On the bias of K-fold cross validation with stable learners

Anass Aghbalou
LTCI, Télécom Paris
Université Paris-Saclay

François Portier
CREST
Ensai

Anne Sabourin
CNRS, MAP5
Université Paris Cité

Abstract

This paper investigates the efficiency of the K-fold cross-validation (CV) procedure and a debiased version thereof as a means of estimating the generalization risk of a learning algorithm. We work under the general assumption of uniform algorithmic stability. We show that the K-fold risk estimate may not be consistent under such general stability assumptions, by constructing non vanishing lower bounds on the error in realistic contexts such as regularized empirical risk minimisation and stochastic gradient descent. We thus advocate the use of a debiased version of the K-fold and prove an error bound with exponential tail decay regarding this version. Our result is applicable to the large class of uniformly stable algorithms, contrarily to earlier works focusing on specific tasks such as density estimation. We illustrate the relevance of the debiased K-fold CV on a simple model selection problem and demonstrate empirically the usefulness of the promoted approach on real world classification and regression datasets.

1 INTRODUCTION

Introduced in Stone (1974), cross-validation (CV) is a popular tool in statistics for estimating the generalization risk of a learning algorithm. It is also the mainstream approach for model and parameter selection. Despite its widespread use, it has been shown in several contexts that CV schemes fail to select the correct model unless the test fraction is negligible in front of the sample size. Unfortunately, this excludes the widely used K-fold CV. This suboptimality has been pinned in the linear regression framework by Burman (1989); Shao (1997); Yang (2007), then in other specific frameworks such as density estimation (Arlot, 2008)

and classification (Yang, 2006). The theoretical properties of CV procedures for model selection in wider settings are notoriously difficult to establish and remain the subject of active research (Bayle et al., 2020; Wager, 2020).

To tackle the suboptimality of K-fold, Burman (1989); Fushiki (2011) have proposed to add some debiasing correction terms to the K-fold CV estimate in order to improve the convergence rate. However, the analysis conducted in these works is purely asymptotic and focuses only on ordinary linear regression. More recently Arlot and Lerasle (2016) conduct a non asymptotic study for the bias corrected K-fold in the density estimation framework and show the benefits of such a correction. Nonetheless, the latter study relies on closed-form formulas for risk estimates which are valid only for histogram rules. To summarize, the statistical consistency of the debiased version has been established only in specific frameworks pertaining to classical statistics. In addition, to our best knowledge, the consistency of K-fold without correction has not been proved nor disproved in the existing literature.

The main purpose of this paper is to establish non-asymptotic results (upper and lower bounds) regarding the error of the K-fold risk estimate, under realistic assumptions which are valid for a wide class of modern algorithms (regularized empirical risk minimization, neural networks, bagging, SGD, *etc...*), namely an *algorithmic stability* assumption discussed below. In other words, the question we seek to answer is as follows:

- Is K-fold cross-validation consistent under algorithmic stability assumptions? If not, how about the bias corrected K-fold?

The notion of *algorithmic stability* and its consequences in learning theory has received much attention since its introduction in Devroye and Wagner (1979). This property allows to obtain generalization bounds for a large class of a learning algorithms such as k-nearest-neighbors (Devroye and Wagner, 1979), empirical risk minimizers (Kearns and Ron, 1999), regularization networks (Bousquet and Elisseeff, 2001), bagging (Elisseeff et al., 2005) to name but a few. For an exhaustive review of the different notions of *stability* and their consequences on the generalization

risk of a learning algorithms, the reader is referred to Kutin and Niyogi (2002). Our working assumption in this paper is *uniform stability*, which encompasses many algorithms such as Support Vector Machine (Bousquet and Elisse-eff, 2002), regularized empirical risk minimization (Zhang, 2004; Wibisono et al., 2009), stochastic gradient descent (Hardt et al., 2016) and neural networks with a simple architecture (Charles and Papailiopoulos, 2018).

Related Work on K-fold CV with Stable Learners. In Kale et al. (2011); Kumar et al. (2013), K-fold CV for risk estimation is envisioned under stability assumptions regarding the algorithm. It is shown that the K-fold risk estimate has a much smaller variance than the simple hold-out estimate and the amount of variance reduction is quantified. Another related work is Abou-Moustafa and Szepesvári (2017) who builds upon a variant of algorithmic stability, namely L^q-*stability* to derive PAC upper bounds for K-fold CV error estimates. Other results regarding the asymptotic behavior of K-fold estimates can be found in Austern and Zhou (2020); Bayle et al. (2020).

However, none of the results mentioned above imply a universal upper bound regarding the K-fold neither for risk estimation nor for model selection. Indeed their focus is on the the *variance term* of the K-fold error, while they do not take into account the high *bias* generally induced by this CV scheme (see Shao (1997); Arlot and Lerasle (2016) for instance). To our best knowledge, the literature on algorithmic stability is silent about the consistency of K-fold CV – the most widely used CV scheme – in a generic stability setting. Filling this gap is the main purpose of the present paper.

Contributions and Outline. We introduce the necessary background and notations about CV risk estimation and algorithmic stability in Section 2. Section 3 is intended to give some context about provable guarantees regarding K-fold CV scheme, namely we state and prove a generic upper bound on the error of the generalization risk estimate for uniformly stable algorithms. However, with realistic stability constants, the obtained upper bound is not satisfactory, in so far as it does not vanish as the sample size $n \to \infty$.

Our main contributions are gathered in sections 4 to 6 and may be summarized as follows:

1. One may wonder whether the looseness of the bound for the K-fold CV error is just an artifact from our proof. We answer in the negative by deriving a lower bound on the K-fold error (Section 4) in two different contexts, specifically, regularized empirical risk minimization and stochastic gradient optimization. The latter bound shows that under the uniform stability assumption alone, K-fold CV is inefficient in so far as it

can fail in estimating the generalization risk of a uniformly stable algorithm.

2. We analyze a corrected K-fold procedure and prove a PAC generalization upper bound covering the general case of uniformly stable learners. As a consequence, the corrected version of the K-fold is shown to be efficient in contrast to the standard version. The corrected K-fold scheme has been investigated in Burman (1989, 1990); Fushiki (2011); Arlot and Lerasle (2016) in the particular frameworks of ordinary linear regression and density estimation. Furthermore, the analysis in the latter references relies on strong regularity assumptions (further details are given in Section 5) which aren't satisfied by many modern learning rules like Support Vector Machine (SVM), stochastic gradient descent methods, bagging, etc. Instead our upper bound covers the general case of uniformly stable learners. As an example of application, we show that the debiased K-fold permits to select a model within a finite collection in a risk consistent manner (Section 6). In other words, the excess risk of the selected model tends to 0 as $n \to \infty$. Finally we demonstrate empirically the added value of the debiased K-fold compared with the standard one in terms of the test error of the selected model.

2 BACKGROUND, NOTATIONS AND WORKING ASSUMPTIONS

2.1 Notations

We place ourselves in the following general learning setting. One receives a collection of independent and identically distributed random vectors $\mathcal{D} = (O_1, \ldots, O_n)$ lying in a sample space $\mathcal{Z}$, with common distribution P. For any $n \in \mathbb{N}$, let $[n]$ denote the set of integers $\{1, 2, \ldots, n\}$. Consider a class of predictors $\mathcal{G}$ and a loss function $\ell : \mathcal{G} \times \mathcal{Z} \to \mathbb{R}$, so that $\ell(g, O)$ be the error of g on the observation $O \in \mathcal{Z}$. As an example, in the supervised learning setting $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$, g is a mapping $\mathcal{X} \to \mathcal{Y}$ and for $o = (x, y)$ the loss function writes as $\ell(g, o) = \ell(g(x), y)$. However our results are not limited to the supervised setting. Given a subsample $\mathcal{D}_T = \{O_i \mid i \in T\}$ indexed by $T \subset [n]$ and an algorithm (or learning rule) $\mathcal{A}$, we denote by $\mathcal{A}(T) \in \mathcal{G}$ the predictor obtained by training $\mathcal{A}$ on $\mathcal{D}_T$. We consider in this paper deterministic algorithms, that is, given a subsample $\mathcal{D}_T$, the output of the algorithm $\mathcal{A}(T)$ is non random. We thereby neglect the randomness brought *e.g.* by optimization routines. The case of random algorithms can be covered at the price of additional notational burden. For the sake of readability we restrict ourselves to deterministic algorithms in the main paper and show how to relax it in the supplementary material (Section D), in order to cover the case of random algorithms such as stochastic gradient descent (SGD) or neural networks. This extension

is in particular necessary to one of our counter-examples (Section 4.2) where we build a lower bound for the SGD algorithm.

The generalization risk of the predictor $\mathcal{A}(T)$ is then

$$\mathcal{R}\big[\mathcal{A}(T)\big] = \mathbb{E}\left[\ell\left(\mathcal{A}\left(T\right), O\right) \mid \mathcal{D}_T\right],$$

where O is independent from $\mathcal{D}_T$. Notice that the randomness in the latter expectation stems from the novel observation O only while the trained algorithm $\mathcal{A}(T)$ is fixed. The quantity of interest here is the generalization risk of the learning rule trained on the full dataset, $\mathcal{R}\big[\mathcal{A}([n])\big]$. The hold-out estimate of the latter involves a validation index set V disjoint from T and writes as

$$\widehat{\mathcal{R}}\big[\mathcal{A}(T), V\big] = \frac{1}{n_V} \sum_{i \in V} \ell(\mathcal{A}(T), O_i),$$

where $n_V = card(V)$.

Given a family of validation sets in $[n]$, $V_{1:K} = (V_j)_{j=1,\ldots,K}$, the K-fold CV estimator of the generalization risk of $\mathcal{A}([n])$ is

$$\widehat{\mathcal{R}}_{CV}\left[\mathcal{A}, V_{1:K}\right] = \frac{1}{K} \sum_{j=1}^{K} \widehat{\mathcal{R}}\big[\mathcal{A}(T_j), V_j\big], \qquad (2.1)$$

where $T_j = [n] \backslash V_j$. For clarity reasons, we suppose further that n is divisible by K so that n/K is an integer. This condition guarantees, that all validation sets have the same cardinal $n_V = n/K$.

2.2 Algorithmic Stability

An algorithm $\mathcal{A}$ is called stable if removing a training point O_i from $\mathcal{D}_T$ ($i \in T$) or replacing O_i with an independent observation O' drawn from the same distribution does not change much the risk of the output. Formally, for $i \in T \subset [n]$ as above, let $T^{\backslash i} = T \backslash \{i\}$, so that $\mathcal{A}(T^{\backslash i})$ is the output of $\mathcal{A}$ trained on $\mathcal{D}_T \backslash \{O_i\}$. Denote similarly $\mathcal{A}(T^i)$ the output of $\mathcal{A}$ trained on $\mathcal{D}_T \backslash \{O_i\} \cup \{O'\}$. The notion of *hypothesis stability* was first introduced in Devroye and Wagner (1979) to derive non asymptotic guarantees for the leave-one-out CV (*l.o.o.*). In this paper, we consider instead *uniform stability*, an assumption used in Bousquet and Elisseeff (2002); Wibisono et al. (2009); Hardt et al. (2016); Feldman and Vondrak (2019) to derive probability upper bounds for the training error and *l.o.o.* estimates. With the above notations, uniform stability is defined as follows.

Definition 2.1. An algorithm $\mathcal{A}$ is said to be $(\beta_t)_{1 \le t \le n}$ uniformly stable with respect to a loss function ℓ if, for any $T \subset [n], i \in T$ it holds that

$$\left|\ell\left(\mathcal{A}(T), O\right) - \ell\left(\mathcal{A}(T^{\backslash i}), O\right)\right| \le \beta_{n_T}, \qquad (2.2)$$

with P-probability one.

Many widely used Machine Learning algorithms are uniformly stable in the sense of Definition 2.1. In particular $\beta_n \le \frac{C}{n}$ for SVM and least square regression with the usual mean squared error, and for SVM classification with the soft margin loss (Bousquet and Elisseeff, 2002). Up to a minor definition of uniform stability accounting for randomness (see Section D in the supplement), many extensively used stochastic gradient methods are also uniformly stable, such as *e.g.* SGD with convex and non convex losses (Hardt et al., 2016) or RGD (randomized coordinate descent) and SVRG (stochastic variance reduced gradient method) with loss functions verifying the Polyak-Łojasiewicz condition (Charles and Papailiopoulos, 2018).

