Near-Optimal Bounds for Cross-Validation via Loss Stability

Ravi Kumar RAVI.K53@GMAIL.COM
Google, 1600 Amphitheater Parkway, Mountain View, CA 94043 USA

Daniel Lokshtanov DANIELLO@II.UIB.NO
Department of Computer Science and Engineering, UC San Diego, La Jolla, CA 92093 USA

Sergei Vassilvitskii SERGEIV@GOOGLE.COM
Google, 1600 Amphitheater Parkway, Mountain View, CA 94043 USA

Andrea Vattani AVATTANI@CS.UCSD.EDU
Department of Computer Science and Engineering, UC San Diego, La Jolla, CA 92093 USA

Abstract

Multi-fold cross-validation is an established practice to estimate the error rate of a learning algorithm. Quantifying the variance reduction gains due to cross-validation has been challenging due to the inherent correlations introduced by the folds. In this work we introduce a new and weak measure called *loss stability* and relate the cross-validation performance to this measure; we also establish that this relationship is near-optimal. Our work thus quantitatively improves the current best bounds on cross-validation.

1. Introduction

Cross-validation is a classical tool in the machine learner's repertoire for obtaining good estimates of a learning algorithm's performance (Rosset, 2009; Mullin & Sukthankar, 2000; Blockeel & Struyf, 2002). By repeatedly slicing the data into test and training sets, one obtains k dependent estimates of the average error that are usually further averaged to obtain an estimate of the performance of the algorithm. Although it was initially motivated by the lack of labeled examples (and the cost in obtaining labels), cross-validation remains extremely popular and pertinent, even in the current age of 'big data.'

Despite its ubiquity in practice, cross-validation has largely eluded formal analysis. Under investigation is the decrease in the variance of the generalization error, i.e., the variance in the estimate of the algorithm's performance. In their seminal work, Blum, Kalai, and Langford (Blum et al., 1999), proved *insanity check* bounds, and showed that cross-validation always helps to reduce this variance, although they did not quantify the magnitude of the improvement. Such a task is non-trivial since cross-validation involves subtle correlations between the hypotheses learned on the different overlapping slices of the input.

Recently, Kale, Kumar, and Vassilvitskii (Kale et al., 2011) showed that under a suitable notion of stability (called mean-square stability), k-fold cross-validation achieves a significant variance reduction. Specifically, they showed that in a 'noisy' setting, the variance reduction is near-optimal (a factor of k) and, for weakly stable algorithms, one obtains a sub-optimal $O(1/\sqrt{k})$ reduction (for more details, see (Kale et al., 2011)). Even though they improved upon previous results, their bound was far from tight, for example, simple calculations illustrate the gap in the bound even in the toy setting of estimating the mean of a Gaussian.

In this work we introduce a new stability measure called *loss stability*, which is weaker than the mean-square stability and other stability measures defined in the past. Our main contribution is to show that this new notion serves as an *additive* factor to the optimal variance reduction obtained by cross-validation. In other words, an algorithm's loss stability precisely captures the degradation of the reduction of generalization errors due to the dependencies between the different folds. Thus, we obtain an improved bound on the performance of cross-validation; in addition, we prove that this bound is near-optimal. We illustrate an

Proceedings of the 30^{th} International Conference on Machine Learning, Atlanta, Georgia, USA, 2013. JMLR: W&CP volume 28.

application of our findings to the problem of estimating the mean of a distribution and show that loss stability is able to precisely capture the variance reduction in the generalization error due to cross-validation. We then show that, in the noisy setting, many algorithms such as t-nearest neighbor rules obtain a near-optimal $\Theta(k)$ variance reduction.

2. Related work

Cross-validation has been studied (Moore & Lee, 1994; Ng, 1997; Bengio & Grandvalet, 2004; Kohavi, 1995; Kearns, 1996), and has been used in practice for many decades; see also the references in `http://masi.cscs.lsa.umich.edu/~crshalizi/notabene/cross-validation.html`. However, analyzing the improvements offered by cross-validation has been tricky. Blum, Kalai, and Langford (Blum et al., 1999) showed under mild assumptions on the learning algorithm that the variance of the cross-validation estimate is never more than that of a single holdout estimate. Kale, Kumar, and Vassilvitskii (Kale et al., 2011) generalized this considerably, quantifying the variance reduction as a function of the algorithm's stability. Our results can be viewed as an improvement over their work: we obtain a near-optimal connection between cross-validation and an appropriately defined notion of algorithmic stability.

The notion of algorithmic stability, once again, has been studied in various contexts over the past many years. Rogers and Wagner (Rogers & Wagner, 1978) and Devroye and Wagner (Devroye & Wagner, 1979) implicitly defined weak hypothesis stability in their work on leave-one-out cross-validation. Kearns and Ron (Kearns & Ron, 1999) and Anthony and Holden (Anthony & Holden, 1998) defined weak-error stability in the context of proving sanity check bounds. Kutin and Niyogi (Kutin & Niyogi, 2002) defined the weak-L_1 stability notion; see also the work of Bousquet and Elisseeff (Bousquet & Elisseeff, 2002). The notion of mean-square-stability introduced by (Kale et al., 2011) is weaker than all the previous notions except for that of weak-error stability and is closely related to the efficacy of cross-validation. Our new definition of loss stability is weaker still, but can be used to obtain near-optimal results. In contrast, we show that the notion of weak-error stability is not enough to obtain a significant variance reduction.

3. Preliminaries

Let $\mathcal{X}$ be the input space and $\mathcal{Y}$ be the set of labels and let $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$. Labeled examples are drawn from an unknown, fixed distribution $\mathcal{D}$ over $\mathcal{Z}$. We will denote by $\mathcal{D}_{\mathcal{X}}$ the marginal distribution of $\mathcal{D}$ over $\mathcal{X}$, i.e., $x \in \mathcal{X}$ is sampled from $\mathcal{D}_{\mathcal{X}}$ with probability $\sum_{y \in \mathcal{Y}} \Pr_z[z = (x, y)]$. To help readability, every time we refer to some examples $z \in \mathcal{Z}$ or $x \in \mathcal{X}$, we will implicitly assume that z and x are respectively drawn from $\mathcal{D}$ and $\mathcal{D}_{\mathcal{X}}$.

A *hypothesis* is a function $h : \mathcal{X} \to \mathcal{Y}$. The *loss* of a hypothesis h on an example $z = (x, y)$ is defined as $\ell_h(z) = \ell(y, h(x))$, where $\ell : \mathcal{Y}^2 \to \mathbb{R}^{\geq 0}$ is a fixed loss function. This leads to the following two notions: the *expected loss* of a hypothesis h defined as $\bar{\ell}_h = \mathrm{E}_z[\ell_h(z)]$ and the (empirical) loss of a hypothesis h on a test set T defined as $\ell_h(T) = \frac{1}{|T|} \sum_{z \in T} \ell_h(z)$; note that the latter is an unbiased estimator of the former. As in (Blum et al., 1999), we denote the *discrepancy* in the estimation by $\mathtt{disc}_h(T) = \ell_h(T) - \bar{\ell}_h$.

