BEYOND REAL DATA: SYNTHETIC DATA THROUGH THE LENS OF REGULARIZATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Synthetic data can improve generalization when real data is scarce, but excessive reliance may introduce distributional mismatches that degrade performance. In this paper, we present a learning-theoretic framework to quantify the trade-off between synthetic and real data. Our approach leverages algorithmic stability to derive generalization error bounds, characterizing the optimal synthetic-to-real data ratio that minimizes expected test error as a function of the Wasserstein distance between the real and synthetic distributions. We motivate our framework in the setting of kernel ridge regression with mixed data, offering a detailed analysis that may be of independent interest. Our theory predicts the existence of an optimal ratio, leading to a U-shaped behavior of test error with respect to the proportion of synthetic data. Empirically, we validate this prediction on CIFAR-10 and a clinical brain MRI dataset. Our theory extends to the important scenario of domain adaptation, showing that carefully blending synthetic target data with limited source data can mitigate domain shift and enhance generalization. We conclude with practical guidance for applying our results to both in-domain and out-of-domain scenarios.

1 INTRODUCTION

The success of modern Machine Learning (ML) and Artificial Intelligence (AI) heavily depends on the availability of large-scale training datasets (Sun et al., 2017; Radford et al., 2021). However, in many critical domains such as healthcare, data collection is often prohibitively expensive, time-consuming, or constrained by privacy concerns (Esteva et al., 2019; Kaissis et al., 2021). Similar challenges arise in scientific domains where obtaining labeled data requires high-fidelity physical simulations or specialized experimental setups. For instance, generating data in molecular dynamics (Hollingsworth & Dror, 2018; Hansson et al., 2002) often demands significant computational resources, or structural biology techniques like cryo-electron microscopy (Murata & Wolf, 2018; Milne et al., 2013) involve costly and complex instrumentation. In these scenarios, ML models are trained on small datasets and as a result frequently suffer from poor generalization, limiting their practical applicability (Recht et al., 2019; Maleki et al., 2022; Schmidt et al., 2018; Brigato & Iocchi, 2021).

To address this challenge, several strategies have been proposed, including data augmentation (Shorten & Khoshgoftaar, 2019; Cubuk et al., 2020) and the use of synthetic data (Frid-Adar et al., 2018; Karras et al., 2020; Lu et al., 2023). Although these methods can improve model accuracy, their success depends critically on how well the synthetic data approximates the real data distribution (Bowles et al., 2018). With the emergence of powerful generative models such as diffusion models (Ho et al., 2020; Song et al., 2021; Lipman et al., 2022; De Bortoli et al., 2021), there is renewed interest in using synthetic data to supplement limited real data (Trabucco et al., 2023; Voetman et al., 2023; Alemohammad et al., 2024b). Empirical evidence suggests that, when properly generated, synthetic data can substantially boost the downstream model performance in low-data regimes (Azizi et al., 2023).

However, the integration of synthetic data introduces a critical trade-off as synthetic data may deviate from the true data distribution. If the synthetic dataset grows disproportionately large, the training algorithm may overlook the real data, introducing bias (Alemohammad et al., 2024a; Briesch et al., 2023; Betzalel et al., 2022; Dohmatob et al., 2025; Bertrand et al., 2024). See Appendix I for an extensive literature review on this topic. This issue motivates a central question:

"What is the optimal balance between real and synthetic data to minimize generalization error?"

In this work, we address this question from a learning-theoretic perspective, establishing that an optimal ratio of synthetic to real data exists for maximizing generalization performance. We first motivate our analysis through a simple yet insightful case study in kernel ridge regression (Singh & Vijaykumar, 2023), which may be of independent interest. We then extend our theoretical framework to more general settings, deriving generalization bounds via stability analysis (Bousquet & Elisseeff, 2002; Shalev-Shwartz & Ben-David, 2014; Hardt et al., 2016). Our theoretical insights are empirically validated on two distinct datasets: CIFAR-10 (a standard benchmark) (Krizhevsky et al., 2009) and a real-world brain imaging dataset for Multiple Sclerosis (MS) (Carass et al., 2017).

Furthermore, we extend our framework to domain adaptation settings (Ben-David et al., 2010; Ganin et al., 2016; Wilson & Cook, 2020), where synthetic data from a target domain is used to enhance limited real data from a source domain. This broadens the scope of our approach, highlighting its relevance for data-scarce scenarios in diverse ML applications like healthcare (Zhuang et al., 2021).

Contributions and paper structure Our main contributions are as follows:

- We provide a learning-theoretic analysis demonstrating the existence of an optimal ratio between synthetic and real data that minimizes generalization error. Our approach is grounded in stability-based generalization bounds and is first illustrated through a tractable kernel ridge regression model. See Sections 2 and 3.
- We empirically validate our theoretical predictions using both benchmark (CIFAR-10, Appendix H.2) and real-world (brain MRI for Multiple Sclerosis, Section 4) datasets, confirming that an appropriate balance of synthetic data improves performance in low-data regimes.
- We extend our framework to domain adaptation, showing how synthetic data from a target domain can be effectively combined with limited real data from a source domain, thereby broadening the applicability of our results (Section 5). We also provide practical guidance for applying our theory to both in-domain and out-of-domain generalization tasks (Section 6).

2 Motivation: Synthetic data in kernel ridge regression

We study the effect of incorporating synthetic data into kernel regression (Singh & Vijaykumar, 2023; Allerbo, 2023; Wang & Jing, 2022; Smale & Zhou, 2005) as a simple yet illustrative setting to gain insight into the key factors influencing the generalization bound. Our goal is to identify theoretical and empirical conditions under which synthetic data improves or degrades generalization, highlighting the trade-offs involved in leveraging such data to enhance learning performance.

We consider kernel regression, where a function is learned by minimizing a regularized empirical risk over a Reproducing Kernel Hilbert Space (RKHS) denoted by $\mathcal{H}_K$. Given training data $\{(\mathrm{x}_n, \mathrm{y}_n)\}_{n=1}^N$ with $\mathrm{x}_n \in \mathcal{X} \sim p_{\mathbf{x}}$ and $\mathrm{y}_n \in \mathbb{R}$, the objective is to find a function $f \in \mathcal{H}_K$ that best fits the data while controlling complexity through a regularization term. We assume that $\mathrm{y}_n = f_\star(\mathrm{x}_n) + \varepsilon_n$, where ε_n are Independent and Identically Distributed (i.i.d.) samples from a zero-mean Gaussian distribution with variance σ^2. Unlike the standard setup, we regularize towards a synthetic data generator $g \in \mathcal{H}_K$, effectively corresponding to the case of having an infinite number of synthetic samples. See Appendix D for an analysis of this asymptotic behavior, along with a discussion of the finite-sample alternative. The resulting Empirical Risk Minimization (ERM) problem is:

$$f_N = \underset{f \in \mathcal{H}_k}{\arg\min} \frac{1}{N} \sum_{n=1}^{N} (\mathrm{y}_n - f(\mathrm{x}_n))^2 + \lambda \|f - g\|_{\mathcal{H}_k}^2 , \tag{1}$$

where $\lambda > 0$ is the regularization strength. By *Representer Theorem* (Kimeldorf & Wahba, 1971; Schölkopf et al., 2001), the learned function takes the form $f_N(\mathrm{x}) = \sum_{n=1}^N \alpha_n K(\mathrm{x}, \mathrm{x}_n)$, where K is a positive definite kernel function, and α_n are coefficients obtained from a regularized least squares problem. Let $\tilde{\mathcal{H}} = \mathrm{span}\{K(\cdot, \mathrm{x}_1), \ldots, K(\cdot, \mathrm{x}_N)\}$, and decompose $\mathcal{H}_K = \tilde{\mathcal{H}} \oplus \tilde{\mathcal{H}}^\perp$,where $\tilde{\mathcal{H}}^\perp$ is the orthogonal complement in $\mathcal{H}_K$. By Representer Theorem, the synthetic data generator $g \in \mathcal{H}_K$ can be written as $g(\mathrm{x}) = \sum_{n=1}^N \beta_n K(\mathrm{x}, \mathrm{x}_n) + g_\perp(\mathrm{x})$, where $g_\perp \in \tilde{\mathcal{H}}^\perp$. Setting $g = 0$ recovers the standard kernel ridge regression. We establish the following lemma (proof in Appendix E.1), which characterizes the solution to Equation 1.

Lemma 2.1. *Let $K_N \in \mathbb{R}^{N \times N}$ be the empirical kernel matrix with entries $(K_N)_{ij} = K(\mathrm{x}_i, \mathrm{x}_j)$. Define the integral operator $T_K : L^2(p_x) \to L^2(p_x)$ by $(T_K f)(\mathrm{x}) = \int K(\mathrm{x}, x') f(x')\, dp_x(x') =$*

$\mathbb{E}_{\mathrm{x}'}\left[K(\mathrm{x},\mathrm{x}')f(\mathrm{x}')\right]$. *Let $\lambda_N = N\lambda$. Then the solution to Equation 1 has the closed-form representation:*

$$\boldsymbol{\alpha} = \left(K_N + \lambda_N I\right)^{-1}\left(K_N\boldsymbol{\alpha}_\star + \lambda_N\boldsymbol{\beta} + \boldsymbol{\varepsilon}\right) \,,$$

where $\boldsymbol{\alpha}_\star$, $\boldsymbol{\beta}$, and $\boldsymbol{\varepsilon}$ are the coefficients of $f_\star$, g, and the noise vector in the training basis.

We now recall the Mercer decomposition (Mercer, 1909) of the kernel. The operator T_K defined in Lemma 2.1 is compact, self-adjoint, and positive semi-definite, and thus admits a spectral decomposition. That is, there exist eigenfunctions $\{\phi_j\}_{j=1}^\infty$ forming an orthonormal basis of $L^2(p_x)$ and corresponding non-negative eigenvalues $\mu_1 \geq \mu_2 \geq \cdots \to 0$ such that $T_K\phi_j = \mu_j\phi_j$. The eigenfunctions ϕ_j can be interpreted as the natural coordinates of the function space with respect to the kernel, and the eigenvalues μ_j encode their relative importance. In this basis, we can write

$$f_\star = \sum_{j=1}^{\infty}\theta_j\phi_j \,, \qquad g = \sum_{j=1}^{\infty}\omega_j\phi_j \,, \qquad \text{where } \theta_j = \langle f_\star, \phi_j\rangle \,, \text{and } \omega_j = \langle g, \phi_j\rangle \,. \tag{2}$$

Assumption 2.1 (Polynomial eigendecay and smoothness)**.** *We assume the kernel K exhibits $2r$-polynomial eigendecay for some $r \geq \frac{1}{2}$. Given the expansions in Equation 2, we assume:*

$$\text{(a) } \theta_j^2 \asymp \mu_j^s \asymp j^{-2rs} \text{ for some } s > 0, \qquad \text{(b) } \omega_j^2 \asymp \mu_j^{s'} \asymp j^{-2rs'} \text{ for some } s' > 0.$$

Assumption 2.1 quantifies how well $f_\star$ and g align with the eigenfunctions of T_K. It ensures that $\sum_j \theta_j^2/\mu_j^s < \infty$ and $\sum_j \omega_j^2/\mu_j^{s'} < \infty$, i.e. $f_\star$ and g decay sufficiently fast in the eigenbasis; larger rate corresponds to greater smoothness. Such assumptions are standard in kernel regression analysis; see, e.g., Cheng et al. (2024); Bartlett et al. (2019); Cui et al. (2021); Barzilai & Shamir (2024).

Definition 2.1 (Bias-Variance Decomposition)**.** *Define the test error $\mathcal{R}_N(\lambda; g)$ to be the population mean squared error between the regressor and the true label averaged over noise:*

$$\mathcal{R}_N(\lambda; g) = \mathbb{E}_{\mathbf{x},\varepsilon}\left[\left(f_\star(\mathbf{x}) - f_N(\mathbf{x})\right)^2\right] \,.$$

We decompose the test error into a bias $\mathcal{B}$ and variance $\mathcal{V}$, with $\mathcal{R}_N(\lambda; g) = \mathcal{B}^2 + \mathcal{V}$, such that:

$$\mathcal{B}^2 = \mathbb{E}_{\mathbf{x}}\left[f_\star(\mathbf{x}) - \mathbb{E}_\varepsilon\left[f_N(\mathbf{x})\right]\right]^2 \,, \qquad \mathcal{V} = \mathbb{E}_{\mathbf{x},\varepsilon}\left[\left(f_N(\mathbf{x}) - \mathbb{E}_\varepsilon\left[f_N(\mathbf{x})\right]\right)^2\right] \,.$$

We now present a bias–variance decomposition of Equation 1, along with a corollary characterizing the optimal number of synthetic samples. Proofs are in Appendices E.2 and E.3.

Theorem 2.2 (Generalization Error Bound)**.** *Under Assumption 2.1, for the kernel regression problem defined in Equation 1 and any fixed regularization parameter $\lambda \geq 0$, the test error admits the bound:*

$$\mathcal{R}_N(\lambda; g) = \mathcal{O}\left(\frac{\mathcal{D}(f_\star, g)}{N\lambda^2} + \lambda^{2-\frac{1}{4r}}\mathcal{D}(f_\star, g) + \frac{\sigma^2}{N}\lambda^{-\frac{1}{2r}}\right) \,,$$

where $\mathcal{D}(f_\star, g)^2 = \sum_{j=1}^{\infty}\frac{1}{\mu_j^2}(\theta_j - \omega_j)^2$ denotes the discrepancy between the target function $f_\star$ and the synthetic generator g.

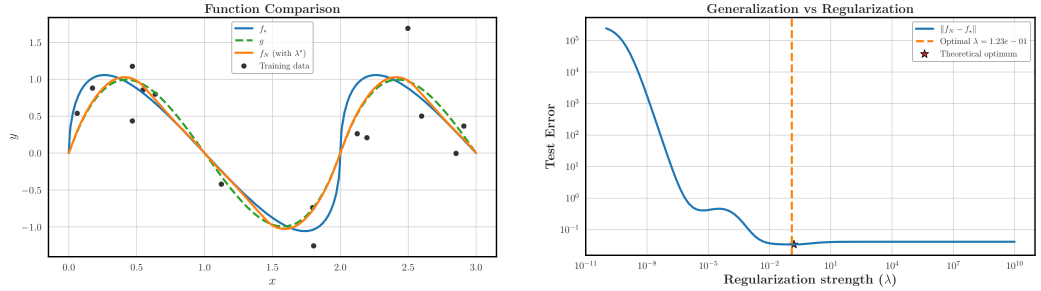
Corollary 2.2.1 (Optimal Regularization and Synthetic Sample Size)**.** *Under the assumptions of Theorem 2.2, the optimal regularization parameter that minimizes the test error is given by*

$$\lambda^\star \asymp \left(\frac{\sigma^2}{N\mathcal{D}(f_\star, g)}\right)^{\frac{4r}{8r+1}} \,.$$

Setting $\lambda = \frac{M}{N}$, the optimal number of synthetic samples satisfies:

$$M^\star \asymp \left(\frac{\sigma^2}{\mathcal{D}(f_\star, g)}\right)^{\frac{4r}{8r+1}} N^{\frac{4r+1}{8r+1}} .$$

We empirically validate our theory in Figure 1, observing a U-curve as predicted by Theorem 2.2, with error minimized near the theoretical λ^*. See Appendix H.1 for details.



(a) Function comparisons.

(b) Generalization vs regularization.

Figure 1: *(a)* Comparison of the true function $f_\star$ (blue), the synthetic generator g (green), and the learned estimator f_N (orange), obtained via Lemma 2.1, with parameters $r = 2.0$, $s = 0.8$, and $s' = 1.5$. *(b)* Prediction error $|f_N - f_\star|_{L_2}$ as a function of the regularization strength λ. The U-shaped curve attains its minimum at $\lambda^\star$ (orange dashed line), which closely matches the theoretical optimum (star marker).

3 Generalization error with synthetic data as a regularizer

We begin by introducing the notation and formal setting used throughout the remainder of the paper. We consider learning on a separable complete metric space $(\mathcal{X}, d_{\mathcal{X}})$. We define the sample space $\mathcal{S}_N = \mathcal{X}^N$ and the random training dataset of N i.i.d. samples from p_{x} over $\mathcal{X}$ is denoted by $\mathbf{S} = \{\mathrm{x}_1, \dots, \mathrm{x}_N\} \in \mathcal{S}_N$, with joint law $p_{\mathbf{S}}$. Consider some measurable hypothesis space $\mathcal{H}$ and a loss function $\ell : \mathcal{H} \times \mathcal{X} \to \mathbb{R}$ that quantifies the performance of a hypothesis, and we assume $\ell(h, \cdot) \in L^1(p_{\mathrm{x}})$ for each $h \in \mathcal{H}$. We define the empirical and population risks as,

$$\mathcal{L}_{\mathbf{S}}(h) = \frac{1}{N} \sum_{i=1}^{N} \ell(h, \mathrm{x}_i), \qquad \mathcal{L}_{\mathcal{X}}(h) = r(h) = \mathbb{E}_{p_{\mathrm{x}}}[\ell(h, \mathrm{x})].$$

For $r \in [1, \infty)$, the Wasserstein r-distance between two probability measures p and q on $\mathcal{X}$ with finite r-moments is defined as $\mathcal{W}_r\,(p\,, q) = \inf_{\gamma \in \Gamma(p,q)} (\mathbb{E}_{(x,y)\sim\gamma}[d(x, y)^r])^{1/r}$, where $\Gamma(p, q)$ is the set of all couplings of p and q. See Appendix C for more detailed notation.

3.1 Population error bounds

Following the motivation in Section 2, we consider training on a mixture of real and synthetic data, where the synthetic data acts as a form of regularization. We focus on the following mixed loss, which slightly differs from that in the previous section in its regularization formulation:

$$\mathcal{R}_\lambda(h, \mathbf{S}) = (1 - \lambda)\mathcal{L}_{\mathbf{S}}(h) + \lambda \mathbb{E}_{\mathbf{x} \sim p'_{\mathbf{x}}} [\ell(h, \mathbf{x})],$$

where $p'_{\mathbf{x}}$ denotes the distribution of synthetic data, which may differ from the real distribution $p_{\mathbf{x}}$. We are interested in upper-bounding the generalization error of the algorithm that minimizes the mixed-loss. Our approach leverages a strategy from the learning theory literature known as algorithmic stability.

Definition 3.1 (Uniform Stability). *Let $\mathcal{A} : \mathcal{S} \mapsto \mathcal{H}$ denote an algorithm. Algorithm $\mathcal{A}$ is ε-uniformly stable if for all $\mathbf{S}, \mathbf{S}' \in \mathcal{X}^N$ such that $\mathbf{S}, \mathbf{S}'$ differ in at most one example, the corresponding outputs $\mathcal{A}(\mathbf{S})$ and $\mathcal{A}(\mathbf{S}')$ satisfy $\sup_{\mathbf{x} \in \mathcal{X}} |\ell(\mathcal{A}(\mathbf{S}); \mathbf{x}) - \ell(\mathcal{A}(\mathbf{S}'); \mathbf{x})| \leq \varepsilon$.*

This notion of algorithmic stability captures sensitivity of an algorithm on individual changes in the dataset. Under this property, it has been shown that generalization gap bounds in both expectation and high probability can be obtained (Bousquet & Elisseeff, 2000; 2002).

In our analysis we consider the general case where $\mathcal{H}$ consists of set of functions between $\mathcal{X}$ and some metric space $\mathcal{Y}$. We make the assumption that it is a compact subset $L^\infty(p_{\mathrm{x}})$.

Assumption 3.1. *The hypothesis class $\mathcal{H}$ is a set of measurable functions of the form $\mathcal{X} \to \mathcal{Y}$ and there exists $D > 0$ such that for any $h, h' \in \mathcal{H}$, $\|h - h'\|_{L^\infty(p_{\mathrm{x}})} \leq D$.*

Standard generalization bounds (e.g., Russo & Zou (2020); Lopez & Jog (2018); Clerico et al. (2022)) rely on regularity conditions on the loss function ℓ. We now recall the regularity conditions adopted in this work. We recall that a differentiable function $\phi : \mathcal{Y} \to \mathbb{R}$ is m-strongly convex for some constant $m > 0$ if it satisfies $\phi(x) \geq \phi(y) + \langle \nabla\phi(y), y - x\rangle + \frac{m}{2}\|x - y\|^2$ and is M-smooth if it satisfies $\phi(x) \leq \phi(y) + \langle \nabla\phi(y), y - x\rangle + \frac{M}{2}\|x - y\|^2$.

Assumption 3.2. *The loss function takes the form* $\ell(h, \mathrm{x}) = c(h(\mathrm{x}), \mathrm{x})$ *for a function* $c : \mathcal{Y} \times \mathcal{X} \to \mathbb{R}^+$*, where for every* $\mathrm{x} \in \mathcal{X}$*, the function* $c(\cdot, \mathrm{x})$ *is differentiable,* m*-strongly convex,* M_1*-smooth and satisfies* $\inf_{y\in\mathcal{Y}} c(y, \mathrm{x}) = 0$*. Furthermore, for any* $y \in \mathcal{Y}$*,* $c(y, \cdot)$ *is* M_2*-smooth.*

This is satisfied by many common learning objectives, including regression with mean squared error and classification with cross-entropy loss. Furthermore, the use of smoothness and strong-convexity is standard within algorithmic stability and generalization (e.g., Bousquet & Elisseeff (2002; 2000); Charles & Papailiopoulos (2018); Bousquet et al. (2019); Yang et al. (2023); Shalev-Shwartz et al. (2010); Feldman & Vondrák (2019); Attia & Koren (2022); Farghly & Rebeschini (2021)).

We now state a result showing that the mixed-loss algorithm is uniformly stable and provides a bound on the generalization gap.