The following simple fact concerns the effect of removing n' training points on uniformly stable algorithms.

Fact 2.1. *Let $\mathcal{A}$ be a decision rule which is $(\beta_t)_{1 \le t \le n}$ uniformly stable, additionally suppose that the sequence $(\beta_t)_{1 \le t \le n}$ is decreasing, then for any $T \subset [n]$, one has,*

$$\left|\ell\left(\mathcal{A}\left([n]\right), O\right) - \ell\left(\mathcal{A}(T), O\right)\right| \le \sum_{i=n_T+1}^{n} \beta_i.$$

Remark 2.1. Definition 2.1 and Fact 2.1 play a key role in our proofs. Namely we use Fact 2.1 to control the bias of the CV risk estimate and Definition 2.1 to derive a probability upper bound on its deviations via McDiarmid's inequality.

We also rely on the fact that the training and validation sets of K-fold CV verify a certain balance condition.

Fact 2.2. *For the K-fold CV the validation sets $V_1, V_2, \ldots V_K$ satisfies*

$$card(V_j) = n_V \quad \forall j \in \llbracket 1, K \rrbracket, \qquad (2.3)$$

for some $n_V \in \llbracket 1, n \rrbracket$. Moreover it holds that

$$\frac{1}{K} \sum_{j=1}^{K} \mathbb{1}\{l \in V_j\} = \frac{n_V}{n} \quad \forall l \in [n]. \qquad (2.4)$$

Because $T_j = [n] \backslash V_j$, if (2.4) holds, then the training sets T_j verify a similar equation, that is,

$$\frac{1}{K} \sum_{j=1}^{K} \mathbb{1}\{l \in T_j\} = \frac{n_T}{n} \quad \forall l \in [n].$$

We prove Fact 2.2 in the supplement (Lemma A.1).

Throughout this paper we work under the following uniform stability assumption combined with a boundedness assumption regarding the cost function.

Assumption 1 (Stable algorithm). The algorithm $\mathcal{A}$ is $(\beta_t)_{1 \le t \le n}$ uniformly stable with respect to a cost function ℓ that satisfies

$$\forall O \in \mathcal{Z}, \ \forall T \subset [n], \ |\ell\left(\mathcal{A}\left(T\right), O\right)| \le L.$$

3 UPPER BOUNDS FOR K-FOLD RISK ESTIMATION

Our first result Theorem 3.1 is a generic upper bound on the error of the generalization risk estimate for stable algorithms satisfying Assumption 1. Our upper bound is of the same order of magnitude as existing results in the literature which apply to other contexts, *e.g.* in Bousquet and Elisseeff (2002) for *l.o.o.* CV or in Cornec (2017) for the specific case empirical risk minimizers, although our techniques of proof are different. The fact that the upper bound does not vanish with large sample sizes n with realistic stability constants (Corollary 3.1) gives some context and motivates the rest of this work.

Theorem 3.1. *Consider a stable learning algorithm $\mathcal{A}$ satisfying Assumption 1. Then, we have with probability $1 - 2\delta$,*

$$\left| \widehat{\mathcal{R}}_{CV}\left[\mathcal{A}, V_{1:K}\right] - \mathcal{R}\left[\mathcal{A}([n])\right] \right| \leq \sum_{i=n_T+1}^{n} \beta_i$$
$$+ (4\beta_{n_T} n_T + 2L) \sqrt{\frac{\log(1/\delta)}{2n}}.$$

Where L is the upper bound on the cost function ℓ from Assumption 1.

Sketch of proof. Define the average risk of the family $\left(\mathcal{A}(\mathcal{D}_{T_j})\right)_{1 \leq j \leq K}$

$$\mathcal{R}_{CV}\left[\mathcal{A}, V_{1:K}\right] = \frac{1}{K} \sum_{j=1}^{K} \mathcal{R}\left[\mathcal{A}(T_j)\right], \qquad (3.1)$$

then write the following decomposition

$$\widehat{\mathcal{R}}_{CV}\left[\mathcal{A}, V_{1:K}\right] - \mathcal{R}\left[\mathcal{A}([n])\right] = D_{CV} + \text{Bias}, \qquad (3.2)$$

with

$$D_{CV} = \widehat{\mathcal{R}}_{CV}\left[\mathcal{A}, V_{1:K}\right] - \mathcal{R}_{CV}\left[\mathcal{A}, V_{1:K}\right], \qquad (3.3)$$
$$\text{Bias} = \mathcal{R}_{CV}\left[\mathcal{A}, V_{1:K}\right] - \mathcal{R}\left[\mathcal{A}([n])\right]. \qquad (3.4)$$

The proof consists in bounding each term of the above decomposition independently. The term D_{CV} measure the deviations of $\widehat{\mathcal{R}}_{CV}$ from its mean and it can be controlled using McDiarmid's inequality (Proposition A.1). The second term Bias is controlled using Fact 2.1. The detailed proof is deferred to the appendix. □

As discussed in the background section 2.2, typical uniform stability constants β_n for standard algorithms satisfy $\beta_n \leq \frac{C}{n}$. In this case, Theorem 3.1 yields the following corollary.

Corollary 3.1. Consider a stable learning algorithm $\mathcal{A}$ satisfying Assumption 1 with stability parameter $\beta_n \leq \frac{C}{n}$.

Then, we have with probability $1 - 2\delta$,

$$\left| \widehat{\mathcal{R}}_{CV}\left[\mathcal{A}, V_{1:K}\right] - \mathcal{R}\left[\mathcal{A}([n])\right] \right| \leq C \log\left(\frac{K}{K-1}\right)$$
$$+ (4C + 2L)\sqrt{\frac{\log(1/\delta)}{2n}},$$

wich does not converge to 0 as $n \to \infty$.

Proof. Recall that $\frac{n}{n_T} = \frac{K}{K-1}$ and write

$$\sum_{i=n_T+1}^{n} \beta_i \leq C \sum_{i=n_T+1}^{n} \frac{1}{i} \leq C \log\left(\frac{K}{K-1}\right).$$

The result follows from Theorem 3.1. □

Remark 3.1 (Bias and Variance of K-fold CV). The two terms of the sum in the upper bound of Theorem 3.1 correspond respectively to a variance and a bias term, namely D_{CV} and Bias in the error decomposition (3.2).

On the one hand, the term $(4\beta_{n_T} n_T + 2L)\sqrt{\frac{\log(1/\delta)}{2n}}$ reflects the variance of the CV procedure. When $\beta_n \leq \frac{C}{n}$ it yields the usual rate $1/\sqrt{n}$. On the other hand, the term $\sum_{i=n_T+1}^{n} \beta_i$ reflects the bias of K-fold CV. Contrarily to the variance term, it does not vanish as $n \to \infty$, even when $\beta_n \leq C/n$. Finally, Notice that, as the number of folds K increases, the training size n_T gets closer to the sample size n and the bias of K-fold vanishes. However, for computational efficiency, increasing the number of folds is not always desirable.

One may wonder whether the looseness of the bias term $\sum_{i=n_T+1}^{n} \beta_i$ is just an artifact from our proof. In the next theorem we answer in the negative by deriving a lower bound for the K-fold. The latter bound shows that under the uniform stability assumption alone, K-fold CV is inefficient in so far as it can fail in estimating the generalization risk of a uniformly stable algorithm.

4 LOWER BOUND FOR THE K-FOLD ERROR UNDER ALGORITHMIC STABILITY

To construct lower bounds on K-fold CV error, we consider two families of algorithms that satisfy the uniform stability hypothesis with parameter β_n scaling as $1/n$. Namely, regularized empirical risk minimizers (RERM) and stochastic gradient descent (SGD).

4.1 Regularized Empirical Risk Minimization

The first counter-example that we build to prove a lower bound is formulated in a regression framework. In partic-

ular, we consider L^2-regularized empirical risk minimization (L^2-RERM)

$$\mathcal{A} = \arg\min_{g \in \mathcal{G}} \left\{ \frac{1}{n} \sum_{i=1}^{n} \ell\left(g, O_i\right) + \lambda_n \left\|g\right\|_{\mathcal{G}}^2 \right\}, \quad (4.1)$$

where ℓ is a convex loss function and $\mathcal{G}$ is a hypothesis space.

Under some mild assumptions on ℓ and the input space $\mathcal{Z}$, an L^2-RERM algorithm is uniformly stable with $\beta_n \leq \frac{C}{n}$ (see *e.g.* Zhang (2004); Wibisono et al. (2009); Liu et al. (2017) for further details).

The next result confirms that uniform stability alone is not sufficient to ensure the consistency of K-fold CV for RERM.

Theorem 4.1 (Non vanishing lower bound on the K-fold error). *Consider the regression problem for a random pair $O = (X, Y) \in \mathcal{Z} = \mathcal{X} \times \mathcal{Y}$, loss function $\ell(g, o) = (g(x) - y)^2$ and hypothesis class $\mathcal{G}$ consisting of linear predictors . Set $M \geq 1$ and $n \in [\![2, e^M]\!]$. Then, there exists an input space $\mathcal{Z}$ with distribution P and a regularization parameter λ_n, such that the RERM algorithm $\mathcal{A}$ (Equation 4.1) satisfies Assumption 1 with $L = M$ and $\beta_n \leq 2/n$. Furthermore the K-fold CV error satisfies,*

$$\mathbb{E}\left[\left|\widehat{\mathcal{R}}_{CV}\left[\mathcal{A}, V_{1:K}\right] - \mathcal{R}\left[\mathcal{A}([n])\right]\right|\right] \geq$$
$$2 \log\left(\frac{K}{K-1}\right)\left(1 - \frac{1}{M}\right),$$

and

$$\mathbb{E}\left[\left|\widehat{\mathcal{R}}_{CV}\left[\mathcal{A}, V_{1:K}\right] - \mathcal{R}\left[\mathcal{A}([n])\right]\right|\right] \leq 2 \log\left(\frac{K}{K-1}\right).$$

Sketch of proof. First set $\mathcal{Z} = \mathcal{X} \times \mathcal{Y} \subset \mathbb{R} \times \mathbb{R}$ and consider the hypothesis class of linear regressors,

$$\mathcal{G} = \left\{g_b \mid b \in \mathbb{R}\right\},$$

where $g_b(x) = bx$ for $x \in \mathbb{R}$. With the loss $\ell(g_b, o) = (y - bx)^2$, it can be shown that the solution to problem 4.1 writes as $\mathcal{A}([n]) = g_{b_n}$ with

$$b_n = f(\lambda_n), \quad (4.2)$$

for some decreasing function f. Now, by carefully picking $\mathcal{Z}$ and λ_n we construct a problem such that, for any $O \in \mathcal{Z}$ and $T \subset T' \subset [n]$

1. λ_n is decreasing.

2. $\ell\left(\mathcal{A}(T), O\right) \geq \ell\left(\mathcal{A}(T'), O\right)$.

3. $\ell\left(\mathcal{A}(T^{\backslash i}), O\right) - \ell\left(\mathcal{A}(T), O\right) \leq f\left(\lambda_n\right) - f\left(\lambda_{n-1}\right)$.

4. $f(\lambda_n) - f(\lambda_{n-1}) \leq \frac{2}{n}$.

The result easily follows from the three latter facts. For the detailed proof, the reader is deferred to Appendix C.2. □

4.2 Stochastic Gradient Descent

In this section, we consider the SGD update rule

$$\forall t \geq 0, \ \mathcal{A}_{t+1} = \mathcal{A}_t - \alpha_{t,n} \nabla_{\mathcal{A}} \ell\left(\mathcal{A}_t, X_{i_t}\right), \quad (4.3)$$

where $\nabla_{\mathcal{A}} \ell$ denotes the derivative of ℓ with respect to the first argument, i_t is the index picked by SGD at step t and $\alpha_{t,n} \geq 0$ is the step size.

It is well known that SGD verifies the uniform stability assumption with respect to many losses (Hardt et al., 2016; Liu et al., 2017; Charles and Papailiopoulos, 2018). For instance, when the loss function is convex, β smooth, and σ-Lipschitz, it has been shown that SGD algorithm (Hardt et al., 2016) is uniformly stable with parameter $\beta_n \leq \frac{2\sigma^2}{n} \sum_{k=1}^{t} \alpha_{k,n}$.

