A Bias-Variance-Covariance Decomposition of Kernel Scores for Generative Models

Sebastian G. Gruber [1 2 3] **Florian Buettner** [1 2 3 4]

Abstract

Generative models, like large language models, are becoming increasingly relevant in our daily lives, yet a theoretical framework to assess their generalization behavior and uncertainty does not exist. Particularly, the problem of uncertainty estimation is commonly solved in an ad-hoc and task-dependent manner. For example, natural language approaches cannot be transferred to image generation. In this paper, we introduce the first bias-variance-covariance decomposition for kernel scores. This decomposition represents a theoretical framework from which we derive a kernel-based variance and entropy for uncertainty estimation. We propose unbiased and consistent estimators for each quantity which only require generated samples but not the underlying model itself. Based on the wide applicability of kernels, we demonstrate our framework via generalization and uncertainty experiments for image, audio, and language generation. Specifically, kernel entropy for uncertainty estimation is more predictive of performance on CoQA and TriviaQA question answering datasets than existing baselines and can also be applied to closed-source models.

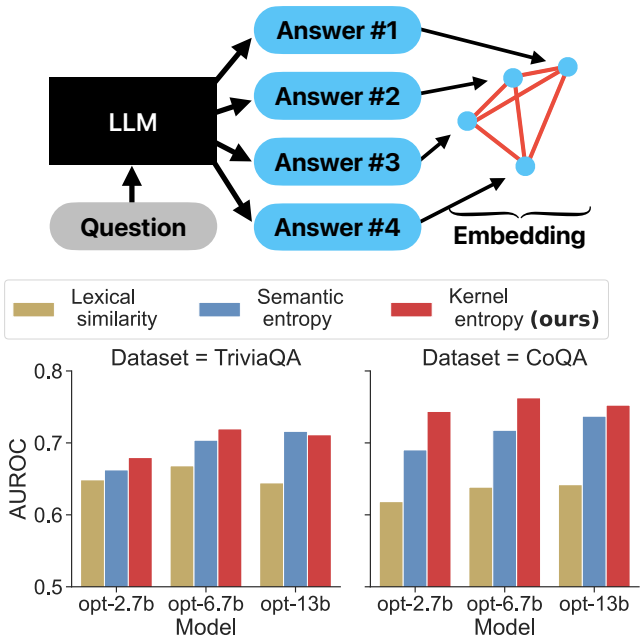


Figure 1. **Top:** Illustration of predictive kernel entropy for a generative model. A kernel measures the pairwise similarities (red lines) of outputs in a vector space. The predictive kernel entropy is then the negative average kernel value. **Bottom:** The predictive kernel entropy shows the best performance among uncertainty approaches for single-model settings (c.f. Section 5.3).

1. Introduction

In recent years, generative models have revolutionized daily lives well beyond the field of machine learning (Kasneci et al., 2023; Meskó & Topol, 2023). These models have found applications in diverse domains, including image creation (Ramesh et al., 2021), natural language generation (OpenAI, 2023), drug discovery (Paul et al., 2021), and speech synthesis (Ning et al., 2019). While generative models have demonstrated remarkable capabilities in generating data that closely resemble real-world samples, they often fall short in providing the vital and often overlooked aspect of uncertainty estimation (Wu & Shang, 2020). Uncertainty estimation in machine learning is a critical component of model performance assessment and deployment (Hekler et al., 2023). It addresses the inherent limitations and challenges associated with machine learning based decisions. For generative models, this may include their propensity to generate improbable or nonsensical samples ("hallucinations"). Even though uncertainty estimation methods for natural language question answering tasks exist (Kuhn et al., 2023), they are ad-hoc without theoretical grounding and

[1]German Cancer Consortium (DKTK), partner site Frankfurt/Mainz, a partnership between DKFZ and UCT Frankfurt-Marburg, Germany, Frankfurt am Main, Germany [2]German Cancer Research Center (DKFZ), Heidelberg, Germany [3]Goethe University Frankfurt, Germany [4]Frankfurt Cancer Institute (FCI), Germany. Correspondence to: Sebastian G. Gruber <sebastian.gruber@dkfz.de>.

Proceedings of the 41st International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024. Copyright 2024 by the author(s).

are not transferable to other data generation tasks .