A *learning algorithm* $\mathcal{A}$ is a (symmetric) function whose input is a set S of labeled training examples and whose output is a hypothesis $\mathcal{A}(S)$. In a standard k-fold cross-validation setting we are given a set of nk examples drawn from $\mathcal{D}$, which we denote by U. Let $T_1, \ldots, T_k$ be a random equipartition of U into k parts, called *folds*, with $|T_i| = n$. We learn k different hypotheses with $h_i = \mathcal{A}(U \setminus T_i)$ the hypothesis learned on all of the data *except* for the ith fold; we denote by $m = n(k-1)$ the size of the training set for each of these k hypotheses. Following the work of (Blum et al., 1999) we focus on the cross-validated hypothesis, $h_{\mathtt{cv}}$, which picks one of the $\{h_i\}_{i=1}^k$ uniformly at random.

Specifically, we want to relate the discrepancy of $h_{\mathtt{cv}}$ to that of the discrepancy of a hypothesis trained on a single fold. Formally, let $\mathtt{disc}_i = \mathtt{disc}_i(T_i) = \ell_{h_i}(T_i) - \bar{\ell}_{h_i}$ be the discrepancy of the ith hypothesis. We are interested in:

$$\mathtt{disc}_{\mathtt{cv}} = \frac{1}{k} \sum_{i=1}^{k} \ell_{h_i}(T_i) - \frac{1}{k} \sum_{i=1}^{k} \bar{\ell}_{\mathcal{A}(h_i)} = \frac{1}{k} \sum_{i=1}^{k} \mathtt{disc}_i.$$

We assume that each of the hypotheses is unbiased, $\mathrm{E}_U[\mathtt{disc}_i] = \mathrm{E}_U[\mathtt{disc}_{\mathtt{cv}}] = 0$. Therefore, we focus on higher moments, specifically the variance of the discrepancy, $\mathbf{var}_U(\mathtt{disc}_{\mathtt{cv}})$. Note that if each of the k hypothesis was trained on an independently drawn set of training and test examples, it would be easy to conclude that $\mathbf{var}(\mathtt{disc}_{\mathtt{cv}})$ is smaller than $\mathbf{var}(\mathtt{disc}_1)$ by a factor of k. The goal of this work is to quantify the relationship between these two quantities while taking into account the dependencies between the folds.

For the rest of the paper, we assume that the training set T is drawn i.i.d. from $\mathcal{D}$ and the loss function ℓ is arbitrary but fixed. To simplify notation, for a set T of examples and an example $z \notin T$, we will use T^z

to denote the set of examples obtained by replacing an example chosen uniformly at random from T by z. We will also use shorthand notation such as $\sum_{x\in A} g(x)$ to denote $\sum_{(x,y)\in A} g(x)$.

4. Loss stability

Kale, Kumar, and Vassilvitskii (Kale et al., 2011) introduced the concept of mean-squared stability (MSS), which was based on the expected variance of the loss on a random test example under a change of a single element in the training set. We recall their definition.

Definition 1 (Mean-square stability (Kale et al., 2011))**.** *The* mean-squared stability *of a learning algorithm $\mathcal{A}$ trained on m examples and with respect to a loss ℓ is defined as*
$$\mathrm{mss}_{m,\ell}(\mathcal{A}) = \mathop{\mathrm{E}}_{T:|T|=m,z',z}\left[\left(\ell_{\mathcal{A}(T)}(z) - \ell_{\mathcal{A}(T^{z'})}(z)\right)^2\right].$$

Since m and ℓ are fixed, we will drop them for clarity. We refine and weaken the MSS notion further and introduce the concept of loss stability, which tightly captures the variance reduction possible in cross-validation.

We begin by considering an unbiased loss of a learning algorithm on a test set example. Given a training set T, let the *unbiased loss* of the algorithm on example x be $\ell'_{\mathcal{A}(T)}(z) = \ell_{\mathcal{A}(T)}(z) - \bar{\ell}_{\mathcal{A}(T)}$. When the algorithm is clear from the context we will use the shorthand $\ell'_T(z)$ for $\ell'_{\mathcal{A}(T)}(z)$.

Definition 2 (Loss stability)**.** *The* loss stability *of a learning algorithm $\mathcal{A}$ trained on m examples and with respect to a loss ℓ is defined as*
$$\mathrm{ls}_{m,\ell}(\mathcal{A}) = \mathop{\mathrm{E}}_{T:|T|=m,z',z}\left[\left(\ell'_{\mathcal{A}(T)}(z) - \ell'_{\mathcal{A}(T^{z'})}(z)\right)^2\right].$$
A learning algorithm $\mathcal{A}$ is γ-loss stable *if* $\mathrm{ls}_{m,\ell}(\mathcal{A}) \le \gamma$.

By a simple rearrangement of the terms, and using the fact that $2\,\mathbf{var}(X) = \mathrm{E}_{x,y\sim X}(x-y)^2$, the loss-stability of a learning algorithm can be viewed as the variance of the loss due to a change in a single training example of a typical training set.

Lemma 1. $\mathrm{ls}_{m,\ell}(\mathcal{A}) = 2 \mathop{\mathrm{E}}_{T:|T|=m,z}\left[\mathop{\mathbf{var}}_{z'}(\ell'_{\mathcal{A}(T^{z'})}(z))\right].$

As before, we will drop the subscripts m and ℓ. We first show that loss stability is upper bounded by mean-squared stability.

Lemma 2. *For any algorithm $\mathcal{A}$,* $\mathrm{ls}(\mathcal{A}) \le \mathrm{mss}(\mathcal{A})$.

Proof. Using Definitions 1 and 2, we have $\mathrm{mss}(\mathcal{A}) =$

$$\begin{aligned}
&= \mathop{\mathrm{E}}_{T,z',z}\left[(\ell_{\mathcal{A}(T)}(z) - \ell_{\mathcal{A}(T^{z'})}(z))^2\right] \\
&= \mathop{\mathrm{E}}_{T,z',z}\left[\left((\ell'_T(z) - \ell'_{T^{z'}}(z)) + (\bar{\ell}_{\mathcal{A}(T)} - \bar{\ell}_{\mathcal{A}(T^{z'})})\right)^2\right] \\
&= \mathop{\mathrm{E}}_{T,z',z}\left[(\ell'_T(x) - \ell'_{T^{z'}}(z))^2\right] + \mathop{\mathrm{E}}_{T,z',z}\left[(\bar{\ell}_{\mathcal{A}(T)} - \bar{\ell}_{\mathcal{A}(T^{z'})})^2\right] \\
&\ge \mathop{\mathrm{E}}_{T,z',z}\left[(\ell'_T(z) - \ell'_{T^{z'}}(x))^2\right] \quad = \quad \mathrm{ls}(\mathcal{A}),
\end{aligned}$$

where the second step follows from the facts that $E_x[\ell'_{\mathcal{A}(T)}(z)] = 0$ for any T and neither $\bar{\ell}_{\mathcal{A}(T)}$ nor $\bar{\ell}_{\mathcal{A}(T^{z'})}$ depends on z. □

Kale et al. (Kale et al., 2011) showed that many previously studied notions of stability, for example, weak-error stability (Kearns & Ron, 1999), uniform stability (Bousquet & Elisseeff, 2002), and others imply a bounded MSS. Therefore all of their bounds apply to loss stability as well; for the sake of brevity, we do not repeat them.