Remark 4.1. For a fixed data sequence $\mathcal{D}$, the output of SGD is random. Hence, the definition of *random uniform stability* is introduced in Appendix D and is slightly different than Definition 2.1. However, most if not all the properties of deterministic uniformly stable rules (discussed before) are preserved by random uniformly stable learners (see *e.g.* Elisseeff et al. (2005); Hardt et al. (2016); Liu et al. (2017)).

Theorem 4.2. *Let $M > 1$ be a real number , $2 \leq n \leq e^M$ an integer, and $t \geq 1$ a maximum number of iterations . Set the initialization $\mathcal{A}_0 = 0$. Then, There exists an input space $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$, a convex loss function ℓ and a sequence of step sizes $(\alpha_{k,n})_{k \leq t}$ such as,*

$$\mathbb{E}\left[\left|\widehat{\mathcal{R}}_{CV}\left[\mathcal{A}_t, V_{1:K}\right] - \mathcal{R}\left[\mathcal{A}_t([n])\right]\right|\right] \geq \frac{1}{3} \log\left(\frac{K}{K-1}\right).$$

Furthermore, $\mathcal{A}_t$ fulfills Assumption 1 with $L = M$ and

$$\beta_n \leq \frac{3}{n-1} \sum_{k=1}^{t} \alpha_{k,n} \leq \frac{3M}{n-1}.$$

Proof. The proof is deferred to the appendix (section D.2). □

Remark 4.2. In both examples, we suppose that the input is a binary random variable. Such a restrictive setting serves as a corner case to derive lower bounds (Bousquet et al., 2020; Zhang et al., 2022), suggesting the necessity of additional assumptions to ensure the consistency of K-fold scheme.

Remark 4.3 (BOUNDEDNESS OF n). With the current assumptions, the lower bounds don't ensure the inconsistency as the sample size n grows to infinity. However, Theorems 4.1 and 4.2 in their current state prove that, one cannot obtain a standard universal vanishing upper bound on the expected estimation error of K-fold ($\mathbb{E}[\text{error}_{CV}]$) which would be valid for all sample spaces ($\mathcal{Z}$), all distributions

(P) and all stable algorithms. In other words, one cannot construct a function $h(n)$ such that

$$\exists n_0 \in \mathbb{N} \,, \; \forall n \geq n_0 \,, \; \forall (\mathcal{Z}, P),$$

$$\forall \mathcal{A} \text{ stable with parameter } \beta_n \leq \frac{1}{n} \,, \; \mathbb{E}\left[\text{error}_{\text{CV}}\right] \leq h(n),$$

and $h(n) \xrightarrow[n \to \infty]{} 0$.

An additional remark that ought to be made: if we relax Assumption 1 into : "the loss function $\ell(\mathcal{A}([n]), \cdot)$ is $L \log(n)$-bounded and the algorithm $\mathcal{A}$ is $\frac{\log(n)}{n}$ uniformly stable", then all the upper bounds from the present paper remain valid (up to a $\log(n)$ factor), in particular the upper bound on the bias-corrected K-fold estimation error (Corollary 5.1 below). Furthermore, under this new assumption, the lower bound in Theorem 4.2 is valid $\forall n \in \mathbb{N}$, which yields the inconsistency of K-fold CV. More precisely, Theorem 4.2 becomes

$$\exists (\mathcal{Z}, P), \forall n_0 \in \mathbb{N} \,, \; \forall n \geq n_0 \,,$$

$$\exists \mathcal{A} \text{ stable with parameter } \beta_n \leq \frac{\log(n)}{n}, \; \mathbb{E}\left[\text{error}_{\text{CV}}\right] \geq \kappa,$$

with $\kappa = \frac{1}{3} \log\left(\frac{K}{K-1}\right)$. One must note that the additional $\log(n)$ factor in the upper bound weakens the tightness of the lower bound, however this is compatible with existing lower bounds from the stability literature (Bousquet et al., 2020).

Discussion. *Theorems 4.1 and 4.2 reveals the suboptimality of K-fold CV in the general stability framework for risk estimation. For the purpose of model selection, which is arguably a harder problem, existing works have shown the suboptimality of K-fold CV. For example, in a regression framework, under the linear model assumption, Yang (2007) (See also Yang (2006) for classification problems) has shown that the usual K-fold procedure may not select the best model. For efficient model selection, where the performance is measured in terms of generalization risk of the selected model, Arlot (2008) shows (See Theorem 1 in the latter reference) that K-fold CV can be suboptimal, i.e. an example is provided where the risk ratio between the selected model and the optimal one is uniformly greater (for all n) than $1 + \kappa$ for some $\kappa > 0$. In Theorem 4.1 (resp. 4.2) of the present paper we show a stronger result, in so far as we consider the easier task of risk estimation, and we show that K-fold CV with fixed K does not enjoy sanity-check guarantees because for any n, there exists a regression (resp. optimization) problem where the error is at least $2 \log(\frac{K}{K-1})$ (resp. $\frac{1}{3} \log \frac{K}{K-1}$). It is worth mentioning at this stage that even if the uniform stability ensures the low variance of K-fold (Kumar et al. (2013); Bayle et al. (2020), etc.), it is not sufficient to control the bias.*

In the next section, we show that adding correction terms (Burman, 1989) to K-fold CV addresses the inconsistency

issues underlined by Theorems 4.1 and 4.2. The resulting CV scheme enjoys both the computational efficiency of the K-fold (compared with the *l.o.o.*) and finite sample guarantees comparable to those of the *l.o.o.* in view of the upper bound stated in Theorem 5.1.

5 BIAS CORRECTED K-FOLD WITH STABLE LEARNERS

A key ingredient of the lack of guarantee regarding the K-fold risk estimate is its bias for finite sample sizes, an issue pointed out by Burman (1989) who proposes a corrected version of the standard K-fold with a reduced bias, see also Burman (1990) for applications to model selection. In the present work we follow in the footsteps of Burman (1989) and consider the same corrected CV estimate of the generalization risk

$$\begin{aligned}\widehat{\mathcal{R}}_{CV}^{corr}\left[\mathcal{A}, V_{1:K}\right] &= \widehat{\mathcal{R}}_{CV}\left[\mathcal{A}, V_{1:K}\right] \\ &+ \widehat{\mathcal{R}}\left[\mathcal{A}([n]), [n]\right] - \frac{1}{K}\sum_{j=1}^{K}\widehat{\mathcal{R}}\left[\mathcal{A}(T_j), [n]\right].\end{aligned} \tag{5.1}$$

The correcting term (second line in the above display) is the average difference between $\mathcal{R}\left[\mathcal{A}([n])\right]$ and the empirical risks $\mathcal{R}\left[\mathcal{A}(T_j)\right]$'s. In Burman (1989) the analysis is carried out in an asymptotic framework and focuses on the asymptotic bias and variance of the estimator for different CV schemes. The results are obtained under differentiability assumptions regarding the loss function (See also the appendix section in Fushiki (2011) where these assumptions are explicitly listed) which are typically not satisfied by SVM, stochastic gradient methods or L^1-regularized risk minimization algorithms. In Burman (1990) the working assumption is that of a linear model and the specific task is ordinary linear regression. In contrast, we conduct here a non asymptotic analysis (valid for any sample size) which applies to any uniformly stable algorithm.

Theorem 5.1. *Suppose that Assumption 1 holds. Then, we have, with probability $1 - 6\delta$,*

$$\left|\widehat{\mathcal{R}}_{CV}^{corr}\left[\mathcal{A}, V_{1:K}\right] - \mathcal{R}\left[\mathcal{A}([n])\right]\right| \leq 2\left(\beta_n + \beta_{n_T}\right)$$

$$+ 3(4\beta_{n_T}n_T + 2L)\sqrt{\frac{\log(1/\delta)}{2n}}.$$

Sketch of proof. By simple algebra write the corrected CV estimator as

$$\begin{aligned}\widehat{\mathcal{R}}_{CV}^{corr}\left[\mathcal{A}, V_{1:K}\right] &= \widehat{\mathcal{R}}\left[\mathcal{A}([n]), [n]\right] \\ &+ \frac{n_T}{nK}\sum_{j=1}^{K}\left[\widehat{\mathcal{R}}\left[\mathcal{A}(T_j), V_j\right] - \widehat{\mathcal{R}}\left[\mathcal{A}(T_j), T_j\right]\right].\end{aligned}$$

From this, deduce the following error decomposition

$$\widehat{\mathcal{R}}_{CV}^{corr}\left[\mathcal{A}, V_{1:K}\right] - \mathcal{R}\left[\mathcal{A}[n]\right] = D_{\mathcal{A}[n]}$$
$$+ \frac{n_T}{n}(D_{CV} - D_{\mathcal{A}[T_{1:K}]}), \quad (5.2)$$

where D_{CV} is defined in (3.3) and

$$D_{\mathcal{A}[n]} = \widehat{\mathcal{R}}\left[\mathcal{A}[n], [n]\right] - \mathcal{R}\left[\mathcal{A}[n]\right], \quad (5.3)$$

$$D_{\mathcal{A}[T_{1:K}]} = \frac{1}{K}\sum_{j=1}^{K}\left(\widehat{\mathcal{R}}\left[\mathcal{A}(T_j), T_j\right] - \mathcal{R}\left[\mathcal{A}(T_j)\right]\right). \quad (5.4)$$

Notice the absence of a bias term in the above display, contrarily to the error decomposition (3.2) for the standard CV estimate in the proof of Theorem 3.1. The remaining technical arguments for bounding each term of the decomposition above are gathered in the supplement. Namely, using McDiarmid's inequality (Proposition A.1), we derive concentration bound both for D_{CV} as stated in Lemma B.1 and for $D_{\mathcal{A}[T_{1:K}]}$ as given in Lemma B.2. Finally the deviations of the empirical risk $D_{\mathcal{A}[n]}$ are controlled using Remark B.1, a consequence of Lemma B.2, in the supplement. $\square$

Theorem 5.1 yields immediately a consistent upper bound for the K-fold CV error when $\beta_n \leq C/n$ for some $C > 0$. For the proof it suffices to notice that $n_T = \frac{K-1}{K}n$ for the K-fold scheme and to use that $(2K-1)/(K-1) \leq 3$ whenever $K \geq 2$.

Corollary 5.1. *Suppose that Assumption 1 holds with $\beta_n \leq \frac{C}{n}$, for some $C > 0$. Then, for $K \geq 2$, the error of the corrected K-fold CV estimate of the generalization risk satisfies with probability at least $1 - 6\delta$,*

$$\left|\widehat{\mathcal{R}}_{CV}^{corr}\left[\mathcal{A}, V_{1:K}\right] - \mathcal{R}\left[\mathcal{A}([n])\right]\right| \leq \frac{6C}{n}$$
$$+ 3(4C + 2L)\sqrt{\frac{\log(1/\delta)}{2n}}.$$

Discussion. *Corollary 5.1 confirms the relevance of the bias corrected K-fold for risk estimation in the broad context of uniformly stable learners satisfing $\beta_n \leq \frac{C}{n}$, such as SVM, stochastic gradient methods or regularized empirical risk minimizers.*

Our main results Theorem 5.1 and Corollary 5.1 concern the problem of risk estimation by means of K-fold CV. However in practice, CV is widely used in the context of model- or parameter *selection*. It is precisely the purpose of the next section to illustrate how our results shed light on such practice. Namely we consider the practical problem of selecting a penalty parameter among a finite collection of candidates for Support Vector Regression and Classification.

6 APPLICATION TO HYPER-PARAMETER SELECTION AND NUMERICAL EXPERIMENTS

Selecting a penalty parameter in regularized empirical risk minimization algorithms such as SVM's may be viewed as a particular instance of a model selection problem, where one identifies a model with an algorithm equipped with a particular choice of regularization parameter. We start-off this section with a brief introduction to this topic. We provide minimal theoretical guarantees (Proposition 6.1) regarding model selection within a finite collection of models by means of the debiased K-fold procedure. We describe our experimental setting in Subsections 6.2, 6.3 and we report our results in Section 6.4.