Theorem 3.1 (Mixed-data Generalization Bound)**.** *Let* $\mathcal{H}$ *be a class of* L*-Lipschitz functions. Suppose Assumptions 3.1, and 3.2 hold and let* $h_{\mathbf{S}} \in \arg\min_{h\in\mathcal{H}} \mathcal{R}_\lambda(h, \mathbf{S})$*. Then, there exists a universal constant* $C > 0$ *and a sample size threshold* $N_0 > 0$ *such that for all* $N \geq N_0$*, the algorithm* $\mathcal{A}(\mathbf{S}) = h_{\mathbf{S}}$ *is uniformly stable with stability constant*

$$\varepsilon \lesssim \frac{1}{\lambda}\mathcal{R}_\lambda(h_{\mathbf{S}}) + C\xi\left(\frac{M_1}{m^2L^2\lambda}\mathcal{R}_\lambda(h_{\mathbf{S}}) + \frac{\sqrt{M_1M_2}(1-\lambda)D^2}{mL^2\lambda N}\right)^{\frac{1}{d_\star+1}},$$

where $d_\star$ *denotes the upper packing dimension of the measure* p'_{x} *(see Appendix F.2 for details),* $\xi = M_1L^2 + M_2$*,* $\eta = M_1/m^2$*, and* $\tau = \sqrt{M_1M_2}/m$*. Let* $r^\star = \min_{h\in\mathcal{H}} r(h)$ *be the true population risk minimizer. For any* $\lambda \in (0, 1)$*, the generalization gap satisfies*

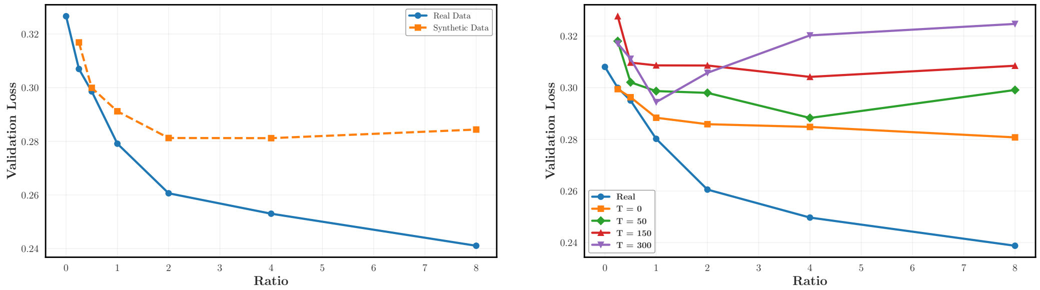
$$\mathbb{E}[r(h_{\mathbf{S}})] - r^\star \lesssim \lambda\xi\mathcal{W}_2\left(p_{\mathbf{x}}, p'_{\mathbf{x}}\right)^2 + C(1-\lambda)\xi\left(\frac{\eta}{L^2\lambda}r^\star + \frac{\eta\xi}{L^2}\mathcal{W}_2\left(p_{\mathbf{x}}, p'_{\mathbf{x}}\right)^2 + \frac{\tau(1-\lambda)D^2}{L^2\lambda N}\right)^{\frac{1}{d_\star+1}}.$$

The proof of Theorem 3.1, along with another result of stability for the mixed loss is discussed in Appendix F. The packing dimension $d_\star$ can be intuitively understood as the intrinsic dimension of the real data manifold. Notably, the generalization bound exhibits a U-shaped dependence on λ, similar to Theorem 2.2: for a fixed distributional discrepancy $\mathcal{W}_2\left(p_{\mathbf{x}}, p'_{\mathbf{x}}\right)$, there exists an optimal mixing parameter λ. This reflects a trade-off between algorithmic stability (which improves with more synthetic data) and distributional mismatch. In particular, when $\mathcal{W}_2\left(p_{\mathbf{x}}, p'_{\mathbf{x}}\right) = 0$, the optimal λ is 1, suggesting that it is beneficial to generate as much synthetic data as possible. We refer to the ratio $\frac{\lambda}{1-\lambda}$ as the *synthetic-to-real* ratio, which approximates $\frac{M}{N}$ in the finite-sample setting.

4 Experiments: Real-World Medical Images

Multiple sclerosis (MS) is a chronic neurological disease affecting millions worldwide (Tullman, 2013). T2-hyperintense lesions in MRI reflect neuroinflammatory damage and serve as key biomarkers for diagnosis, monitoring, and prognosis (McGinley et al., 2021). Accurate segmentation of MS lesions in MRI remains a challenging problem. ML methods must contend with substantial variability in image characteristics, lesion appearance, and domain shift, arising from differences in scanners, acquisition protocols, and imaging parameters between training and test sets (Zeng et al., 2020). Furthermore, publicly available datasets for lesion segmentation remain limited in size and diversity, as acquiring labeled, heterogeneous MRI data is both costly and time-consuming.

Therefore, our work is motivated by the need to improve lesion segmentation performance under limited and heterogeneous training data and distributional shift. Specifically, we consider the setting where a synthetic data generator is available to address data scarcity and mitigate domain shift between the source (training) and target (test) domains, while potentially introducing additional distributional discrepancies. Following our theoretical result in Section 3, we study the in-domain setup in this section and refer to Section 5 for out-of-domain scenario.



(a) The effect of synthetic data.

(b) The effect of distributional distance.

Figure 2: *(a)* Validation loss decreases consistently as more real data is added (blue line), while increasing synthetic data (orange dashed line) produces a U-shaped curve, indicating an optimal mixing ratio λ, as predicted by Theorem 3.1. *(b)* Effect of distributional distance: varying the diffusion model timestep $T \in 0, 50, 150, 300$ controls the noise level of synthetic samples. The U-shaped trend persists across all T but becomes sharper with increased discrepancy between real and synthetic distributions, further supporting Theorem 3.1.

We conduct our experiments on the *NO.MS* dataset (Dahlke et al., 2021), one of the largest and most comprehensive clinical trial datasets for MS. It comprises over 200,000 MRI scans from more than 11,000 patients. Ground-truth lesion annotations are generated using an automated tool and subsequently refined by expert radiologists. The data originates from two Contract Research Organizations (CROs), NeuroRx and MIAC, introducing inherent domain variability. For the downstream segmentation task, we use a training set of 100 NeuroRx scans ($\sim$4,500 slices) and a fixed validation set of 20 NeuroRx scans ($\sim$1,000 slices). In addition, we train a conditional diffusion model on NeuroRx data as our synthetic data generator. To empirically validate the theoretical insights from Theorem 3.1, we design two experiments:

1. **Effect of synthetic data:** We augment the training set by varying the synthetic-to-real ratio from 0.25 to 8, and compare performance against a control case in which the real dataset is scaled up accordingly.
2. **Distributional distance:** While we do not have direct access to the distance between the true and synthetic distributions, we study the effect of this discrepancy by varying the sampling timestep of the diffusion model ($T = 50$, 150, 300) out of 600 total denoising steps. We expect that samples from noisier timesteps exhibit greater distributional distance from the real data.

More details on the segmentation model architecture and hyperparameters are provided in Appendix H.3. Figure 2 shows that an appropriately chosen synthetic-to-real data ratio improves performance on the downstream segmentation task. Figure 2a compares validation loss when increasing the amount of synthetic data versus scaling up real data. As expected, adding more real data consistently improves performance. In contrast, synthetic data exhibits a U-shaped effect: moderate amounts enhance generalization, while excessive amounts degrade it, indicating the existence of an optimal interpolation parameter λ.

Figure 2b further examines how this behavior depends on the distributional distance between real and synthetic data. By varying the diffusion timestep T which controls the noise level in generated samples, we observe that the U-shape persists but becomes sharper as the synthetic data diverges further from the real distribution. These findings support our results that the generalization gap is influenced by both the mixing ratio and the distributional discrepancy between data sources. The relationship between the optimal synthetic-to-real ratio and distributional distance is further illustrated in Figure 8 in Appendix H.3.

5 Synthetic data for domain adaptation

In this section, we study the learning problem under *domain shift* (Zhang et al., 2019; Stacke et al., 2019; Redko et al., 2020; Shui et al., 2022): the real training data consist of samples from a source domain $\mathcal{X}$ with distribution $p_{\mathbf{x}}$, while the goal is to evaluate the learned model on a distinct target domain $\mathcal{X}^\star$ with distribution $p_{\mathbf{x}}^\star$, from which no real data are available. To address this distribution mismatch, we assume access to synthetic data generated on the target domain $\mathcal{X}^\star$, though drawn from a potentially imperfect distribution $p'_{\mathbf{x}} \neq p^\star_{\mathbf{x}}$. As in previous sections, this synthetic data is used to regularize the ERM objective, aiming to improve generalization to the target domain in the absence of real samples from $p^\star_{\mathbf{x}}$.

We first analyze this setting within the kernel framework (Section 2). Specifically, we consider a dataset of N real training pairs $\mathbf{y}_n = \tilde{f}(\mathbf{x}_n) + \varepsilon_n$, and a synthetic data generator g as defined earlier. The test error is measured with respect to a ground truth function $f_\star$. The main difference from Section 2 is that the training function $\tilde{f}$ differs from $f_\star$, capturing the domain shift. Although the empirical estimator remains unchanged (Equation 1), the generalization behavior is affected by the discrepancy between the training and target domains. Our result shows that stronger regularization can improve performance when the synthetic data more accurately approximates the target domain than the source data, providing a principled guideline for tuning λ under domain shift. The bound is formalized below; see Appendix G.1 for the proof.

Theorem 5.1 (Generalization under Domain Shift). *Under Assumption 2.1, for the kernel regression problem defined in Equation 1 and any fixed regularization parameter $\lambda \geq 0$, the test error under domain shift satisfies the bound:*

$$\mathcal{R}_N(\lambda; g) = \mathcal{O}\left(\mu_{\max}\lambda^{r+1}\mathcal{D}(f_\star, \tilde{f}) + \lambda^{\max\{2-\frac{1}{4r}, r+1\}}\mathcal{D}(f_\star, g) + \frac{\sigma^2}{N}\lambda^{-\frac{1}{2r}}\right),$$

where $\mu_{\max} = \max_j \mu_j$, and $\mathcal{D}(\cdot,\cdot)$ denotes the distributional discrepancy, as in Theorem 2.2.

We now extend this result to the setup in Section 3, where test error is measured with respect to $p^\star_{\mathbf{x}}$. The resulting generalization gap is stated below; see Appendix G.2 for the proof.

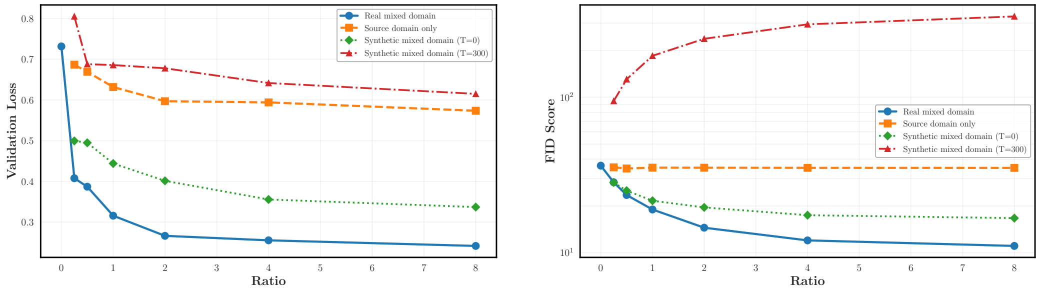
Theorem 5.2 (Mixed-data Generalization under Domain Shift). *Let $\mathcal{H}$ be a class of L-Lipschitz functions. Suppose Assumptions 3.1, and 3.2 hold and let $h_{\mathbf{S}} \in \arg\min_{h\in\mathcal{H}} \mathcal{R}_\lambda(h, \mathbf{S})$. Then, for any $\lambda \in (0, 1)$, the generalization gap under the domain shift satisfies*

$$\mathbb{E}[r(h_{\mathbf{S}})] - r^\star \lesssim \lambda\xi\mathcal{W}_2\left(p^\star_{\mathbf{x}}, p'_{\mathbf{x}}\right)^2 + (1-\lambda)\xi\mathcal{W}_2\left(p^\star_{\mathbf{x}}, p_{\mathbf{x}}\right)^2$$
$$+ C(1-\lambda)\xi\left(\frac{M_1}{m^2L^2\lambda}r^\star + \frac{M_1\xi}{m^2L^2}\mathcal{W}_2\left(p^\star_{\mathbf{x}}, p_{\mathbf{x}}\right)^2 + \frac{\sqrt{M_1M_2}(1-\lambda)D^2}{mL^2\lambda N}\right)^{\frac{1}{d_\star+1}}$$

where $r^\star = \min_{h\in\mathcal{H}} r^\star(h)$ is the true population risk minimizer of the target domain.

As expected, compared to Theorems 2.2 and 3.1, these bounds include an additional term that captures the mismatch between the source and target distributions. Consequently, the optimal choice of λ depends on the relative magnitudes of $\mathcal{D}(f_\star, \tilde{f})$ and $\mathcal{D}(f_\star, g)$ or similarly $\mathcal{W}_2\left(p^\star_{\mathbf{x}}, p_{\mathbf{x}}\right)$ and $\mathcal{W}_2\left(p^\star_{\mathbf{x}}, p'_{\mathbf{x}}\right)$. Intuitively, when the synthetic data generator more closely approximates the target domain, it is beneficial to choose a larger regularization parameter λ. In the special case where $f_\star = \tilde{f}$ or $\mathcal{W}_2\left(p^\star_{\mathbf{x}}, p_{\mathbf{x}}\right) = 0$, the bound reduces to the previous results, albeit with potentially larger constants for the kernel regression case, arising from the more general proof strategy.

Experimental setup We follow the experimental setup for medical brain MRI scans described in Section 4 to study the effect of domain shift. Since we have two data sources (MIAC and NeuroRx), we can naturally adapt the setup to introduce domain shift: we treat MIAC as the source domain and NeuroRx as the target domain. The synthetic data generator, a conditional diffusion model, is trained on NeuroRx, thus approximating the target distribution. As before, we vary the synthetic-to-real data ratio in the range 0.25 to 8, and compare the resulting performance. Results are shown in Figure 3a. We include two baselines in this experiment: (1) access to real data from the target domain for training the downstream segmentation task on NeuroRx (blue line), and (2) no access to either target or synthetic data, with only increased source domain data available (orange line). To examine the impact of distributional discrepancy between synthetic and target data, we adopt the same approach as in Section 4, sampling from the diffusion model at two timesteps, $T = 0$ and $T = 300$. We expect $T = 0$ (green dashed line) to closely match the target distribution, while $T = 300$ (red dashed line) reflects a greater distributional distance. As observed, synthetic data can significantly improve performance when the distributional distance between the synthetic and target is small. However, when the synthetic generator induces a large distributional shift, using additional source data alone can be more effective, but if the source domain itself is far from the target, neither synthetic nor source data is likely to help. This observation aligns with our theoretical understanding of the trade-offs in generalization error, where the benefit of additional data depends critically on the distributional closeness to the target domain.

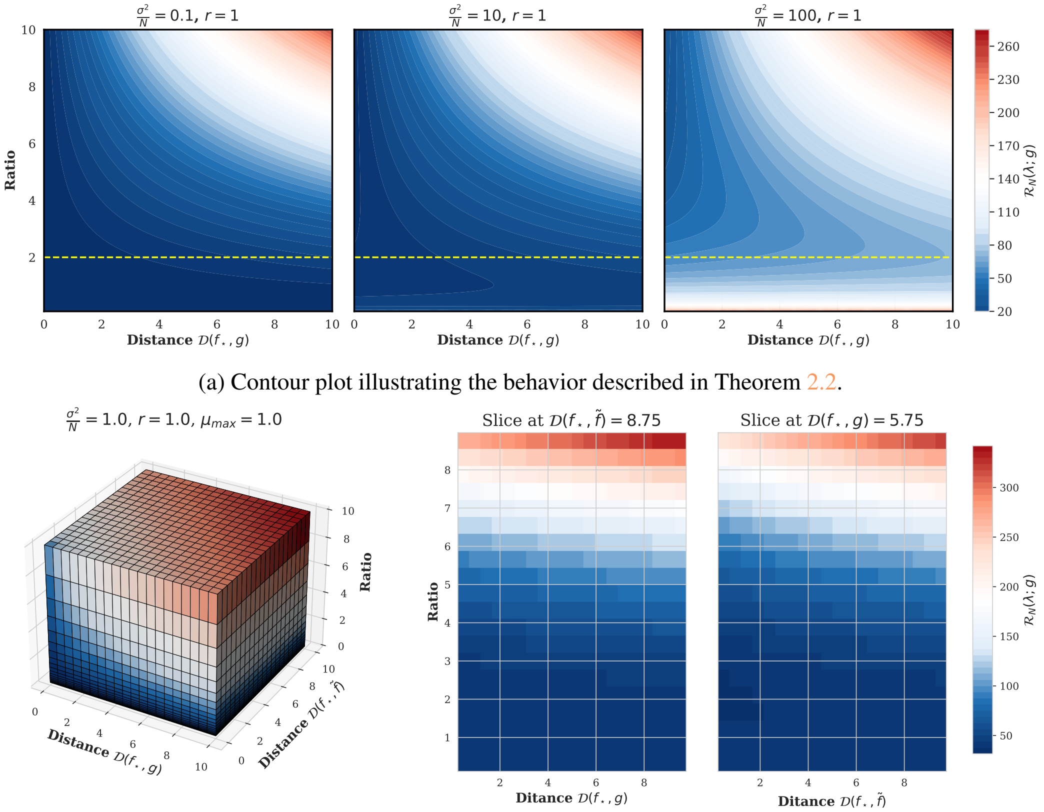


(a) The effect of synthetic data in domain shift.

(b) FID score vs. synthetic-to-real data ratio

Figure 3: *(a)* Effect of synthetic data from distributions close to (green dashed) or far from (red dashed) the target, compared to target (blue) and source (orange) baselines. Results show the trade-off between distributional shift and regularization predicted by Theorem 5.2. *(b)* FID as a proxy for distributional shift: $T = 0$ (green) aligns with the target, while noisy (red) and source (orange) data show higher FID and reduced utility.

6 Insights for practitioners



(a) Contour plot illustrating the behavior described in Theorem 2.2.

(b) Contour plot illustrating the behavior described in Theorem 5.1.

Figure 4: Effect of the synthetic-to-real data ratio and distributional distance(s) on the error rate: *(a)* in-domain scenario across various signal-to-noise ratios for the real dataset; *(b)* out-of-domain scenario. In both cases, one should ideally choose λ such that it lies within the blue regions, which correspond to lower error rates.

To apply our theoretical results in practice, practitioners must estimate key quantities that influence the generalization bound, such as distributional distances, noise levels, and the complexity of the hypothesis class. This section provides practical guidance on how to approximate these quantities and offers heuristic strategies based on empirical observations.

Exact distributional distances between real and synthetic data, or between source and target domains, are typically unavailable. Therefore, practical applications require accessible proxies. In our experiments (Sections 4 and 5), we initially used diffusion timesteps as a proxy. Here, we explore a more broadly applicable alternative: the widely adopted Fréchet Inception Distance (FID) metric. Although FID does not measure true distributional distance, it offers a practical approximation: computing the Wasserstein distance between multivariate Gaussians fitted to the Inception embeddings of real and generated images. When the synthetic distribution closely aligns with the target domain, FID serves as a useful tool for comparing generators or estimating alignment. As shown in Figure 3b, adding synthetic data from $T = 0$ (green) reduces the FID score in a manner similar to adding real target-domain data (blue), reflecting performance trends seen in Figure 3a. In contrast, source-domain data or noisy diffusion samples either leave FID unchanged or worsen it. Depending on the application, other approximations such as cross-validation loss or Kullback-Leibler Divergence (KLD) may also be used.

To guide empirical decisions, we visualize our theoretical bounds in terms of two user-controllable factors: data heterogeneity and the distance between real and synthetic distributions. Heterogeneity is captured by the ratio σ^2/N, reflecting data variance and sample size, and plays a key role in practical experiments. Practitioners can have control over heterogeneity through N. Figure 4 illustrates how the bound changes under different scenarios, offering heuristic strategies for choosing the optimal mix of real and synthetic data. In particular, lower bounds (shown in cooler colors) indicate better generalization, while higher bounds (in red) highlight settings that risk overfitting, and should be avoided. See Appendix H.4 for additional results.

In-domain. When data heterogeneity is small to moderate and the generator produces high-quality synthetic data (i.e., $D(f_*, g)$ is small to moderate), augmenting the dataset with up to twice the amount of real data is an effective strategy (Figure 4a). While adding more synthetic data can slightly reduce the bound, the marginal gains diminish, and additional data primarily increases computational cost without notable improvements in downstream performance. Even when heterogeneity is high, e.g., small N and high σ^2, as often encountered in biomedical datasets such as brain MRI, a 1:2 real-to-synthetic ratio remains a reasonable choice. In such cases, the generalization bound is naturally higher, yet augmenting with synthetic data still provides substantial benefit. However, further increases in the synthetic ratio are only effective if the generator is exceptionally accurate. Otherwise, collecting more real data to reduce heterogeneity is a better strategy.

Out-of-domain. In domain shift scenarios (Figure 4b), the same 1:2 ratio remains an effective and robust choice, provided the synthetic data generator is of good quality. Unlike the in-domain case, increasing the synthetic ratio beyond this point can harm performance, especially when the domain shift is large. For severe shifts (e.g., $\mathcal{D}(f_\star, \tilde{f}) \approx 8.75$), even a 1:2 ratio works reasonably well, but larger ratios degrade performance. For moderate shifts (e.g., $\mathcal{D}(f_\star, g) \approx 5$), a 1:2 ratio continues to be a reliable default. Overall, we recommend ratios in the range of 1:1 to 1:2. While a good generator allows modest augmentation even under domain shift, excessive use of synthetic data, especially when not well aligned with the target distribution, can lead to degraded performance. Careful tuning of the augmentation ratio is thus crucial in the out-of-domain case.

Finally, our theory assumes Lipschitz continuity, which can be estimated via gradient norms or controlled by clipping. It scales the bound by a constant but does not affect the order of the optimal ratio.

7 Conclusion

The integration of synthetic data into machine learning is critical in domains where real data is limited, expensive, or sensitive, such as healthcare. Optimizing this integration is essential for improving generalization without distorting the true data distribution.