Predictive uncertainty is an informal concept, but it is implied that it relates to the prediction error without requiring access to target outcomes. A formal approach to this is the bias-variance decomposition, a central concept in statistical learning theory (Bishop & Nasrabadi, 2006; Hastie et al., 2009; Murphy, 2022). It helps to understand the generalization behavior of models and naturally raises uncertainty terms by isolating the target prediction into a bias term (Gruber & Buettner, 2023).
Ueda & Nakano (1996) discovered the bias-variance-covariance decomposition of the mean squared error, which is the foundation of negative correlation learning (Liu & Yao, 1999b;a; Brown, 2004) and for reducing correlations in weight averaging (Rame et al., 2022). Though the bias-variance decomposition has been generalized to distributions (Gruber & Buettner, 2023), the current theory does not include a covariance term and relies on having access to the predicted distribution. But, many generative models only indirectly fit the training distribution by learning how to generate samples. Others, such as large language models (LLMs), do explicitly fit the target distribution, but the prevalence of closed source models means the predictive distribution is often not available to the practitioner (OpenAI, 2023). This makes it infeasible to apply the powerful framework of the bias-variance decomposition in these cases.
Contrary, kernels allow to quantify differences between distributions only based on their samples without requiring access to these distributions (Gretton et al., 2012a). They are used in kernel scores to assess the goodness-of-fit for predicted distributions (Gneiting & Raftery, 2007).
As **contribution** in this work, we...

- introduce the first extension of the bias-variance-covariance decomposition beyond the mean squared error to kernel scores in Section 3, and propose unbiased and consistent estimators only requiring generated samples in Section 4.
- examine the generalisation behavior of generative models for image and audio generation and investigate how bias, variance and kernel entropy relate to the generalisation error in Section 5. This includes evidence that mode collapse of underrepresented minority groups is expressed purely in the bias.
- demonstrate how kernel entropy in combination with text embeddings outperforms existing methods for estimating the uncertainty of LLMs on common question answering datasets (c.f. Figure 1 and Section 5.3).

2. Background

In this section, we give a brief introduction into kernel scores, followed up by other bias-variance decompositions and approaches for assessing the uncertainty in natural language generation.

2.1. Kernel Scores

Kernel scores are a class of loss functions for distribution predictions (Eaton, 1981; Eaton et al., 1996; Dawid, 2007). For simplicity, we omit complex-valued kernels. We refer to a symmetric kernel $k\colon \mathscr{Y} \times \mathscr{Y} \to \mathbb{R}$ defined on a set $\mathscr{Y}$ as **positive definite** (p.d.) if $\sum_{i=1}^{n}\sum_{j=1}^{n} a_i k(x_i, x_j) a_j > 0$ for all $x_1, \ldots, x_n \in \mathscr{Y}$ and $a_1, \ldots, a_n \neq 0$ with $n \in \mathbb{N}$. **Positive semi-definite** (p.s.d.) refers to the case when only '$\geq$' holds. Assume $\mathscr{P}$ is a set of distributions defined on $\mathscr{X}$ such that for a kernel k the operator $\langle P \,|\, k \,|\, Q \rangle := \int_{\mathscr{Y}} \int_{\mathscr{Y}} k(x, y) \,\mathrm{d}P(x) \,\mathrm{d}Q(y)$ is finite for all $P, Q \in \mathscr{P}$ (Eaton, 1981). It follows that $\langle . \,|\, k \,|\, . \rangle$ is a symmetric bilinear form and induces the semi-norm $\|P\|_k = \sqrt{\langle P \,|\, k \,|\, P \rangle}$.
A **kernel score** $S_k\colon \mathscr{P} \times \mathscr{Y} \to \mathbb{R}$ based on a p.s.d. kernel $k\colon \mathscr{Y} \times \mathscr{Y} \to \mathbb{R}$ is defined as (Steinwart & Ziegel, 2021)

$$S_k(P, y) = \|P\|_k^2 - 2\,\mathbb{E}_{\hat{Y}\sim P}\left[k\left(\hat{Y}, y\right)\right]. \tag{1}$$