The discrepancy introduced earlier has a simple characterization using the unbiased loss.

Lemma 3. *For any learning algorithm $\mathcal{A}$ and $i \in [k]$,*
$$\mathop{\mathbf{var}}_{U}(\mathtt{disc}_i) = \frac{1}{n}\mathop{\mathrm{E}}_{T}[\mathop{\mathbf{var}}_{z}(\ell'_{\mathcal{A}(T)}(z))].$$

Proof.

$$\begin{aligned}
\mathop{\mathbf{var}}_{U}(\mathtt{disc}_i) &= \mathop{\mathrm{E}}_{U\setminus T_i}[\mathop{\mathbf{var}}_{T_i}(\mathtt{disc}_i \mid U \setminus T_i)] \\
&= \frac{1}{n}\mathop{\mathrm{E}}_{U\setminus T_i}[\mathop{\mathbf{var}}_{z}(\ell_{h_i}(z) - \bar{\ell}_{h_i} \mid U \setminus T_i)] \\
&= \frac{1}{n}\mathop{\mathrm{E}}_{U\setminus T_i}[\mathop{\mathbf{var}}_{z}(\ell'_{h_i}(z))].
\end{aligned}$$
□

5. Cross-validation from loss stability

We now prove our main result regarding the variance reduction obtained using k-fold cross-validation for an algorithm that is γ-loss stable. We first state a characterization of the variance of the discrepancy when using k-fold cross-validation in terms of the variance and covariance of the discrepancy on the folds.

Lemma 4 ((Kale et al., 2011))**.** $\mathbf{var}_U(\mathtt{disc_{cv}}) = \frac{1}{k}\mathbf{var}_U(\mathtt{disc}_1) + \left(1-\frac{1}{k}\right)\mathbf{cov}_U(\mathtt{disc}_1, \mathtt{disc}_2)$.

We next state our main result.

Theorem 1. *Consider any learning algorithm $\mathcal{A}$ that is γ-loss stable with respect to ℓ. Then*
$$\mathop{\mathbf{var}}_{U}(\mathtt{disc_{cv}}) \le \frac{1}{k}\mathop{\mathbf{var}}_{U}(\mathtt{disc}_1) + \left(1-\frac{1}{k}\right)\gamma.$$

By Lemma 4 it is enough to analyze the covariance $\mathbf{cov}_U(\mathtt{disc}_1, \mathtt{disc}_2)$ between the discrepancies on two different folds. The following key lemma provides a more manageable form for $\mathbf{cov}_U(\mathtt{disc}_1, \mathtt{disc}_2)$. To simplify notation, let $S = U \setminus (T_1 \cup T_2)$ be the set of training examples shared by the first two folds.

Lemma 5. *Let a fold T_i consist of a single element z_i and elements $\tilde{T}_i$. Then,*

$$\mathop{\mathbf{cov}}_U(\mathtt{disc}_1, \mathtt{disc}_2) = \mathop{\mathrm{E}}_{\substack{S,\tilde{T}_1,\tilde{T}_2 \\ z_1,z_2,z_1',z_2'}} \Big[(\ell'_{S\cup\tilde{T}_1\cup z_1}(z_2) - \ell'_{S\cup\tilde{T}_1\cup z_1'}(z_2)) \cdot (\ell'_{S\cup\tilde{T}_2\cup z_2}(z_1) - \ell'_{S\cup\tilde{T}_2\cup z_2'}(z_1)) \Big].$$

Proof. We first reduce the covariance over the folds to the covariance over a change in a single example.

$$\begin{aligned}
&\mathop{\mathbf{cov}}_U(\mathtt{disc}_1, \mathtt{disc}_2) \\
&= \mathop{\mathrm{E}}_{S,T_1,T_2}\left[\left(\frac{1}{n}\sum_{z\in T_2} \ell_{\mathcal{A}(S\cup T_1)}(z) - \bar{\ell}_{\mathcal{A}(S\cup T_1)}\right) \cdot \left(\frac{1}{n}\sum_{z\in T_1} \ell_{\mathcal{A}(S\cup T_2)}(z) - \bar{\ell}_{\mathcal{A}(S\cup T_2)}\right)\right] \\
&= \mathop{\mathrm{E}}_{S,T_1,T_2}\left[\left(\frac{1}{n}\sum_{z\in T_2} (\ell_{\mathcal{A}(S\cup T_1)}(z) - \bar{\ell}_{\mathcal{A}(S\cup T_1)})\right) \cdot \left(\frac{1}{n}\sum_{z\in T_1} (\ell_{\mathcal{A}(S\cup T_2)}(z) - \bar{\ell}_{\mathcal{A}(S\cup T_2)})\right)\right].
\end{aligned}$$

By definition of unbiased loss,

$$\begin{aligned}
\mathop{\mathbf{cov}}_U(\mathtt{disc}_1, \mathtt{disc}_2) &= \mathop{\mathrm{E}}_{S,T_1,T_2}\left[\left(\frac{1}{n}\sum_{z\in T_2} \ell'_{\mathcal{A}(S\cup T_1)}(z)\right) \cdot \left(\frac{1}{n}\sum_{z\in T_1} \ell'_{\mathcal{A}(S\cup T_2)}(z)\right)\right] \\
&= \mathop{\mathrm{E}}_{S,T_1,T_2}\left[\mathop{\mathrm{E}}_i[\ell'_{S\cup T_1}(z_2^i)] \mathop{\mathrm{E}}_j[\ell'_{S\cup T_2}(z_1^j)]\right],
\end{aligned}$$

where i, j, are distributed uniformly in $\{1, \ldots, n\}$, and z_2^i, z_1^j represent the ith element of T_2 and the jth element of T_1 (for some ordering of their elements). We can now write:

$$\begin{aligned}
\mathop{\mathbf{cov}}_U(\mathtt{disc}_1, \mathtt{disc}_2) &= \mathop{\mathrm{E}}_{S,T_1,T_2}\left[\mathop{\mathrm{E}}_i[\ell'_{S\cup T_1}(z_2^i)] \mathop{\mathrm{E}}_j[\ell'_{S\cup T_2}(z_1^j)]\right] \\
&= \mathop{\mathrm{E}}_{i,j}\left[\mathop{\mathrm{E}}_{S,T_1,T_2}[\ell'_{S\cup T_1}(z_2^i) \cdot \ell'_{S\cup T_2}(z_1^j) \mid i, j]\right] \\
&= \mathop{\mathrm{E}}_{S,T_1,T_2}[\ell'_{S\cup T_1}(z_2^1) \cdot \ell'_{S\cup T_2}(z_1^1)],
\end{aligned}$$

where the last step uses the fact that the elements in T_1 and T_2 are i.i.d.