We present a principled learning-theoretic framework that characterizes the trade-off between real and synthetic data. Under standard regularity assumptions, we prove the existence of a non-trivial optimal synthetic-to-real ratio that minimizes generalization error, first in kernel ridge regression, then more broadly using algorithmic stability. Empirical results on both benchmark and real-world datasets validate our predictions, revealing a non-monotonic relationship between performance and synthetic data proportion. Our results extend to domain adaptation scenarios involving distribution shift, demonstrating its broader applicability and underscoring the importance of data balancing strategies in real-world, data-constrained machine learning pipelines. We provide a heuristic guideline based on our theoretical and empirical results for practitioners.

References

Sina Alemohammad, Josue Casco-Rodriguez, Lorenzo Luzi, Ahmed Imtiaz Humayun, Hossein Babaei, Daniel LeJeune, Ali Siahkoohi, and Richard G. Baraniuk. Self-consuming generative models go MAD. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024a. URL https://openreview.net/forum?id=ShjMHfmPs0.

Sina Alemohammad, Ahmed Imtiaz Humayun, Shruti Agarwal, John Collomosse, and Richard Baraniuk. Self-improving diffusion models with synthetic data. *arXiv preprint arXiv:2408.16333*, 2024b.

Oskar Allerbo. Solving kernel ridge regression with gradient descent for a non-constant kernel. *arXiv preprint arXiv:2311.01762*, 2023.

Martin Anthony and Peter L. Bartlett. *Neural Network Learning - Theoretical Foundations*. Cambridge University Press, 2002. ISBN 978-0-521-57353-5. URL http://www.cambridge.org/gb/knowledge/isbn/item1154061/?site_locale=en_GB.

Amir-Reza Asadi, Emmanuel Abbe, and Sergio Verdú. Chaining mutual information and tightening generalization bounds. In Samy Bengio, Hanna M. Wallach, Hugo Larochelle, Kristen Grauman, Nicolò Cesa-Bianchi, and Roman Garnett (eds.), *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pp. 7245–7254, 2018. URL https://proceedings.neurips.cc/paper/2018/hash/8d7628dd7a710c8638dbd22d4421ee46-Abstract.html.

Amit Attia and Tomer Koren. Uniform stability for first-order empirical risk minimization. In Po-Ling Loh and Maxim Raginsky (eds.), *Conference on Learning Theory, 2-5 July 2022, London, UK*, volume 178 of *Proceedings of Machine Learning Research*, pp. 3313–3332. PMLR, 2022. URL https://proceedings.mlr.press/v178/attia22a.html.

Shekoofeh Azizi, Laura Culp, Jan Freyberg, Basil Mustafa, Simon Baur, Simon Kornblith, Ting Chen, Nenad Tomasev, Jovana Mitrović, Patricia Strachan, et al. Synthetic data in healthcare. *arXiv preprint arXiv:2306.08037*, 2023.

Peter L. Bartlett, Philip M. Long, Gábor Lugosi, and Alexander Tsigler. Benign overfitting in linear regression. *CoRR*, abs/1906.11300, 2019. URL http://arxiv.org/abs/1906.11300.

Daniel Barzilai and Ohad Shamir. Generalization in kernel regression under realistic assumptions. In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=PY3bKuorBI.

Mikhail Belkin, Siyuan Ma, and Soumik Mandal. To understand deep learning we need to understand kernel learning. In Jennifer G. Dy and Andreas Krause (eds.), *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, volume 80 of *Proceedings of Machine Learning Research*, pp. 540–548. PMLR, 2018. URL http://proceedings.mlr.press/v80/belkin18a.html.

Shai Ben-David, John Blitzer, Koby Crammer, and Fernando Pereira. A theory of learning from different domains. *Machine learning*, 79(1):151–175, 2010.

Quentin Bertrand, Avishek Joey Bose, Alexandre Duplessis, Marco Jiralerspong, and Gauthier Gidel. On the stability of iterative retraining of generative models on their own data. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria,* *May 7-11, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=JORAfH2xFd.

Eyal Betzalel, Coby Penso, Aviv Navon, and Ethan Fetaya. A study on the evaluation of generative models. *arXiv preprint arXiv:2206.10935*, 2022.

Olivier Bousquet and André Elisseeff. Algorithmic stability and generalization performance. In Todd K. Leen, Thomas G. Dietterich, and Volker Tresp (eds.), *Advances in Neural Information Processing Systems 13, Papers from Neural Information Processing Systems (NIPS) 2000, Denver, CO, USA*, pp. 196–202. MIT Press, 2000. URL https://proceedings.neurips.cc/paper/2000/hash/49ad23d1ec9fa4bd8d77d02681df5cfa-Abstract.html.

Olivier Bousquet and André Elisseeff. Stability and generalization. *J. Mach. Learn. Res.*, 2:499–526, 2002. URL https://jmlr.org/papers/v2/bousquet02a.html.

Olivier Bousquet, Stéphane Boucheron, and Gábor Lugosi. Introduction to statistical learning theory. In Olivier Bousquet, Ulrike von Luxburg, and Gunnar Rätsch (eds.), *Advanced Lectures on Machine Learning, ML Summer Schools 2003, Canberra, Australia, February 2-14, 2003, Tübingen, Germany, August 4-16, 2003, Revised Lectures*, volume 3176 of *Lecture Notes in Computer Science*, pp. 169–207. Springer, 2003. doi: 10.1007/978-3-540-28650-9_8. URL https://doi.org/10.1007/978-3-540-28650-9_8.

Olivier Bousquet, Yegor Klochkov, and Nikita Zhivotovskiy. Sharper bounds for uniformly stable algorithms. *CoRR*, abs/1910.07833, 2019. URL http://arxiv.org/abs/1910.07833.

Christopher Bowles, Liang Chen, Ricardo Guerrero, Paul Bentley, Roger Gunn, Alexander Hammers, David A Dickie, Maria Valdés Hernández, Joanna Wardlaw, and Daniel Rueckert. Gan augmentation: Augmenting training data using generative adversarial networks. *arXiv preprint arXiv:1810.10863*, 2018. URL https://arxiv.org/abs/1810.10863.

Martin Briesch, Dominik Sobania, and Franz Rothlauf. Large language models suffer from their own output: An analysis of the self-consuming training loop. *CoRR*, abs/2311.16822, 2023. doi: 10.48550/ARXIV.2311.16822. URL https://doi.org/10.48550/arXiv.2311.16822.

Lorenzo Brigato and Luca Iocchi. A close look at deep learning with small data. In *2020 25th international conference on pattern recognition (ICPR)*, pp. 2490–2497. IEEE, 2021.

Aaron Carass, Snehashis Roy, Amod Jog, Jennifer L. Cuzzocreo, Elizabeth M. Sweeney, Adrian Gherman, Julia Button, James Nguyen, Ferran Prados, Carole H. Sudre, Manuel Jorge Cardoso, Niamh Cawley, Olga Ciccarelli, Claudia A. M. Wheeler-Kingshott, Sébastien Ourselin, Laurence Catanese, Hrishikesh Deshpande, Pierre Maurel, Olivier Commowick, Christian Barillot, and Xavier Tomas-Fernandez. Longitudinal multiple sclerosis lesion segmentation: Resource and challenge. *NeuroImage*, 148:77–102, 2017. doi: 10.1016/J.NEUROIMAGE.2016.12.064. URL https://doi.org/10.1016/j.neuroimage.2016.12.064.

Zachary Charles and Dimitris S. Papailiopoulos. Stability and generalization of learning algorithms that converge to global optima. In Jennifer G. Dy and Andreas Krause (eds.), *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, volume 80 of *Proceedings of Machine Learning Research*, pp. 744–753. PMLR, 2018. URL http://proceedings.mlr.press/v80/charles18a.html.

Tin Sum Cheng, Aurélien Lucchi, Anastasis Kratsios, and David Belius. A comprehensive analysis on the learning curve in kernel ridge regression. *CoRR*, abs/2410.17796, 2024. doi: 10.48550/ARXIV.2410.17796. URL https://doi.org/10.48550/arXiv.2410.17796.

Eugenio Clerico, Amitis Shidani, George Deligiannidis, and Arnaud Doucet. Chained generalisation bounds. In Po-Ling Loh and Maxim Raginsky (eds.), *Conference on Learning Theory, 2-5 July 2022, London, UK*, volume 178 of *Proceedings of Machine Learning Research*, pp. 4212–4257. PMLR, 2022. URL https://proceedings.mlr.press/v178/clerico22a.html.

Andoni Cortés, Clemente Rodríguez, Gorka Vélez, Javier Barandiarán, and Marcos Nieto. Analysis of classifier training on synthetic data for cross-domain datasets. *CoRR*, abs/2410.22748, 2024. doi: 10.48550/ARXIV.2410.22748. URL https://doi.org/10.48550/arXiv.2410.22748.

Ekin D Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V Le. Randaugment: Practical automated data augmentation with a reduced search space. *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pp. 702–703, 2020.

Hugo Cui, Bruno Loureiro, Florent Krzakala, and Lenka Zdeborová. Generalization error rates in kernel regression: The crossover from the noiseless to noisy regime. *CoRR*, abs/2105.15004, 2021. URL https://arxiv.org/abs/2105.15004.

F. Dahlke, D. L. Arnold, P. Aarden, H. Ganjgahi, D. A. Häring, J. Čuklina, T. E. Nichols, S. Gardiner, R. Bermel, and H. Wiendl. Characterisation of MS phenotypes across the age span using a novel data set integrating 34 clinical trials (NO.MS cohort): Age is a key contributor to presentation. *Multiple Sclerosis Journal*, 27(13):2062–2076, 2021.

Valentin De Bortoli, James Thornton, Jeremy Heng, and Arnaud Doucet. Diffusion schrödinger bridge with applications to score-based generative modeling. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, volume 34, pp. 17695–17709. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper_files/paper/2021/file/940392f5f32a7ade1cc201767cf83e31-Paper.pdf.

Elvis Dohmatob, Yunzhen Feng, and Julia Kempe. Model collapse demystified: The case of regression. *CoRR*, abs/2402.07712, 2024. doi: 10.48550/ARXIV.2402.07712. URL https://doi.org/10.48550/arXiv.2402.07712.

Elvis Dohmatob, Yunzhen Feng, Arjun Subramonian, and Julia Kempe. Strong model collapse. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=et5l9qPUhm.

Cynthia Dwork and Aaron Roth. The algorithmic foundations of differential privacy. *Found. Trends Theor. Comput. Sci.*, 9(3-4):211–407, 2014. doi: 10.1561/0400000042. URL https://doi.org/10.1561/0400000042.

Andre Esteva, Alexandre Robicquet, Bharath Ramsundar, Volodymyr Kuleshov, Mark DePristo, Katherine Chou, Claire Cui, Greg Corrado, Sebastian Thrun, and Jeffrey Dean. A guide to deep learning in healthcare. *Nature Medicine*, 25(1):24–29, 2019. URL https://www.nature.com/articles/s41591-018-0316-z.

Tyler Farghly and Patrick Rebeschini. Time-independent generalization bounds for sgld in non-convex settings. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, volume 34, pp. 19836–19846. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper_files/paper/2021/file/a4ee59dd868ba016ed2de90d330acb6a-Paper.pdf.

Vitaly Feldman and Jan Vondrák. High probability generalization bounds for uniformly stable algorithms with nearly optimal rate. *CoRR*, abs/1902.10710, 2019. URL http://arxiv.org/abs/1902.10710.

Damien Ferbach, Quentin Bertrand, Avishek Joey Bose, and Gauthier Gidel. Self-consuming generative models with curated data provably optimize human preferences. *CoRR*, abs/2407.09499, 2024. doi: 10.48550/ARXIV.2407.09499. URL https://doi.org/10.48550/arXiv.2407.09499.

Maayan Frid-Adar, Eyal Klang, Michal Amitai, Jacob Goldberger, and Hayit Greenspan. Gan-based synthetic medical image augmentation for increased cnn performance in liver lesion classification. *arXiv preprint arXiv:1803.01229*, 2018. URL https://arxiv.org/abs/1803.01229.

Borja Rodríguez Gálvez, Germán Bassi, Ragnar Thobaben, and Mikael Skoglund. On random subset generalization error bounds and the stochastic gradient langevin dynamics algorithm. *CoRR*, abs/2010.10994, 2020. URL https://arxiv.org/abs/2010.10994.

Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *The journal of machine learning research*, 17(1):2096–2030, 2016.

Federica Gerace, Luca Saglietti, Stefano Sarao Mannelli, Andrew M. Saxe, and Lenka Zdeborová. Probing transfer learning with a model of synthetic correlated datasets. *Mach. Learn. Sci. Technol.*, 3(1):15030, 2022. doi: 10.1088/2632-2153/AC4F3F. URL https://doi.org/10.1088/2632-2153/ac4f3f.

Matthias Gerstgrasser, Rylan Schaeffer, Apratim Dey, Rafael Rafailov, Henry Sleight, John Hughes, Tomasz Korbak, Rajashree Agrawal, Dhruv Pai, Andrey Gromov, Daniel A. Roberts, Diyi Yang, David L. Donoho, and Sanmi Koyejo. Is model collapse inevitable? breaking the curse of recursion by accumulating real and synthetic data. *CoRR*, abs/2404.01413, 2024. doi: 10.48550/ARXIV.2404.01413. URL https://doi.org/10.48550/arXiv.2404.01413.

Benjamin Guedj. A primer on pac-bayesian learning. *CoRR*, abs/1901.05353, 2019. URL http://arxiv.org/abs/1901.05353.

Mahdi Haghifam, Jeffrey Negrea, Ashish Khisti, Daniel M. Roy, and Gintare Karolina Dziugaite. Sharpened generalization bounds based on conditional mutual information and an application to noisy, iterative algorithms. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin (eds.), *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/712a3c9878efeae8ff06d57432016ceb-Abstract.html.

Tomas Hansson, Chris Oostenbrink, and WilfredF van Gunsteren. Molecular dynamics simulations. *Current opinion in structural biology*, 12(2):190–196, 2002.

Moritz Hardt, Ben Recht, and Yoram Singer. Train faster, generalize better: Stability of stochastic gradient descent. In Maria Florina Balcan and Kilian Q. Weinberger (eds.), *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pp. 1225–1234, New York, New York, USA, 20–22 Jun 2016. PMLR.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *arXiv preprint arXiv:2006.11239*, 2020. URL https://arxiv.org/abs/2006.11239.

Scott A Hollingsworth and Ron O Dror. Molecular dynamics simulation for all. *Neuron*, 99(6): 1129–1143, 2018.

Songyan Hou, Parnian Kassraie, Anastasis Kratsios, Andreas Krause, and Jonas Rothfuss. Instance-dependent generalization bounds via optimal transport. *J. Mach. Learn. Res.*, 24:349:1–349:51, 2023. URL http://jmlr.org/papers/v24/22-1293.html.

Benedikt T. Imbusch, Max Schwarz, and Sven Behnke. Synthetic-to-real domain adaptation using contrastive unpaired translation. In *18th IEEE International Conference on Automation Science and Engineering, CASE 2022, Mexico City, Mexico, August 20-24, 2022*, pp. 595–602. IEEE, 2022. doi: 10.1109/CASE49997.2022.9926640. URL https://doi.org/10.1109/CASE49997.2022.9926640.

Ayush Jain, Andrea Montanari, and Eren Sasoglu. Scaling laws for learning with real and surrogate data. *CoRR*, abs/2402.04376, 2024. doi: 10.48550/ARXIV.2402.04376. URL https://doi.org/10.48550/arXiv.2402.04376.

Georgios Kaissis, Alexander Ziller, Jonathan Passerat-Palmbach, Théo Ryffel, Dmitrii Usynin, Andrew Trask, Ivan Lima Jr, Jason Mancuso, Friederike Jungmann, Marc-Matthias Steinborn, et al. Secure, privacy-preserving and federated machine learning in medical imaging. *Nature Machine Intelligence*, 3(6):473–484, 2021.

Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan. *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 8110–8119, 2020.

Yoonsik Kim, Jae Woong Soh, Gu Yong Park, and Nam Ik Cho. Transfer learning from synthetic to real-noise denoising with adaptive instance normalization. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pp. 3479–3489. Computer Vision Foundation / IEEE, 2020. doi: 10.1109/CVPR42600.2020.00354.

URL https://openaccess.thecvf.com/content_CVPR_2020/html/Kim_Transfer_Learning_From_Synthetic_to_Real-Noise_Denoising_With_Adaptive_Instance_CVPR_2020_paper.html.

George Kimeldorf and Grace Wahba. Some results on tchebycheffian spline functions. *Journal of mathematical analysis and applications*, 33(1):82–95, 1971.

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. Technical report, Citeseer, 2009.

Hyungtae Lee, Yan Zhang, Heesung Kwon, and Shuvra S. Bhattacharrya. Exploring the potential of synthetic data to replace real data. *CoRR*, abs/2408.14559, 2024. doi: 10.48550/ARXIV.2408.14559. URL https://doi.org/10.48550/arXiv.2408.14559.

Dongyue Li and Hongyang R. Zhang. Improved regularization and robustness for fine-tuning in neural networks. In Marc'Aurelio Ranzato, Alina Beygelzimer, Yann N. Dauphin, Percy Liang, and Jennifer Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pp. 27249–27262, 2021. URL https://proceedings.neurips.cc/paper/2021/hash/e4a93f0332b2519177ed55741ea4e5e7-Abstract.html.

Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022.

Adrian Tovar Lopez and Varun S. Jog. Generalization error bounds using wasserstein distances. In *IEEE Information Theory Workshop, ITW 2018, Guangzhou, China, November 25-29, 2018*, pp. 1–5. IEEE, 2018. doi: 10.1109/ITW.2018.8613445. URL https://doi.org/10.1109/ITW.2018.8613445.

Yingzhou Lu, Minjie Shen, Huazheng Wang, Xiao Wang, Capucine van Rechem, Tianfan Fu, and Wenqi Wei. Machine learning for synthetic data generation: a review. *arXiv preprint arXiv:2302.04062*, 2023.

Farhad Maleki, Katie Ovens, Rajiv Gupta, Caroline Reinhold, Alan Spatz, and Reza Forghani. Generalizability of machine learning models: quantitative evaluation of three methodological pitfalls. *Radiology: Artificial Intelligence*, 5(1):e220028, 2022.

David A. McAllester. Pac-bayesian model averaging. In Shai Ben-David and Philip M. Long (eds.), *Proceedings of the Twelfth Annual Conference on Computational Learning Theory, COLT 1999, Santa Cruz, CA, USA, July 7-9, 1999*, pp. 164–170. ACM, 1999. doi: 10.1145/307400.307435. URL https://doi.org/10.1145/307400.307435.

Marisa P McGinley, Carolyn H Goldschmidt, and Alexander D Rae-Grant. Diagnosis and treatment of multiple sclerosis: a review. *Jama*, 325(8):765–779, 2021.

J. Mercer. Functions of positive and negative type and their connection with the theory of integral equations. *Phil. Trans. Roy. Soc. Lond. A*, 209:415, 1909. doi: 10.1098/rsta.1909.0016.

Jacqueline LS Milne, Mario J Borgnia, Alberto Bartesaghi, Erin EH Tran, Lesley A Earl, David M Schauder, Jeffrey Lengyel, Jason Pierson, Ardan Patwardhan, and Sriram Subramaniam. Cryo-electron microscopy–a primer for the non-microscopist. *The FEBS journal*, 280(1):28–45, 2013.

Samarth Mishra, Rameswar Panda, Cheng Perng Phoo, Chun-Fu Richard Chen, Leonid Karlinsky, Kate Saenko, Venkatesh Saligrama, and Rogério Schmidt Feris. Task2sim: Towards effective pre-training and transfer from synthetic data. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, pp. 9184–9194. IEEE, 2022. doi: 10.1109/CVPR52688.2022.00898. URL https://doi.org/10.1109/CVPR52688.2022.00898.

Theodor Misiakiewicz and Basil Saeed. A non-asymptotic theory of kernel ridge regression: deterministic equivalents, test error, and GCV estimator. *CoRR*, abs/2403.08938, 2024. doi: 10.48550/ARXIV.2403.08938. URL https://doi.org/10.48550/arXiv.2403.08938.

Wenlong Mou, Yuchen Zhou, Jun Gao, and Liwei Wang. Dropout training, data-dependent regularization, and generalization bounds. In Jennifer G. Dy and Andreas Krause (eds.), *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, volume 80 of *Proceedings of Machine Learning Research*, pp. 3642–3650. PMLR, 2018. URL http://proceedings.mlr.press/v80/mou18a.html.

Koustav Mullick, Harshil Jain, Sanchit Gupta, and Amit Arvind Kale. Domain adaptation of synthetic driving datasets for real-world autonomous driving. *CoRR*, abs/2302.04149, 2023. doi: 10.48550/ARXIV.2302.04149. URL https://doi.org/10.48550/arXiv.2302.04149.

Kazuyoshi Murata and Matthias Wolf. Cryo-electron microscopy for structural analysis of dynamic biological macromolecules. *Biochimica et Biophysica Acta (BBA) - General Subjects*, 1862(2):324–334, 2018. ISSN 0304-4165. doi: https://doi.org/10.1016/j.bbagen.2017.07.020. URL https://www.sciencedirect.com/science/article/pii/S0304416517302374. Biophysical Exploration of Dynamical Ordering of Biomolecular Systems.

Xingchao Peng, Ben Usman, Kuniaki Saito, Neela Kaushik, Judy Hoffman, and Kate Saenko. Syn2real: A new benchmark forsynthetic-to-real visual domain adaptation. *CoRR*, abs/1806.09755, 2018. URL http://arxiv.org/abs/1806.09755.

Yury Polyanskiy and Yihong Wu. Wasserstein continuity of entropy and outer bounds for interference channels. *CoRR*, abs/1504.04419, 2015. URL http://arxiv.org/abs/1504.04419.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. *arXiv preprint arXiv:2103.00020*, 2021.

Maxim Raginsky, Alexander Rakhlin, and Matus Telgarsky. Non-convex learning via stochastic gradient langevin dynamics: a nonasymptotic analysis. *CoRR*, abs/1702.03849, 2017. URL http://arxiv.org/abs/1702.03849.

Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. Do imagenet classifiers generalize to imagenet? *arXiv preprint arXiv:1902.10811*, 2019. URL https://arxiv.org/abs/1902.10811.