Note that Eaton (1981) and Dawid (2007) use a slightly less general definition. If $\mathscr{P}$ only consists of Borel probability measures, then the expected kernel score $\mathbb{E}\left[S_k(P, Y)\right]$ based on a target $Y \sim Q \in \mathscr{P}$ is minimized when $P = Q$ (Gneiting & Raftery, 2007). Following Dawid (2007), we refer to $-\|Q\|_k^2 = \mathbb{E}\left[S_k(Q, Y)\right]$ as the **kernel entropy** function for any Q. The kernel score is connected to the **maximum mean discrepancy** (MMD) via $\mathrm{MMD}_k^2(P, Q) = \mathbb{E}\left[S_k(P, Y)\right] + \|Q\|_k^2$ (Steinwart & Ziegel, 2021). The MMD is used for non-parametric two-sample testing (Gretton et al., 2012a) and generative image modelling (Li et al., 2015; Bińkowski et al., 2018). Compared to the MMD, kernel scores are applicable to a wider range of scenarios, since one sample of the target distribution is sufficient for evaluation. For example, MMDs cannot be computed for question-answering pairs when there is only one answer for each question in the dataset. We offer a more extensive discussion of related work in Appendix B.

2.2. Bias-Variance (-Covariance) Decompositions

Ueda & Nakano (1996) introduced the bias-variance-covariance decomposition for the mean squared error. For a real-valued ensemble prediction $\hat{P}^{(n)} = \frac{1}{n}\sum_{i=1}^{n}\hat{P}_i$ with identically distributed $\hat{P}_1, \ldots, \hat{P}_n$ and real-valued target Y it is given by

$$\begin{aligned}\underbrace{\mathbb{E}\left[\left(\hat{P}^{(n)} - Y\right)^2\right]}_{\text{Expected Squared Error}} &= \underbrace{\mathbb{V}(Y)}_{\text{Noise}} + \underbrace{\left(\mathbb{E}\left[\hat{P}\right] - \mathbb{E}[Y]\right)^2}_{\text{Bias}} \\ &+ \underbrace{\frac{1}{n}\mathbb{V}\left(\hat{P}\right)}_{\text{Variance}} + \underbrace{\frac{n-1}{n}\,\mathrm{Cov}\left(\hat{P}, \hat{P}'\right)}_{\text{Covariance}},\end{aligned} \tag{2}$$

with $\hat{P} := \hat{P}_1$ and $\hat{P}' := \hat{P}_2$. We use $\mathbb{V}$ and Cov to denote the textbook definitions of variance and covariance for real-valued random variables (Capiński & Kopp, 2004). Throughout this work, we imply that the pairwise covariance of identically distributed variables is the same. Rame et al. (2022) propose an approximate bias-variance-covariance decomposition for hard-label classification but it only holds in an infinitesimal locality around the prediction. To our best knowledge, the mean squared error is the only case so far with a non-approximated decomposition. Gruber & Buettner (2023) introduce a bias-variance decomposition for loss functions of general distributions. They demonstrated that the variance term is a meaningful measure of the model uncertainty similar to confidence scores in classification. However, they require a loss-specific transformation of the distributions into a dual vector space and a covariance term is not given.

2.3. Uncertainty in Natural Language Generation

In the following, we give a brief overview of uncertainty estimation in natural language generation.

A common approach is predictive entropy, which is the Shannon entropy $-\int \log \hat{p}\,(y \mid x)\,\mathrm{d}\hat{p}\,(y \mid x)$ of the predicted distribution $\hat{p}$ given an input x (Malinin & Gales, 2020). For a generated token sequence $\mathbf{s} = (s_1, \ldots, s_l) \in \mathbb{N}^l$ of length $l \in \mathbb{N}$ it is computed via $\sum_{i=1}^{l} \log \hat{p}\,(s_i \mid s_1, \ldots, s_{i-1})$, where $\hat{p}$ is the predicted distribution of the generating language model. Note that the predicted distribution is not always available for closed-source models. The computation also scales linearly with the length of the generated text, making it costly for larger text generations. Malinin & Gales (2020) propose to use length-normalisation of the predictive entropy since the Shannon entropy is systematically affected by the sequence length. Kuhn et al. (2023) propose *semantic entropy* to ease the computation of the predictive entropy by finding clusters of semantically similar generations. Another approach is *lexical similarity* (Fomicheva et al., 2020), which quantifies the average pairwise similarity between generated answers according to a similarity measure, like RougeL (Lin & Och, 2004; Kuhn et al., 2023). Kadavath et al. (2022) propose the baseline *p(True)*, which asks the model itself if the generated answer is correct. Alternative approaches exist, which require an ensemble of models (Lakshminarayanan et al., 2017; Malinin & Gales, 2020). However, ensembles are practically less relevant due to the high computational cost of training even a single model.