6.1 Cross-Validation for Parameter Selection

Following the terminology of Arlot and Lerasle (2016), we consider here the problem of *efficient* model selection, aiming at selecting a model for which the generalization risk of the learnt predictor is close to the smallest possible risk. The goal of efficient model selection is in general easier to attain than *identification* of the best possible model. It is a known fact that cross-validation is in general sub-optimal for model *identification* purpose. A major reason for this is that different models (or algorithms) may have comparable performance. However if one aims only at selecting a model for which the generalization risk is close to that of the optimal one, lack of identifiability is not necessarily an issue anymore, which is precisely the approach we take here.

Model selection is a prominent topic in statistical learning theory which is by far too broad to be extensively covered here. We refer the reader to the monograph of Massart (2007). Cross-validation is one of several possible candidate methods for this problem, which has been the subject of a wealth of literature as discussed in the introduction, see also Arlot and Celisse (2010) for a review.

Given a family of models (or algorithms) $\mathcal{A}^{(m)}$ indexed by $m \in \mathcal{M}$ and a dataset $\mathcal{D}$ of size n, an optimal model $\mathcal{A}^{(m^*)}$ called an oracle is any model such that

$$\mathcal{A}^{(m^*)} \in \arg\min_{m \in \mathcal{M}} \mathcal{R}\left[\mathcal{A}^{(m)}([n])\right]. \quad (6.1)$$

Since the true risk is unknown, an empirical criterion must be used instead to select an efficient model. One of the most popular tools for model selection is K-fold CV. However, as discussed earlier, the latter procedure may not be consistent. To tackle this issue, we propose to use the corrected K-fold and select a model $\mathcal{A}^{\widehat{m}}$ such that

$$\mathcal{A}^{(\widehat{m})} \in \arg\min_{m \in \mathcal{M}} \widehat{\mathcal{R}}_{\text{Kfold}}^{corr}\left[\mathcal{A}^{(m)}, V_{1:K}\right]. \quad (6.2)$$

Proposition 6.1 provides a sanity check guarantee regarding the consistency of the corrected K-fold CV procedure for this purpose, in the form of an upper bound in probability for the excess risk $\mathcal{R}\left[\mathcal{A}^{(\hat{m})}[n]\right] - \mathcal{R}\left[\mathcal{A}^{(m^*)}([n])\right]$. The result applies to the case where the family of models $\mathcal{M}$ is finite.

Proposition 6.1. *Let $(\mathcal{A}^{(m)})_{m \in \mathcal{M}}$ be a family of algorithms where each learner $\mathcal{A}^{(m)}$ is $(\beta_{m,t})_{1 \leq t \leq n}$ uniformly stable with respect to a loss function $0 \leq \ell(g, O) \leq L$. Additionally, assume that,*

$$\forall m \in \mathcal{M} \, ; \, \beta_{m,t} \leq \frac{M}{t},$$

for some $M > 0$. Then one has, with probability at least $1 - 6\delta$,

$$\mathcal{R}\left[\mathcal{A}^{(\hat{m})}([n])\right] - \mathcal{R}\left[\mathcal{A}^{(m^*)}([n])\right] \leq \frac{12M}{n}$$
$$+ 6(4M + 2L)\sqrt{\frac{\log(|\mathcal{M}|/\delta)}{n}}.$$

Proof. The proof consists in applying a union bound combined with the exponential tail bound of Theorem 5.1, which yields a multiplicative constant which only depends logarithmically on the number of models. The details are gathered in the supplementary material, Section C.1. □

Discussion (bounded stability parameters $\beta_{m,t}$'s)**.** *The assumption $\beta_{m,t} \leq \frac{M}{t}$ for all m is indeed verified in many applications. For example, in regularized SVM, where each learner $\mathcal{A}^{(m)}$ is trained using a regularization parameter λ_m, Bousquet and Elisseeff (2002) show that $\beta_{m,t} = \frac{1}{t}\sqrt{\frac{C}{\lambda_m}}$ where C is a positive constant. Thus, if one performs a grid search for λ_m on a grid $[a, b]$ with $a > 0$, then $\beta_{m,t} \leq \frac{1}{t}\sqrt{\frac{C}{a}}$ for any $\lambda_m \in [a, b]$. In other words, since the search space is generally bounded, the boundedness assumption regarding the $\beta_{m,t}$'s is not too restrictive in practice.*

6.2 Support Vector Machines and Experimental Setting

The aim of our experiments is to illustrate empirically the added value of the corrected K-fold compared with the standard one in terms of efficiency in model selection. In other words, we perform model selection with the K-fold and the corrected version that we promote and we compare the generalization risks of the selected trained models.

We consider a finite family of SVM regressors and classifiers trained with a regularization parameter λ_m ranging in a finite grid in an interval $[a, b]$, where $m \in \mathcal{M}$ a finite index set. Namely we set $[a, b] = [0.1, 100]$, and the grid is constructed with a constant step size equal to $\Delta = 0.1$, so that

$$(\lambda_m)_{m \in \mathcal{M}} = \{a + j\Delta \mid 0 \leq j < 1000\}.$$

In this SVM framework,

$$\mathcal{A}^{(m)}(T) = \arg\min_{f \in \mathcal{F}} \frac{1}{n_T} \sum_{i \in T} \ell(f, O_i) + \lambda_m \|f\|_k^2$$

where $\mathcal{F}$ is a reproducing kernel Hilbert space with kernel k. The kernel k is chosen here as the sigmoid kernel $\tanh(\tau\langle x, \rangle)$. Following standard practice we set $\tau = \frac{1}{d}$, where d is the dimension of the dataset. We use the quadratic loss for regression problems and the hinge loss $\ell(g, (x, y)) = (1 - yg(x))_+$ for classification, where $(f(x))_+ = max(0, f(x))$. Since the training datasets are bounded, we may consider that both these losses as bounded as well. For both these losses the algorithm $\mathcal{A}^{(m)}$ is $\frac{C}{n}$ uniformly stable (see Bousquet and Elisseeff (2002) for further details). The assumptions of Theorem 6.1 are thus satisfied, as pointed out in the discussion following the theorem's statement.

6.3 Datasets

Eight reference datasets from UCI [1] are considered: four classification datasets and four regression datasets listed below.

Regression datasets *Real estate valuation* (REV, 414 house price of unit area with 5 covariates); *QSAR fish toxicity* (906 toxic chemical concentration with 6 attributes); *Energy efficiency* (EE, 768 heating loads with 8 features.); *Concrete Compressive Strength* (CS, 1030 measure of the compressive strength with 8 attributes).

Classification datasets *Ionosphere dataset* (IO, 351 radar returns with 34 attributes), *Raisin dataset* (RS, 900 Keciman/Besni raisin with 7 attributes), *Audit risk dataset* (AR, 777 firm evaluation (fraudulent/non fraudulent) with 18 risk factors) and *QSAR bio degradation Data Set* (BIODEG, 1055 chemicals categorization with 12 descriptive features).

For each dataset one third of the data are removed ($\mathcal{S}$) and reserved for testing, *i.e.* for evaluating the generalization risk of the model selected using the remaining two thirds ($\mathcal{D}$). Let $\hat{m}_{\text{Kf}}$ (*resp.* $\hat{m}_{\text{Kf-corr}}$) denote the model selected using K-fold CV (*resp.* corrected K-fold CV) on the train set $\mathcal{D}$. In other words

$$\mathcal{A}^{(\hat{m}_{\text{Kf-corr}})} = \arg\min_{m \in \mathcal{M}} \widehat{\mathcal{R}}_{\text{Kfold}}^{corr}\left[\mathcal{A}^{(m)}, V_{1:K}\right],$$
$$\mathcal{A}^{(\hat{m}_{\text{Kf}})} = \arg\min_{m \in \mathcal{M}} \widehat{\mathcal{R}}_{\text{Kfold}}\left[\mathcal{A}^{(m)}, V_{1:K}\right].$$

In the end, in line with Section 6, the performance of both models are compared in terms of the mean squared error (or hinge loss for classification) and its estimated standard deviation on the test set $\mathcal{S}$.

[1] https://archive.ics.uci.edu

6.4 Numerical Results

We use the implementation provided by the python library `scikit-learn`. Tables 1 and 2 gather the results obtained respectively with the regression and classification datasets, for different numbers of folds K varying between 3 and 5. In all cases, the model selected by the bias corrected K-fold has a lower generalization risk than the one selected by the standard K-fold. As expected, the standard K-fold procedure behaves generally better with $K = 5$ than with $K = 3$. Indeed larger values of K decrease the bias of the K-fold CV. The benefit of the bias correction is thus all the more important for small values of K.

Table 1: Regression mean squared errors for the K-fold and the bias corrected K-fold on various data sets. Estimated standard deviations are reported between parentheses.

DATASET	K-FOLD	BIAS CORRECTED K-FOLD
REV; K=3	74.198 (12.57)	**68.958 (11.63)**
K=4	74.189 (12.48)	**68.958 (11.63)**
K=5	73.359 (12.38)	**68.958 (11.63)**
EE; K=3	15.501 (1.81)	**14.405 (1.86)**
K=4	15.825 (1.84)	**14.350 (1.77)**
K=5	14.730 (1.73)	**14.298 (1.79)**
QSAR; K=3	1.183 (0.16)	**1.035 (0.14)**
K=4	1.112 (0.15)	**1.035 (0.14)**
K=5	1.112 (0.15)	**1.035 (0.14)**
CS; K=3	146.881 (13.81)	**126.492 (10.23)**
K=4	144.195 (13.16)	**124.205 (9.46)**
K=5	137.060 (11.48)	**123.641 (9.30)**

Table 2: Hinge losses for the K-fold and the bias corrected K-fold on various data sets. Estimated standard deviations are reported between parentheses.

DATASET	K-FOLD	BIAS CORRECTED K-FOLD
RS; K=3	0.470 (0.048)	**0.419 (0.038)**
K=4	0.420 (0.039)	**0.418 (0.039)**
K=5	0.420 (0.039)	**0.419 (0.038)**
IO; K=3	0.454 (0.081)	**0.414 (0.072)**
K=4	0.447 (0.092)	**0.425 (0.072)**
K=5	0.477 (0.091)	**0.464 (0.095)**
BIODEG; K=3	0.361 (0.039)	**0.357 (0.036)**
K=4	0.363 (0.037)	**0.357 (0.036)**
K=5	0.381 (0.041)	**0.357 (0.036)**
AR; K=3	0.112 (0.023)	**0.109 (0.021)**
K=4	0.108 (0.027)	**0.105 (0.025)**
K=5	0.107 (0.025)	**0.106 (0.024)**

7 Conclusion

This paper demonstrates the limitations of the standard K-fold procedure for risk estimation *via* a lower bound on its error with a uniformly stable learner. We show that the corrected version of the K-fold for uniformly stable algorithms does not suffer the same drawbacks through a sanity-check upper bound and leverage this result to obtain guarantees regarding efficient model selection. This paves the way towards two possible research directions. A relevant follow-up would be to relax our uniform stability assumption in order to cover still a wider class of algorithms such as k-nearest-neighbors (Devroye and Wagner, 1979), Adaboost (Freund and Schapire, 1997) and Lasso regression (Celisse and Guedj, 2016). A second promising avenue would be to consider an extension of the K-fold *penalization* proposed by Arlot and Lerasle (2016) to the class of stable learners.

References

K. Abou-Moustafa and C. Szepesvári. An a priori exponential tail bound for k-folds cross-validation. *arXiv preprint*, 2017.

S. Arlot. V-fold cross-validation improved: V-fold penalization. 40 pages, plus a separate technical appendix., Feb. 2008.

S. Arlot and A. Celisse. A survey of cross-validation procedures for model selection. *Statistics surveys*, 4:40–79, 2010.

S. Arlot and M. Lerasle. Choice of v for v-fold cross-validation in least-squares density estimation. *Journal of Machine Learning Research*, 17(208):1–50, 2016.

M. Austern and W. Zhou. Asymptotics of cross-validation. *arXiv preprint*, 2020.

P. Bayle, A. Bayle, L. Janson, and L. Mackey. Cross-validation confidence intervals for test error. In H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 16339–16350, 2020.

