridge penalty $\|\boldsymbol{\theta}^{(j)}\|_2^2$ so that the training criterion becomes

$$\frac{1}{2}\|y - g(X|\boldsymbol{\theta})\|^2_T + \sum_{j=1}^{J}\lambda_j\left(P_j(\boldsymbol{\theta}^{(j)}) + \frac{w}{2}\|\boldsymbol{\theta}^{(j)}\|_2^2\right) \tag{25}$$

The proofs for all the examples follow a similar recipe. We determine the gradient of the fitted model with respect to the penalty parameter vector by implicitly differentiating the KKT conditions. We can then bound the norm of the gradient to get the Lipschitz constant.

For illustration, we present the proof for Lemma 1 in the case where there is only one penalty parameter. The case with multiple penalty parameters is given in Section 6.

Proof of Lemma 1. fix me if we still want this here

By the KKT conditions, we have

$$\langle y - g(\boldsymbol{\theta}), \nabla_{\boldsymbol{\theta}} g(\boldsymbol{\theta})\rangle_T + \lambda\nabla_{\boldsymbol{\theta}} P(\boldsymbol{\theta}) + \lambda w\boldsymbol{\theta} = \mathbf{0}$$

Its implicit derivative...

finish proof □

3.1.2 Parametric regression with non-smooth penalties

If the regression problem contains non-smooth penalty functions, similar results do not necessarily hold. Nonetheless we find that for many popular non-smooth penalty functions, such as the lasso (CITE) and group lasso (CITE), the fitted functions are still smoothly parameterized by $\boldsymbol{\lambda}$ almost everywhere. To characterize such problems, we use the approach in Feng & Simon (TBD- CITE?). We begin with the following definitions:

Definition 2. *The differentiable space of a real-valued function f at $\boldsymbol{\theta}$ is*

$$\Omega^f(\boldsymbol{\theta}) = \left\{ \boldsymbol{\beta} \middle| \lim_{\epsilon \to 0} \frac{f(\boldsymbol{\theta} + \epsilon\boldsymbol{\beta}) - f(\boldsymbol{\theta})}{\epsilon} \text{ exists} \right\} \tag{26}$$

Definition 3. *S is a local optimality space for a convex function $f(\cdot, \boldsymbol{\lambda})$ over the W if for every $\boldsymbol{\lambda} \in W$,*

$$\arg\min_{\boldsymbol{\theta} \in \mathbb{R}^p} f(\boldsymbol{\theta}, \boldsymbol{\lambda}) = \arg\min_{\boldsymbol{\theta} \in S} f(\boldsymbol{\theta}, \boldsymbol{\lambda}) \tag{27}$$

We can now characterize a set $\Lambda_{smooth} \subseteq \Lambda$ over which the fitted functions are well-behaved. Λ_{smooth} must satisfy the following conditions:

Condition 1. *For every $\boldsymbol{\lambda} \in \Lambda_{smooth}$, there exists a ball $B(\boldsymbol{\lambda})$ with nonzero radius centered at $\boldsymbol{\lambda}$ such that*

- *For all $\boldsymbol{\lambda}' \in B(\boldsymbol{\lambda})$, the training criterion $L_T(\cdot, \cdot)$ is twice differentiable along directions in $\Omega^{L_T(\cdot,\cdot)}\left(\hat{\boldsymbol{\theta}}_\lambda\right)$.*
- *The differentiable space $\Omega^{L_T(\cdot,\boldsymbol{\lambda})}(\boldsymbol{\theta})$ at $\hat{\boldsymbol{\theta}}(\boldsymbol{\lambda})$ is a local optimality space for $L_T(\cdot, \boldsymbol{\lambda})$ over $B(\boldsymbol{\lambda})$.*

Condition 2. *For every $\boldsymbol{\lambda^{(1)}}, \boldsymbol{\lambda^{(2)}} \in \Lambda_{smooth}$, let the line segment between the two points be denoted*

$$\mathcal{L}(\boldsymbol{\lambda^{(1)}}, \boldsymbol{\lambda^{(2)}}) = \left\{ \alpha\boldsymbol{\lambda^{(1)}} + (1-\alpha)\boldsymbol{\lambda^{(2)}} : \alpha \in [0,1] \right\}$$

Suppose the intersection $\mathcal{L}(\boldsymbol{\lambda^{(1)}}, \boldsymbol{\lambda^{(2)}}) \cap \Lambda^C_{smooth}$ is countable.

In lasso and group lasso problems, it is hypothesized that almost every penalty parameter satisfies these properties. (CITE?) Equipped with these conditions, we can characterize the smoothness of the fitted functions when the penalties are non-smooth. In fact the Lipschitz constant is exactly the same as that in Lemma 1.