3. A Bias-Variance-Covariance Decomposition of Kernel Scores

In this section, we state our main theoretical contribution. All proofs are presented in Appendix E. To highlight the similarity to the mean squared error case, we introduce the novel definitions for distributional variance and distributional covariance. The latter also implies a distributional correlation, which we define later in Section 4. Note that textbook variance and covariance are based on multiplication of two scalar components. Multiplication of two scalars is a special case of an inner product based on k. Thus, we interpret $\langle . \mid k \mid . \rangle$ as a generalization of this multiplication, which directly implies the following.

Definition 3.1. Assume we have a p.s.d. kernel k and random variables P and Q with outcomes in a distribution space as defined above. We define the **distributional variance** generated by k of P as

$$\mathrm{Var}_k\,(P) = \mathbb{E}\left[\|P - \mathbb{E}\,[P]\|_k^2\right] \tag{3}$$

and the **distributional covariance** generated by k between P and Q as

$$\mathrm{Cov}_k\,(P, Q) = \mathbb{E}\left[\langle P - \mathbb{E}\,[P] \mid k \mid Q - \mathbb{E}\,[Q] \rangle\right]. \tag{4}$$

If P is deterministic, i.e. is a random variable with only one outcome, then $\mathrm{Var}_k\,(P) = 0$. Further, we have $\mathrm{Cov}_k\,(P, P) = \mathrm{Var}_k\,(P)$, and, if P and Q are independent, then $\mathrm{Cov}_k\,(P, Q) = 0$. Note that the terms *kernel variance* and *kernel covariance* already exist in the literature and should not be confused with our definitions (Gretton et al., 2003).

We now have the necessary tools to state our main theoretical contribution in a concise manner.

Theorem 3.2. *Let S_k be a kernel score based on a p.s.d. kernel k and $\hat{P}$ a predicted distribution for a target $Y \sim Q$, then*

$$\underbrace{\mathbb{E}\left[S_k\left(\hat{P}, Y\right)\right]}_{\textit{Generalization Error}} = \underbrace{-\|Q\|_k^2}_{\textit{Noise}} + \underbrace{\left\|\mathbb{E}\left[\hat{P}\right] - Q\right\|_k^2}_{\textit{Bias}} + \underbrace{\mathrm{Var}_k\left(\hat{P}\right)}_{\textit{Variance}}. \tag{5}$$

If we have an ensemble prediction $\hat{P}^{(n)} := \frac{1}{n}\sum_{i=1}^{n} \hat{P}_i$ with identically distributed members $\hat{P}_1, \ldots, \hat{P}_n$, then

$$\mathrm{Var}_k\left(\hat{P}^{(n)}\right) = \frac{1}{n}\,\mathrm{Var}_k\left(\hat{P}_1\right) + \frac{n-1}{n}\,\mathrm{Cov}_k\left(\hat{P}_1, \hat{P}_2\right). \tag{6}$$

In the same sense, a bias-variance-covariance decomposition of the expected MMD is implied since $\mathbb{E}[\mathrm{MMD}_k^2(\hat{P}, Q)] = \mathbb{E}[S_k(\hat{P}, Y)] + \|Q\|_k^2$. Theorem 3.2 allows bias-variance and ensemble evaluations of increasingly more common generative models since kernels can be used for almost all data scenarios via vector embeddings (Liu et al., 2020). For example, in Section 5, we evaluate diffusion models for images, flow-based models for text-to-speech synthesis, and transformers for natural language generation. It is also possible to offer Theorem 3.2 in terms of an associated reproducing kernel Hilbert space or if the ensemble members

are not identically distributed (c.f. Appendix C).

To illustrate Theorem 3.2, we give as example the Brier score used in classification, which also recovers the original decomposition of Ueda & Nakano (1996).