A. E. Bilgrau, C. F. Peeters, P. S. Eriksen, M. Bøgsted, and W. N. van Wieringen. Targeted fused ridge estimation of inverse covariance matrices from multiple high-dimensional data classes. *J. Mach. Learn. Res.*, 21(26): 1–52, 2020.

O. Bousquet and A. Elisseeff. Algorithmic stability and generalization performance. In T. Leen, T. Dietterich, and V. Tresp, editors, *Advances in Neural Information Processing Systems*, volume 13. MIT Press, 2001.

O. Bousquet and A. Elisseeff. Stability and generalization. *The Journal of Machine Learning Research*, 2:499–526, 2002.

O. Bousquet, Y. Klochkov, and N. Zhivotovskiy. Sharper bounds for uniformly stable algorithms. In *Conference on Learning Theory*, pages 610–626. PMLR, 2020.

P. Burman. A comparative study of ordinary cross-validation, v-fold cross-validation and the repeated learning-testing methods. *Biometrika*, 76(3):503–514, 1989.

P. Burman. Estimation of optimal transformations using v-fold cross validation and repeated learning-testing methods. *Sankhyā: The Indian Journal of Statistics, Series A*, pages 314–345, 1990.

A. Celisse and B. Guedj. Stability revisited: new generalisation bounds for the leave-one-out, 2016.

Z. Charles and D. Papailiopoulos. Stability and generalization of learning algorithms that converge to global optima. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80, pages 745–754. PMLR, 10–15 Jul 2018.

M. Cornec. Concentration inequalities of the cross-validation estimator for empirical risk minimizer. *Statistics*, 51(1):43–60, 2017.

L. Devroye and T. Wagner. Distribution-free performance bounds for potential function rules. *IEEE Transactions on Information Theory*, 25(5):601–604, 1979.

A. Elisseeff, T. Evgeniou, and M. Pontil. Stability of randomized learning algorithms. *Journal of Machine Learning Research*, 6(3):55–79, 2005.

V. Feldman and J. Vondrak. High probability generalization bounds for uniformly stable algorithms with nearly optimal rate. In *Conference on Learning Theory*, pages 1270–1279. PMLR, 2019.

Y. Freund and R. E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55(1):119–139, 1997.

T. Fushiki. Estimation of prediction error by using k-fold cross-validation. *Statistics and Computing*, 21(2):137–146, 2011.

M. Hardt, B. Recht, and Y. Singer. Train faster, generalize better: Stability of stochastic gradient descent. In *Proceedings of The 33rd International Conference on Machine Learning*, volume 48, pages 1225–1234. PMLR, 20–22 Jun 2016.

S. Kale, R. Kumar, and S. Vassilvitskii. Cross-validation and mean-square stability. In *In Proceedings of the Second Symposium on Innovations in Computer Science (ICS2011*. Citeseer, 2011.

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

R. Kumar, D. Lokshtanov, S. Vassilvitskii, and A. Vattani. Near-optimal bounds for cross-validation via loss stability. In *International Conference on Machine Learning*, pages 27–35. PMLR, 2013.

S. Kutin and P. Niyogi. Almost-everywhere algorithmic stability and generalization error. In *Proceedings of the Eighteenth Conference on Uncertainty in Artificial Intelligence*, page 275–282, 2002.

T. Liu, G. Lugosi, G. Neu, and D. Tao. Algorithmic stability and hypothesis complexity. In *International Conference on Machine Learning*, pages 2159–2167. PMLR, 2017.

P. Massart. *Concentration inequalities and model selection: Ecole d'Eté de Probabilités de Saint-Flour XXXIII-2003*. Springer, 2007.

C. McDiarmid. Concentration. In M. Habib, C. McDiarmid, J. Ramirez-Alfonsen, and B. Reed, editors, *Probabilistic Methods for Algorithmic Discrete Mathematics*, pages 195–248. 1998.

J. Shao. An asymptotic theory for linear model selection. *Statistica Sinica*, 7(2):221–242, 1997.

M. Stone. Cross-validatory choice and assessment of statistical predictions. *Journal of the Royal Statistical Society. Series B (Methodological)*, 36(2):111–147, 1974.

R. J. Tibshirani and J. Taylor. The solution path of the generalized lasso. *The annals of statistics*, 39(3):1335–1371, 2011.

S. Wager. Cross-validation, risk estimation, and model selection: Comment on a paper by rosset and tibshirani. *Journal of the American Statistical Association*, 115(529):157–160, 2020.

A. Wibisono, L. Rosasco, and T. Poggio. Sufficient conditions for uniform stability of regularization algorithms. 2009.

Y. Yang. Comparing learning methods for classification. *Statistica Sinica*, 16(2):635–657, 2006.

Y. Yang. Consistency of cross validation for comparing regression procedures. *The Annals of Statistics*, 35(6):2450 – 2473, 2007.

T. Zhang. Statistical behavior and consistency of classification methods based on convex risk minimization. *The Annals of Statistics*, 32(1):56–85, 2004.

Y. Zhang, W. Zhang, S. Bald, V. Pingali, C. Chen, and M. Goswami. Stability of sgd: Tightness analysis and improved bounds. In *Uncertainty in Artificial Intelligence*, pages 2364–2373. PMLR, 2022.

A MAIN TOOLS

First we recall McDiarmid's inequality (Theorem 3.1 in McDiarmid (1998)).

Proposition A.1 (McDiarmid's inequality). *let* $Z = f(\mathcal{D})$ *for some measurable function* f *and define*

$$\Delta_l(\mathcal{D}, O') = f(\mathcal{D}) - f(\mathcal{D}^l),$$

where $\mathcal{D}^l$ *is obtained by replacing the* l*'th element of* $\mathcal{D}$ *by a sample* $O' \in \mathcal{Z}$. *In addition, suppose that*

$$\forall l \in [n], \ \sup_{\mathcal{D} \in \mathcal{Z}^n} \sup_{O' \in \mathcal{Z}} |\Delta_l(\mathcal{D}, O')| \leq c_l.$$

Then for any $t \geq 0$,

$$\mathbb{P}(Z - \mathbb{E}(Z) \geq t) \leq \exp\left(\frac{-2t^2}{\sum_{l=1}^n c_l^2}\right).$$

The following lemma guarantees that Fact 2.2 is verified for K-fold CV.

Lemma A.1. *For the* K*-fold procedure, the training samples* $T_{1:K}$ *and validation samples* $V_{1:K}$ *satisfy Fact 2.2 i.e*

$$\frac{1}{K}\sum_{j=1}^K \frac{\mathbb{1}\{l \in T_j\}}{n_T} = \frac{1}{K}\sum_{j=1}^K \frac{\mathbb{1}\{l \in V_j\}}{n_V} = \frac{1}{n} \quad \forall l \in [\![1, n]\!].$$

Proof. The validation sets of K-fold verify the following property

$$\bigcup_{j=1}^K V_j = [\![1, n]\!] \ and \ V_j \bigcap V_k = \varnothing, \ \forall j \neq k \in [\![1, K]\!]. \tag{A.1}$$

Under the hypothesis that $card(V_j) = n_V$ for all the validation sets, (A.1) implies that

$$n = \sum_{j=1}^K card(V_j) = K n_V. \tag{A.2}$$

Furthermore, under (A.1), an index $l \in [\![1, n]\!]$ belongs to a unique validation test V_j' and to all the train sets $T_j = V_j^c$ with $j \neq j'$. Hence, we have

$$\begin{cases} \sum_{j=1}^K \mathbb{1}\{l \in T_j\} = K - 1, \\ \sum_{j=1}^K \mathbb{1}\{l \in V_j\} = 1. \end{cases}$$

Using (A.1) and the fact that $n_T = n - n_V = (K-1)n_V$ yields the desired result.

$\square$

B INTERMEDIATE RESULTS

First we provide a concentration bound for D_{CV} which has been defined in (3.3) as

$$D_{CV} = \widehat{\mathcal{R}}_{CV}[\mathcal{A}, V_{1:K}] - \mathcal{R}_{CV}[\mathcal{A}, V_{1:K}].$$

Lemma B.1. *Suppose that Assumption 1 holds. Then, we have*

$$\mathbb{P}(|D_{CV}| \geq t) \leq \exp\left(\frac{-2nt^2}{(4\beta_{n_T} n_T + 2L)^2}\right).$$

Proof. Let $O' \in \mathcal{X}$ be an independent copy of $O_1, O_2, \ldots, O_n$, for any $l \in [\![1, n]\!]$ define

$$\Delta_l(\mathcal{D}, O') = \big| D_{CV}(\mathcal{D}) - D_{CV}(\mathcal{D}^l) \big|,$$

where $\mathcal{D}^l$ is obtained by replacing the l'th element of $\mathcal{D}$ by O'. Now we derive an upper bound on $\mathbb{P}(|D_{CV}| \geq t)$ using Proposition A.1. Namely, we will bound the maximum deviation of Δ_l by $\Delta_l \leq \frac{4\beta_{n_T} n_T}{n} + \frac{L}{n}$. To do so, write

$$
\begin{aligned}
D_{CV} &= \widehat{\mathcal{R}}_{CV}\left[\mathcal{A}, V_{1:K}\right] - \mathcal{R}_{CV}\left[\mathcal{A}, V_{1:K}\right] \\
&= \frac{1}{Kn_{val}} \sum_{j=1}^{K} \sum_{i \in V_j} \left(\ell(\mathcal{A}(T_j), O_i) - \mathbb{E}_O\left[\ell(\mathcal{A}(T_j), O) \mid \mathcal{D}_{T_j}\right] \right) \\
&= \frac{1}{Kn_{val}} \sum_{j=1}^{K} \sum_{i \in V_j} h(\mathcal{A}(T_j), O_i),
\end{aligned}
$$

where the last is used to define h. For a training set $T_j \subset [n]$, let $\mathcal{A}(T_{j,l})$ denote the algorithm $\mathcal{A}$ trained on the sequence $\mathcal{D}_{T_j,l} = \{O_i \in \mathcal{D}^l \mid i \in T_j\}$. Note that, for all $\mathcal{D}_{T_j}, o \in \mathcal{Z}^{n_T} \times \mathcal{Z}$ one has

$$
\begin{cases}
\ell\big(\mathcal{A}(T_j), o\big) = \ell\big(\mathcal{A}(T_{j,l}), o\big) \text{ if } l \notin T_j, \\
\left| \ell\big(\mathcal{A}(T_j), o\big) - \ell\big(\mathcal{A}(T_{j,l}), o\big) \right| \leq 2\beta_{n_T} \text{ otherwise.}
\end{cases}
\tag{B.1}
$$

The first equation follows from the fact that $\mathcal{D}_{T_j,l} = \mathcal{D}_{T_j}$ if $l \notin T_j$, indeed, if the training set $\mathcal{D}_{T_j}$ doesn't contain the index l then changing the l'th element of $\mathcal{D}$ won't affect $\mathcal{D}_{T_j}$. The second inequality is obtained using the *uniform* stability of $\mathcal{A}$. Furthermore, using Equation B.1 write

$$
\begin{cases}
\mathbb{E}\left[\ell\big(\mathcal{A}(T_j), O\big) \mid \mathcal{D}_{T_j}\right] = \mathbb{E}\left[\ell\big(\mathcal{A}(T_{j,l}), O\big) \mid \mathcal{D}_{T_{j,l}}\right] \text{ if } l \notin T_j \\
\left| \mathbb{E}\left[\ell\big(\mathcal{A}(T_j), O\big) \mid \mathcal{D}_{T_j}\right] - \mathbb{E}\left[\ell\big(\mathcal{A}(T_{j,l}), O\big) \mid \mathcal{D}_{T_{j,l}}\right] \right| \leq 2\beta_{n_T}, \text{ otherwise.}
\end{cases}
\tag{B.2}
$$