Ievgen Redko, Emilie Morvant, Amaury Habrard, Marc Sebban, and Younès Bennani. A survey on domain adaptation theory: learning bounds and theoretical guarantees. *arXiv preprint arXiv:2004.11829*, 2020.

Daniel Russo and James Zou. How much does your data exploration overfit? controlling bias via information usage. *IEEE Trans. Inf. Theory*, 66(1):302–323, 2020. doi: 10.1109/TIT.2019.2945779. URL https://doi.org/10.1109/TIT.2019.2945779.

Amir Hossein Saberi, Amir Najafi, Ala Emrani, Amin Behjati, Yasaman Zolfimoselo, Mahdi Shadrooy, Abolfazl S. Motahari, and Babak H. Khalaj. Gradual domain adaptation via manifold-constrained distributionally robust optimization. *CoRR*, abs/2410.14061, 2024a. doi: 10.48550/ARXIV.2410.14061. URL https://doi.org/10.48550/arXiv.2410.14061.

Seyed Amir Hossein Saberi, Amir Najafi, Alireza Heidari, Mohammad Hosein Movasaghinia, Abolfazl S. Motahari, and Babak H. Khalaj. Out-of-domain unlabeled data improves generalization. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024b. URL https://openreview.net/forum?id=Bo6GpQ3B9a.

Mert Bülent Sariyildiz, Karteek Alahari, Diane Larlus, and Yannis Kalantidis. Fake it till you make it: Learning transferable representations from synthetic imagenet clones. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023, Vancouver, BC, Canada, June 17-24, 2023*, pp. 8011–8021. IEEE, 2023. doi: 10.1109/CVPR52729.2023.00774. URL https://doi.org/10.1109/CVPR52729.2023.00774.

Ludwig Schmidt, Shibani Santurkar, Dimitris Tsipras, Kunal Talwar, and Aleksander Madry. Adversarially robust generalization requires more data. In S. Bengio,

H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018. URL https://proceedings.neurips.cc/paper_files/paper/2018/file/f708f064faaf32a43e4d3c784e6af9ea-Paper.pdf.

Bernhard Schölkopf, Ralf Herbrich, and Alexander J. Smola. A generalized representer theorem. In David P. Helmbold and Robert C. Williamson (eds.), *Computational Learning Theory, 14th Annual Conference on Computational Learning Theory, COLT 2001 and 5th European Conference on Computational Learning Theory, EuroCOLT 2001, Amsterdam, The Netherlands, July 16-19, 2001, Proceedings*, volume 2111 of *Lecture Notes in Computer Science*, pp. 416–426. Springer, 2001. doi: 10.1007/3-540-44581-1_27. URL https://doi.org/10.1007/3-540-44581-1_27.

Viktor Seib, Benjamin Lange, and Stefan Wirtz. Mixing real and synthetic data to enhance neural network training - A review of current approaches. *CoRR*, abs/2007.08781, 2020. URL https://arxiv.org/abs/2007.08781.

Siamak Shakeri, Cícero Nogueira dos Santos, Henry Zhu, Patrick Ng, Feng Nan, Zhiguo Wang, Ramesh Nallapati, and Bing Xiang. End-to-end synthetic data generation for domain adaptation of question answering systems. *CoRR*, abs/2010.06028, 2020. URL https://arxiv.org/abs/2010.06028.

Shai Shalev-Shwartz and Shai Ben-David. *Understanding Machine Learning - From Theory to Algorithms*. Cambridge University Press, 2014. ISBN 978-1-10-705713-5. URL http://www.cambridge.org/de/academic/subjects/computer-science/pattern-recognition-and-machine-learning/understanding-machine-learning-theory-algorithms.

Shai Shalev-Shwartz, Ohad Shamir, Nathan Srebro, and Karthik Sridharan. Learnability, stability and uniform convergence. *J. Mach. Learn. Res.*, 11:2635–2670, 2010. doi: 10.5555/1756006.1953019. URL https://dl.acm.org/doi/10.5555/1756006.1953019.

Connor Shorten and Taghi M Khoshgoftaar. A survey on image data augmentation for deep learning. *Journal of Big Data*, 6(1):1–48, 2019. URL https://arxiv.org/abs/1904.08324.

Ashish Shrivastava, Tomas Pfister, Oncel Tuzel, Joshua Susskind, Wenda Wang, and Russell Webb. Learning from simulated and unsupervised images through adversarial training. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pp. 2242–2251. IEEE Computer Society, 2017. doi: 10.1109/CVPR.2017.241. URL https://doi.org/10.1109/CVPR.2017.241.

Changjian Shui, Qi Chen, Jun Wen, Fan Zhou, Christian Gagné, and Boyu Wang. A novel domain adaptation theory with jensen–shannon divergence. *Knowledge-Based Systems*, 257:109808, 2022. ISSN 0950-7051. doi: https://doi.org/10.1016/j.knosys.2022.109808. URL https://www.sciencedirect.com/science/article/pii/S0950705122009200.

Ilia Shumailov, Zakhar Shumaylov, Yiren Zhao, Nicolas Papernot, Ross J. Anderson, and Yarin Gal. AI models collapse when trained on recursively generated data. *Nat.*, 631(8022):755–759, 2024. doi: 10.1038/S41586-024-07566-Y. URL https://doi.org/10.1038/s41586-024-07566-y.

Rahul Singh and Suhas Vijaykumar. Kernel ridge regression inference. *arXiv preprint arXiv:2302.06578*, 2023.

Steve Smale and Ding-Xuan Zhou. Shannon sampling ii: Connections to learning theory. *Applied and Computational Harmonic Analysis*, 19(3):285–302, 2005. ISSN 1063-5203. doi: https://doi.org/10.1016/j.acha.2005.03.001. Computational Harmonic Analysis - Part 1.

Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2021.

Karin Stacke, Gabriel Eilertsen, Jonas Unger, and Claes Lundström. A closer look at domain shift for deep learning in histopathology. *arXiv preprint arXiv:1909.11575*, 2019.

Chen Sun, Abhinav Shrivastava, Saurabh Singh, and Abhinav Gupta. Revisiting unreasonable effectiveness of data in deep learning era. *arXiv preprint arXiv:1707.02968*, 2017. URL https://arxiv.org/abs/1707.02968.

Brandon Trabucco, Kyle Doherty, Max Gurinas, and Ruslan Salakhutdinov. Effective data augmentation with diffusion models. *arXiv preprint arXiv:2302.07944*, 2023.

Mark J Tullman. Overview of the epidemiology, diagnosis, and disease progression associated with multiple sclerosis. *Am J Manag Care*, 19(2 Suppl):S15–20, 2013.

Vladimir Naumovich Vapnik. *The Nature of Statistical Learning Theory, Second Edition*. Statistics for Engineering and Information Science. Springer, 2000. ISBN 978-0-387-98780-4.

Roy Voetman, Maya Aghaei, and Klaas Dijkstra. The big data myth: Using diffusion models for dataset generation to train deep detection models. *arXiv preprint arXiv:2306.09762*, 2023.

Wenjia Wang and Bing-Yi Jing. Gaussian process regression: Optimality, robustness, and relationship with kernel ridge regression. *Journal of Machine Learning Research*, 23(193):1–67, 2022. URL http://jmlr.org/papers/v23/21-0570.html.

Garrett Wilson and Diane J Cook. A survey of unsupervised deep domain adaptation. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 11(5):1–46, 2020.

Qinyu Wu, Jonathan Yu-Meng Li, and Tiantian Mao. On generalization and regularization via wasserstein distributionally robust optimization. *arXiv preprint arXiv:2212.05716*, 2022.

Ming Yang, Xiyuan Wei, Tianbao Yang, and Yiming Ying. Stability and generalization of stochastic compositional gradient descent algorithms. *CoRR*, abs/2307.03357, 2023. doi: 10.48550/ARXIV.2307.03357. URL https://doi.org/10.48550/arXiv.2307.03357.

Chenyi Zeng, Lin Gu, Zhenzhong Liu, and Shen Zhao. Review of deep learning approaches for the segmentation of multiple sclerosis lesions on brain mri. *Frontiers in Neuroinformatics*, 14:610967, 2020.

Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning requires rethinking generalization. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017. URL https://openreview.net/forum?id=Sy8gdB9xx.

Xi Zhang, Yanwei Fu, Andi Zang, Leonid Sigal, and Gady Agam. Learning classifiers from synthetic data using a multichannel autoencoder. *CoRR*, abs/1503.03163, 2015. URL http://arxiv.org/abs/1503.03163.

Yuchen Zhang, Tianle Liu, Mingsheng Long, and Michael Jordan. Bridging theory and algorithm for domain adaptation. In Kamalika Chaudhuri and Ruslan Salakhutdinov (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 7404–7413. PMLR, 09–15 Jun 2019. URL https://proceedings.mlr.press/v97/zhang19i.html.

Alexey Zhezherau and Alexei Yanockin. Hybrid training approaches for llms: Leveraging real and synthetic data to enhance model performance in domain-specific applications. *arXiv preprint arXiv:2410.09168*, 2024.

Fuzhen Zhuang, Zhiyuan Qi, Keyu Duan, Dongbo Xi, Yongchun Zhu, Hengshu Zhu, Hui Xiong, and Qing He. A comprehensive survey on transfer learning. *Proceedings of the IEEE*, 109(1):43–76, 2021.

Appendices

A Broader impact

This work shows how synthetic data can be effectively integrated with real data to improve the performance and generalization of downstream tasks in both in-domain and out-of-domain settings.

There are several potential benefits of our work:

1. Our framework enables practitioners to use an effective synthetic-to-real data ratio that yields improved performance at a reduced computational cost, therefore reducing carbon footprint.
2. Although our experiments focus on lesion segmentation, the underlying theory and insights are broadly applicable. Practitioners in various domains can leverage our framework to address challenges related to low-data regimes and domain shifts by exploiting powerful generative models to synthesize data, issues that are common across many applied fields.
3. We identify key factors necessary for evaluating the impact of synthetic data. This is particularly relevant in the current landscape, where a wide range of generative and foundation models are available to generate synthetic data. Our findings can help the community make more informed decisions about incorporating the generated samples from these models, particularly their quantity and quality.
4. Our results highlight the importance of distributional shift in achieving better performance, which in turn underscores the potential value of incorporating human feedback into the synthetic data generation process.
5. In scenarios involving biased datasets—closely related to our distribution shift setup—our framework offers a principled way to generate an adequate number of synthetic samples to improve model performance. This is particularly useful not only in data-scarce domains such as healthcare but also in datasets lacking diversity.

We also acknowledge potential risks and undesirable consequences associated with our approach. In efforts to maximize downstream task performance, practitioners may be incentivized to collect additional data or train more powerful generative models. This introduces several challenges:

1. Collecting extensive data about a subject raises concerns about responsible data acquisition.
2. Training larger generative models requires increased computational resources, which may have a greater environmental impact.

B Limitations

While our work provides theoretical insights and practical guidelines for combining synthetic and real data, several limitations remain:

1. Our analysis involves certain approximations to key parameters that affect the generalization bound and the optimal synthetic-to-real data ratio. The sensitivity of the results to our approach, and studying other ways of approximating them needs further investigation.
2. Although our theory aligns with empirical trends observed in lesion segmentation, we have not validated the proposed bounds across a broader range of applications. Extending the empirical evaluation to diverse domains would help assess the generality of our framework.
3. We focus on providing theoretical and practical insights and do not present a concrete algorithm that integrates a specific real dataset with a synthetic data generator. Developing such an algorithm would facilitate adoption in real-world settings.
4. Our experiments are restricted to the image modality. Investigating how the framework extends to other data types, such as text, audio, or multimodal settings, remains an open and promising direction for future work.

C Other notation

We denote scalar or vector-valued Random Variable (RV) by x, and collections of RVs by $\mathbf{X}$, with corresponding probability densities p_{x} and $p_{\mathbf{X}}$. Realizations of these variables are denoted by $\boldsymbol{x}$

and $\boldsymbol{X}$, respectively, with $\boldsymbol{x}$ taking values in a measurable space $\mathcal{X}$. The conditional distribution of a random variable $\mathbf{y}$ given $\mathbf{x} = \boldsymbol{x}$ is denoted by $p_{\mathbf{y}|\mathbf{x}=\boldsymbol{x}}$. The expectation of a measurable function $f : \mathcal{X} \to \mathbb{R}$ is written as $\mathbb{E}[f(\mathbf{x})] = \mathbb{E}_{\boldsymbol{x}\sim p_{\mathbf{x}}}[f(\boldsymbol{x})]$. For integers $a \leq c \leq b$, we denote by $\mathbf{X}_{a:b} = \{\mathbf{x}_a, \mathbf{x}_{a+1}, \ldots, \mathbf{x}_b\}$ a finite collection of RVs, and by $\mathbf{X}_{a:b}^{(\neq c)} = \mathbf{X}_{a:b} \setminus \{\mathbf{x}_c\}$ the subset excluding $\mathbf{x}_c$. The density $p_{\mathbf{X}_{a:b}}$ denotes the joint density of the variables in $\mathbf{X}_{a:b}$.

We use $[K]$ to denote the index set $\{1, \ldots, K\}$, and reserve Latin letters for samples and Greek letters for parameters or distributions.

Definition C.1 (Lipschitz Continuity)**.** *A function $f : \mathcal{Z} \to \mathbb{R}^q$, for $(\mathcal{Z}, d_{\mathcal{Z}})$ a metric space, is ξ-Lipschitz if for all $z, z' \in \mathcal{Z}$, $\|f(z) - f(z')\| \leq \xi \, d_{\mathcal{Z}}(z, z')$.*

Let $\mathcal{R}_\lambda(h) = \mathbb{E}_{\mathbf{S}}[\mathcal{R}_\lambda(h, \mathbf{S})]$ denote the expected mixed loss, and $r_\lambda(h)$ the hybrid population risk:

$$r_\lambda(h_{\mathbf{S}}) = (1-\lambda) r(h) + \lambda \mathbb{E}_{\mathbf{x}\sim p'_{\mathbf{x}}}\left[\ell(h, \mathbf{x})\right]. \tag{3}$$

Definition C.2 (Generalization Gap)**.** *The* generalization error *of a hypothesis h is defined as the absolute difference between its population and empirical risks:*

$$g_{\mathbf{S}}(h) = \left|\mathcal{L}_{\mathcal{X}}(h) - \mathcal{L}_{\mathbf{S}}(h)\right|.$$

The generalization gap *of a learning algorithm is the expected generalization error:*

$$\mathcal{G} = \mathbb{E}_{p_{\mathrm{h},\mathbf{S}}}[g_{\mathbf{S}}(\mathrm{h})] = \mathbb{E}_{p_{\mathrm{h},\mathbf{S}}}\left[\left|\mathcal{L}_{\mathcal{X}}(\mathrm{h}) - \mathcal{L}_{\mathbf{S}}(\mathrm{h})\right|\right].$$

We can now define the generalization gap in the mixed-data setting as:

$$\mathcal{G} = \mathbb{E}_{p_{\mathrm{h},\mathbf{S}}}[g_{\mathbf{S}}(\mathrm{h})] = \mathbb{E}_{p_{\mathrm{h},\mathbf{S}}}\left[\left|r(\mathrm{h}) - \mathcal{R}_\lambda(\mathrm{h}, \mathbf{S})\right|\right].$$

D Asymptotic effect of synthetic data in kernel ridge regression

Suppose we have M synthetic samples $\{(\tilde{x}_m, \tilde{y}_m)\}_{m=1}^M$, where $\tilde{x}_m \sim p(x)$ i.i.d., and $\tilde{y}_m = g(\tilde{x}_m)$. We assume these synthetic samples are noiseless, reflecting access to the exact synthetic data generator. Then the ERM objective

$$f_N = \operatorname*{arg\,min}_{f\in\mathcal{H}_k} \frac{1}{N}\sum_{n=1}^N (y_n - f(x_n))^2 + \lambda \frac{1}{M}\sum_{m=1}^M \big(f(\tilde{x}_m) - g(\tilde{x}_m)\big)^2$$

satisfies, by the (strong) law of large numbers,

$$\lim_{M\to\infty} \frac{1}{M}\sum_{m=1}^M \big(f(\tilde{x}_m) - g(\tilde{x}_m)\big)^2 = \mathbb{E}_{\tilde{x}\sim p}\big[(f(\tilde{x}) - g(\tilde{x}))^2\big] = \big\langle f-g,\, T_K(f-g)\big\rangle_{\mathcal{H}_k},$$

where the kernel integral operator $T_K : \mathcal{H}_k \to \mathcal{H}_k$ is defined by

$$(T_K\, h)(\cdot) = \int K(\cdot, x)\, h(x)\, p(x)\, \mathrm{d}x.$$

Note that while the synthetic covariates $\tilde{x}_m$ are drawn i.i.d. from the same marginal distribution as the real data, the synthetic labels $\tilde{y}_m$ follow a potentially different mapping g, as determined by the data generator.

Equivalence of $L^2(p)$ and RKHS norms By Mercer's theorem (Mercer, 1909), the operator T_K admits the spectral decomposition

$$K(x, y) = \sum_{i=1}^\infty \mu_i\, \phi_i(x)\, \phi_i(y),$$

where $\{\phi_i\}$ form an orthonormal basis in $L^2(p)$ and $\mu_i > 0$ are the eigenvalues. Any function $h \in \mathcal{H}_k$ can be written as $h = \sum_i a_i \sqrt{\mu_i}\, \phi_i$, yielding

$$\|h\|_{L^2(p)}^2 = \sum_{i=1}^\infty \mu_i\, a_i^2, \qquad \|h\|_{\mathcal{H}_k}^2 = \sum_{i=1}^\infty a_i^2.$$

If the nonzero eigenvalues satisfy $0 < \mu_{\min} \leq \mu_i \leq \mu_{\max} < \infty$, then

$$\mu_{\min}\|h\|^2_{\mathcal{H}_k} \leq \|h\|^2_{L^2(p)} \leq \mu_{\max}\|h\|^2_{\mathcal{H}_k},$$

so the norms are equivalent up to constants:

$$\|h\|^2_{L^2(p)} \asymp \|h\|^2_{\mathcal{H}_k}.$$

Under this spectral assumption, the $L^2(p)$ term $\mathbb{E}_{\tilde{x}\sim p}[(f(\tilde{x}) - g(\tilde{x}))^2]$ appearing in the infinite-M limit is thus proportional to $\|f - g\|^2_{\mathcal{H}_k}$.

Note that we use this equivalence to motivate our setup, we study the effect of limited synthetic data in Section 3 more precisely.

E Technical proofs of modified kernel regression

E.1 Proof of Lemma 2.1

Lemma 2.1. *Let $K_N \in \mathbb{R}^{N\times N}$ be the empirical kernel matrix with entries $(K_N)_{ij} = K(\mathrm{x}_i, \mathrm{x}_j)$. Define the integral operator $T_K : L^2(p_x) \to L^2(p_x)$ by $(T_K f)(\mathrm{x}) = \int K(\mathrm{x}, x')f(x')\, dp_x(x') = \mathbb{E}_{\mathrm{x}'}[K(\mathrm{x}, \mathrm{x}')f(\mathrm{x}')]$. Let $\lambda_N = N\lambda$. Then the solution to Equation 1 has the closed-form representation:*

$$\boldsymbol{\alpha} = (K_N + \lambda_N I)^{-1}(K_N\boldsymbol{\alpha}_\star + \lambda_N\boldsymbol{\beta} + \boldsymbol{\varepsilon})\ ,$$

where $\boldsymbol{\alpha}_\star$, $\boldsymbol{\beta}$, and $\boldsymbol{\varepsilon}$ are the coefficients of $f_\star$, g, and the noise vector in the training basis.

Proof. Let us first rewrite the ERM using the Representer theorem as following:

$$\boldsymbol{\alpha} = \arg\min_{\hat{\boldsymbol{\alpha}}} \frac{1}{N}\|\mathbf{y} - K_N\hat{\boldsymbol{\alpha}}\|^2 + \lambda\|\hat{\boldsymbol{\alpha}} - \boldsymbol{\beta}\|^2_{\mathcal{H}_k} + \lambda\|g_\perp\|^2_{\mathcal{H}_k}\,.$$

Finite-sample solution. Taking the derivation with respect to $\boldsymbol{\alpha}$, we have

$$K_N\left[(K_N + \lambda_N I)\boldsymbol{\alpha} - \mathbf{y} - \lambda_N\boldsymbol{\beta}\right] = 0 \tag{4}$$

Solving the optimization, similarly to the standard regularized kernel regression, we achieve:

$$\boldsymbol{\alpha} = (K_N + \lambda_N I)^{-1}(\mathbf{y} + \lambda_N\boldsymbol{\beta})\,,$$

where we conclude the proof by noting that $\mathbf{y} = K_N\boldsymbol{\alpha}_\star + \boldsymbol{\varepsilon}$. This also results in the fact that $f_N(\mathrm{x}) = K_{\mathrm{x}}(K_N + \lambda_N I)^{-1}(\mathbf{y} + \lambda_N\boldsymbol{\beta})$, where $K_{\mathrm{x}} = (K(\mathrm{x}, \mathrm{x}_1), \ldots, K(\mathrm{x}, \mathrm{x}_N))$.

We now study the behavior of this closed-form solution in the population limit, which becomes useful later.