Denote by $\tilde{T}_1, \tilde{T}_2$ sets of $(n-1)$ i.i.d. samples from $\mathcal{D}$, and z_1, z_2, z_1', z_2' i.i.d. samples from $\mathcal{D}$. We have

$$\begin{aligned}
\mathop{\mathbf{cov}}_U(\mathtt{disc}_1, \mathtt{disc}_2) &= \mathop{\mathrm{E}}_{\substack{S,\tilde{T}_1,\tilde{T}_2 \\ z_1,z_2}}[\ell'_{S\cup\tilde{T}_1\cup z_1}(z_2) \cdot \ell'_{S\cup\tilde{T}_2\cup z_2}(z_1)] \\
&= \mathop{\mathrm{E}}_{\substack{S,\tilde{T}_1,\tilde{T}_2 \\ z_1,z_2,z_1',z_2'}}[(\ell'_{S\cup\tilde{T}_1\cup z_1}(z_2) - \ell'_{S\cup\tilde{T}_1\cup z_1'}(z_2)) \cdot (\ell'_{S\cup\tilde{T}_2\cup z_2}(z_1) - \ell'_{S\cup\tilde{T}_2\cup z_2'}(z_1))],
\end{aligned}$$

since

$$\begin{aligned}
&\mathop{\mathrm{E}}_{\substack{S,\tilde{T}_1,\tilde{T}_2 \\ z_1,z_2,z_1',z_2'}}[\ell'_{S\cup\tilde{T}_1\cup z_1'}(z_2) \cdot \ell'_{S\cup\tilde{T}_2\cup z_2}(z_1)] \\
&= \mathop{\mathrm{E}}_{\substack{S,\tilde{T}_1,\tilde{T}_2 \\ z_1,z_2,z_1',z_2'}}[\mathop{\mathrm{E}}_{z_1}[\ell'_{S\cup\tilde{T}_1\cup z_1'}(z_2) \cdot \ell'_{S\cup\tilde{T}_2\cup z_2}(z_1) \mid S, \tilde{T}_1, \tilde{T}_2, z_2, z_1', z_2']] \\
&= \mathop{\mathrm{E}}_{\substack{S,\tilde{T}_1,\tilde{T}_2 \\ z_1,z_2,z_1',z_2'}}[\ell'_{S\cup\tilde{T}_1\cup z_1'}(z_2) \cdot \mathop{\mathrm{E}}_{z_1}[\ell'_{S\cup\tilde{T}_2\cup z_2}(z_1) \mid S, \tilde{T}_1, \tilde{T}_2, z_2, z_1', z_2']] \quad = \quad 0;
\end{aligned}$$

similar arguments hold for the other cross-terms. □

The proof of our main theorem is now a simple application of the Cauchy–Schwarz inequality.

Proof of Theorem 1. By Lemma 4 it is enough to show that $\mathbf{cov}_U(\mathtt{disc}_1, \mathtt{disc}_2) \le \gamma$. By Lemma 5 and Cauchy–Schwarz,

$$\begin{aligned}
&\mathop{\mathbf{cov}}_U(\mathtt{disc}_1, \mathtt{disc}_2) \\
&= \mathop{\mathrm{E}}_{\substack{S,\tilde{T}_1,\tilde{T}_2 \\ z_1,z_2,z_1',z_2'}}\Big[(\ell'_{S\cup\tilde{T}_1\cup z_1}(z_2) - \ell'_{S\cup\tilde{T}_1\cup z_1'}(z_2)) \cdot (\ell'_{S\cup\tilde{T}_2\cup z_2}(z_1) - \ell'_{S\cup\tilde{T}_2\cup z_2'}(z_1))\Big] \\
&\le \sqrt{\mathop{\mathrm{E}}_{\substack{S,\tilde{T}_1,\tilde{T}_2 \\ z_1,z_2,z_1',z_2'}}[(\ell'_{S\cup\tilde{T}_1\cup z_1}(z_2) - \ell'_{S\cup\tilde{T}_1\cup z_1'}(z_2))^2]} \cdot \sqrt{\mathop{\mathrm{E}}_{\substack{S,\tilde{T}_1,\tilde{T}_2 \\ z_1,z_2,z_1',z_2'}}[(\ell'_{S\cup\tilde{T}_2\cup z_2}(z_1) - \ell'_{S\cup\tilde{T}_2\cup z_2'}(z_1))^2]} \\
&= \mathop{\mathrm{E}}_{S,\tilde{T}_1,z_1,z_2,z_1'}[(\ell'_{S\cup\tilde{T}_1\cup z_1}(z_2) - \ell'_{S\cup\tilde{T}_1\cup z_1'}(z_2))^2] \\
&= \mathop{\mathrm{E}}_{T_1,z_1',z_2}[(\ell'_{T_1}(z_2) - \ell'_{T_1^{z_1'}}(z_2))^2] \quad \le \quad \gamma.
\end{aligned}$$

□

6. Does cross-validation always help?

Previous work left open the question whether an assumption on the stability of the algorithm is at all

necessary for k-fold cross-validation to yield a significant variance reduction. Intuitively, one may try to argue that discrepancy estimates obtained from very unstable algorithms are essentially independent. On the other hand, one can try to weaken the notion of stability needed to obtain $O(k)$ variance reduction. In studying the effect of mean square stability, Kale et al. (Kale et al., 2011) asked whether the weakest such notion, that of weak-error[1] stability introduced in (Kearns & Ron, 1999) is enough to obtain a significant variance reduction.

In this section we provide an answer to both questions by showing that there are instances that are $(0,0)$-weak-error stable for which the variance reduction is only a constant *independent* of k. We observe that such instances have hypothesis class with VC dimension only 2, hence assumptions based only on the VC dimension are also insufficient.

Theorem 2. *For every $4\sqrt{n} \leq t \leq \sqrt{n(k-2)}/2$, there is a loss function ℓ and a learning algorithm $\mathcal{A}$ such that $\mathbf{var}_U(\mathtt{disc}_1) = 1/n$ and $\mathbf{var}_U(\mathtt{disc}_{\mathtt{cv}}) = \Omega(\frac{1}{t\sqrt{n}})$. Furthermore the algorithm $\mathcal{A}$ is $(\frac{2}{\sqrt{n(k-2)}} + \frac{2}{t})$-loss stable and $(0,0)$-weak-error stable.*

Proof. The input space $\mathcal{X}$ and labels $\mathcal{Y}$ are both $\{-1,1\}$, with uniform distribution over $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$. The loss function is $\ell(y, h(x)) = h(x)$, i.e., it is independent of y. (While having negative loss may be non-standard, it greatly simplifies notation, and changing the loss function to $\ell(y, h(x)) = h(x) + c$ for any constant c would not change the obtained results.)