Combining (B.2) and (B.1) gives

$$
\begin{cases}
|\mathbb{1}\{l \in T_j\}(h(\mathcal{A}(T_j), O_i)) - h(\mathcal{A}(T_{j,l}), O_i))| \leq 4\beta_{n_T}, \\
|\mathbb{1}\{l \notin T_j\}(h(\mathcal{A}(T_j), O_i)) - h(\mathcal{A}(T_{j,l}), O_i))| \leq 2L\mathbb{1}\{i = l\},
\end{cases}
\tag{B.3}
$$

so that

$$|\Delta_l(\mathcal{D}, O')| \leq \frac{1}{Kn_{val}} \sum_{j=1}^{K} \sum_{i \in V_j} |h(\mathcal{A}(T_j), O_i)) - h(\mathcal{A}(T_{j,l}), O_i))|$$

$$\text{(From the fact that } [n] \backslash T_j = V_j) \quad = \frac{1}{Kn_{val}} \sum_{j=1}^{K} \sum_{i \in V_j} |h(\mathcal{A}(T_j), O_i)) - h(\mathcal{A}(T_{j,l}), O_i))| \, (\mathbb{1}\{l \in T_j\} + \mathbb{1}\{l \in V_j\})$$

$$\text{(By Equation B.3)} \quad \leq \frac{4\beta_{n_T}}{K} \sum_{j=1}^{K} \mathbb{1}\{l \in T_j\} + \frac{2L}{n_{val}K} \sum_{j=1}^{K} \mathbb{1}\{l \in V_j\}$$

$$\text{(By Fact 2.2)} \quad \leq \frac{4\beta_{n_T} n_T}{n} + \frac{2L}{n}.$$

Using Mcdiarmid's inequality (Proposition A.1) gives

$$\mathbb{P}(D_{CV} \geq t) \leq \exp\left(\frac{-2nt^2}{(4\beta_{n_T} n_T + 2L)^2} \right).$$

Symmetrically, one has,

$$\mathbb{P}(D_{CV} \leq -t) \leq \exp\left(\frac{-2nt^2}{(4\beta_{n_T} n_T + 2L)^2} \right).$$

Thus,

$$\mathbb{P}(|D_{CV}| \geq t) \leq 2 \exp\left(\frac{-2nt^2}{(4\beta_{n_T} n_T + 2L)^2}\right),$$

which is the desired result. $\qquad\square$

In the next lemma we obtain a similar concentration bound for the term $D_{\mathcal{A}[T_{1:K}]}$ defined in Eq 5.4 as

$$D_{\mathcal{A}[T_{1:K}]} = \frac{1}{K} \sum_{j=1}^{K} \left(\widehat{\mathcal{R}}\left[\mathcal{A}(T_j), T_j\right] - \mathcal{R}\left[\mathcal{A}(T_j)\right]\right).$$

Lemma B.2. *Suppose that Assumption 1 holds. Then, one has,*

$$\mathbb{P}(|D_{\mathcal{A}[T_{1:K}]}| \geq t + 2\beta_{n_T}) \leq \exp\left(\frac{-2nt^2}{(4\beta_{n_T} n_T + 2L)^2}\right),$$

where L is the upper bound of the cost function defined in Assumption 1.

Proof. Though the proof bears resemblance with the one of Lemma B.1, we provide the full details details for compeleteness. We use McDiarmid's inequality with $f(\mathcal{D}) = D_{\mathcal{A}[T_{1:K}]}$, that is,

$$f(\mathcal{D}) = \frac{1}{K} \sum_{j=1}^{K} \left(\widehat{\mathcal{R}}\left[\mathcal{A}(T_j), T_j\right] - \mathcal{R}\left[\mathcal{A}(T_j)\right]\right)$$

$$f(\mathcal{D}^l) = \frac{1}{K} \sum_{j=1}^{K} \left(\widehat{\mathcal{R}}\left[\mathcal{A}(T_{j,l}), T_{j,l}\right] - \mathcal{R}\left[\mathcal{A}(T_{j,l})\right]\right).$$

Since for $l \notin T_j$, $T_j = T_{j,l}$, we find that

$$|\Delta_l(\mathcal{D}, O')| \leq \frac{1}{K n_T} \sum_{j=1}^{K} \mathbb{1}\{l \in T_j\} \left(\widehat{\mathcal{R}}\left[\mathcal{A}(T_j), T_j\right] - \widehat{\mathcal{R}}\left[\mathcal{A}(T_{j,l}), T_{j,l}\right] + \mathcal{R}\left[\mathcal{A}(T_{j,l})\right] - \mathcal{R}\left[\mathcal{A}(T_j)\right]\right)$$

$$\leq \frac{1}{K n_T} \sum_{j=1}^{K} \mathbb{1}\{l \in T_j\} \sum_{i \in T_j} |h(\mathcal{A}(T_j), O_i) - h(\mathcal{A}(T_{j,l}), O_i^l)|,$$

with $h(\mathcal{A}(T), o) = \ell(\mathcal{A}(T), o) - \mathbb{E}_O\left[\ell(\mathcal{A}(T), O) \mid \mathcal{D}_T\right]$ and $(O_i^l)_{i=1,\ldots,n}$ is the same as $(O_i)_{i=1,\ldots,n}$ except the l-th element, O_l, which is replaced by O'. Whenever $l \in T_j$, it holds that

$$|\ell(\mathcal{A}(T_j), O_i) - \ell(\mathcal{A}(T_{j,l}), O_i^l)| = |\ell(\mathcal{A}(T_j), O_i) - \ell(\mathcal{A}(T_{j,l}), O_i)|\mathbb{1}\{i \neq l\} + |\ell(\mathcal{A}(T_j), O_l) - \ell(\mathcal{A}(T_{j,l}), O')|\mathbb{1}\{i = l\}$$

$$\leq 2\beta_{n_T} + L\mathbb{1}\{i = l\},$$

and that

$$\mathbb{E}\left[|\ell(\mathcal{A}(T_j), O) - \ell(\mathcal{A}(T_{j,l}), O)| \mid \mathcal{D}_T, O'\right] \leq 2\beta_{n_T}.$$

It follows from the definition of h that

$$|h(\mathcal{A}(T_j), O_i) - h(\mathcal{A}(T_{j,l}), O_i^l)| \leq 4\beta_{n_T} + 2L\mathbb{1}\{i = l\}.$$

By using the identity $\frac{1}{K} \sum_{j=1}^{K} \mathbb{1}\{l \in T_j\} = \frac{n_T}{n}$ we get

$$\Delta_l(\mathcal{D}, O') \leq \frac{4\beta_{n_T} n_T}{n} + \frac{2L}{n}.$$

Thus by Mcdiarmid's (Proposition A.1), we obtain, for $Z = f(\mathcal{D}) - \mathbb{E}[f(\mathcal{D})]$, that

$$\mathbb{P}(Z \geq t) \leq \exp\left(\frac{-2nt^2}{(4\beta_{n_T} n_T + 2L)^2}\right).$$

Symmetrically, the event $-Z \geq t$ is subject to the same probability bound. It follows that

$$\mathbb{P}(|Z| \geq t) \leq 2 \exp\left(\frac{-2nt^2}{(4\beta_{n_T} n_T + 2L)^2}\right). \tag{B.4}$$

To derive an upper bound for $\mathbb{E}\left[D_{\mathcal{A}[T_{1:K}]}\right]$, we use the fact that all the training sets have the same length so that

$$\mathbb{E}(D_{\mathcal{A}[T_{1:K}]}) = \mathbb{E}\left[\mathcal{R}\left[\mathcal{A}(T_1)\right] - \widehat{\mathcal{R}}\left[\mathcal{A}(T_1), T_1\right]\right].$$

Then, we use Lemma 7 from Bousquet and Elisseeff (2002) ensuring that

$$\forall T \subset [n] \quad , \quad \mathbb{E}\left[\mathcal{R}\left[\mathcal{A}(T)\right] - \widehat{\mathcal{R}}\left[\mathcal{A}(T), T\right]\right] = \mathbb{E}\left[\ell\left(\mathcal{A}(T), O'\right) - \ell\left(\mathcal{A}(T^l), O'\right)\right].$$

Where $\mathcal{A}(T^l)$ is the learning rule $\mathcal{A}$ trained on the sample $\mathcal{D}_T \setminus \{O_l\} \cup \{O'\}$. Replacing the left side term by $\mathbb{E}\left[D_{\mathcal{A}[T_{1:K}]}\right]$ and using Definition 2.1 gives

$$|\mathbb{E}\left[f(\mathcal{D})\right]| = |\mathbb{E}\left[D_{\mathcal{A}[T_{1:K}]}\right]| \leq 2\beta_{n_T}.$$

Since $|f(\mathcal{D})| \leq |Z| + 2\beta_{n_T}$, we simply use (B.4) to reach the conclusion. $\qquad\square$

Remark B.1. Applying Lemma B.2 with $T = [n]$ and $K = 1$ gives the following probability upper bound

$$\mathbb{P}\left(\left|\widehat{\mathcal{R}}\left[\mathcal{A}([n]), [n]\right] - \mathcal{R}\left[\mathcal{A}([n])\right]\right| \geq t + 2\beta_n\right) \leq 2\exp\left(\frac{-2nt^2}{(2L + 4\beta_n n)^2}\right).$$

Thus we retrieve the bound of Theorem 12 in Bousquet and Elisseeff (2002).

C   DETAILED PROOFS

C.1   Proof of Theorem 3.1

We proceed as described in the sketch of proof. Using Equation 3.2, write

$$\left|\widehat{\mathcal{R}}_{CV}\left[\mathcal{A}, V_{1:K}\right] - \mathcal{R}\left[\mathcal{A}([n])\right]\right| \leq |D_{CV}| + |\text{Bias}|.$$

It remains to combine Lemma B.1 with Fact 2.1 to obtain the desired result.

C.2   Proof of Theorem 4.1

Consider the regression problem for a random pair $O = (X, Y)$ consisting of a covariate $X = \frac{\epsilon}{M}$ where ϵ is a Rademacher variable and a response $Y = M \operatorname{sign} X$ for some $M > 1$. Namely $\epsilon \in \{+1, -1\}$ and $P(\epsilon = \pm 1) = 1/2$.

Now, let $n \leq e^M$ and define the algorithm $\mathcal{A}$ as a regularized empirical risk minimizer, more precisely $\mathcal{A}(\mathcal{D}, X) = \hat{\beta}(\mathcal{D}) X$ with

$$\hat{\beta}(\mathcal{D}) = \arg\min_{\beta \in \mathbb{R}} \frac{1}{n} \sum_{i=1}^{n} (Y_i - \beta X_i)^2 + \lambda_n |\beta|^2,$$

and $\lambda_n = \frac{1}{\log(n)} - \frac{1}{M^2}$.

Algorithmic Stability   It's easy to check that $\hat{\beta} = \frac{\overline{X_n Y_n}}{(X_n)^2 + \lambda_n}$ where $\overline{X_n Y_n} = \frac{1}{n}\sum_{i=1}^{n} X_i Y_i$. Moreover, using the fact that $X_i Y_i = 1$, one obtains

$$\hat{\beta}(\mathcal{D}) = \frac{1}{1/M^2 + \lambda_n} = \log(n).$$

On the other hand, write

$$\ell\big(\mathcal{A}\,(\mathcal{D})\,,O\big) - \ell\big(\mathcal{A}(\mathcal{D}^{\backslash i}),O\big) = (\beta_{n-1} - \beta_n)X\,(2Y - (\beta_n + \beta_{n-1})X)$$
$$= (\beta_{n-1} - \beta_n)\left(2 - \frac{(\beta_n + \beta_{n-1})}{M^2}\right),$$
$$= (\log(n-1) - \log(n))\left(2 - \frac{\log(n) + \log(n-1)}{M^2}\right),$$

where the second line follows from the fact that $XY = 1$ and the last follows by replacing β and λ by their expression. To conclude this part, we use the fact that $\log(1 + x) \le x$ for all $x \le 1$, to obtain

$$\left|\ell\big(\mathcal{A}\,(\mathcal{D})\,,O\big) - \ell\big(\mathcal{A}(\mathcal{D}^{\backslash i}),O\big)\right| \le \frac{2}{n}.$$

Bias Lower Bound   Using the same equation as before we have

$$\ell\big(\mathcal{A}\,(\mathcal{D})\,,O\big) - \ell\big(\mathcal{A}(\mathcal{D}_T),O\big) = (\log(n_T) - \log(n))\left(2 - \frac{\log(n) + \log(n_T)}{M^2}\right)$$
$$= \log(\frac{K-1}{K})\left(2 - \frac{\log(n) + \log(n_T)}{M^2}\right).$$

Thus, since $n_T \le n \le e^M$ we obtain

$$\mathcal{R}\,[\mathcal{A}\,(\mathcal{D}_T)] - \mathcal{R}\,[\mathcal{A}\,(\mathcal{D})] \ge 2\log(\frac{K}{K-1})\left(1 - \frac{1}{M}\right).$$

It remains to notice that

$$\mathbb{E}\left[\left|\widehat{\mathcal{R}}_{CV}\,[\mathcal{A}, V_{1:K}] - \mathcal{R}\,[\mathcal{A}[n]]\,\right|\right] \ge \left|\mathbb{E}\left[\widehat{\mathcal{R}}_{CV}\,[\mathcal{A}, V_{1:K}] - \mathcal{R}\,[\mathcal{A}[n]]\right]\right|$$
$$= \mathcal{R}\,[\mathcal{A}\,(\mathcal{D}_T)] - \mathcal{R}\,[\mathcal{A}\,(\mathcal{D})]\,,$$

and the proof is complete.