Population limit and expectation. We use Equation 4, so we have:

$$K_N\left[(K_N + \lambda_N I)(\boldsymbol{\alpha} - \boldsymbol{\beta}) - \mathbf{y} + K_N\boldsymbol{\beta}\right] = 0$$
$$\boldsymbol{\alpha} - \boldsymbol{\beta} = (K_N + \lambda_N I)^{-1}(\mathbf{y} - K_N\boldsymbol{\beta})$$
$$f_N(\mathrm{x}) - g(\mathrm{x}) = K_{\mathrm{x}}(K_N + \lambda_N I)^{-1}(\mathbf{y} - K_N\boldsymbol{\beta})\,.$$

Now, consider the population limit where the sample size $N \to \infty$. The empirical kernel matrix K_N converges to the integral operator T_K, which is a classic approach in kernel ridge regression (Singh & Vijaykumar, 2023). Therefore, $T_K \approx \frac{1}{N}\sum_{n=1}^N K(\cdot, \mathrm{x}_n) \otimes K(\cdot, \mathrm{x}_n)$, which means $K_N \approx NT_K$. So, we have

$$f_N - g = (T_K + \lambda I)^{-1}\left(T_K f_\star - T_K g + \frac{1}{N}\boldsymbol{\varepsilon}\right)$$
$$\mathbb{E}_\varepsilon\left[f_N - g\right] = (T_K + \lambda I)^{-1}T_K(f_\star - g)\,.$$

Noting that $\mathbb{E}_\varepsilon[g] = g$, we get the following result in the population limit:

$$\mathbb{E}_\varepsilon\left[f_N\right] = g + (T_K + \lambda I)^{-1}T_K(f_\star - g)\,. \tag{5}$$

□

E.2 Proof of Theorem 2.2

Theorem 2.2. *Under Assumption 2.1, for the kernel regression problem defined in Equation 1 and any fixed regularization parameter $\lambda \geq 0$, the test error admits the bound:*

$$\mathcal{R}_N(\lambda; g) = \mathcal{O}\left(\frac{\mathcal{D}(f_\star, g)}{N\lambda^2} + \lambda^{2-\frac{1}{4r}}\mathcal{D}(f_\star, g) + \frac{\sigma^2}{N\lambda^2}\right),$$

where $\mathcal{D}(f_\star, g)^2 = \sum_{j=1}^{\infty} \frac{1}{\mu_j^2}(\theta_j - \omega_j)^2$ denotes the discrepancy between the target function $f_\star$ and the synthetic generator g.

Proof. To bound the test error, we use the bias-variance decomposition in Definition 2.1. We start with the variance term. We follow the approach of Misiakiewicz & Saeed (2024). So, we have:

$$\begin{aligned}
\mathcal{V} &= \mathbb{E}_{\mathbf{x},\varepsilon}\left[\left(f_N(\mathbf{x}) - \mathbb{E}_\varepsilon\left[f_N(\mathbf{x})\right]\right)^2\right] && (6)\\
&= \mathbb{E}_{\mathbf{x},\varepsilon}\left[\left(\frac{1}{N}K_{\mathbf{x}}\left(T_K + \lambda I\right)^{-1}\boldsymbol{\varepsilon}\right)\right] && (7)\\
&= \frac{\sigma^2}{N^2}\mathrm{tr}\left(NT_K^2\left(T_K + \lambda I\right)^{-2}\right) && (8)\\
&= \frac{\sigma^2}{N}\sum_{j=1}^{\infty}\frac{\mu_j^2}{(\mu_j + \lambda)^2}\,. && (9)
\end{aligned}$$

Note that the above discussion assumes the population limit. An analogous behaviour holds in the finite-sample setting. Define $\kappa = \sup_{\boldsymbol{x}\in\mathcal{X}}\sqrt{K(\boldsymbol{x}, \boldsymbol{x})}$. Then

$$\begin{aligned}
\mathcal{V} &= \mathbb{E}_{\mathbf{x},\varepsilon}\left[(f_N(\mathbf{x}) - \mathbb{E}_\varepsilon[f_N(\mathbf{x})])^2\right] && (10)\\
&= \mathbb{E}_{\mathbf{x},\varepsilon}\left[\left(\tfrac{1}{N}K_{\mathbf{x}}(K_N + \lambda I)^{-1}\boldsymbol{\varepsilon}\right)^2\right] && (11)\\
&\leq \frac{\kappa^2\sigma^2}{N\lambda^2}\,, && (12)
\end{aligned}$$

where the inequality follows from $\|(K_N + \lambda I)^{-1}\| \leq \frac{1}{\lambda}$ and the bound $\|K_{\mathbf{x}}\|^2 \leq N\kappa^2$. Now, let us bound the bias term. We have:

$$\begin{aligned}
\mathcal{B}^2 &= \mathbb{E}_{\mathbf{x}}\left[(f_\star - \mathbb{E}_\varepsilon\left[f_N\right])(\mathrm{x})\right]^2\\
&\leq \mathbb{E}_{\mathbf{S}}\left[\|\mathbb{E}_\varepsilon\left[f_N\right] - f_\lambda\|_{\mathcal{H}_K}^2\right] + \|f_\star - f_\lambda\|_{p_{\mathbf{x}}}^2\,,
\end{aligned}$$

where the last line is resulted from Jensen's inequality and triangle inequality. Moreover, note that $\|f_\star - f_\lambda\|_{p_{\mathbf{x}}}^2 \leq \kappa^2\|f_\star - f_\lambda\|_{\mathcal{H}}^2$. We define f_λ as the population limit of f_N:

$$f_\lambda = g + (T_K + \lambda I)^{-1}T_K(f_\star - g)\,. \qquad (13)$$

To bound the bias term, we also define an additional auxiliary function $f_{N,\lambda}$:

$$f_{N,\lambda} = g + (K_N + \lambda I)^{-1}T_K(f_\star - g)\,.$$

This function helps us compute the first term of bias.

Population and sampling bias $\mathbb{E}_{\mathbf{S}}\left[\|\mathbb{E}_\varepsilon\left[f_N\right] - f_\lambda\|_{\mathcal{H}_K}^2\right]$. We rewrite this term as follows:

$$\begin{aligned}
\mathbb{E}_{\mathbf{S}}\left[\|\mathbb{E}_\varepsilon\left[f_N\right] - f_\lambda\|_{\mathcal{H}_K}^2\right] &\leq \mathbb{E}_{\mathbf{S}}\left[\|\mathbb{E}_\varepsilon\left[f_N\right] - f_{N,\lambda}\|_{\mathcal{H}_K}^2\right] + \mathbb{E}_{\mathbf{S}}\left[\|f_{N,\lambda} - f_\lambda\|_{\mathcal{H}_K}^2\right]\\
&\leq \mathbb{E}_{\mathbf{S}}\left[\|(K_N + \lambda I)^{-1}(K_N - T_K)(f_\star - g)\|_{\mathcal{H}_K}^2\right] + \mathbb{E}_{\mathbf{S}}\left[\|f_{N,\lambda} - f_\lambda\|_{\mathcal{H}_K}^2\right]\\
&\leq \frac{1}{\lambda^2}E_{\mathbf{S}}\left[\|(K_N - T_K)(f_\star - g)\|_{\mathcal{H}_K}^2\right] + \mathbb{E}_{\mathbf{S}}\left[\|f_{N,\lambda} - f_\lambda\|_{\mathcal{H}_K}^2\right]\\
&\leq \frac{\kappa^2\|f_\star - g\|_{p_{\mathbf{x}}}^2}{N\lambda^2} + \mathbb{E}_{\mathbf{S}}\left[\|f_{N,\lambda} - f_\lambda\|_{\mathcal{H}_K}^2\right]\,,
\end{aligned}$$

where we used the fact that $\|(K_N + \lambda I)^{-1}\| \leq \frac{1}{\lambda}$, and the last inequality is proved in Smale & Zhou (2005)[Theorem 3]. We continue the bound by first noticing that Equation 13 gives us $(T_K + \lambda I)(f_\lambda - g) = T_K(f_\star - g)$:

$$\begin{aligned}
\mathbb{E}_{\mathbf{S}}\left[\|\mathbb{E}_\varepsilon\left[f_N\right] - f_\lambda\|_{\mathcal{H}_K}^2\right] &\leq \frac{\kappa^2\|f_\star - g\|_{p_{\mathbf{x}}}^2}{N\lambda^2} + \mathbb{E}_{\mathbf{S}}\left[\|f_{N,\lambda} - f_\lambda\|_{\mathcal{H}_K}^2\right] \\
&\leq \frac{\kappa^2\|f_\star - g\|_{p_{\mathbf{x}}}^2}{N\lambda^2} + \mathbb{E}_{\mathbf{S}}\left[\|(K_N + \lambda I)^{-1}(T_K + \lambda I)(f_\lambda - g) - (f_\lambda - g)\|_{\mathcal{H}_K}^2\right] \\
&\leq \frac{\kappa^2\|f_\star - g\|_{p_{\mathbf{x}}}^2}{N\lambda^2} + \mathbb{E}_{\mathbf{S}}\left[\|(K_N + \lambda I)^{-1}(T_K - K_N)(f_\lambda - g)\|_{\mathcal{H}_K}^2\right] \\
&\leq \frac{\kappa^2\|f_\star - g\|_{p_{\mathbf{x}}}^2}{N\lambda^2} + \frac{1}{\lambda^2}\mathbb{E}_{\mathbf{S}}\left[\|(T_K - K_N)(f_\lambda - g)\|_{\mathcal{H}_K}^2\right] \\
&\leq \frac{\kappa^2\|f_\star - g\|_{p_{\mathbf{x}}}^2}{N\lambda^2} + \frac{\kappa^2\|f_\lambda - g\|_{p_{\mathbf{x}}}^2}{N\lambda^2} \,,
\end{aligned}$$

where we have used Smale & Zhou (2005)[Theorem 3] once more. Moreover, we note that Equation 13 is also the solution to the following Kernel optimization:

$$f_\lambda = g + \underset{f \in \mathcal{H}_K}{\arg\min}\left\{\|f - (f_\star - g)\|_{p_{\mathbf{x}}}^2 + \lambda\|f\|_{\mathcal{H}_K}^2\right\} \,.$$

Therefore, setting f to zero, we have $\|f_\lambda - (f_\star - g)\|_{p_{\mathbf{x}}}^2 + \lambda\|f_\lambda\|_{\mathcal{H}_K}^2 \leq \|f_\star - g\|_{p_{\mathbf{x}}}^2$, from which we can conclude that $\|f_\lambda - g\|_{p_{\mathbf{x}}}^2 \leq 2\|f_\star - g\|_{p_{\mathbf{x}}}^2$. Putting all these results together and the fact that $\|f_\star - g\|_{p_{\mathbf{x}}}^2 \leq \sup_{\boldsymbol{x}} K(x,x)\|f_\star - g\|_{\mathcal{H}_K}^2$, we have:

$$\mathbb{E}_{\mathbf{S}}\left[\|\mathbb{E}_\varepsilon\left[f_N\right] - f_\lambda\|_{\mathcal{H}_K}^2\right] \leq \frac{3\kappa^4\|f_\star - g\|_{\mathcal{H}}^2}{N\lambda^2} \,. \tag{14}$$

Population bias $\|f_\star - f_\lambda\|_{\mathcal{H}}^2$. To bound this term, we substitute the Mercer decomposition of $f_\star$ and g, and the fact that the eigenvalues of $(T_K + \lambda I)^{-1}$ are $1/(\lambda + \mu_j)$ as following:

$$\begin{aligned}
f_\lambda &= g + \left(T_K + \lambda I\right)^{-1} T_K(f_\star - g) \\
&= \sum_j^\infty \left(\frac{\mu_j}{\mu_j + \lambda}\theta_j + \frac{\lambda}{\mu_j + \lambda}\omega_j\right)\phi_j \,.
\end{aligned}$$

Therefore, we have

$$\|f_\star - f_\lambda\|_{\mathcal{H}}^2 = \|\sum_{j=1}^\infty \frac{\lambda}{\mu_j + \lambda}(\theta_j - \omega_j)\phi_j\|^2 \,,$$

We can bound this bias term as follows by noting that $\{\phi_j\}_j$ consist the orthonormal basis:

$$\|f_\star - f_\lambda\|_{\mathcal{H}}^2 = \sum_{j=1}^\infty \frac{\lambda^2}{(\mu_j + \lambda)^2}(\theta_j - \omega_j)^2 \,. \tag{15}$$

Now, combining the results of Equations 14 and 15 gives us an upper-bound for $\mathcal{B}^2$, and combining them with Equation 12 gets a bound for the test error. We have:

$$\begin{aligned}
\mathcal{R}_N(\lambda; g) &= \frac{3\kappa^4\|f_\star - g\|_{\mathcal{H}}^2}{N\lambda^2} + \sum_{j=1}^\infty \frac{\lambda^2}{(\mu_j + \lambda)^2}(\theta_j - \omega_j)^2 + \frac{\kappa^2\sigma^2}{N\lambda^2} \\
&\leq \frac{3\kappa^4\|f_\star - g\|_{\mathcal{H}}^2}{N\lambda^2} + \lambda^2\sqrt{\left(\sum_{j=1}^\infty \frac{\mu_j^2}{(\mu_j + \lambda)^2}\right)\left(\sum_{j=1}^\infty \frac{1}{\mu_j^2}(\theta_j - \omega_j)^2\right)} + \frac{\kappa^2\sigma^2}{N\lambda^2} \,,
\end{aligned}$$

where the inequality is due to the Cauchy-Schwarz inequality. Since $\sum_{j=1}^{\infty} \frac{1}{\mu_j^2}(\theta_j - \omega_j)^2 = \mathcal{D}(f_\star, g)^2$, we can now write:

$$\begin{aligned}
\mathcal{R}_N(\lambda; g) &= \mathcal{O}\left(\frac{3\kappa^4 \|f_\star - g\|_{\mathcal{H}}^2}{N\lambda^2} + \lambda^2 \sqrt{\mathcal{D}(f_\star, g)^2 \left(\sum_{j=1}^{\infty} \frac{\mu_j^2}{(\mu_j + \lambda)^2} \right)} + \frac{\kappa^2\sigma^2}{N\lambda^2} \right) \\
&= \mathcal{O}\left(\frac{3\kappa^4 \|f_\star - g\|_{\mathcal{H}}^2}{N\lambda^2} + \lambda^2 \mathcal{D}(f_\star, g) \sqrt{\int_0^\infty \left(\frac{x^{-2r}}{x^{-2r} + \lambda} \right)^2 dx} + \frac{\kappa^2\sigma^2}{N\lambda^2} \right) \\
&= \mathcal{O}\left(\frac{3\kappa^4 \|f_\star - g\|_{\mathcal{H}}^2}{N\lambda^2} + \lambda^{2 - \frac{1}{4r}} \mathcal{D}(f_\star, g) C_r + \frac{\kappa^2\sigma^2}{N\lambda^2} \right),
\end{aligned}$$

where

$$C_r = \left(\int_0^\infty \left(\frac{x^{-2r}}{x^{-2r} + \lambda} \right)^2 dx \right)^{1/2} = \left(\frac{1}{2r} \int_0^\infty \frac{v^{1-1/2r}}{(v+1)^2} dv \right)^{1/2} = \sqrt{\frac{B(1/2r, 2 - 1/2r)}{2r}},$$

with $B(z_1, z_2)$ denoting the beta function. We conclude by noting that $\mathcal{D}(f_\star, g) \geq \|f_\star - g\|_{\mathcal{H}}^2$. □

E.3 Proof of Corollary 2.2.1

Corollary 2.2.1. *Under the assumptions of Theorem 2.2, the optimal regularization parameter that minimizes the test error is given by*

$$\lambda^\star \asymp \left(\frac{\sigma^2}{N\mathcal{D}(f_\star, g)} \right)^{\frac{4r}{8r+1}}.$$

Setting $\lambda = \frac{M}{N}$, the optimal number of synthetic samples satisfies:

$$M^\star \asymp \left(\frac{\sigma^2}{\mathcal{D}(f_\star, g)} \right)^{\frac{4r}{8r+1}} N^{\frac{4r+1}{8r+1}}.$$

Proof. The result follows by minimizing the bound in Theorem 2.2:

$$\mathcal{R}_N(\lambda; g) = \mathcal{O}\left(\lambda^{2 - \frac{1}{4r}} \mathcal{D}(f_\star, g) + \frac{\sigma^2}{N} \lambda^{-\frac{1}{2r}} \right).$$

We differentiate the right-hand side with respect to λ and set the derivative to zero:

$$\frac{\partial \mathcal{R}_N}{\partial \lambda} = \left(2 - \frac{1}{4r} \right) \lambda^{1 - \frac{1}{4r}} \mathcal{D}(f_\star, g) - \frac{1}{2r} \cdot \frac{\sigma^2}{N} \lambda^{-\frac{1}{2r} - 1} = 0.$$

Solving for λ gives

$$\lambda^\star = \left(\frac{\sigma^2}{N\mathcal{D}(f_\star, g)} \cdot \frac{1}{\left(2 - \frac{1}{4r}\right) 2r} \right)^{\frac{4r}{8r+1}} \asymp \left(\frac{\sigma^2}{N\mathcal{D}(f_\star, g)} \right)^{\frac{4r}{8r+1}}.$$

Substituting $\lambda^\star = \frac{M^\star}{N}$ yields

$$M^\star = N\lambda^\star \asymp \left(\frac{\sigma^2}{\mathcal{D}(f_\star, g)} \right)^{\frac{4r}{8r+1}} N^{\frac{4r+1}{8r+1}},$$

completing the proof. □

F Generalization bound with mixed real and synthetic data

F.1 Lemmata

Definition F.1. *The upper packing dimension of a measure ν is the quantity d^* defined by:*

$$d^* := \operatorname{ess\,sup}(\Phi^*), \quad \Phi^*(x) := \limsup_{\delta \to 0} \frac{\log p_\delta(x)}{\log \delta}.$$

Definition F.2 ($\mathcal{D}$-Regularity Clerico et al. (2022))**.** *Let $\mathcal{D}$ be a measurable map $\mathcal{P} \times \mathcal{P} \to [0, +\infty]$. Fix $\mu \in \mathcal{P}$ and $\xi \geq 0$. We say that a function $f : \mathcal{Z} \to \mathbb{R}$ is $R_{\mathcal{D}}(\xi)$-regular with respect to μ if $f \in L^1(\mu)$ and for every $\nu \in \mathcal{P}$ such that* $\operatorname{Supp}(\nu) \subseteq \operatorname{Supp}(\mu)$ *and $f \in L^1(\nu)$,*

$$|\mathbb{E}_\mu[f(Z)] - \mathbb{E}_\nu[f(Z)]| \leq \xi\, \mathcal{D}(\mu, \nu).$$

Lemma F.1 (2-Wasserstein Continuity Polyanskiy & Wu (2015); Raginsky et al. (2017); Clerico et al. (2022))**.** *Consider a measurable map $f : \mathcal{Z} \to \mathbb{R}^q$ (with $q \geq 1$). Define the divergence measures*

$$\mathcal{D}_2 : (\mu, \nu) \mapsto \mathcal{W}_2\left(\mu\ , \nu\right)\ .$$

If f is ξ-Lipschitz on $\mathcal{Z}$, then f has regularity $R_{\mathcal{D}_2}(\xi)$ with respect to any $\mu \in \mathcal{P}$ such that $f \in L^1(\mu)$.

Lemma F.2. *Consider a mapping $A : \mathcal{S}_N \to \mathcal{H}$ and define the random variable $\tilde{\mathrm{x}} \sim p_{\mathrm{x}}$ such that $\tilde{\mathrm{x}} \perp\!\!\!\perp \mathbf{S}$. Suppose there exists $\varepsilon \geq 0$ such that for any $i \in \{1, ..., N\}$, it holds that*

$$\mathbb{E}[\ell(h^i, \mathrm{x}_i) - \ell(h, \mathrm{x}_i)] \leq \varepsilon, \tag{16}$$

where $h = A(\mathbf{S}), h^i = A(\mathbf{S}^i)$ and $\mathbf{S}^i = \{\mathrm{x}_1, ..., \mathrm{x}_{i-1}, \tilde{\mathrm{x}}, \mathrm{x}_{i+1}, ..., \mathrm{x}_N\}$. Then it holds that

$$\mathbb{E}[\mathcal{L}_{\mathbf{S}}(A(\mathbf{S})) - \mathcal{L}_{\mathcal{X}}(A(\mathbf{S}))] \leq \varepsilon.$$

Proof. Follows from Lemma 7 of Bousquet & Elisseeff (2002). □

F.2 Stability of the mixed risk minimizer

Denote by $\mathcal{A}$, the algorithm that minimizes the mixed empirical risk:

$$\mathcal{A}(\mathbf{S}) := \operatorname{argmin}_{h \in \mathcal{H}} \mathcal{R}_\lambda(h, \mathbf{S}) = h_{\mathbf{S}}.$$

Lemma F.3. *Suppose that the function $f : \mathcal{Y} \to \mathbb{R}$ is M-smooth, then for any $y \in \mathcal{Y}$,*

$$f(y) - f^* \geq \frac{1}{2M} \|\nabla f(y)\|^2,$$

where $f^* := \inf_{y \in \mathcal{Y}} f(y)$.

Proof. Let $\langle \cdot, \cdot \rangle$ denote the inner product associated with the space $\mathcal{Y}$. From smoothness, it follows that f is differentiable. Setting $z = y - \frac{1}{M} \nabla f(y)$, it further follows from smoothness that,

$$\begin{aligned} f(z) - f(y) &\leq \langle \nabla f(y), z - y \rangle + \frac{M}{2} \|z - y\|^2 \\ &\leq -\frac{1}{M} \langle \nabla f(y), \nabla f(y) \rangle + \frac{1}{2M} \|\nabla f(y)\|^2 \\ &\leq -\frac{1}{2M} \|\nabla f(y)\|^2. \end{aligned}$$

Rearranging and using the fact that $f(z) \geq f^*$ leads to the bound in the statement. □

Lemma F.4. *Suppose Assumption 3.2 holds, then for any $i \in \{1, ..., N\}$ it holds that,*

$$\mathbb{E}\left[\int \|h(x) - h^i(x)\|^2 p'_{\mathrm{x}}(dx)\right] \leq \frac{8M_1}{m^2\lambda} \mathbb{E}[\mathcal{R}_\lambda(h, \mathbf{S})] + \frac{4\sqrt{2M_1} D(1-\lambda)}{mN\lambda} \Big(\mathbb{E}[\mathcal{L}_{\mathbf{S}}(h)]^{1/2} + \mathbb{E}[\mathcal{L}_{\mathcal{X}}(h)]^{1/2}\Big),$$

where $h = \mathcal{A}(\mathbf{S}), h^i = \mathcal{A}(\mathbf{S}^i)$, $\mathbf{S}^i$ is as defined in Lemma F.2.