Example 3.3. *Let $\mathscr{P} = \Delta^l$ be the l-dimensional simplex and define $k_\delta(x, y) := \mathbf{1}_{x=y} = 1$ if $x = y$ and 0 otherwise. Then, for $y \in \{1, \ldots, l\}$ and $P \in \mathscr{P}$, the negative kernel entropy $\|P\|_{k_\delta}^2 = \sum_{i=1}^l P_i^2 =: \|P\|^2$ is the squared euclidean norm and $S_{k_\delta}(P, y) + 1 = \sum_{i=1}^l (P_i - \mathbf{1}_{i=y})^2 =: \mathrm{BS}(P, y)$ is the Brier score (Brier, 1950). Applying Theorem 3.2 decomposes $\mathbb{E}\left[\mathrm{BS}\left(\hat{P}, Y\right)\right]$ into*

$$1 - \|Q\|^2 + \left\|\mathbb{E}\left[\hat{P}\right] - Q\right\|^2 + \sum_{i=1}^{l} \mathbb{V}\left(\hat{P}_i\right). \tag{7}$$

The original decomposition in Equation (2) can be recovered for binary classification since $\mathrm{BS}(P, y) = 2\left(P_1 - \mathbf{1}_{1=y}\right)^2$ for $l = 2$.

In the last example, we omitted the covariance term for simplicity. However, this term has also its practical utility as demonstrated in the next example.

Example 3.4. *Assume we train a generative model and intend to improve the generalization performance via an ensemble approach. Due to computational reasons, we decide to use an ensemble of consecutive training epochs as ensemble members $\hat{P}_1, \ldots, \hat{P}_n$. We empirically perform such an experiment in Section 5.1, which shows that the assumptions of Theorem 3.2 hold approximately after excluding the first 20 epochs (c.f. Figure 3 and Equation (17)). We receive as estimates $\mathrm{Var}_k\left(\hat{P}_1\right) \approx 0.0052$ and $\mathrm{Cov}_k\left(\hat{P}_1, \hat{P}_2\right) \approx 0.0049$. Thus,*

$$\begin{aligned} &\overbrace{\mathbb{E}\left[S_k\left(\hat{P}_1, Y\right)\right]}^{\textit{Single Model Error}} - \overbrace{\mathbb{E}\left[S_k\left(\frac{1}{n}\sum_{i=1}\hat{P}_i, Y\right)\right]}^{\textit{Ensemble Error}} \\ &= \frac{n-1}{n}\left(\mathrm{Var}_k\left(\hat{P}_1\right) - \mathrm{Cov}_k\left(\hat{P}_1, \hat{P}_2\right)\right) \\ &\approx \left(1 - \frac{1}{n}\right) 0.0003. \end{aligned} \tag{8}$$

This example demonstrates how to predict the generalization improvement of arbitrary ensemble sizes even when the ensemble members are not independently distributed. This is not possible without a variance-covariance decomposition.

In the following of this work, we use Theorem 3.2 to study the generalization behavior of generative models and to find ways to estimate the uncertainty of generated data. The presented evaluations and approaches are applicable to almost any data generation task due to the flexibility of kernels and data embeddings.

Predictive Kernel Entropy for Single Models. Historically, the bias-variance decomposition had a large impact on the development of some of the most established machine learning algorithms, like Random Forests (Breiman, 2001) or Gradient Boosting (Friedman, 2002). However, ensemble approaches are not similarly dominant for generative modeling. Estimating $\mathrm{Var}_k(\hat{P})$ requires an ensemble of models, which is not always feasible. Instead, note the decomposition $\mathrm{Var}_k(\hat{P}) = \mathbb{E}[\|\hat{P}\|_k^2] - \|\mathbb{E}[\hat{P}]\|_k^2$ and observe that the distributional variance depends on the **predictive kernel entropy** $-\|\hat{P}\|_k^2$, which is estimated for single models. The predictive kernel entropy also appears in the definition of kernel scores in Equation (1). This suggests that it may have a substantial influence on the generalization error. In Section 5, we will discover that this influence is extremely high (Pearson correlation of approx. 0.95), but the sign of the correlation is task-specific. Further, by using text embeddings, predictive kernel entropy is better than other baselines in predicting the performance of LLMs (c.f. Section 5).

4. Unbiased and Consistent Estimators

If the prediction $\hat{P}$ is available in closed-form, the quantities in Theorem 3.2 can be computed according to conventional approaches (Gruber & Buettner, 2023). However, this is usually not the case for generative models in Deep Learning. For example, Diffusion Models (Ho et al., 2020) or closed-source LLMs (OpenAI, 2023) are also learning the training distribution, but they are often limited to generating samples. In this section, we introduce estimators of the distributional variance and covariance for the case when only samples of the distributions are available. This increases the practical applicability of Theorem 3.2 by a wide margin and allows investigating the most recent and largest generative models without constraints. We assume a minimum of two samples from each distribution is given. All estimators in the following require a two-stage sampling procedure (Särndal et al., 2003): First, distributions are sampled in an outer loop, which can be seen as clusters. In Section 5, this will be an ensemble of generative models. Second, we sample of each distribution multiple times in an inner loop, which can be seen as within-cluster samples. This will be the data generations of each model.