C.3   Proof of Theorem 5.1

We proceed as described in the sketch of proof. First remind the error decomposition 5.2

$$\widehat{\mathcal{R}}_{CV}^{corr}\,[\mathcal{A}, V_{1:K}] - \mathcal{R}\,[\mathcal{A}([n])] = D_{\mathcal{A}[n]} + \frac{n_T}{n}(D_{CV} - D_{\mathcal{A}[T_{1:K}]}),$$

where D_{CV}, $D_{\mathcal{A}[T_{1:K}]}$ and $D_{\mathcal{A}[n]}$ are defined in 3.3, 5.4 and 5.3 respectively. Since $n_T \le n$, using the triangular inequality yields

$$\left|\widehat{\mathcal{R}}_{CV}^{corr}\,[\mathcal{A}, V_{1:K}] - \mathcal{R}\,[\mathcal{A}[n]]\,\right| \le |D_{\mathcal{A}[n]}| + |D_{CV}| + |D_{\mathcal{A}[T_{1:K}]}|.$$

Combining lemma B.1 and B.2 regarding D_{CV} and $D_{\mathcal{A}[T_{1:K}]}$ with Remark B.1 regarding $D_{\mathcal{A}[n]}$, one obtains

$$\mathbb{P}\left(\left|\widehat{\mathcal{R}}_{CV}^{corr}\,[\mathcal{A}, V_{1:K}] - \mathcal{R}\,[\mathcal{A}[n]]\,\right| \ge t + 2(\beta_{n_T} + \beta_n)\right)$$
$$\le \mathbb{P}\,(|D_{CV}| \ge t/3) + \mathbb{P}\left(|D_{\mathcal{A}[n]}| \ge t/3 + 2\beta_n\right) + \mathbb{P}\left(|D_{\mathcal{A}[T_{1:K}]}| \ge t/3 + 2\beta_{n_T}\right)$$
$$\le 6\exp\left(\frac{-2nt^2}{9(4\beta_{n_T}n_T + L)^2}\right). \tag{C.1}$$

By inverting, and using the assumption $\beta_t \le \frac{\lambda}{t}$ one gets, with probability $1 - 6\delta$,

$$|\widehat{\mathcal{R}}_{CV}^{corr}\,[\mathcal{A}, V_{1:K}] - \mathcal{R}\,[\mathcal{A}[n]]| \le 2\lambda(\frac{1}{n} + \frac{1}{n_T}) + 3(4\lambda + L)\sqrt{\frac{\log(1/\delta)}{2n}},$$

which is the desired result.

C.4 Proof of Theorem 6.1

The proof of Theorem 6.1 relies on the following proposition,

Proposition C.1. *Let $(\mathcal{A}^{(m)})_{m \in \mathcal{M}}$ be a family of algorithms where each learner $\mathcal{A}^{(m)}$ is $(\beta_{m,t})_{1 \leq i \leq n}$ uniform stable with respect to loss function $0 \leq \ell(g, O) \leq L$. Additionally, assume that, $|\mathcal{M}| < \infty$ and that*

$$\forall m \in \mathcal{M} \; ; \; \beta_{m,t} \leq \frac{M}{t},$$

for some $M > 0$. Then one has, with probability at least $1 - 6\delta$,

$$\sup_{m \in \mathcal{M}} \left| \widehat{\mathcal{R}}_{K\text{-fold}}^{corr} \left[\mathcal{A}^{(m)}, V_{1:K} \right] - \mathcal{R} \left[\mathcal{A}^{(m)}([n]) \right] \right| \leq \frac{6M}{n} + 4(M + L) \sqrt{\frac{\log(|\mathcal{M}|/\delta)}{n}}.$$

Proof. Using Fact 2.2 for the K-fold scheme ($n_T = n(K-1)/K$ with $K \geq 2$), and the fact that $\beta_{m,n} + \beta_{m,n_T} \leq (M/n)((2K-1)/(K-1))$ as well as $(2K-1)/(K-1) \leq 3$, one gets

$$\forall m \in \mathcal{M} \; , \; \mathbb{P} \left(\left| \widehat{\mathcal{R}}_{K\text{-fold}}^{corr} \left[\mathcal{A}^{(m)}, V_{1:K} \right] - \mathcal{R} \left[\mathcal{A}^{(m)}([n]) \right] \right| \geq t + (6M/n) \right) \leq 6 \exp \left(\frac{-2nt^2}{9(4M + L)^2} \right),$$

which gives by a union bound

$$\mathbb{P} \left(\sup_{m \in \mathcal{M}} \left| \widehat{\mathcal{R}}_{K\text{-fold}}^{corr} \left[\mathcal{A}^{(m)}, V_{1:K} \right] - \mathcal{R} \left[\mathcal{A}^{(m)}([n]) \right] \right| \geq t + (6M/n) \right) \leq 6|\mathcal{M}| \exp \left(\frac{-2nt^2}{9(4M + L)^2} \right).$$

Thus, by inverting, we obtain the desired result. $\qquad\square$

Proof of Theorem 6.1 First, using the definition of $\hat{m}$ (eq. 6.2), write

$$\widehat{\mathcal{R}}_{Kfold}^{corr} \left[\mathcal{A}^{(\hat{m})}, V_{1:K} \right] - \widehat{\mathcal{R}}_{Kfold}^{corr} \left[\mathcal{A}^{(m^*)}, V_{1:K} \right] \leq 0.$$

It follows that

$$\mathcal{R} \left[\mathcal{A}^{(\hat{m})}([n]) \right] - \mathcal{R} \left[\mathcal{A}^{(m^*)}([n]) \right] \leq \mathcal{R} \left[\mathcal{A}^{(\hat{m})}([n]) \right] - \widehat{\mathcal{R}}_{Kfold}^{corr} \left[\mathcal{A}^{(\hat{m})}, V_{1:K} \right] + \widehat{\mathcal{R}}_{Kfold}^{corr} \left[\mathcal{A}^{(m^*)}, V_{1:K} \right] - \mathcal{R} \left[\mathcal{A}^{(m^*)}([n]) \right]$$

$$\text{(C.2)}$$

$$\leq 2 \sup_{m \in \mathcal{M}} \left| \widehat{\mathcal{R}}_{K\text{-fold}}^{corr} \left[\mathcal{A}^{(m)}, V_{1:K} \right] - \mathcal{R} \left[\mathcal{A}^{(m)}([n]) \right] \right|.$$

It remains to use proposition C.1 and the proof is complete.

D UNIFORM STABILITY FOR RANDOMIZED ALGORITHMS

In this section we generalize the results from he main paper to the case of randomized algorithms. Let us start with reminding the concept of uniform stability for randomized learning algorithms introduced in Elisseeff et al. (2005).

Definition D.1. *An algorithm $\mathcal{A}$ is said to be $(\beta_t)_{1 \leq t \leq n}$-uniform stable with respect to a loss function ℓ if, for any $\mathcal{D} \in \mathcal{Z}^n$, $T \subset [n]$, $i \in T$ and $o \in \mathcal{Z}$, the following holds*

$$\left| \mathbb{E}_{\mathcal{A}} \left[\ell \left(\mathcal{A}(T), o \right) \right] - \mathbb{E}_{\mathcal{A}} \left[\ell \left(\mathcal{A}(T^{\backslash i}), o \right) \right] \right| \leq \beta_{n_T}. \tag{D.1}$$

Where the randomness in the latter expectation stems from the algorithm $\mathcal{A}$ while the observation o and the data sequence $\mathcal{D}$ are fixed. Equivalently,

$$\left| \mathbb{E} \left[\ell \left(\mathcal{A}(T), O \right) - \ell \left(\mathcal{A}(T^{\backslash i}), O \right) \middle| \mathcal{D}_T, O \right] \right| \leq \beta_{n_T}. \tag{D.2}$$

Note that a similar version of Fact 2.1 still holds, more precisely, one has

Fact D.1. *Let $\mathcal{A}$ be a decision rule which is $(\beta_t)_{1 \leq t \leq n}$ uniformly stable, additionally suppose that the sequence $(\beta_t)_{1 \leq t \leq n}$ is decreasing, then for any $T \subset [n]$, and $o \in \mathcal{Z}$, one has*

$$\left| \mathbb{E} \left[\ell \left(\mathcal{A}([n]), o \right) - \ell \left(\mathcal{A}(T), o \right) \mid \mathcal{D} \right] \right| \leq \sum_{i=n_T}^{n} \beta_i.$$

D.1 Upper Bounds for K-fold CV under Random Uniform Stability

We now derive upper bounds *-in expectation-* on the error induced by $\widehat{\mathcal{R}}_{CV}$ and $\widehat{\mathcal{R}}_{CV}^{corr}$. As highlighted in Remark 3.1, the main problem with K-fold CV is it's bias. Therefore, for the sake of brevity, we will focus only on the expectation of the estimate .

Theorem D.1. *Suppose that $\mathcal{A}$ is $(\beta_t)_{1 \leq t \leq n}$ uniformly stable. Then we have*

$$\left| \mathbb{E}\left[\widehat{\mathcal{R}}_{CV}\left[\mathcal{A}, V_{1:K} \right] - \mathcal{R}\left[\mathcal{A}([n]) \right] \right] \right| \leq \sum_{i=n_T}^{n} \beta_i.$$

Proof. First, by applying the tower rule one has

$$
\begin{aligned}
\mathbb{E}\left[\widehat{\mathcal{R}}_{CV}\left[\mathcal{A}, V_{1:K} \right] - \mathcal{R}\left[\mathcal{A}([n]) \right] \right] &= \mathbb{E}\left[\frac{1}{K} \sum_{j=1}^{K} \widehat{\mathcal{R}}\left[\mathcal{A}(T_j), V_j \right] - \mathcal{R}\left[\mathcal{A}([n]) \right] \right] \\
&= \frac{1}{K} \sum_{j=1}^{K} \mathbb{E}\left[\mathbb{E}\left[\widehat{\mathcal{R}}\left[\mathcal{A}(T_j), V_j \right] \mid \mathcal{D}_{T_j} \right] - \mathcal{R}\left[\mathcal{A}([n]) \right] \right] \\
&= \frac{1}{K} \sum_{j=1}^{K} \mathbb{E}\left[\mathbb{E}\left[\ell(\mathcal{A}(T_j), O) \mid \mathcal{D}_{T_j} \right] - \mathcal{R}\left[\mathcal{A}([n]) \right] \right] \\
&= \frac{1}{K} \sum_{j=1}^{K} \mathbb{E}\left[\mathbb{E}\left[\ell(\mathcal{A}(T_j), O) - \ell(\mathcal{A}([n], O)) \mid \mathcal{D} \right] \right]. \quad \text{(D.3)}
\end{aligned}
$$

The third line follows from the fact that $\ell(\mathcal{A}(T), O_j)$ and $\ell(\mathcal{A}(T), O)$ has the same law for all $j \in V$. This indeed verified since all the training sets T_j's has the same length and the O_j's are independent from $\mathcal{D}_T$. To obtain the desired result, it remains to combine Equation D.3 with fact D.1. □

Now let us prove that the bias corrected K-fold (cf. Eq (5.1)) has a vanishing bias for randomized algorithms.