Proof. Define the measures,

$$\hat{q}(dx) := \frac{1-\lambda}{N}\sum_{x_j \in \mathbf{S}} \delta_{\mathrm{x}_j}(dx) + \lambda p'_{\mathrm{x}}(dx), \qquad \tilde{q}(dx) := \frac{1-\lambda}{N}\sum_{\mathrm{x}_j \in \mathbf{S}^i} \delta_{\mathrm{x}_j}(dx) + \lambda p'_{\mathrm{x}}(dx).$$

Using the strong convexity of c we obtain,

$$\langle h^i(\mathrm{x}) - h(\mathrm{x}), \nabla_1 c(h^i(\mathrm{x}), \mathrm{x})\rangle \geq c(h^i(\mathrm{x}), \mathrm{x}) - c(h(\mathrm{x}), \mathrm{x}) + \frac{m}{2}\|h^i(\mathrm{x}) - h(\mathrm{x})\|^2.$$

which when integrated with respect to $\hat{q}$, leads to

$$\int \langle h^i(\mathrm{x}) - h(\mathrm{x}), \nabla_1 c(h^i(\mathrm{x}), \mathrm{x})\rangle\, \hat{q}(dx) = \mathcal{R}_\lambda(h^i, \mathbf{S}) - \mathcal{R}_\lambda(h, \mathbf{S}) + \frac{m}{2}\int \|h^i(\mathrm{x}) - h(\mathrm{x})\|^2 \hat{q}(dx).$$

The right-hand side is lower bounded further using $\mathcal{R}_\lambda(h^i, \mathbf{S}) \geq \mathcal{R}_\lambda(h, \mathbf{S})$ and the left-hand side is upper bounded using,

$$\begin{aligned}
&\int \langle h^i(\mathrm{x}) - h(\mathrm{x}), \nabla_1 c(h^i(\mathrm{x}), \mathrm{x})\rangle\, \hat{q}(d\mathrm{x}) \\
&\quad = \int \langle h^i(\mathrm{x}) - h(\mathrm{x}), \nabla_1 c(h^i(\mathrm{x}), \mathrm{x})\rangle\, \tilde{q}(dx) \\
&\qquad + \frac{1-\lambda}{N}\Big(\langle h^i(\mathrm{x}_i) - h(\mathrm{x}_i), \nabla_1 c(h^i(\mathrm{x}_i), \mathrm{x}_i)\rangle - \langle h^i(\tilde{\mathrm{x}}) - h(\tilde{\mathrm{x}}), \nabla_1 c(h^i(\tilde{\mathrm{x}}), \tilde{\mathrm{x}})\rangle\Big) \\
&\quad \leq \Big(\int \|h^i(\mathrm{x}) - h(\mathrm{x})\|^2\, \tilde{q}(dx)\Big)^{1/2}\Big(\int \|\nabla_1 c(h^i(\mathrm{x}), \mathrm{x})\|^2\, \tilde{q}(dx)\Big)^{1/2} \\
&\qquad + \frac{D(1-\lambda)}{N}\Big(\|\nabla_1 c(h^i(\mathrm{x}_i), \mathrm{x}_i)\| + \|\nabla_1 c(h^i(\tilde{\mathrm{x}}), \tilde{\mathrm{x}})\|\Big) \\
&\quad \leq \sqrt{2M_1}\Big(\int \|h^i(\mathrm{x}) - h(\mathrm{x})\|^2\, \tilde{q}(dx)\Big)^{1/2}\mathcal{R}_\lambda(h^i, \mathbf{S}^i)^{1/2} + \frac{\sqrt{2M_1}D(1-\lambda)}{N}\Big(c(h^i(\mathrm{x}_i), \mathrm{x}_i)^{1/2} + c(h^i(\tilde{\mathrm{x}}), \tilde{\mathrm{x}})^{1/2}\Big).
\end{aligned}$$

The first inequality above follows from the Cauchy-Schwarz inequality whereas the seconds from Lemma F.3.

This results in the bound,

$$\begin{aligned}
\int \|h^i(\mathrm{x}) - h(\mathrm{x})\|^2 \hat{q}(dx) \leq{}& \frac{2\sqrt{2M_1}}{m}\Big(\int \|h^i(\mathrm{x}) - h(\mathrm{x})\|^2\, \tilde{q}(dx)\Big)^{1/2}\mathcal{R}_\lambda(h, \mathbf{S})^{1/2} \\
&+ \frac{2\sqrt{2M_1}D(1-\lambda)}{mN}\Big(c(h^i(\mathrm{x}_i), \mathrm{x}_i)^{1/2} + c(h^i(\tilde{\mathrm{x}}), \tilde{\mathrm{x}})^{1/2}\Big).
\end{aligned}$$

Taking the expectation, we use the fact that $(h, h^i, \mathbf{S}^i)$ shares the same law as $(h^i, h, \mathbf{S})$ and thus can be exchanged, as well as the symmetry of the algorithm $\mathcal{A}$ under permutations in the dataset, to obtain,

$$\begin{aligned}
\mathbb{E}\Big[\int \|h^i(\mathrm{x}) - h(\mathrm{x})\|^2 \hat{q}(dx)\Big] &\leq \frac{2\sqrt{2M_1}}{m}\Big(\mathbb{E}\Big[\int \|h^i(\mathrm{x}) - h(\mathrm{x})\|^2\, \hat{q}(dx)\Big]\Big)^{1/2}\mathbb{E}[\mathcal{R}_\lambda(h^i, \mathbf{S})]^{1/2} \\
&\quad + \frac{2\sqrt{2M_1}D(1-\lambda)}{mN}\Big(\mathbb{E}[\mathcal{L}_{\mathbf{S}}(h)]^{1/2} + \mathbb{E}[\mathcal{L}_{\mathcal{X}}(h)]^{1/2}\Big) \\
&\leq \frac{2\sqrt{2M_1}}{m}\Big(\mathbb{E}\Big[\int \|h^i(\mathrm{x}) - h(\mathrm{x})\|^2\, \hat{q}(dx)\Big]\Big)^{1/2}\mathbb{E}[\mathcal{R}_\lambda(h^i, \mathbf{S})]^{1/2} \\
&\quad + \frac{2\sqrt{2M_1}D(1-\lambda)}{mN}\Big(2\mathbb{E}[\mathcal{L}_{\mathbf{S}}(h)]^{1/2} + \varepsilon^{1/2}\Big),
\end{aligned}$$

where in the final inequality, we used Lemma F.2. By solving the quadratic, we deduce that this implies,

$$\begin{aligned}
\mathbb{E}\Big[\int \|h^i(x) - h(x)\|^2 \hat{q}(dx, dy)\Big]^{1/2} \leq{}& \frac{\sqrt{2M_1}}{m}\mathbb{E}[\mathcal{R}_\lambda(h, \mathbf{S})]^{1/2} \\
&+ \sqrt{\frac{2M_1}{m^2}\mathbb{E}[\mathcal{R}_\lambda(h, \mathbf{S})] + \frac{2\sqrt{2M_1}D(1-\lambda)}{mN}\Big(2\mathbb{E}[\mathcal{L}_{\mathbf{S}}(h)]^{1/2} + \varepsilon^{1/2}\Big)}.
\end{aligned}$$

This leads to the bound,

$$\mathbb{E}\Big[\int \|h^i(x) - h(x)\|^2 p'_{\mathrm{x}}(dx)\Big] \le \frac{1}{\lambda}\mathbb{E}\Big[\int \|h^i(x) - h(x)\|^2 \hat{q}(dx, dy)\Big]$$
$$\le \frac{8M_1}{m^2\lambda}\mathbb{E}[\mathcal{R}_\lambda(h, \mathbf{S})] + \frac{4\sqrt{2M_1}D(1-\lambda)}{mN\lambda}\Big(2\mathbb{E}[\mathcal{L}_{\mathbf{S}}(h)]^{1/2} + \varepsilon^{1/2}\Big).$$

□

Lemma F.5. *Suppose that $\mathcal{H}$ consists of L-Lipschitz functions, then for any $c > 1$ and $\delta > 0$ sufficiently small, the assumption in Equation 16 is satisfied with*

$$\varepsilon \le \frac{1}{2}\mathbb{E}[\mathcal{L}_{\mathbf{S}}(\mathcal{A}(\mathbf{S}))] + 4\sqrt{2}M\delta^{-cd^*}\sup_i \mathbb{E}\Big[\int \|h(x) - h^i(x)\|^2 p_{\mathrm{x}}(dx)\Big] + 8ML^2\delta^2.$$

Proof. Define $\varepsilon = \mathbb{E}[c(h^i(\mathrm{x}_i), \mathrm{x}_i) - c(h(\mathrm{x}_i), \mathrm{x}_i)]$, then because of the smoothness of c, we obtain

$$\begin{aligned}\varepsilon &\le \mathbb{E}[\langle h^i(\mathrm{x}_i) - h(\mathrm{x}_i), \nabla_1 c(h(\mathrm{x}_i), \mathrm{x}_i)\rangle] + M_1\mathbb{E}[\|h^i(\mathrm{x}_i) - h(\mathrm{x}_i)\|^2]\\ &\le \sqrt{2M_1}\mathbb{E}[\|h^i(\mathrm{x}_i) - h(\mathrm{x}_i)\|^2]^{1/2}\mathbb{E}[c(h(\mathrm{x}_i), \mathrm{x}_i)]^{1/2} + M_1\mathbb{E}[\|h^i(\mathrm{x}_i) - h(\mathrm{x}_i)\|^2]\\ &\le \sqrt{2M_1}\mathbb{E}[\|h^i(\mathrm{x}_i) - h(\mathrm{x}_i)\|^2]^{1/2}\mathbb{E}[\mathcal{L}_{\mathbf{S}}(h)]^{1/2} + M_1\mathbb{E}[\|h^i(\mathrm{x}_i) - h(\mathrm{x}_i)\|^2]\\ &\le \frac{1}{2}\mathbb{E}[\mathcal{L}_{\mathbf{S}}(h)] + 2M_1\mathbb{E}[\|h^i(\mathrm{x}_i) - h(\mathrm{x}_i)\|^2]\end{aligned}$$

Define the measure,

$$p_{\mathrm{x}}^{\tilde{\mathrm{x}},\delta}(dx) := \mathbb{1}_{B_\delta(\tilde{\mathrm{x}})}(x)\, p'_{\mathrm{x}}(B_\delta(\tilde{\mathrm{x}}))^{-1}\, p'_{\mathrm{x}}(dx).$$

Then we can relate function evaluations to the integral over p'_{x} as follows:

$$\begin{aligned}\|h(\tilde{\mathrm{x}}) - h^i(\tilde{\mathrm{x}})\| &\le \Big(\int \|h(x) - h^i(x)\|^2\, p_{\mathrm{x}}^{\tilde{\mathrm{x}},\delta}(dx)\Big)^{1/2} + \Big(\int \|h(x) - h(\tilde{\mathrm{x}})\|^2\, p_{\mathrm{x}}^{\tilde{\mathrm{x}},\delta}(dx)\Big)^{1/2}\\ &\quad + \Big(\int \|h^i(x) - h^i(\tilde{\mathrm{x}})\|^2\, p_{\mathrm{x}}^{\tilde{\mathrm{x}},\delta}(dx)\Big)^{1/2}\\ &\le p'_{\mathrm{x}}(B_\delta(\tilde{x}))^{-1/2}\Big(\int \|h(x) - h^i(x)\|^2\, p'_{\mathrm{x}}(dx)\Big)^{1/2} + 2L\delta.\end{aligned}$$

Taking the expectation gives,

$$\mathbb{E}[\|h(\tilde{x}) - h^i(\tilde{x})\|^2] \le 2\mathbb{E}_{\tilde{x}\sim p'_{\mathrm{x}}}\Big[p'_{\mathrm{x}}(B_\delta(\tilde{x}))^{-2}\Big]^{1/2}\mathbb{E}\Big[\int \|h(x) - h^i(x)\|^2\, p'_{\mathrm{x}}(dx)\Big]^{1/2} + 8L^2\delta^2.$$

For any $c > 1$, we have that for sufficiently small δ,

$$\frac{\log p'_{\mathrm{x}}(B_\delta(x))}{\log \delta} \le cd^*.$$

From Fatou's Lemma we have

$$\begin{aligned}\limsup_{\delta\to 0^+}\mathbb{E}\Big[p'_{\mathrm{x}}(B_\delta(\tilde{x}))^{-2}\delta^{2d^*c}\Big] &\le \mathbb{E}\Big[\limsup_{\delta\to 0^+}\Big(p'_{\mathrm{x}}(B_\delta(\tilde{x}))^{-2}\delta^{2d^*c}\Big)\Big]\\ &= \mathbb{E}\Big[\limsup_{\delta\to 0^+}\exp\Big(-2\log p'_{\mathrm{x}}(B_\delta(\tilde{x})) + 2d^*c\log\delta\Big)\Big]\\ &= \mathbb{E}\Big[\limsup_{\delta\to 0^+}\exp\Big(2\log(1/\delta)\Big(\frac{\log p'_{\mathrm{x}}(B_\delta(\tilde{x}))}{\log\delta} - cd^*\Big)\Big]\\ &\le 1.\end{aligned}$$

Therefore, for δ sufficiently small, we have the non-asymptotic bound

$$\mathbb{E}\Big[p'_{\mathrm{x}}(B_\delta(\tilde{x}))^{-2}\Big] \le 2\delta^{-2d^*c}.$$

□

F.3 Proof of Theorem 3.1

Theorem 3.1. *Let $\mathcal{H}$ be a class of L-Lipschitz functions. Suppose Assumptions 3.1, and 3.2 hold and let $h_{\mathbf{S}} \in \arg\min_{h\in\mathcal{H}} \mathcal{R}_\lambda(h, \mathbf{S})$. Then, there exists a universal constant $C > 0$ and a sample size threshold $N_0 > 0$ such that for all $N \geq N_0$, the algorithm $\mathcal{A}(\mathbf{S}) = h_{\mathbf{S}}$ is uniformly stable with stability constant*

$$\varepsilon \lesssim \frac{1}{\lambda}\mathcal{R}_\lambda(h_{\mathbf{S}}) + C\xi\left(\frac{M_1}{m^2L^2\lambda}\mathcal{R}_\lambda(h_{\mathbf{S}}) + \frac{\sqrt{M_1M_2}(1-\lambda)D^2}{mL^2\lambda N}\right)^{\frac{1}{d_\star+1}},$$

where $d_\star$ denotes the upper packing dimension of the measure p'_{x} (see Appendix F.2 for details), $\xi = M_1L^2 + M_2$, $\eta = M_1/m^2$, and $\tau = \sqrt{M_1M_2}/m$. Let $r^\star = \min_{h\in\mathcal{H}} r(h)$ be the true population risk minimizer. For any $\lambda \in (0, 1)$, the generalization gap satisfies

$$\mathbb{E}[r(h_{\mathbf{S}})] - r^\star \lesssim \lambda\xi\mathcal{W}_2\left(p_{\mathbf{x}}\ , p'_{\mathbf{x}}\right)^2 + C(1-\lambda)\xi\left(\frac{\eta}{L^2\lambda}r^\star + \frac{\eta\xi}{L^2}\mathcal{W}_2\left(p_{\mathbf{x}}\ , p'_{\mathbf{x}}\right)^2 + \frac{\tau(1-\lambda)D^2}{L^2\lambda N}\right)^{\frac{1}{d_\star+1}}.$$

Proof. We note that the first part of the theorem is satisfied by Lemma F.5, where we showed the stability of algorithm $\mathcal{A}$ for any $\delta > 0$. Optimizing with respect to δ, provides us with

$$\varepsilon \lesssim \frac{1}{\lambda}\mathcal{R}_\lambda(h_{\mathbf{S}}) + C\xi\left(\frac{M_1}{m^2L^2\lambda}\mathcal{R}_\lambda(h_{\mathbf{S}}) + \frac{\sqrt{M_1M_2}(1-\lambda)D^2}{mL^2\lambda N}\right)^{\frac{1}{d_\star+1}}.$$

Now, we use the following decomposition to upper bound the generalization error:

$$r(h) = [r(h) - r_\lambda(h)] + [r_\lambda(h) - \mathcal{R}_\lambda(h)] + \mathcal{R}_\lambda(h).$$

We now bound each of the three terms. We begin with the first and third terms, and then analyze the second term, which we refer to as the *stability* term.

Bounding $r(h) - r_\lambda(h)$: We compute

$$\begin{aligned} r(h) - r_\lambda(h) &= \mathbb{E}_{p_{\mathbf{x}}}[\ell(h, \mathbf{x})] - (1-\lambda)\mathbb{E}_{p_{\mathbf{x}}}[\ell(h, \mathbf{x})] - \lambda\mathbb{E}_{p'_{\mathbf{x}}}[\ell(h, \mathbf{x})] \\ &= \lambda\left(\mathbb{E}_{p_{\mathbf{x}}}[\ell(h, \mathbf{x})] - \mathbb{E}_{p'_{\mathbf{x}}}[\ell(h, \mathbf{x})]\right) \\ &\leq \lambda\xi\,\mathcal{W}_2(p_{\mathbf{x}}, p'_{\mathbf{x}}), \end{aligned}$$

where the inequality follows from Lemma F.1.

Bounding $\mathcal{R}_\lambda(h)$: Let $h_{\mathbf{S}} = \arg\min_{h\in\mathcal{H}} \mathcal{R}_\lambda(h, \mathbf{S})$ be the empirical minimizer. Then, for any $h_\star \in \mathcal{H}$, by optimality of $h_{\mathbf{S}}$, we have

$$\begin{aligned} \mathcal{R}_\lambda(h_{\mathbf{S}}, \mathbf{S}) &\leq \mathcal{R}_\lambda(h_\star, \mathbf{S}), \\ \mathbb{E}_{\mathbf{S}}[\mathcal{R}_\lambda(h_{\mathbf{S}}, \mathbf{S})] &\leq \mathbb{E}_{\mathbf{S}}[\mathcal{R}_\lambda(h_\star, \mathbf{S})], \\ \mathcal{R}_\lambda(h_{\mathbf{S}}) &\leq r_\lambda(h_\star). \end{aligned}$$

From the definition of $r_\lambda(h_\star)$ (see Equation 3), we can write:

$$\begin{aligned} \mathcal{R}_\lambda(h_{\mathbf{S}}) &\leq (1-\lambda)\mathbb{E}_{p_{\mathbf{x}}}[\ell(h_\star, \mathbf{x})] + \lambda\mathbb{E}_{p'_{\mathbf{x}}}[\ell(h_\star, \mathbf{x})] \\ &= r(h_\star) + \lambda\left(\mathbb{E}_{p'_{\mathbf{x}}}[\ell(h_\star, \mathbf{x})] - \mathbb{E}_{p_{\mathbf{x}}}[\ell(h_\star, \mathbf{x})]\right). \end{aligned}$$

If $\ell(h, \mathbf{x})$ is ξ-Lipschitz, then by Lemma F.1,

$$\mathbb{E}_{p'_{\mathbf{x}}}[\ell(h_\star, \mathbf{x})] - \mathbb{E}_{p_{\mathbf{x}}}[\ell(h_\star, \mathbf{x})] \leq \xi\mathcal{W}_2\left(p_{\mathbf{x}}\ , p'_{\mathbf{x}}\right),$$

and thus,

$$\mathcal{R}_\lambda(h_{\mathbf{S}}) \leq r(h_\star) + \xi\lambda\mathcal{W}_2\left(p_{\mathbf{x}}\ , p'_{\mathbf{x}}\right).$$

Finally, we can choose $h_\star = \arg\min_{h\in\mathcal{H}} r(h)$ to tighten the bound. It is now sufficient to study the stability term.

Bounding $r_\lambda(h) - \mathcal{R}_\lambda(h)$: We start by substituting the definition of each term and simplifying them. We have:

$$\begin{aligned}
r_\lambda(h) - \mathcal{R}_\lambda(h) &= (1-\lambda)\left(\mathbb{E}_{p_{\mathbf{x}}}\left[\ell(h,\mathbf{x})\right] - \mathbb{E}_{\mathbf{S}}\left[\mathcal{R}_\lambda(h,\mathbf{S})\right]\right) \\
&= (1-\lambda)\left(\mathbb{E}_{\mathbf{S},\mathbf{x}'}\left[\ell(h_{\mathbf{S}},\mathbf{x}')\right] - \mathbb{E}_{\mathbf{S},i}\left[\mathcal{R}_\lambda(h_{\mathbf{S}},\mathbf{x}_i)\right]\right) \\
&= (1-\lambda)\left(\mathbb{E}_{\mathbf{S},\mathbf{x}'}\left[\ell(h_{\mathbf{S}'},\mathbf{x}_i)\right] - \mathbb{E}_{\mathbf{S},i}\left[\mathcal{R}_\lambda(h_{\mathbf{S}},\mathbf{x}_i)\right]\right) ,
\end{aligned}$$

where the first equality is due to the fact that $\lambda\mathbb{E}_{p'_{\mathbf{x}}}[\ell(h,\mathbf{x})]$ is common in both terms. The second equality is by the definition of each term, and the fact that $\mathbf{x}' \perp\!\!\!\perp \mathbf{S}$. Note that $i \sim \text{Unif}([N])$. The final line results from defining $\mathbf{S}' = \mathbf{S} \cup \{\mathbf{x}'\} \setminus \{\mathbf{x}_i\}$, which is a neighboring set to $\mathbf{S}$. Now, assuming that we have ε-uniformly stable algorithm $\mathcal{A}$, then we can write

$$\begin{aligned}
r_\lambda(h) - \mathcal{R}_\lambda(h) &= (1-\lambda)\mathbb{E}_{\mathbf{S},\mathbf{x}',i}\left[\ell(h_{\mathbf{S}'},\mathbf{x}_i) - \ell(h_{\mathbf{S}},\mathbf{x}_i)\right] \\
&\leq (1-\lambda)\varepsilon \,.
\end{aligned}$$

These combine to give the bound,

$$\mathbb{E}[r(h) - r(h_\star)] \leq 2\lambda C\, \mathcal{W}_2(p_{\mathbf{x}}, p'_{\mathbf{x}}) + (1-\lambda)\varepsilon.$$

We conclude by using the first part of the proof for the stability of the algorithm $\mathcal{A}$. □

G Theoretical results and discussions of domain shift

G.1 Proof of Theorem 5.1

Theorem 5.1. *Under Assumption 2.1, for the kernel regression problem defined in Equation 1 and any fixed regularization parameter $\lambda \geq 0$, the test error under domain shift satisfies the bound:*

$$\mathcal{R}_N(\lambda; g) = \mathcal{O}\left(\mu_{\max}\lambda^{r+1}\mathcal{D}(f_\star, \tilde{f}) + \lambda^{\max\{2-\frac{1}{4r}, r+1\}}\mathcal{D}(f_\star, g) + \frac{\sigma^2}{N}\lambda^{-\frac{1}{2r}}\right),$$

where $\mu_{\max} = \max_j \mu_j$, and $\mathcal{D}(\cdot,\cdot)$ denotes the distributional discrepancy, as in Theorem 2.2.