For $b \in \{\pm 1\}$, let $h_b(x) = b \cdot x$. The expected loss of h_b is

$$\begin{aligned} \bar{\ell}_{h_b} &= \mathop{\mathrm{E}}_{(x,y)}[\ell_{h_b}((x,y))] \\ &= \mathop{\mathrm{E}}_{(x,y)}[\ell(y, h_b(x))] = \mathop{\mathrm{E}}_{(x,y)}[b \cdot x] = 0. \end{aligned} \quad (1)$$

For every $t \in \mathbb{Z}$ and set $S = \{(x_1,y_1), \ldots, (x_t,y_t)\} \subseteq (\mathcal{X} \times \mathcal{Y})^t$, define $\|S\| = \sum_{i \leq t} x_i$. For a function $f : \mathbb{Z} \to \{-1,1\}$ to be defined later we consider the algorithm $\mathcal{A}(S)$ that returns the hypothesis $h_{f(\|S\|)}$. Observe that (1) implies the unbiased loss $\ell'_{\mathcal{A}(S)}(z) = \ell_{\mathcal{A}(S)}(z)$ for all S.

We start by computing $\mathbf{var}_U(\mathtt{disc}_1)$ and $\mathbf{cov}_U(\mathtt{disc}_1, \mathtt{disc}_2) = \mathrm{E}_U[\mathtt{disc}_1 \cdot \mathtt{disc}_2]$. By Lemma 3, the definition of ℓ', and the fact $f(\cdot) \in \{-1,1\}$,

$$\begin{aligned} \mathop{\mathbf{var}}_U(\mathtt{disc}_1) &= \frac{1}{n} \mathop{\mathrm{E}}_{S,T_2}[\mathop{\mathbf{var}}_{(x,y)}(\ell'_{S \cup T_2}(x,y) \mid S, T_2)] \\ &= \frac{1}{n} \mathop{\mathrm{E}}_{S,T_2}[\mathop{\mathbf{var}}_{x}(f(\|S\| + \|T_2\|) \cdot x \mid S, T_2)] \\ &= \frac{1}{n}. \end{aligned}$$

We now proceed with $\mathbf{cov}_U(\mathtt{disc}_1, \mathtt{disc}_2)$. In the following we write $f_{S,T,x}$ as a shorthand for $f(\|S\| + \|T\| + x)$. By Lemma 5 and the definition of ℓ', we have

$$\begin{aligned} &\mathop{\mathbf{cov}}_U(\mathtt{disc}_1, \mathtt{disc}_2) \\ &= \mathop{\mathrm{E}}_{S,T_1,T_2,x_1,x_2,x'_1,x'_2}[x_2(f_{S,T_1,x_1} - f_{S,T_1,x'_1}) \\ &\qquad \cdot x_1(f_{S,T_2,x_2} - f_{S,T_2,x'_2})] \\ &= \mathop{\mathrm{E}}_{S}[\mathop{\mathrm{E}}_{T_1,x_1,x'_1}[x_1(f_{S,T_1,x_1} - f_{S,T_1,x'_1})] \\ &\qquad \cdot \mathop{\mathrm{E}}_{T_2,x_2,x'_2}[x_2(f_{S,T_2,x_2} - f_{S,T_2,x'_2})]] \\ &= \mathop{\mathrm{E}}_{S}[(\mathop{\mathrm{E}}_{T_1,x,x'}[x(f_{S,T_1,x} - f_{S,T_1,x'})])^2] \\ &= \frac{1}{2} \cdot \mathop{\mathrm{E}}_{S}[(\mathop{\mathrm{E}}_{T_1}[f_{S,T_1,1} - f_{S,T_1,-1}])^2], \end{aligned}$$

where the last step follows as $x, x' \in \{1,-1\}$.

Thus to create a setting where cross-validation performs poorly we need to find a function f that maximizes $\mathrm{E}_S[(\mathrm{E}_{T_1}[f_{S,T_1,1} - f_{S,T_1,-1}])^2]$. We now describe such a function f. In fact we will describe a family of functions, with one function f for every choice of a parameter $10\sqrt{n} \leq t \leq \sqrt{n(k-2)}/2$ in order to allow for different values of loss stability for $\mathcal{A}$.

$$f(x) = \begin{cases} -1, & \text{if } x \geq 0 \text{ and } \lfloor \frac{x}{t} \rfloor \text{ is even,} \\ 1, & \text{if } x \geq 0 \text{ and } \lfloor \frac{x}{t} \rfloor \text{ is odd,} \\ -f(-x), & \text{if } x < 0. \end{cases}$$

The function f is a step-wise function: we will call x a *step* of f if $f(x) \neq f(x-1)$.

We now proceed with computing the covariance. Denote by s the size of the overlapping training set, $s = |S| = (k-2)n$. We say that the set S is *active* if the following two properties are satisfied. (a) $-\sqrt{s} \leq \|S\| \leq \sqrt{s}$, and (b) $\|S\| \mod t \in \{0, 1, \ldots, \frac{\sqrt{n}}{2}\}$. Since $\|S\| = \sum_{(x,y) \in S} x$ where the $x's$ are uniform in $\{-1,1\}$, we can interpret $\|S\|$ as a one-dimensional unbiased random walk of length s. The following facts are known for a random walk W of length n starting at zero[2]:

[1] A learning algorithm is $\mathcal{A}$ is (β,δ)-*weak-error stable* w.r.t. ℓ if $\Pr_{S,z'}(|\bar{\ell}_{\mathcal{A}(S)} - \bar{\ell}_{\mathcal{A}(S^{z'})}| \leq \beta) \geq 1 - \delta$.

[2] Such bounds can be easily derived by approximating the distribution of W by a (shifted) binomial distribution which in turn can be approximated by the normal distribution.

1. $\Pr[W = 0] \geq \frac{1}{\sqrt{n}}$.
2. For every $-\sqrt{n} \leq i \leq \sqrt{n}$ with the same parity as n, $\frac{1}{3} \leq \frac{\Pr[W=i]}{\Pr[W=0]} \leq 1$.
3. $\Pr[-\sqrt{n} \leq W \leq \sqrt{n}] \geq \frac{1}{2}$.
4. For $i \geq 0$, $\Pr[|W| = i + \sqrt{n}] \leq \frac{1}{e}\Pr[|W| = i]$.