The procedure differs slightly between the variance and covariance case. For simplicity, we also assume that all within-cluster sample sizes are the same. All estimators can be adjusted if that is not the case and will still be unbiased and consistent. Again, all proofs are presented in Appendix E.

4.1. Distributional Variance

Assume we have a random variable P with outcomes in $\mathscr{P}$ based on an unknown distribution $\mathbb{P}_P$ from which we

can sample. First, we sample distributions $P_1, \dots, P_n \overset{\text{iid}}{\sim} \mathbb{P}_P$. Then, we sample $X_{i1}, \dots, X_{im} \overset{\text{iid}}{\sim} P_i$ for $i = 1 \dots n$. The estimator we are about to propose is directly derived from the conventional variance estimator $\hat{\sigma}^2 := \frac{1}{n-1} \sum_{i=1}^n \left\| P_i - \frac{1}{n} \sum_{s=1}^n P_s \right\|_k^2$. Note that it holds $\hat{\sigma}^2 = \frac{1}{n} \sum_{i=1}^n \|P_i\|_k^2 - \frac{1}{n(n-1)} \sum_{i=1}^n \sum_{\substack{s=1 \\ s \neq i}}^n \langle P_i \,|\, k \,|\, P_s \rangle$, i.e. the estimator is the average of same-index pairs minus the average of the rest. Our extended estimator then uses as plug-ins $\|P_i\|_k^2 \approx \frac{1}{m(m-1)} \sum_{j=1}^m \sum_{\substack{t=1 \\ t \neq j}}^m k(X_{ij}, X_{it})$ and $\langle P_i \,|\, k \,|\, P_s \rangle \approx \frac{1}{m^2} \sum_{j=1}^m \sum_{t=1}^m k(X_{ij}, X_{st})$. The complete estimator of the distributional variance $\mathrm{Var}_k(P)$ is defined by

$$\widehat{\mathrm{Var}}_k^{(n,m)} = \underbrace{\frac{1}{nm(m-1)} \sum_{i=1}^n \sum_{\substack{j,t=1 \\ t \neq j}}^m k(X_{ij}, X_{it})}_{\text{Average similarity within clusters}} - \underbrace{\frac{1}{n(n-1)m^2} \sum_{\substack{s,i=1 \\ s \neq i}}^n \sum_{j,t=1}^m k(X_{ij}, X_{st})}_{\text{Average similarity between clusters}}. \tag{9}$$

An illustration is given on the left in Figure 2. The estimator is unbiased since $\mathbb{E}[\widehat{\mathrm{Var}}_k^{(n,m)}] = \mathrm{Var}_k(P)$. Its runtime complexity is in $\mathscr{O}(m^2 n^2)$. Estimators with lower complexity, like $\mathscr{O}(mn)$, exist but are not recommendable since they have a worse performance and in most applications, generating the samples is far more costly than evaluating the estimator. The variance of the estimator is in $\mathscr{O}\left(\frac{1}{n}\left(1 + \frac{1}{m}\right)\right)$, which proves $\widehat{\mathrm{Var}}_k^{(n,m)} \longrightarrow \mathrm{Var}_k(P)$ in probability with growing n but not m. In words, the estimator is consistent with increasing outer samples but not inner samples. This may suggest to neglect creating inner samples and keep m small, but our analysis in Appendix E.2 shows that there exist sub-terms which converge equally fast in m as in n. In combination with the finite sample simulation in Figure 2, we recommend to use $m \geq n \geq 10$, if no prior information is available.

4.2. Distributional Covariance and Correlation

For the covariance case, assume we have random variables P and Q with outcomes in $\mathscr{P}$ based on an unknown joint distribution $\mathbb{P}_{PQ}$ from which we can sample. We require samples $X_{i1}, \dots, X_{im} \overset{\text{iid}}{\sim} P_i$ and $Y_{i1}, \dots, Y_{im} \overset{\text{iid}}{\sim} Q_i$ with $(P_1, Q_1), \dots, (P_n, Q_n) \overset{\text{iid}}{\sim} \mathbb{P}_{PQ}$. Then, we propose the unbiased and consistent covariance estimator