Proof. The proof follows the proof of Theorem 2.2 in Appendix E.2, using the bias-variance decomposition. We note that the variance term remains the same as it only depends on the noise of the data, while the bias term will have the dependency on all three terms of $f_\star$, $\tilde{f}$ and g. Let us start by the formal definition of bias term:

$$\begin{aligned}
\mathcal{B} &= \|f_\star - f_N\|_{\mathcal{H}_k} \\
&= \|\sum_{j=1}^{\infty}\left(\frac{\lambda}{\mu_j+\lambda}(\theta^\star_j - \theta_j) + \frac{\lambda}{\mu_j+\lambda}(\theta^\star_j - \omega_j)\right)\phi_j\| ,
\end{aligned}$$

where $\theta^\star$, θ, and ω refer to the Mercer coefficient of $f_\star$, $\tilde{f}$, and g, respectively. Therefore, we have:

$$\mathcal{B}^2 = \sum_j \left[\frac{\mu_j^2}{(\mu_j+\lambda)^2}(\theta^\star_j - \theta_j)^2 + \frac{\lambda^2}{(\mu_j+\lambda)^2}(\theta^\star_j, \omega_j) + \frac{\lambda\mu_j}{(\mu_j+\lambda)^2}(\theta^\star_j - \theta)(\theta^\star_j - \omega_j)\right] \tag{17}$$

$$\leq \sum_j \frac{\mu_j^2}{(\mu_j+\lambda)^2}(\theta^\star_j - \theta_j)^2 + \lambda^{2-\frac{1}{4r}}\mathcal{D}(f_\star, g)C_r + \sum_j \frac{\lambda\mu_j}{(\mu_j+\lambda)^2}(\theta^\star_j - \theta)(\theta^\star_j - \omega_j) \tag{18}$$

$$\leq \sum_j \frac{\mu_j^2}{(\mu_j+\lambda)^2}(\theta^\star_j - \theta_j)^2 + \lambda^{2-\frac{1}{4r}}\mathcal{D}(f_\star, g)C_r + \sum_j \frac{\lambda\mu_j}{2(\mu_j+\lambda)^2}\left((\theta^\star_j - \theta)^2 + (\theta^\star_j - \omega_j)^2\right) , \tag{19}$$

where the first inequality is taken from the proof of Theorem 2.2, and the second inequality results from the arithmetic-geometric inequality. Now, we start by bounding the first term. By Cauchy-

Schwarz inequality, we have:

$$\sum_j \frac{\mu_j^2}{(\mu_j+\lambda)^2}(\theta_j^\star-\theta_j)^2 \le \sqrt{\left(\sum_j \frac{\mu_j^4}{(\mu_j+\lambda)^2}\right)\left(\sum_j \frac{1}{\mu_j^2}(\theta_j^\star-\theta_j)^2\right)}$$
$$\le \mathcal{D}(f_\star,\tilde{f})\sqrt{\sum_j \frac{\mu_j^4}{(\mu_j+\lambda)^2}}\,.$$

Since μ_j has polynomial decay, there exists $j^\star$ such that $\mu_{j^\star}\asymp\lambda$, more precisely $j^\star\asymp\lambda^{2r}$. When $j\ll j^\star$, $\mu_j\gg\lambda$ and vice versa. Therefore, we have:

$$\sum_j \frac{\mu_j^2}{(\mu_j+\lambda)^2}(\theta_j^\star-\theta_j)^2 = \mathcal{O}\left(\mathcal{D}(f_\star,\tilde{f})\sqrt{\sum_{j\ll j^\star}\mu_j^2+\sum_{j\gg j^\star}\lambda^{-2}\mu_j^4}\right)$$
$$= \mathcal{O}\left(\mathcal{D}(f_\star,\tilde{f})\sqrt{\mu_{\text{max}}^2\lambda^{2r}+\lambda^{-2}\int_\lambda^\infty x^{-8r}\,dx}\right)$$
$$= \mathcal{O}\left(\mu_{\text{max}}\lambda^r\mathcal{D}(f_\star,\tilde{f})\right)\,.$$

Now, we move to bound the third term in Equation 19. We have:

$$\sum_j \frac{\lambda\mu_j}{2(\mu_j+\lambda)^2}\left((\theta_j^\star-\theta)^2+(\theta_j^\star-\omega_j)^2\right) = \sum_j \frac{\lambda\mu_j^3}{2(\mu_j+\lambda)^2}\,\frac{(\theta_j^\star-\theta)^2+(\theta_j^\star-\omega_j)^2}{\mu_j^2}$$
$$\le \frac{\lambda}{2}\sqrt{\sum_j\frac{\mu_j^3}{(\mu_j+\lambda)^2}}\left(\sqrt{\sum_j\frac{1}{\mu_j^2}(\theta_j^\star-\theta_j)^2}+\sqrt{\sum_j\frac{1}{\mu_j^2}(\theta_j^\star-\omega_j)^2}\right)$$
$$\le \frac{\lambda}{2}\sqrt{\sum_j\frac{\mu_j^3}{(\mu_j+\lambda)^2}}\left(\mathcal{D}(f_\star,g)+\mathcal{D}(f_\star,\tilde{f})\right)$$
$$= \mathcal{O}\left(\frac{\lambda}{2}\sqrt{\sum_{j\ll j^\star}\mu_j+\sum_{j\gg j^\star}\frac{\mu_j^3}{\lambda}}\left(\mathcal{D}(f_\star,g)+\mathcal{D}(f_\star,\tilde{f})\right)\right)$$
$$= \mathcal{O}\left(\frac{\mu_{\text{max}}^{1/2}}{2}\lambda^{r+1}\left(\mathcal{D}(f_\star,g)+\mathcal{D}(f_\star,\tilde{f})\right)\right)\,,$$

where the first inequality is by Cauchy-Schwarz, and the rest are by the definitions and expansion of the terms. Combining all the terms completes the proof. □

G.2 Proof of Theorem 5.2

Theorem 5.2. *Let $\mathcal{H}$ be a class of L-Lipschitz functions. Suppose Assumptions 3.1, and 3.2 hold and let $h_{\mathbf{S}}\in\arg\min_{h\in\mathcal{H}}\mathcal{R}_\lambda(h,\mathbf{S})$. Then, for any $\lambda\in(0,1)$, the generalization gap under the domain shift satisfies*

$$\mathbb{E}[r(h_{\mathbf{S}})]-r^\star \lesssim \lambda\xi\mathcal{W}_2\left(p_{\mathbf{x}}^\star,p'_{\mathbf{x}}\right)^2+(1-\lambda)\xi\mathcal{W}_2\left(p_{\mathbf{x}}^\star,p_{\mathbf{x}}\right)^2$$
$$+C(1-\lambda)\xi\left(\frac{M_1}{m^2L^2\lambda}r^\star+\frac{M_1\xi}{m^2L^2}\mathcal{W}_2\left(p_{\mathbf{x}}^\star,p_{\mathbf{x}}\right)^2+\frac{\sqrt{M_1M_2}(1-\lambda)D^2}{mL^2\lambda N}\right)^{\frac{1}{d_\star+1}}$$

where $r^\star=\min_{h\in\mathcal{H}}r^\star(h)$ is the true population risk minimizer of the target domain.

Proof. For any $h\in\mathcal{H}$, we show $r^\star(h)$, and $r_\lambda^\star(h)$ as

$$r^\star(h)=\mathbb{E}_{p_{\mathbf{x}}^\star}\left[\ell(h,\mathbf{x})\right],\qquad r_\lambda^\star(h)=(1-\lambda)r^\star(h)+\mathbb{E}_{p'_{\mathbf{x}}}\left[\ell(h,\mathbf{x})\right]\,. \tag{20}$$

We now have the following decomposition for the generalization error:

$$r^\star(h)=(r^\star(h)-r_\lambda^\star(h))+(r_\lambda^\star(h)-r_\lambda(h))+(r_\lambda(h)-\mathcal{R}_\lambda(h))+\mathcal{R}_\lambda(h)\,.$$

Similar to Appendix F.3, we bound each of the terms separately. We have:

Bounding $r^\star(h) - r^\star_\lambda(h)$: Let us expand the term by the definition of each component:

$$\begin{aligned}
r^\star(h) - r^\star_\lambda(h) &= \mathbb{E}_{p^\star_{\mathbf{x}}}\left[\ell(h,\mathbf{x})\right] - \big((1-\lambda)r^\star(h) + \mathbb{E}_{p'_{\mathbf{x}}}\left[\ell(h,\mathbf{x})\right]\big) \\
&= \lambda\left(\mathbb{E}_{p^\star_{\mathbf{x}}}\left[\ell(h,\mathbf{x})\right] - \mathbb{E}_{p'_{\mathbf{x}}}\left[\ell(h,\mathbf{x})\right]\right) \\
&\leq \lambda\xi\mathcal{W}_2\left(p^\star_{\mathbf{x}}, p'_{\mathbf{x}}\right) ,
\end{aligned}$$

where the last inequality is by Lemma F.1.

Bounding $r^\star_\lambda(h) - r_\lambda(h)$: We again use the definitions:

$$\begin{aligned}
r^\star_\lambda(h) - r_\lambda(h) &= \big((1-\lambda)r^\star(h) + \mathbb{E}_{p'_{\mathbf{x}}}\left[\ell(h,\mathbf{x})\right]\big) - \big((1-\lambda)r(h) + \mathbb{E}_{p'_{\mathbf{x}}}\left[\ell(h,\mathbf{x})\right]\big) && (21) \\
&= (1-\lambda)(r^\star_\lambda(h) - r(h)) && (22) \\
&= (1-\lambda)\left(\mathbb{E}_{p^\star_{\mathbf{x}}}\left[\ell(h,\mathbf{x})\right] - \mathbb{E}_{p_{\mathbf{x}}}\left[\ell(h,\mathbf{x})\right]\right) && (23) \\
&\leq (1-\lambda)\xi\mathcal{W}_2\left(p^\star_{\mathbf{x}}, p_{\mathbf{x}}\right) , && (24)
\end{aligned}$$

where we have once again used Lemma F.1.

Bounding $r_\lambda(h) - \mathcal{R}_\lambda(h)$: Similar to Appendix F.3, we refer to this term as the stability term. Note that all the conditions for Lemma F.2 hold here, therefore, this term is the same as Appendix F.3 since the stability is uniform. Thus, $r_\lambda(h) - \mathcal{R}_\lambda(h) \leq (1-\lambda)\varepsilon$ for:

$$\varepsilon \leq 2C\xi\left(\frac{1}{\lambda}\mathcal{R}_\lambda(h_{\mathbf{S}}) + \frac{(1-\lambda)D^2}{\lambda N}\right)^{\frac{1}{d_\star+1}} . \tag{25}$$

We now only need to bound $\mathcal{R}_\lambda(h_{\mathbf{S}})$, for both Equations 20 and 25.

Bounding $\mathcal{R}_\lambda(h_{\mathbf{S}})$: Since $h_{\mathbf{S}} = \arg\min_{h\in\mathcal{H}} \mathcal{R}_\lambda(h,\mathbf{S})$ is the empirical minimizer, for any $h' \in \mathcal{H}$, by optimality of $h_{\mathbf{S}}$, we have

$$\begin{aligned}
\mathcal{R}_\lambda(h_{\mathbf{S}},\mathbf{S}) &\leq \mathcal{R}_\lambda(h',\mathbf{S}), \\
\mathbb{E}_{\mathbf{S}}[\mathcal{R}_\lambda(h_{\mathbf{S}},\mathbf{S})] &\leq \mathbb{E}_{\mathbf{S}}[\mathcal{R}_\lambda(h',\mathbf{S})], \\
\mathcal{R}_\lambda(h_{\mathbf{S}}) &\leq r_\lambda(h') \,.
\end{aligned}$$

Now, let $h' = \arg\min_{h\in\mathcal{H}} r_\lambda(h)$. Then, for any $h_\star \in \mathcal{H}$, we have

$$\begin{aligned}
\mathcal{R}_\lambda(h_{\mathbf{S}},\mathbf{S}) \leq r_\lambda(h') &\leq r_\lambda(h_\star) \\
&\leq r^\star_\lambda(h_\star) + (1-\lambda)\xi\mathcal{W}_2\left(p^\star_{\mathbf{x}}, p_{\mathbf{x}}\right) \\
&\leq r^\star(h_\star) + \lambda\left(\mathbb{E}_{p^\star_{\mathbf{x}}}[\ell(h_\star,\mathbf{x})] - \mathbb{E}_{p'_{\mathbf{x}}}[\ell(h_\star,\mathbf{x})]\right) + (1-\lambda)\xi\mathcal{W}_2\left(p^\star_{\mathbf{x}}, p_{\mathbf{x}}\right) \\
&\leq r^\star(h_\star) + \lambda\xi\mathcal{W}_2\left(p^\star_{\mathbf{x}}, p'_{\mathbf{x}}\right) + (1-\lambda)\xi\mathcal{W}_2\left(p^\star_{\mathbf{x}}, p_{\mathbf{x}}\right) ,
\end{aligned}$$

where the first inequality is by Equation 24, and the second inequality is from the definition and the last one is by Lemma F.1. Now, let $h_\star = \arg\min_{h\in\mathcal{H}} r^\star(h)$. Combining all bounds together completes the proof. □

H Experimental setup

H.1 Optimal regularization in kernel ridge regression

We study a nonparametric regression problem wherein the ground truth function $f_\star$ and an auxiliary function g are both defined as truncated series expansions in an orthonormal sine basis, with polynomially decaying coefficients to encode varying degrees of smoothness. The target function is given by $f_\star(x) = \sum_{j=1}^{T_f}(j+1)^{-rs}\sin(\pi(j+1)x)$, while g is constructed analogously using a decay rate s' over the first T_g terms. Training data consists of $N = 15$ i.i.d. samples $\{x_i\}_{i=1}^{n}$ drawn uniformly from $[0,3]$, with noisy observations $y_i = f_\star(x_i) + \varepsilon_i$, where $\varepsilon_i \sim \mathcal{N}(0, 0.1)$, alongside noiseless evaluations of $g(x_i)$. We employ our modified kernel ridge regression (Lemma 2.1) method using a Mercer kernel with eigenvalue decay $\mu_j \asymp j^{-2r}$, incorporating g as a regularization term to enhances the standard kernel estimator. The predictive performance is evaluated on a dense test grid (test set of

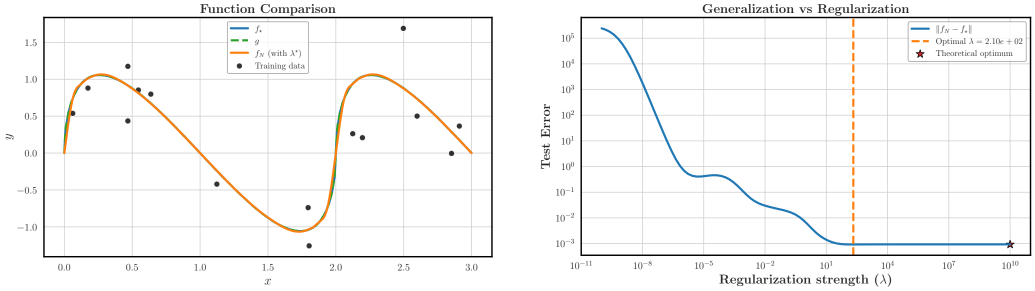
500 points) by computing the empirical L_2-distance between the learned function f_N and the true function $f_\star$. This procedure is repeated across a logarithmically spaced range of regularization parameters $\lambda \in [10^{-10}, 10^{10}]$. In addition, we compute the theoretically optimal regularization parameter by minimizing an upper bound derived from the distance $\mathcal{D}(f_\star, g)$, which depends explicitly on the eigendecay and coefficient mismatch between $f_\star$ and g, based on Theorem 2.2:

$$\lambda^* = \arg\min_{\lambda>0} \left(C_r^2 \frac{\sigma^2}{N} \lambda^{-1/(2r)} + C_r \lambda^{2-1/(4r)} D(f_\star, g) \right) .$$

Our implementation uses SciPy for numerical integration and optimization, with special care given to numerical stability through pseudo-inverses and adaptive regularization. To capture the effect of the difference between $f_\star$ and g, we run the experiment with various values as depicted in Table 1. Figures 1, 5 and 6 illustrate the impact of distributional alignment between the true function $f_\star$ and the synthetic generator g on the behaviour of the estimated function f_N and the choice of regularization strength λ. In Figure 5, the synthetic generator perfectly matches the true distribution ($s = s'$, and $T_f = T_g$), resulting in no discrepancy between $f_\star$, g, and f_N. Consequently, the prediction error $\|f_N - f_\star\|_{L_2}$ is minimized for the largest possible regularization strength, and our algorithm successfully selects this value. In contrast, Figure 6 considers a case with distribution mismatch (large difference between s, s', and T_f, T_g), leading to larger discrepancies between the functions. This results in a characteristic U-shaped prediction error curve, as shown in Figure 6b. While the theoretically chosen regularization strength (star marker) slightly overestimates the empirical optimum (dashed orange line), the difference remains negligible, demonstrating the robustness of our theoretical bound under mismatch. The experimental details are shown in Table 1.

Table 1: Effective parameters for the modified kernel regression.

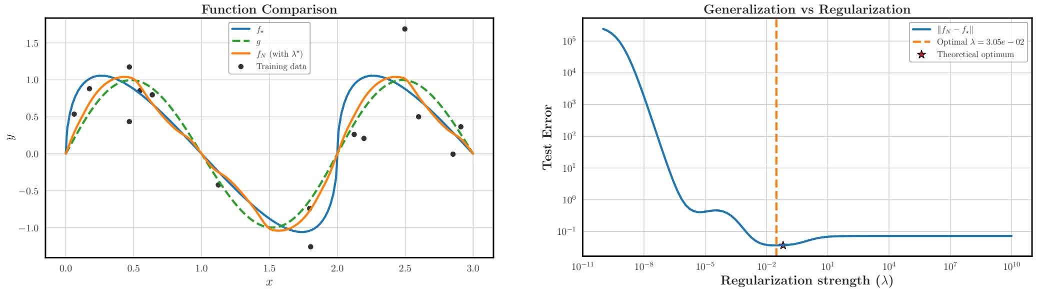
r	s	s'	T_f	T_g	$D(f_\star, g)$	Figure
2.0	0.8	0.8	100	100	0.0000	Figure 5
2.0	0.8	1.5	100	10	737.65	Figure 1
2.0	0.8	2.5	100	10	15509.16	Figure 6



(a) Function comparisons.

(b) Generalization vs Regularization.

Figure 5: *(a)* Comparison of the true function $f_\star$ (blue), the synthetic generator g (green), and the estimated function f_N (orange), obtained via Lemma 2.1, with parameters $r = 2.0$, $s = 0.8$, and $s' = 0.8$. Since $g = f_\star$ in this setting, the RKHS distance is zero and all curves coincide. *(b)* Prediction error $\|f_N - f_\star\|_{L_2}$ as a function of the regularization strength λ. As expected, there is no U-shaped behaviour since the generator fully matches the true distribution. The theoretical optimum selects a large λ (star marker), while the empirical optimum (dashed orange line) selects a smaller value due to numerical precision limits.



(a) Function comparisons.

(b) Generalization vs Regularization.

Figure 6: *(a)* Comparison of the true function $f_\star$ (blue), the synthetic generator g (green), and the estimated function f_N (orange), obtained via Lemma 2.1, with parameters $r = 2.0$, $s = 0.8$, and $s' = 2.5$. The distance between the functions is larger compared to Figure 1. *(b)* Prediction error $\|f_N - f_\star\|_{L_2}$ as a function of the regularization strength λ. A clear U-shaped curve is observed, and while the theoretically optimal λ (star marker) slightly overestimates the empirical optimum (dashed orange line), the difference is negligible.

H.2 Natural images on CIFAR-10

We investigate the effect of synthetic data on classification performance using a conditional diffusion model. Specifically, we train a diffusion model on CIFAR-10 to generate class-conditional synthetic images, which are then used to augment the real training set. We compare two classifiers: one trained solely on real data, and another trained on a mixture of real and synthetic samples. Performance is evaluated across varying synthetic-to-real data ratios, and validation accuracy is reported for each configuration. The real dataset used to train the diffusion model is disjoint from the one used for training and validating the classifier, allowing us to isolate the effect of synthetic data augmentation. Detailed experimental settings are provided in Appendix H.2.1.

In Figure 7, we observe that classification accuracy improves with increasing amounts of mixed training data when the distributional distance between the synthetic generator and real data is small (orange line in Appendix H.2). In contrast, for generators with moderate to high distributional distance (i.e., lower quality - see green and red lines), we observe diminishing returns or even performance degradation. This follows our insights from Section 3. Similar trends are observed at the class level, although the results are noisier due to the reduced amount of data per class—approximately one-tenth of the total. These results indicate that the trained diffusion model captures different classes with varying fidelity, which in turn affects per-class generalization.

This highlights an important practical consideration: when class-wise generalization is a priority, it is crucial to ensure that the synthetic data generator performs well not only in aggregate but also across different classes or groups.