Let A_S be the event that S is active. For $4\sqrt{n} \leq t \leq \frac{\sqrt{s}}{2}$, the number of values for $\|S\|$ that make S active is at least $(\lfloor \frac{2\sqrt{s}}{t} \rfloor - 1)\frac{\sqrt{n}}{2} \geq \frac{\sqrt{sn}}{2t}$. Thus, using properties 2–3 above,

$$\Pr[A_S] \geq \frac{\sqrt{sn}}{2t} \cdot \frac{1}{2\sqrt{s}} \cdot \frac{1}{3} \geq \frac{\sqrt{n}}{12t}.$$

We will derive a bound for the covariance by simply considering the case in which S is active.

$$\begin{aligned}&\operatorname*{\mathbf{cov}}_U(\mathtt{disc}_1, \mathtt{disc}_2)\\ &\geq \operatorname*{E}_S[(\operatorname*{E}_{T_1}[f_{S,T_1,1} - f_{S,T_1,-1} \mid A_S])^2] \cdot \Pr[A_S]. \qquad (2)\end{aligned}$$

Observe that $f_{S,T_1,1} - f_{S,T_1,-1} \in \{2, -2\}$ when $\|S\| + \|T\|$ lands on a step of f, i.e., when $\|S\| + \|T\| \in \{x^*, x^* + 1\}$ where x^* is a step[3], and zero otherwise. For a specific S, let $x_0^*, x_1^*, \ldots$ be the steps of f sorted by the distance to $\|S\|$. Let $E_S^j(T_1)$ be the event that, conditioned on S, T_1 is such that $\|S\| + \|T_1\|$ lands on the jth step from $\|S\|$. Recall that $\|T_1\|$ is distributed as a random walk of length n. When S is active, the zeroth step is at distance at most $\sqrt{n}$ from $\|S\|$ and therefore properties 1 and 2 above imply that $\Pr_{T_1}(E_S^0[T_1] \mid A_S) \geq \frac{1}{3\sqrt{n}}$. For $j \geq 1$ and S active, the distances of x_{2j-1}^* and x_{2j}^* to $\|S\|$ are at least $(t - \frac{\sqrt{n}}{2}) + (j-1)t \geq \frac{7}{2}j\sqrt{n}$ as $t \geq 4\sqrt{n}$. Thus, property 4 yields $\Pr[E_S^{2j-1}[T_1] \mid A_S] \leq \frac{\exp(-7j/2)}{\sqrt{n}}$ and similarly for $\Pr[E_S^{2j}[T_1] \mid A_S]$. We can conclude

$$\begin{aligned}&\left|\operatorname*{E}_{T_1}[f_{S,T_1,1} - f_{S,T_1,-1} \mid A_S]\right|\\ &\quad\geq 2\Pr_{T_1}[E_S^0[T_1]] - \left|2\sum_{j\geq 1}\Pr_{T_1}[E_S^j[T_1]]\right|\\ &\quad\geq \frac{2}{\sqrt{n}}\left(\frac{1}{3} - 2\sum_{j\geq 1} e^{-7j/2}\right).\end{aligned}$$

Simple algebra now gives $|\operatorname{E}_{T_1}[f_{S,T_1,1} - f_{S,T_1,-1}]| \geq \frac{1}{2\sqrt{n}}$. Combining this with (2) and the bound on $\Pr[A_S]$, we have $\operatorname{\mathbf{cov}}_U(\mathtt{disc}_1, \mathtt{disc}_2) \geq \frac{\sqrt{n}}{12t} \cdot \left(\frac{1}{2\sqrt{n}}\right)^2 \geq \frac{1}{48t\sqrt{n}}$. The bound on $\operatorname{\mathbf{var}}_U(\mathtt{disc}_{\mathtt{cv}})$ now follows by Lemma 4.

It remains to analyze the loss stability and the weak error stability of the algorithm. Since both hypotheses output by the algorithm have expected loss 0, the algorithm is $(0, 0)$-weak error stable. Finally,

$$\begin{aligned}\mathrm{ls}(\mathcal{A}) &= \operatorname*{E}_{T,z',x}\left[\left(\ell'_{\mathcal{A}(T)}(x) - \ell'_{\mathcal{A}(T^{z'})}(x)\right)^2\right]\\ &= \operatorname*{E}_{T,z,z',x}\left[\left(\ell_{\mathcal{A}(T)}(x) - \ell_{\mathcal{A}(T^{z'})}(x)\right)^2\right].\end{aligned}$$

Inserting the definition for the loss function yields

$$\begin{aligned}\mathrm{ls}(\mathcal{A}) &= \operatorname*{E}_{T,z,z',x}\left[(f(\|T\|)\cdot x - f(\|T\| - z + z')\cdot x)^2\right]\\ &= \operatorname*{E}_{T,z,z'}\left[(f(\|T\|) - f(\|T\| - z + z'))^2\right]\\ &\leq 2\cdot\Pr_T[f(\|T\|) \neq f(\|T\| + 2)] \leq \frac{2}{\sqrt{s}} + \frac{2}{t}. \qquad \square\end{aligned}$$

7. Near-optimality

In this section we show that the bound derived in Theorem 1 is nearly tight. We begin with a discussion of our main result and then show a specific instance where it cannot be improved by more than a factor of $3/2$.

Theorem 1 shows that the reduction in the variance of the discrepancy is bounded by the loss stability of the learning algorithm. In fact, the reduction in error depends on how the variance of an average training set on a single training example compares in magnitude to the variance due to a change in a single training example on an 'average' training set. To see this, observe that using Lemma 1 and Lemma 3 we can rewrite the statement of our main theorem as

$$\begin{aligned}&\operatorname*{\mathbf{var}}_U(\mathtt{disc}_{\mathtt{cv}}) \leq \frac{1}{k}\operatorname*{\mathbf{var}}_U(\mathtt{disc}_1) + \frac{k-1}{k}\gamma\\ &= \frac{1}{kn}\operatorname*{E}_{S,T}[\operatorname*{\mathbf{var}}_z(\ell'_{S\cup T}(z))] + \frac{2(k-1)}{k}\operatorname*{E}_{S,T,z}[\operatorname*{\mathbf{var}}_{z'}(\ell'_{S\cup T^{z'}}(z))]\\ &= \frac{1+\rho}{k}\operatorname*{\mathbf{var}}_U(\mathtt{disc}_1),\end{aligned}$$

where

$$\rho = 2n(k-1)\frac{\operatorname{E}_{S,T,z}[\mathbf{var}_{z'}(\ell'_{S\cup T^{z'}}(z))]}{\operatorname{E}_{S,T}[\mathbf{var}_z(\ell'_{S\cup T}(z))]},$$

is the *exchange ratio*. Thus the lower the exchange ratio of a learning algorithm, the higher the reduction due to cross-validation.

[3]Note that only one of $\{x^*, x^* + 1\}$ is obtainable by $\|S\| + \|T\|$ because of parity.

Regression. We begin by considering the case of regression under squared loss and show that for the empirical risk minimization algorithm, $\rho = 2$, and therefore k-fold cross-validation achieves a $k/3$ reduction in the variance of the discrepancy.

Suppose the examples are drawn from a one-dimensional distribution and we are interested in predicting the mean. We will consider the setting of a squared-loss function. Specifically, suppose that the examples are drawn from a distribution $\mathcal{D}_x$ with mean μ and variance σ^2. Let h_T be the hypothesis output by the algorithm $\mathcal{A}$ trained on T. The loss of a hypothesis on an example x is then $\ell_{\mathcal{A}(T)}(x) = (h_T - x)^2$.