Lemma 2. *Define* $\hat{\boldsymbol{\theta}}^{(j)}(\boldsymbol{\lambda})$ *as in* (21).

Suppose $g_j(\cdot|\boldsymbol{\theta}^{(j)})$ *are L-Lipschitz in* $\boldsymbol{\theta}^{(j)}$ *with respect to* $\|\cdot\|_\infty$ *for all* $j = 1, .., J$.

Suppose $P_j(\boldsymbol{\theta}^{(j)})$ *and* $g_j(\cdot|\boldsymbol{\theta}^{(j)})$ *are convex with respect to* $\boldsymbol{\theta}^{(j)}$ *for all* $j = 1, .., J$ *and* $L_T(y, \boldsymbol{\theta}|\boldsymbol{\lambda})$ *is convex with respect to* $\boldsymbol{\theta}$.

Let U_λ *be an orthonormal matrix with columns forming a basis for the differentiable space of* $L_T(\cdot|\boldsymbol{\lambda})$ *at* $\hat{\boldsymbol{\theta}}(\boldsymbol{\lambda})$. *Suppose there is a* $m > 0$ *such that the Hessian of the penalized training criterion with respect to the differentiable space at the minimizer satisfies*

$$ {}_{U_\lambda}\nabla^2_\theta L_T(y, \boldsymbol{\theta}|\boldsymbol{\lambda})\big|_{\theta=\hat{\theta}(\boldsymbol{\lambda})} \succeq mI \tag{28} $$

Suppose $\Lambda_{smooth} \subseteq \Lambda \coloneqq [\lambda_{\min}, \lambda_{\max}]^J$ *satisfies Conditions 1 and 2.*

Then any $\boldsymbol{\lambda}^{(1)}, \boldsymbol{\lambda}^{(2)} \in \Lambda_{smooth}$ *satisfies* (24).

3.2 Nonparametric additive models

We now generalize the results to nonparametric additive models. We consider estimators of the form

$$ \{\hat{g}_j(\cdot|\boldsymbol{\lambda})\}_{j=1}^J = \arg\min_{g\in\mathcal{G}} \left\| \boldsymbol{y} - \sum_{j=1}^J g_j(\boldsymbol{x}_j) \right\|_T^2 + \sum_{j=1}^J \lambda_j P_j(g_j) \tag{29} $$

where P_j are now penalty functionals. The following lemma states that the fitted functions are Lipschitz with respect to $\|\cdot\|_D$, which satisfies the Lipschitz condition in Theorem 1.

Lemma 3. *Let* $\mathcal{G}$ *be a convex function class.* $\hat{g}_j(\cdot|\boldsymbol{\lambda})$ *is defined in 29.*

Suppose the penalty functions P_j *are twice Gateaux differentiable and convex over* $\mathcal{G}$. *Suppose there is a* $m > 0$ *such that the training criterion has a twice Gateaux derivative at* $\hat{g}_j(\cdot|\boldsymbol{\lambda})$ *for all* $j = 1, ..., J$ *satisfies*

$$ \left\langle D^2 \left(\left\| \boldsymbol{y} - \sum_{j=1}^J g_j(\boldsymbol{x}_j) \right\|_T^2 + \sum_{j=1}^J \lambda_j P_j(g_j) \right) \circ h, h \right\rangle \geq m \quad \forall h \in \mathcal{G}, \|h\|_D = 1 \tag{30} $$

Let $\lambda_{\max} > \lambda_{\min} > 0$. *Let*

$$ C_{\theta^*,\Lambda} = \frac{1}{2} \left\| y - \sum_{j=1}^J g_j^*(\cdot|\boldsymbol{\lambda}) \right\|_T^2 + \lambda_{max} \sum_{j=1}^J P_j(g_j^*) \tag{31} $$

For any $\boldsymbol{\lambda}^{(1)}, \boldsymbol{\lambda}^{(2)} \in \Lambda \coloneqq [\lambda_{\min}, \lambda_{\max}]^J$*, we have*

$$\left\| \sum_{j=1}^{J} \hat{g}_j \left(\cdot | \boldsymbol{\lambda}^{(1)} \right) - \hat{g}_j \left(\cdot | \boldsymbol{\lambda}^{(2)} \right) \right\|_D \leq \frac{J}{m\lambda_{min}} \sqrt{2C_{\theta^*,\Lambda} \frac{n}{n_T} \left(1 + \frac{J\lambda_{max}}{\lambda_{min}} \right)} \left\| \boldsymbol{\lambda}^{(1)} - \boldsymbol{\lambda}^{(2)} \right\| \quad (32)$$

4 Simulations

We now provide a simulation study for the prediction error bound given in Theorem 1. The penalty parameters are chosen by a training/validation split. We show that the error of the selected model converges to that of the oracle model at the near-parametric rate.