H.2.1 Experimental details for CIFAR-10

Dataset and preprocessing We conduct experiments on CIFAR-10 Krizhevsky et al. (2009), a dataset of 60,000 colour images (32×32 pixels) across 10 object categories, with 50,000 training and 10,000 test samples. For each run, we stratify the training set to construct three disjoint subsets: a labelled training set $\mathcal{D}_{\text{train}}$ containing N examples, a validation set $\mathcal{D}_{\text{val}}$ of 5,000 examples, and a separate set $\mathcal{D}_{\text{diff}}$ of 50,000 examples used for training the diffusion model. Stratified sampling ensures class balance across all subsets. All images are linearly rescaled to the $[-1, 1]$ range. During classifier training, we apply random horizontal flips as the only form of data augmentation. No augmentations are used during diffusion model training or validation.

Conditional diffusion model Our synthetic data generator is a class-conditional diffusion model trained on $\mathcal{D}_{\text{diff}}$. The architecture is a UNet2D with six downsampling and upsampling blocks, using channel sizes [128, 128, 256, 256, 512, 512]. Self-attention layers are included at the 16×16 spatial resolution. Class conditioning is achieved via a learnable embedding table of dimension 512. We train the model using a linear noise schedule over $T = 1000$ diffusion steps. Optimization is performed with AdamW using a learning rate of 10^{-4} and $(\beta_1, \beta_2) = (0.9, 0.999)$. We apply cosine learning

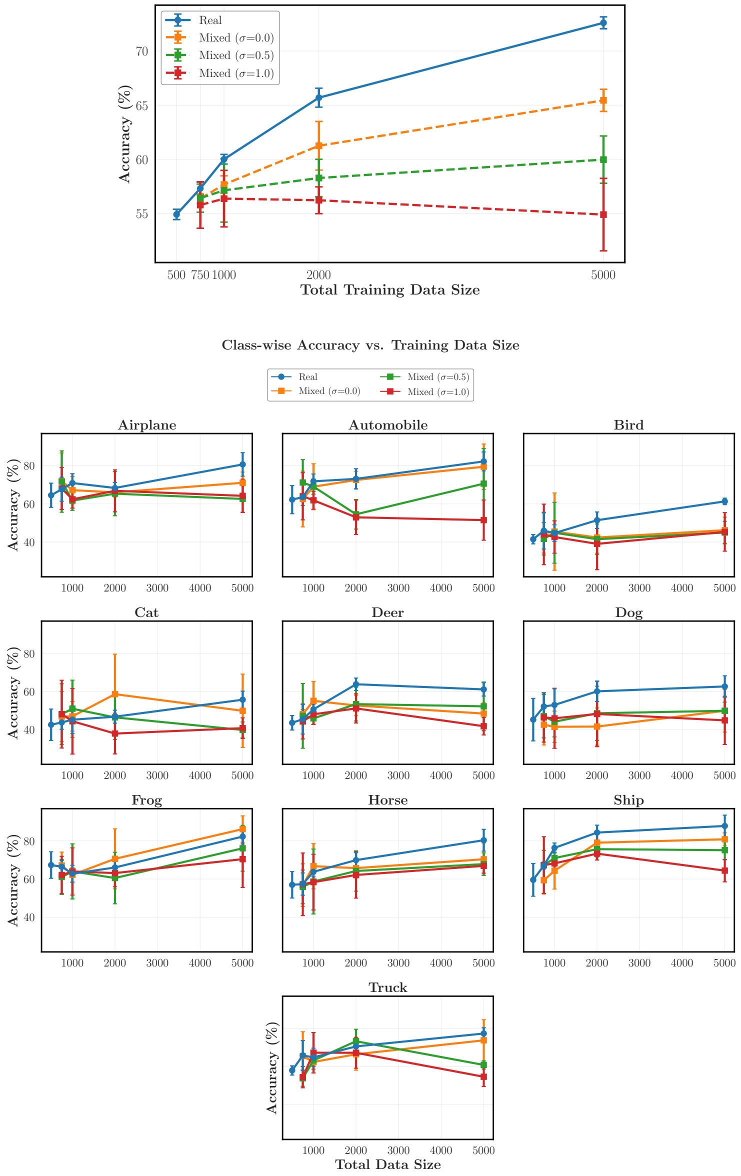


Figure 7: (a) Average accuracy vs. training data size. Increasing the amount of real data (blue line) consistently increases the accuracy of the classification, while for the mixed-data it depends on the quality of the generated samples. (b) Accuracy of each class vs. training data size. We observe a similar pattern, however noisier.

rate decay with 500 warmup steps, use mixed-precision training (FP16), and set the batch size to 64. Each model is trained for 100 epochs.

Classification task For the downstream task, we use a compact convolutional neural network. It consists of two convolutional layers with 3×3 kernels and output channels 32 and 64, respectively, each followed by ReLU activation and max pooling. The output is flattened and passed through a fully connected layer with 512 units, followed by ReLU, a dropout layer with rate 0.25, and a final fully connected layer with 10 outputs. We train this classifier using the Adam optimizer with a learning rate of 10^{-3}, a batch size of 64, and up to 20 epochs with early stopping based on validation performance. Cross-entropy loss is used for optimization.

Experimental protocol For each configuration (N, M), where M denotes the number of synthetic samples to generate, we first train the conditional diffusion model on $\mathcal{D}_{\text{diff}}$. We then sample M class-conditional synthetic images to form $\mathcal{D}_{\text{synth}}$. Two classifiers are trained: f_{real} on $\mathcal{D}_{\text{train}}$ alone, and f_{aug} on the augmented dataset $\mathcal{D}_{\text{train}} \cup \mathcal{D}_{\text{synth}}$. Both classifiers are evaluated on the same validation set $\mathcal{D}_{\text{val}}$ using the classification accuracy metric:

$$\text{Acc} = \frac{1}{|\mathcal{D}_{\text{val}}|} \sum_{(x,y)\in\mathcal{D}_{\text{val}}} \mathbb{I}\left(f(x) = y\right).$$

Hyperparameter configurations We explore several synthetic-to-real data ratios $M/N \in \{0.25, 0.5, 0.75, 1.0, 1.25, 1.5, 2.0, 5.0\}$, with each configuration repeated across multiple random seeds. A summary of the hyperparameters is provided in Table 2. All experiments are implemented with the HuggingFace Diffusers library and executed on NVIDIA A100 GPUs with 40GB memory.

Table 2: Hyperparameter configurations for CIFAR-10 experiments

Parameter	Values
Real data size (N)	500
Synthetic-to-real ratio (M/N)	0.5, 1.0, 3, 9
Diffusion steps (T)	1000
Total noise variances	0.0, 0.5, 1.0

H.2.2 Additional results

H.3 Real-world medical imaging

In this section, we provide additional details for the experimental setup of Section 4.

Diffusion model training Our conditional diffusion model synthesises MRI slices conditioned on anatomical tissue masks. The architecture begins with a 1×1 convolutional layer for initial feature projection, followed by a sinusoidal positional embedding to encode timestep information. The model includes four down-sampling stages, each consisting of two ResNet blocks, a linear attention layer with residual connection, and a 3×3 convolutional down-sampling layer. This is followed by a bottleneck module comprising two additional ResNet blocks and another linear attention layer. The up-sampling path mirrors the down-sampling structure, replacing down-sampling layers with convolutional up-sampling layers of the same kernel size. Finally, a 1×1 convolutional layer projects the features to the desired output channels.

The model follows a hierarchical channel structure, starting with 64 channels, doubling at each down-sampling stage (64 → 128 → 256 → 512 at the bottleneck), then halving symmetrically during up-sampling back to 64. Conditioning is achieved by concatenating a four-channel binary mask (GM, WM, CSF, lesion) with the timestep embedding and spatial inputs at the input layer. The model is trained on slices from the NeuroRx dataset using mean squared error loss over 600 denoising timesteps. All modalities (T1, T2, PD) are trained independently.

Segmentation Model and Task A vanilla U-Net is used for lesion segmentation. The network comprises three downsampling and upsampling layers with skip connections, ReLU activations, and max pooling. Feature channels double in the downsampling path (64 → 128 → 256) and halve in the upsampling path symmetrically. For all experiments, the models train for up to 800 epochs or until plateau, validated on a fixed NeuroRx set.

Hyperparameter Configuration We perform a targeted grid search for hyperparameters considering our hardware constraints. The search space and the final configuration is shown in Table 3. All experiments are executed on NVIDIA A100 GPUs with 40GB memory, with no gradient clipping or additional augmentations.

Table 3: Hyperparameter configurations for medical imaging experiments

Parameter	Values / Search Space
Batch size	16, 32, 64, 128 (selected: 128)
Learning rate	{1e-4, 5e-4, 1e-3} (selected: 1e-4)
Epochs	Up to 800 or until loss plateau
Optimizer	Adam
LR Scheduler	Exponential decay ($\gamma = 0.99$)
Weight Init	Kaiming Uniform
Loss Function	Focal + Tversky loss (equal weight)
Focal Loss Params	$\delta = 0.25$, $\lambda = 2$
Tversky Loss Params	$\alpha = 0.7$, $\beta = 0.3$
Gradient Clipping	None

H.3.1 Additional Results

From Theorem 3.1, we expect that as the quality of synthetic samples deteriorates, i.e., as the distributional distance between the synthetic data generator and the real data increases, the optimal synthetic-to-real ratio should decrease, placing greater emphasis on the real data. Consequently, we anticipate an increase in the validation loss. Figure 8 empirically supports this expectation. In our setting, this distributional distance is modulated by the sampling timestep of the diffusion model: higher timesteps correspond to noisier, and thus less realistic, synthetic samples.

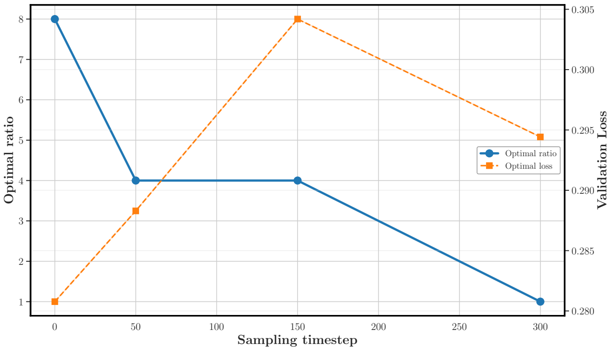
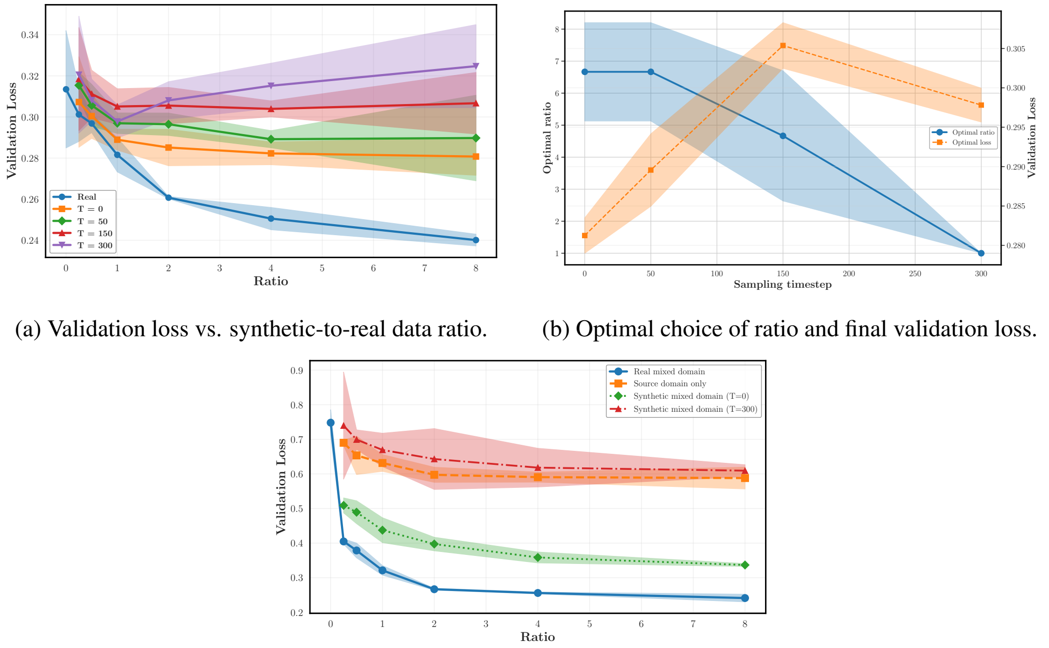


Figure 8: Optimal synthetic-to-real ratio (blue line) and the optimal validation loss (orange dashed line) as the distributional distance or equivalently the diffusion sampling timestep grows.

For reproducibility, we repeat each experiment using three different random seeds to account for variability introduced by stochastic elements in the training and sampling processes. The reported results in Figure 9 represent the mean performance across these runs, with corresponding confidence intervals to capture the variability. This approach ensures that our conclusions are not driven by a particular random initialization and provides a more robust estimate of model behavior.



(a) Validation loss vs. synthetic-to-real data ratio.

(b) Optimal choice of ratio and final validation loss.

(c) Validation loss vs ratio with domain shift.

Figure 9: (a) Validation loss vs. synthetic-to-real data ratio across different sampling timesteps, representing varying distributional distances. We observe a sharper U-turn effect as the distributional distance increases. (b) The optimal synthetic-to-real data ratio decreases as synthetic samples become noisier or deviate further from the true distribution. (c) Incorporating synthetic data improves out-of-domain generalization when the synthetic data distribution is closer to the target than the source. All experiments are repeated with three different seeds. The results align with those shown in the main text for single experimental runs.

H.4 Practical Insights

In this section, we investigate the effects of signal-to-noise ratio, i.e., heterogeneity, and the regularity of the problem. As shown in Figure 10, varying the noise level across different values of r exhibits a pattern consistent with our observations in Section 6. We again find that a 1:2 ratio of real to synthetic data performs well across these scenarios. While changes in the regularity of the objectives (i.e., r) influence the scale of the test error, the overall behavior remains consistent.

Similarly, under domain shift (see Figure 11), the effect of the signal-to-noise ratio aligns with the in-domain behavior reported in Section 6. Specifically, higher heterogeneity necessitates more careful selection of the real-to-synthetic ratio, as higher ratios can degrade performance when the distributional distance from the target domain is large.

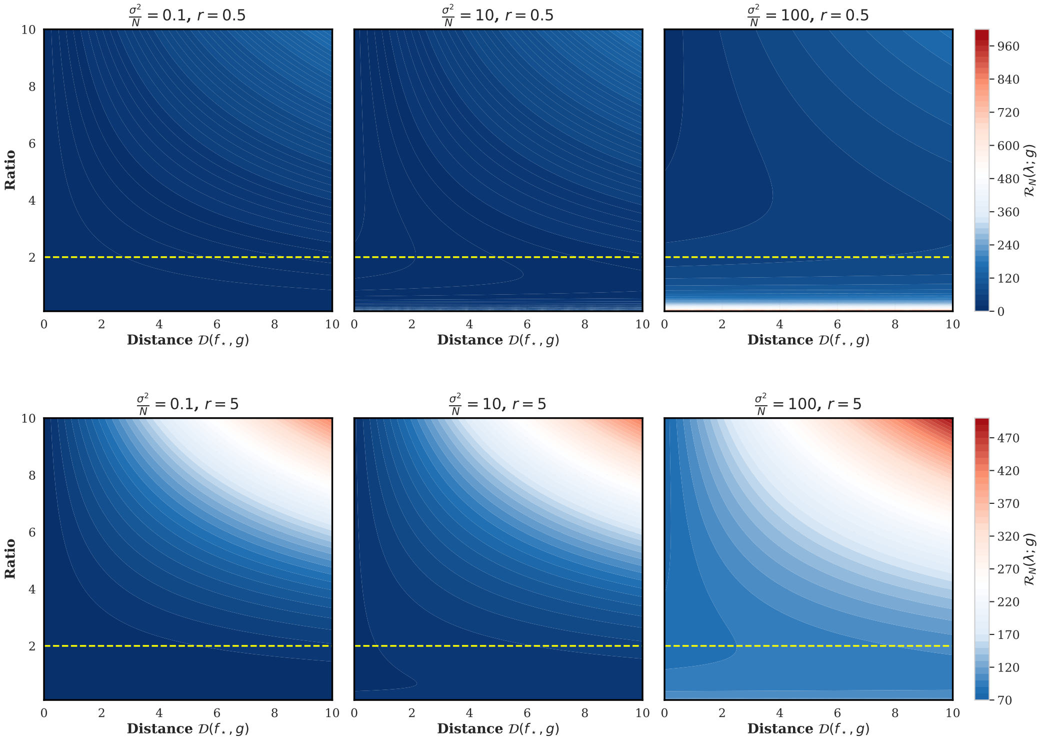


Figure 10: Effect of signal-to-noise ratio on the choice of optimal synthetic-to-real data ratio, across two different values of $r \in \{0.5, 5\}$.

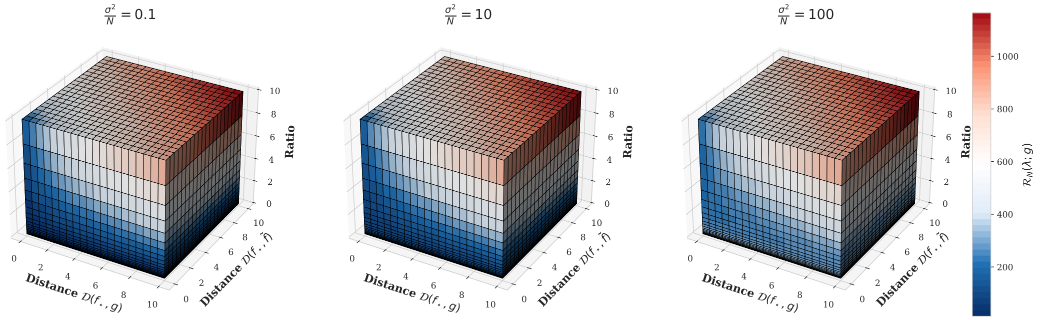


Figure 11: Effect of signal-to-noise ratio on the generalization error. All the plots are with $r = 1, \mu_{\text{max}} = 1$.

I Expanded related work

Synthetic Data The rapid advancement of generative models has significantly improved data generation quality, making it increasingly difficult to distinguish between synthetic and real data. Many previous works (Zhang et al., 2015; Cortés et al., 2024; Shrivastava et al., 2017; Lee et al., 2024; Seib et al., 2020; Zhezherau & Yanockin, 2024) have demonstrated the effectiveness of synthetic data in enhancing the performance of deep learning methods and stabilizing training, both in supervised and unsupervised settings, through augmentation and various applications. Alemohammad et al. (2024a); Shumailov et al. (2024); Briesch et al. (2023) analyzed the effects of training generative models on synthetic data across multiple iterations, creating a self-consuming loop, and found that without sufficient fresh real data, model quality or diversity deteriorates over time, leading to model collapse. To further investigate this, Dohmatob et al. (2024) studied model collapse theoretically in the regression case. Similarly, Bertrand et al. (2024); Gerstgrasser et al. (2024) looked at iteratively training generative models on mixed datasets, concluding that stability is maintained if the initial model is accurate and the real data proportion is sufficiently high. Ferbach et al. (2024) expanded on this by investigating the impact of user-curated synthetic data on iterative retraining, framing it as an implicit preference optimization process and exploring its theoretical effects on model stability and quality. In contrast, our paper does not examine iterative learning but instead focuses on a single-step approach, framing synthetic data as a regularizer. Possibly closest to our work is Jain et al. (2024), where the authors use a weighted empirical risk minimization approach to integrate surrogate data, reducing test error even when unrelated to the original data. However, their work differs from ours in two main aspects: (1) they focus on the scaling law of the test error, while we aim to determine the optimal synthetic-to-real data ratio or regularizer weight; (2) they do not account for the distance between synthetic and real data distributions, while our work specifically provides a bound based on this difference.

Domain Adaptation and Transfer Learning Recent research has explored the intersection of domain adaptation and synthetic data, showing how synthetic data can bridge the gap between source and target domains, thereby enhancing model transferability and generalization across tasks (Mullick et al., 2023; Peng et al., 2018; Imbusch et al., 2022; Shakeri et al., 2020). A major challenge in transfer learning is the distribution gap, and several studies address this by using synthetic data to fine-tune models, improving generalizability (Mishra et al., 2022; Kim et al., 2020; Sariyildiz et al., 2023). Gerace et al. (2022) propose synthetic data as a framework for modeling correlations between datasets, showing improvements in generalization when transferring learned features from source to target tasks. On a more theoretical level, several works connect domain adaptation to distributionally robust learning, demonstrating that adding unlabeled or labeled data improves generalization; these setups can be easily extended to include synthetic data (Saberi et al., 2024a;b; Wu et al., 2022; Hou et al., 2023).

Generalization Bounds The first generalization bounds were based on characterizations of the hypothesis space's complexity, such as the VC dimension or Rademacher complexity (Bousquet et al., 2003; Vapnik, 2000; Shalev-Shwartz & Ben-David, 2014). However, due to their algorithm-independent nature, these bounds must hold even for the worst algorithm within a given hypothesis space, making them often inadequate for modern over-parameterized neural networks, where the complexity measure typically scales exponentially with the architecture's depth (Anthony & Bartlett, 2002; Zhang et al., 2017; Belkin et al., 2018). To address this issue, recent approaches focus on providing algorithm-dependent generalization bounds. The underlying intuition is that a hypothesis less dependent on the input dataset is less prone to overfitting and, therefore, generalizes better. Among the results building on this idea are bounds based on uniform stability (Bousquet & Elisseeff, 2002; Attia & Koren, 2022), differential privacy (Dwork & Roth, 2014), PAC-Bayesian bounds (Guedj, 2019; McAllester, 1999), information-theoretic bounds (Russo & Zou, 2020; Gálvez et al., 2020; Haghifam et al., 2020), and chained bounds (Clerico et al., 2022; Asadi et al., 2018). Our work mainly uses the previously established ideas of generalization bounds for mixed of real and synthetic data when the synthetic data acts as regularizer. More related to our work and on the importance of regularization, Li & Zhang (2021) analyzes the generalization properties of fine-tuning in transfer learning and proposes a PAC-Bayes generalization bound, combining regularization and self-labeling. Mou et al. (2018) provides generalization guarantees for dropout training by bounding the error using offset Rademacher complexities, capturing data-dependent regularization and the effect of perturbation variance.