We begin by computing the unbiased loss,

$$\begin{aligned}\ell'_T(x) = \ell_T(x) - \bar{\ell}_T(x) &= (x - h_T)^2 - \mathop{\mathrm{E}}_x[x - h_T]^2 \\ &= x^2 - 2h_T(x - \mu) - \mathop{\mathrm{E}}_x[x^2].\end{aligned}$$

We can now proceed with computing ρ. For the numerator,

$$\begin{aligned}&\mathop{\mathrm{E}}_{T,x}[\mathop{\mathbf{var}}_{z'}(\ell'_{T^{z'}}(x))] \\ &= \mathop{\mathrm{E}}_{T,x}[\mathop{\mathbf{var}}_{z'}(x^2 - 2(x-\mu)h_{T^{z'}} - \mathop{\mathrm{E}}_x[x^2])] \\ &= \mathop{\mathrm{E}}_{T,x}[4(x-\mu)^2 \mathop{\mathbf{var}}_{z'}(h_{T^{z'}})] \\ &= 4\sigma^2 \mathop{\mathrm{E}}_T[\mathop{\mathbf{var}}_{z'}(h_{T^{z'}})].\end{aligned}$$

For the denominator, we have

$$\begin{aligned}&\mathop{\mathrm{E}}_T[\mathop{\mathbf{var}}_x(\ell'_T(x))] \\ &= \mathop{\mathrm{E}}_T[\mathop{\mathbf{var}}_x(x^2 - 2h_T x)] \\ &= \mathop{\mathrm{E}}_T[\mathop{\mathbf{var}}_x(x^2) + 4h_T^2 \mathop{\mathbf{var}}_x(x) - 4h_T \mathop{\mathbf{cov}}_x(x, x^2)] \\ &= \Big(\mathop{\mathbf{var}}_x(x^2) + 4\mathop{\mathbf{var}}_x(x)(\mathop{\mathrm{E}}_T[h_T])^2 - 4\mathop{\mathbf{cov}}_x(x,x^2)\mathop{\mathrm{E}}_T[h_T]\Big) \\ &\quad + 4\mathop{\mathbf{var}}_x(x)\mathop{\mathbf{var}}_T(h_T).\end{aligned}$$

We claim that the first term is non-negative which will imply that

$$\mathop{\mathrm{E}}_T[\mathop{\mathbf{var}}_x(\ell'_T(x))] \geq 4\sigma^2 \mathop{\mathbf{var}}_T(h_T).$$

To prove the claim, define $g(z) = \mathbf{var}_x(x^2) + 4\mathbf{var}_x(x)z^2 - 4\mathbf{cov}_x(x, x^2)z$.

We have that $g'(z) = 0$ if and only if $z = z^* = \mathbf{cov}_x(x, x^2)/(2\mathbf{var}_x(x))$ and $g''(z) = 8\mathbf{var}_x(x) > 0$. Hence, it is enough that $g(z^*)$ is non-negative. We have

$$\begin{aligned}g(z^*) &= \mathop{\mathbf{var}}_x(x^2) + \frac{(\mathbf{cov}_x(x,x^2))^2}{\mathbf{var}_x(x)} - \frac{2(\mathbf{cov}_x(x,x^2))^2}{\mathbf{var}_x(x)} \\ &= \mathop{\mathbf{var}}_x(x^2) - \frac{(\mathbf{cov}_x(x,x^2))^2}{\mathbf{var}_x(x)} \geq 0,\end{aligned}$$

where the last step follows by the well-known fact that $\mathbf{cov}(X,Y) \leq \sqrt{\mathbf{var}(X)\cdot\mathbf{var}(Y)}$, for any two random variables X and Y.

Therefore, the exchange ratio, ρ is bounded by:

$$\rho \leq 2n(k-1)\frac{\mathrm{E}_T[\mathbf{var}_{z'}(h_{T^{z'}})]}{\mathbf{var}_T(h_T)}. \tag{3}$$

Consider the bound in Equation 3. For algorithms where the effect of a single training example on the hypothesis is independent of the rest of the training set, we expect that the variance of the hypothesis when replacing a single sample in the training set T is a factor $|T| = n(k-1)$ smaller than the variance of the hypothesis over the whole training set. This results in $\rho = 2$. For instance, in case of empirical risk minimization (ERM) algorithm that given a training set T returns hypothesis $\mathcal{A}(T) = h_T = \frac{1}{|T|}\sum_{x\in T} x$, it is easy to check that:

$$\mathop{\mathrm{E}}_T[\mathop{\mathbf{var}}_{z'}(h_{T^{z'}})] = \frac{\sigma^2}{n^2(k-1)^2}, \text{ and } \mathop{\mathbf{var}}_T(h_T) = \frac{\sigma^2}{n(k-1)},$$

which give $\rho = 2$. Hence, cross-validation achieves a $k/3$ reduction in the variance of the discrepancy. On the other hand, an algorithm that considers more interactions between training examples will have a higher value of ρ.

Lower bound. We now show that the bound in Theorem 1 is nearly optimal.

Theorem 3. *There exists a loss function ℓ and a γ-loss stable learning algorithm $\mathcal{A}$ such that $\mathbf{cov}_U(\mathtt{disc}_1, \mathtt{disc}_2) \geq \gamma/2$. Hence,*

$$\mathop{\mathbf{var}}_U(\mathtt{disc}_{\mathtt{cv}}) \geq \frac{1}{k}\mathop{\mathbf{var}}_U(\mathtt{disc}_1) + \left(1 - \frac{1}{k}\right)\frac{\gamma}{2}.$$

Proof. Consider again the setting of estimating the mean of a distribution using the ERM algorithm with squared-loss function, and assume $\mu = 0$. Due to symmetry, we again focus on the covariance between the first two folds. First we write down the discrepancies.

$$\begin{aligned}\mathtt{disc}_1 &= \ell_{h_{S\cup T_1}}(T_2) - \mathop{\mathrm{E}}_T[\ell_{h_{S\cup T_2}}(T)] \\ &= \frac{1}{n}\sum_{y\in T_2}\Big((y^2 - 2yh_{S\cup T_1} + h^2_{S\cup T_1}) \\ &\qquad - \mathop{\mathrm{E}}_y[y^2 - 2yh_{S\cup T_1} + h^2_{S\cup T_1}]\Big) \\ &= \frac{1}{n}\sum_{y\in T_2} y^2 - 2h_{T_2}h_{S\cup T_1} - \sigma^2.\end{aligned}$$

Similarly, $\texttt{disc}_2 = \frac{1}{n}\sum_{y\in T_1} y^2 - 2h_{T_1}h_{S\cup T_2} - \sigma^2$. By the law of total covariance (Kale et al., 2011):

$$\underset{U}{\mathbf{cov}}(\texttt{disc}_1, \texttt{disc}_2) = \mathop{\mathrm{E}}_{S,T_1}\left[\underset{T_2}{\mathbf{cov}}(\texttt{disc}_1, \texttt{disc}_2 \mid S, T_1)\right].$$