Theorem D.2 (Corrected K-fold bias). *Suppose that $\mathcal{A}$ is $(\beta_t)_{1 \leq t \leq n}$ uniformly stable and $\beta_t \leq \frac{\lambda}{t}$, for some $\lambda > 0$. Then, we have*
$$\left| \mathbb{E}\left[\widehat{\mathcal{R}}_{\text{Kfold}}^{corr}\left[\mathcal{A}, V_{1:K} \right] - \mathcal{R}\left[\mathcal{A}([n]) \right] \right] \right| \leq \frac{2\lambda(2K-1)}{(K-1)n}.$$

Proof. Combining the error decomposition (5.2) with the fact that $n_T = \frac{K-1}{K}n$ we obtain

$$\widehat{\mathcal{R}}_{\text{Kfold}}^{corr}\left[\mathcal{A}, V_{1:K} \right] - \mathcal{R}\left[\mathcal{A}[n] \right] = D_{\mathcal{A}[n]} + \frac{K-1}{K}(D_{CV} - D_{\mathcal{A}[T_{1:K}]}). \quad \text{(D.4)}$$

Using Theorem 2.2 in Hardt et al. (2016), one obtains the twin inequality

$$\left| \mathbb{E}\left[D_{\mathcal{A}[n]} \right] \right| \leq 2\beta_n$$

$$\left| \mathbb{E}\left[D_{\mathcal{A}[T_{1:K}]} \right] \right| \leq 2\beta_{n_T}.$$

Since $\mathbb{E}\left[D_{CV} \right] = 0$, Equation D.4 combined with the triangular inequality gives

$$
\begin{aligned}
\left| \mathbb{E}\left[\widehat{\mathcal{R}}_{\text{Kfold}}^{corr}\left[\mathcal{A}, V_{1:K} \right] - \mathcal{R}\left[\mathcal{A}([n]) \right] \right] \right| &\leq 2\left(\beta_n + \frac{(K-1)\beta_{n_T}}{K} \right) \\
&\leq 2\left(\beta_n + \beta_{n_T} \right). \quad \text{(D.5)}
\end{aligned}
$$

It remains to use the assumption $\beta_t \leq \frac{\lambda}{t}$ and the proof is complete. □

We conclude this section by deriving an upper bound for the model selection problem,

Theorem D.3. *Let $(\mathcal{A}^{(m)})_{m \in \mathcal{M}}$ be a family of algorithms where each learner $\mathcal{A}^{(m)}$ is $(\beta_{m,t})_{1 \le i \le n}$ uniform stable with respect to loss function ℓ. Additionally, assume that*

$$\beta_{m,t} \le \frac{M}{t}$$

for some $M > 0$. Then one has

$$\mathbb{E}\left[\mathcal{R}\big[\mathcal{A}^{(\hat{m})}([n])\big] - \mathcal{R}\big[\mathcal{A}^{(m^*)}([n])\big]\right] \le \frac{4M(2K-1)}{(K-1)n},$$

where m^* and $\hat{m}$ are defined by Equations 6.1, 6.2 respectively.

Proof. First, using Equation D.5, one obtains, for all $m \in \mathcal{M}$,

$$\left|\mathbb{E}\left[\widehat{\mathcal{R}}_{\text{Kfold}}^{corr}\big[\mathcal{A}^{(m)}, V_{1:K}\big] - \mathcal{R}\big[\mathcal{A}^{(m)}([n])\big]\right]\right| \le 2\left(\beta_{m,n} + \beta_{m,n_T}\right)$$

$$\le \frac{2M(2K-1)}{(K-1)n}.$$

So that

$$\sup_{m \in \mathcal{M}} \left|\mathbb{E}\left[\widehat{\mathcal{R}}_{\text{Kfold}}^{corr}\big[\mathcal{A}^{(m)}, V_{1:K}\big] - \mathcal{R}\big[\mathcal{A}^{(m)}([n])\big]\right]\right| \le \frac{2M(2K-1)}{(K-1)n}. \tag{D.6}$$

On the other hand, Inequality C.2 yields

$$\mathbb{E}\left[\mathcal{R}\big[\mathcal{A}^{(\hat{m})}([n])\big] - \mathcal{R}\big[\mathcal{A}^{(m^*)}([n])\big]\right] \le \mathbb{E}\left[\mathcal{R}\big[\mathcal{A}_{\mathcal{D}}^{(\hat{m})}([n])\big] - \widehat{\mathcal{R}}_{\text{Kfold}}^{corr}\big[\mathcal{A}^{(\hat{m})}, V_{1:K}\big]\right]$$

$$+ \mathbb{E}\left[\widehat{\mathcal{R}}_{\text{Kfold}}^{corr}\big[\mathcal{A}^{(m^*)}, V_{1:K}\big] - \mathcal{R}\big[\mathcal{A}^{(m^*)}([n])\big]\right]$$

$$\le 2\sup_{m \in \mathcal{M}} \left|\mathbb{E}\left[\widehat{\mathcal{R}}_{\text{Kfold}}^{corr}\big[\mathcal{A}^{(m)}, V_{1:K}\big] - \mathcal{R}\big[\mathcal{A}^{(m)}([n])\big]\right]\right|.$$

Thus, by Inequality D.6, one has

$$\mathbb{E}\left[\mathcal{R}\big[\mathcal{A}^{(\hat{m})}([n])\big] - \mathcal{R}\big[\mathcal{A}^{(m^*)}([n])\big]\right] \le \frac{4M(2K-1)}{(K-1)n},$$

which concludes the proof. □

D.2   Proof of Theorem 4.2

Following the line of Zhang et al. (2022), consider the following convex function ,

$$f(w, o) = \frac{1}{2}w^\top A w - y x^\top w,$$

where A is a positive semi definite penalization matrix (PSD) in $\mathbb{R}^{d \times d}$ with rank $p < d$, $w \in \mathbb{R}^d$ and $x, y \in \mathbb{R}^d \times \mathbb{R}$. Such a rank deficient penalization matrix can be found in multiple contexts like fused lasso (Tibshirani and Taylor, 2011), fused ridge (Bilgrau et al., 2020), etc.

In the next lemma, by picking carefully the input space $\mathcal{X} \times \mathcal{Y}$ and the distribution P, we construct an example where we control exactly the amount of *instability* of SGD . Theorem 4.2 follows directly from the following proposition.

Proposition D.1. *Let $M > 1$, $n \in [\![1, e^M]\!]$. Suppose that $O = (X, Y) \in \mathcal{X} \times \mathcal{Y} = \{v, -v\} \times \{1\}$ and $P(X = v) = \frac{2}{3}$ where v is a unit vector in $\mathbb{R}^d$ such as $Av = 0$. For $t \ge 1$, there exist a sequence of step sizes $(\alpha_{k,n})_{1 \le k \le t}$, such as the SGD algorithm (Definition 4.3) with $\ell = f$ and $\mathcal{A}_0 = 0$ verifies,*

$$\mathbb{E}\left[\left|\widehat{\mathcal{R}}_{CV}\left[\mathcal{A}, V_{1:K}\right] - \mathcal{R}\left[\mathcal{A}[n]\right]\right|\right] \ge \frac{\log\left(K/K-1\right)}{3}.$$

Furthermore SGD satisfies Assumption 1 with respect to ℓ with $L = M$ and

$$\beta_n \le \frac{3}{n-1}\sum_{k=1}^{t} \alpha_{k,n} \le \frac{3M}{n-1}.$$

Proof. Computing the gradient of $f(\cdot, o)$ for $o = (x, y)$ yields,

$$\nabla_w f(w, o) = Aw - yx.$$

Now set $w_t = \mathcal{A}_t([n])$ and write using the definition of SGD (Equation 4.3) :

$$w_{t+1} = (I - \alpha_t A) w_t + \alpha_{t,n} y X_{it}. \tag{D.7}$$

Thus by induction we obtain $w_t = \theta_t v$ for some $\theta_t \in \mathbb{R}$. Consequently, Equation D.7 yields,

$$w_{t+1} = w_t + \alpha_{t,n} y X_{it},$$

so that,

$$w_t = \sum_{k=1}^{t} \alpha_{k,n} y X_{it}. \tag{D.8}$$

Furthermore SGD picks $X_{it} = v$ with probability n_+/n where n_+ (*resp.* n_-) is the number of samples such as $X = v$ (*resp.* $-v$), hence

$$\mathbb{E}_{\mathcal{A}}[w_t] = \sum_{k=1}^{t} \alpha_{k,n} \left(\frac{n_+}{n} - \frac{n_-}{n} \right) v. \tag{D.9}$$

On the other hand, since $w_t = \theta_t v$ we obtain,

$$f(w_t, o) = -yx^\top w_t. \tag{D.10}$$

Set $w'_t = \mathcal{A}\left([n]^{\backslash j}\right)$ and consider the case where j is such as $X_j = v$. Note that the other case is similar and thus omitted. Since $\|v\| = 1$, one has by Equations D.9 and D.10

$$\forall o \in \mathcal{Z}, \ |\mathbb{E}_{\mathcal{A}}[f(w'_t, o)] - \mathbb{E}_{\mathcal{A}} f[(w_t, o)]| = \left| \frac{n_+ - n_-}{n} \sum_{k=1}^{t} \alpha_{k,n} - \frac{n_+ - n_- - 1}{n-1} \sum_{k=1}^{t} \alpha_{k,n-1} \right|.$$

Now for $t \geq 0$, take $\alpha_{k,n} = \frac{\log(n)}{t}$ for all $k \leq t$ so that,

$$\forall o \in \mathcal{Z}, \ |\mathbb{E}_{\mathcal{A}}[f(w'_t, o)] - \mathbb{E}_{\mathcal{A}} f[(w_t, o)]| = \left| \frac{n_+ - n_-}{n} \log(n) - \frac{n_+ - n_- - 1}{n-1} \log(n-1) \right|,$$

which yields by simple algebra

$$\begin{aligned} \forall o \in \mathcal{Z}, \ |\mathbb{E}_{\mathcal{A}}[f(w'_t, o)] - \mathbb{E}_{\mathcal{A}} f[(w_t, o)]| &\leq \frac{2\log(n) + 1}{n-1} \\ &\leq \frac{3\log(n)}{n-1} \\ &= \frac{3}{n-1} \sum_{k=1}^{t} \alpha_{k,n} \\ &\leq \frac{3M}{n-1}. \end{aligned} \tag{D.11}$$

Now, using Equations D.8 and D.10 and the expression of $\alpha_{k,n}$ we get,

$$\forall o \in \mathcal{Z}, \ |f(w_t, o)| = \sum_{k=1}^{t} \alpha_{k,n} = \log(n) \leq M,$$

The latter equation combined with Equation D.11 confirms that SGD verifies Assumption 1 with $L = M$ and

$$\beta_n \leq \frac{3}{n-1} \sum_{k=1}^{t} \alpha_{k,n} \leq \frac{3M}{n-1}.$$

For the lower bound, let $T \subset [n]$ and set $w_t^T = \mathcal{A}_t(T)$. Using D.9 yields

$$\mathbb{E}\left[\mathbb{E}_{\mathcal{A}}\left[w_t^T\right]\right] = \sum_{k=1}^{t} \alpha_{k,n_T} \frac{v}{3},$$

so that by D.10 ,

$$\left|\mathbb{E}\left[\mathbb{E}_{\mathcal{A}}\left[f\left(w_t, o\right)\right]\right] - \left[\mathbb{E}_{\mathcal{A}}\left[f\left(w_t^T, o\right)\right]\right]\right| = \frac{\sum_{k=1}^{t} \alpha_{k,n} - \sum_{k=1}^{t} \alpha_{k,n_T}}{3}$$

$$\left(\alpha_{k,n} = \log(n)/t\right) = \frac{\log\left(K/K - 1\right)}{3}.$$

It remains to notice that

$$\mathbb{E}\left[\left|\widehat{\mathcal{R}}_{CV}\left[\mathcal{A}, V_{1:K}\right] - \mathcal{R}\left[\mathcal{A}[n]\right]\right|\right] \geq \left|\mathbb{E}\left[\widehat{\mathcal{R}}_{CV}\left[\mathcal{A}, V_{1:K}\right] - \mathcal{R}\left[\mathcal{A}[n]\right]\right]\right|$$
$$= \left|\mathcal{R}\left[\mathcal{A}\left(\mathcal{D}_T\right)\right] - \mathcal{R}\left[\mathcal{A}\left(\mathcal{D}\right)\right]\right|,$$

and the proof is complete. $\qquad\square$