Observations were generated from the model

$$y = \exp(x_1) + x_2^2 + \sigma\epsilon \quad (33)$$

where $\epsilon \sim N(0, 1)$ and σ scaled the error term such that the signal to noise ratio was 2. The covariates x_1 and x_2 were uniformly distributed over the interval $(-1, 1)$.

We fit a smoothing splines using the Sobolev penalty (De Boor et al. 1978, Wahba 1990, Green & Silverman 1994). The training criterion was

$$\|y - f_1(x_1) - f_2(x_2)\|_T^2 + \lambda_1 \int_0^6 (f_1^{(2)}(x))^2 dx + \lambda_2 \int_0^6 (f_2^{(2)}(x))^2 dx \quad (34)$$

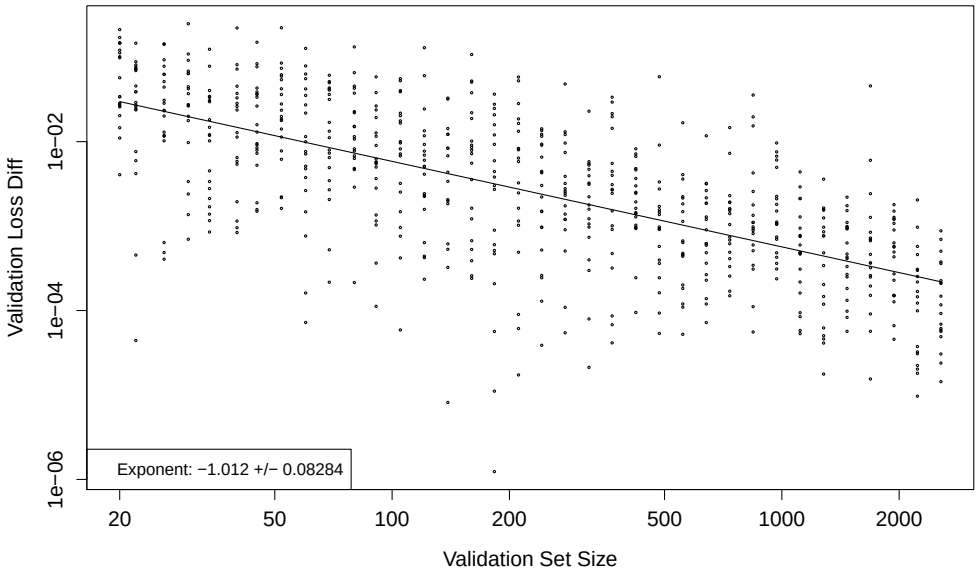
The training set contained 100 samples and models were fitted with 10 knots. A grid search was performed over the penalty parameter values $\{10^{-9+0.05i} : i = 0, ..., 140\}$. We tested 36 validation set sizes $n_V = \lfloor 20 * 2^i \rfloor$ for equally log-spaced intervals from $i = 0$ to $i = 7$. A total of 20 simulations were run for each validation set size.

Figure 4 plots the difference of between the model loss and the oracle loss

$$\left\| \hat{g}(\cdot | \hat{\boldsymbol{\lambda}}) - g^* \right\|_V^2 - \left\| \hat{g}(\cdot | \tilde{\boldsymbol{\lambda}}) - g^* \right\|_V^2$$

as the validation set size increases. The difference of the validation losses drops at a rate of about n^{-1}. This rate is in fact faster than that in Theorem 1 since the geometric term seems to play no role. We conjecture that there may be additional regularity conditions that allow the geometric term to be completely discarded.

Figure 1: Validation loss difference between oracle and selected model as validation set size grows



5 Discussion

In this paper, we have established oracle inequalities for penalty parameter selection using a training/validation split framework or k-fold cross-validation. The results address the concern in Bengio (2000) regarding "the amount of overfitting that can be brought when too many hyperparameters are optimized." Our results show that this should not be a major concern. In a non-parametric setting or parametric setting where p grows with n, the oracle error is the dominating term in the upper bound. At worst, the tuning penalty parameter problem contributes an error that is on the same order as the oracle error, say in a parametric setting where p is fixed.

There is recent interest in combining regularization methods, but seems to be an artificial restriction to two or three penalty parameters. The area of penalized regression methods with tens or hundreds of penalty parameters remains largely unexplored. Our results suggest that this direction of research could be fruitful. As shown in Feng and Simon (TBD), un-pooling the penalty parameters in a sparse group lasso model is surprisingly effective.