We are concerned with the covariance over the fold T_2, and therefore can drop terms independent of T_2:

$$\begin{aligned}
&\underset{T_2}{\mathbf{cov}}(\texttt{disc}_1, \texttt{disc}_2 \mid S, T_1)\\
&= \underset{T_2}{\mathbf{cov}}(\frac{1}{n}\sum_{y\in T_2} y^2 - 2h_{S\cup T_1}h_{T_2}, -2h_{S\cup T_2}h_{T_1})\\
&= 4h_{S\cup T_1}h_{T_1}\underset{T_2}{\mathbf{cov}}(h_{T_2}, h_{S\cup T_2})\\
&\quad - \frac{2h_{T_1}}{n}\underset{T_2}{\mathbf{cov}}(\sum_{y\in T_2} y^2, h_{S\cup T_2}).
\end{aligned}$$

Since $h_{S\cup T_1} = \frac{\left(n(k-2)h_S + nh_{T_1}\right)}{n(k-1)}$, we have

$$\begin{aligned}
\underset{T_2}{\mathbf{cov}}(h_{T_2}, h_{S\cup T_2}) &= \underset{T_2}{\mathbf{cov}}(h_{T_2}, \frac{1}{k-1}h_{T_2})\\
&= \frac{1}{k-1}\underset{T_2}{\mathbf{var}}(h_{T_2}) = \frac{\sigma^2}{n(k-1)}.
\end{aligned}$$

Therefore, the first term of the covariance is:

$$\begin{aligned}
&\frac{4\sigma^2}{n(k-1)}\mathop{\mathrm{E}}_{S,T_1}[h_{S\cup T_1}h_{T_1}]\\
&= \frac{4\sigma^2}{n(k-1)}\mathop{\mathrm{E}}_{S,T_1}\left[\frac{1}{n(k-1)}(n(k-2)h_S + nh_{T_1})h_{T_1}\right]\\
&= \frac{4\sigma^4}{n^2(k-1)^2}.
\end{aligned}$$

To bound the second term, we want to compute:

$$\mathop{\mathrm{E}}_{S,T_1}\left[\frac{2h_{T_1}}{n}\underset{T_2}{\mathbf{cov}}(\sum_{y\in T_2} y^2, h_{S\cup T_2})\right].$$

Note that the covariance is independent of T_1, therefore, we can rewrite as

$$\mathop{\mathrm{E}}_{T_1}[h_{T_1}]\mathop{\mathrm{E}}_{S}\left[\frac{2}{n}\underset{T_2}{\mathbf{cov}}(\sum_{y\in T_2} y^2, h_{S\cup T_2})\right] = 0.$$

Therefore, $\mathbf{cov}_U(\texttt{disc}_1, \texttt{disc}_2) = \frac{4\sigma^4}{n^2(k-1)^2}$. On the other hand, Lemma 1 implies that the loss stability of the algorithm is $\gamma = \frac{8\sigma^4}{n^2(k-1)^2}$. □

8. Noisy setting

In previous work Kale et al. (Kale et al., 2011) considered the noisy setting, where every label is corrupted with a small probability. We recall their definition below:

Definition 3. *An instance of the cross-validation problem composed of the algorithm $\mathcal{A}$, the loss function ℓ, and the distribution $\mathcal{D}$ is δ-volatile, if $\mathbf{var}_U(\texttt{disc}_1) \geq \frac{\Omega(\delta)}{n}$.*

They showed that $O(1/m)$-uniform stable algorithms achieve optimal variance reduction of $1/k$. However, uniform stability is a very strong assumption and many algorithms such as t-nearest neighbor learning rules do not have non-trivial uniform stability but have small $(O(\sqrt{t}/m), O(\sqrt{t}/m))$-weak L_1 stability (Devroye & Wagner, 1979). This setting of weak L_1 stability implies that these algorithms are $O(1/m)$-mean square stable, and thus by Lemma 2, are $O(1/m)$-loss stable.

The following Lemma is immediate and allows us to conclude that these learning algorithms achieve a near-optimal variance reduction.

Lemma 6. *In an $\Omega(1)$-volatile setting, a learning algorithm $\mathcal{A}$ that is either $(O(1/m), O(1/m))$-weak-hypothesis or weak-L_1 stable has:*

$$\underset{U}{\mathbf{var}}(\texttt{disc}_{\texttt{cv}}) \leq \frac{O(1)}{k}\underset{U}{\mathbf{var}}(\texttt{disc}_1).$$

9. Conclusions

We studied the decrease in the variance of cross-validation through the lens of a new algorithmic stability notion, namely, the loss stability. We also showed that this is a near-tight connection. It will be interesting to empirically compute the loss stability of popular learning algorithms such as decision trees and SVM, for which an analysis similar to the one in Section 7 seems daunting.

References

Anthony, M. and Holden, S. B. Cross-validation for binary classification by real-valued functions: theoretical analysis. In *Proc. COLT*, pp. 218–229, 1998.

Bengio, Yoshua and Grandvalet, Yves. No unbiased estimator of the variance of k-fold cross-validation. *JMLR*, 5:1089–1105, 2004.

Blockeel, Hendrik and Struyf, Jan. Efficient algorithms for decision tree cross-validation. *JMLR*, 3:621–650, 2002.

Blum, A., Kalai, A., and Langford, J. Beating the hold-out: Bounds for k-fold and progressive cross-validation. In *Proc. COLT*, pp. 203–208, 1999.

Bousquet, O. and Elisseeff, A. Stability and generalization. *JMLR*, 2:499–526, 2002.

Devroye, L. P. and Wagner, T. J. Distribution-free performance bounds. *IEEE TOIT*, 25:601–604, 1979.

Kale, S., Kumar, R., and Vassilvitskii, S. Cross-validation and mean-square stability. In *Proc. ICS*, pp. 487–495, 2011.

Kearns, M. A bound on the error of cross validation with consequences for the training-test split. In *Proc. NIPS*, pp. 183–189, 1996.

Kearns, M. J. and Ron, D. Algorithmic stability and sanity-check bounds for leave-one-out cross-validation. *Neural Computation*, 11(6):1427–1453, 1999.

Kohavi, R. A study of cross-validation and bootstrap for accuracy estimation and model selection. In *Proc. IJCAI*, pp. 1137–1143, 1995.

Kutin, Samuel and Niyogi, Partha. Almost-everywhere algorithmic stability and generalization error. In *Proc. UAI*, pp. 275–282, 2002.

Moore, A. W. and Lee, M. S. Efficient algorithms for minimizing cross validation error. In *Proc. ICML*, pp. 190–198, 1994.

Mullin, M. D. and Sukthankar, R. Complete cross-validation for nearest neighbor classifiers. In *Proc. ICML*, pp. 639–646, 2000.

Ng, A. Y. Preventing "overfitting" of cross-validation data. In *Proc. ICML*, pp. 245–253, 1997.

Rogers, W. and Wagner, T. A finite sample distribution-free performance bound for local discrimination rules. *Annals of Statistics*, 6(3):506–514, 1978.

Rosset, Saharon. Bi-level path following for cross validated solution of kernel quantile regression. *JMLR*, 10:2473–2505, 2009.