One major caveat to our results is that we have assumed that the penalty parameters can be tuned such that the validation loss is minimized. However it is difficult to find the global

minimizer since the validation loss is not convex in the penalty parameters. Optimization methods need to be developed to effectively solve the bilevel optimization problems in (**??**). In addition, it would be worthwhile to understand the performance of models that are only local minimizers of the validation loss.

Finally, there are still many open questions to explore. Our results assume that the fitted models are smoothly parameterized with respect to the penalty parameters and we provide a number of examples that satisfy these conditions. There are probably many more examples of regression problems that satisfy the smoothness condition and the smoothness condition itself can probably be generalized. In addition, it would be interesting to bound the distance between the selected and oracle penalty parameters

$$\left\| \hat{\boldsymbol{\lambda}} - \tilde{\boldsymbol{\lambda}} \right\|_2 \tag{35}$$

Such a result would perhaps give a more intuitive understanding of penalty parameter selection methods.

6 The Proof

In this paper, we will measure the the complexity of $\mathcal{G}(T)$ by its metric entropy. Let us recall its definition here:

Definition 4. *Let the covering number $N(u, \mathcal{G}, \|\cdot\|)$ be the smallest set of u-covers of $\mathcal{G}$ with respect to the norm $\|\cdot\|$. The metric entropy of $\mathcal{G}$ is defined as the log of the covering number:*

$$H(u, \mathcal{G}, \|\cdot\|) = \log N(u, \mathcal{G}, \|\cdot\|) \tag{36}$$

Theorem 3.

Theorem 4. *This is reproduced from the Mitchell paper*

Proof. Chaining and peeling. □

Proof of Theorem 1

Proof. □

Proof of Theorem 2

Proof of Lemma 1

Proof of Lemma 2

Proof of Lemma 3

References

Arlot, S., Celisse, A. et al. (2010), 'A survey of cross-validation procedures for model selection', *Statistics surveys* **4**, 40–79.

Barron, A., Birgé, L. & Massart, P. (1999), 'Risk bounds for model selection via penalization', *Probability theory and related fields* **113**(3), 301–413.

Bengio, Y. (2000), 'Gradient-based optimization of hyperparameters', *Neural computation* **12**(8), 1889–1900.

Chatterjee, S. & Jafarov, J. (2015), 'Prediction error of cross-validated lasso', *arXiv preprint arXiv:1502.06291* .

Chetverikov, D. & Liao, Z. (2016), 'On cross-validated lasso', *arXiv preprint arXiv:1605.02214* .

De Boor, C., De Boor, C., Mathématicien, E.-U., De Boor, C. & De Boor, C. (1978), *A practical guide to splines*, Vol. 27, Springer-Verlag New York.

Foo, C.-s., Do, C. B. & Ng, A. Y. (2008), Efficient multiple hyperparameter learning for log-linear models, *in* 'Advances in neural information processing systems', pp. 377–384.

Golub, G. H., Heath, M. & Wahba, G. (1979), 'Generalized cross-validation as a method for choosing a good ridge parameter', *Technometrics* **21**(2), 215–223.

Green, P. & Silverman, B. (1994), 'Nonparametric regression and generalized linear models, vol. 58 of', *Monographs on Statistics and Applied Probability* .

Györfi, L., Kohler, M., Krzyzak, A. & Walk, H. (2006), *A distribution-free theory of nonparametric regression*, Springer Science & Business Media.

Lecué, G., Mitchell, C. et al. (2012), 'Oracle inequalities for cross-validation type procedures', *Electronic Journal of Statistics* **6**, 1803–1837.

Simon, N., Friedman, J., Hastie, T. & Tibshirani, R. (2013), 'A sparse-group lasso', *Journal of Computational and Graphical Statistics* **22**(2), 231–245.

Snoek, J., Larochelle, H. & Adams, R. P. (2012), Practical bayesian optimization of machine learning algorithms, *in* 'Advances in neural information processing systems', pp. 2951–2959.

Van Der Laan, M. J. & Dudoit, S. (2003), 'Unified cross-validation methodology for selection among estimators and a general cross-validated adaptive epsilon-net estimator: Finite sample oracle inequalities and examples'.

van der Laan, M. J., Dudoit, S. & Keles, S. (2004), 'Asymptotic optimality of likelihood-based cross-validation', *Statistical Applications in Genetics and Molecular Biology* **3**(1), 1–23.

Wahba, G. (1990), *Spline models for observational data*, Vol. 59, Siam.

Wegkamp, M. (2003), 'Model selection in nonparametric regression', *Annals of Statistics* pp. 252–273.

Zou, H. & Hastie, T. (2003), 'Regression shrinkage and selection via the elastic net', *Journal of the Royal Statistical Society: Series B. v67* pp. 301–320.