$$\widehat{\mathrm{Cov}}_k^{(n,m)}(\mathbf{X}, \mathbf{Y}) = \frac{1}{nm^2} \sum_{i=1}^n \sum_{j,t=1}^m k(X_{ij}, Y_{it}) - \frac{1}{(n-1)nm^2} \sum_{\substack{i,s=1 \\ s \neq i}}^n \sum_{j,t=1}^m k(X_{ij}, Y_{st}) \tag{10}$$

with $\mathbf{X} := (X_{ij})_{i=1\dots n, j=1\dots m}$ and $\mathbf{Y} := (Y_{ij})_{i=1\dots n, j=1\dots m}$. It has the same runtime complexity and convergence rate as the variance estimator of Equation (9) (c.f. Appendix E.3). While the distributional covariance is directly implied by Theorem 3.2, it is difficult to interpret since it is not bounded. Consequently, we propose the distributional correlation estimator based on Equation (10) given by

$$\widehat{\mathrm{Corr}}_k^{(n,m)} = \frac{\widehat{\mathrm{Cov}}_k^{(n,m)}(\mathbf{X}, \mathbf{Y})}{\sqrt{\widehat{\mathrm{Cov}}_k^{(n,m)}(\mathbf{X}, \mathbf{X}) \, \widehat{\mathrm{Cov}}_k^{(n,m)}(\mathbf{Y}, \mathbf{Y})}}, \tag{11}$$

which is in $[-1, 1]$. It is consistent since continuous transformations of consistent estimators are also consistent (Shao, 2003), i.e. for $n \to \infty$ in probability

$$\widehat{\mathrm{Corr}}_k^{(n,m)} \longrightarrow \mathrm{Corr}_k(P, Q) := \frac{\mathrm{Cov}_k(P, Q)}{\sqrt{\mathrm{Var}_k(P) \, \mathrm{Var}_k(Q)}}. \tag{12}$$

We use the covariance estimator for the variance terms since the variance estimator can be negative and the correlation estimator is only asymptotically unbiased no matter the choice. Note that Pearson correlation is recovered by using the kernel k_δ of Example 3.3.

In the next section, we show that distributional correlation, as implied by Theorem 3.2, is a natural tool to gain new insights into the fitting process of generative models. For example, the correlations between epochs indicate how stable the convergence during training is.

5. Applications

In this section, we apply the proposed statistical tools to assess generative models across a variety of different data generation tasks. Specifically, we put emphasis on instance-level uncertainty estimation. A meaningful measure of uncertainty is able to predict the loss for a given prediction. We use the Pearson correlation coefficient to quantify how well an uncertainty measure predicts a continuous loss, and the area under receiver operator characteristic (AUROC) for a binary loss. We first start in Section 5.1 with diffusion models for image generation on the synthetic InfiMNIST dataset for a detailed examination of their generalization behavior. In Section 5.2, we evaluate an ensemble of Glow-TTS models for text-to-speech synthesis on the SpeechLJ

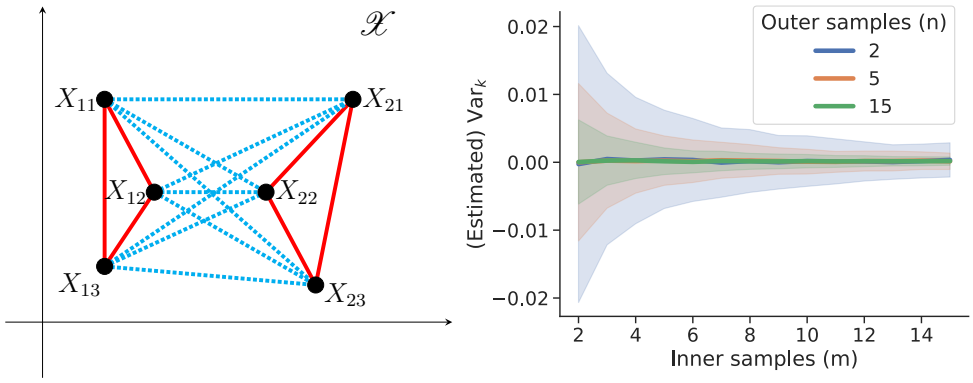


Figure 2. **Left:** Illustration of the estimator $\widehat{\mathrm{Var}}_k^{(n,m)}$ in the sample space $\mathscr{X}$ for $n = 2$ outer samples and $m = 3$ inner samples. The estimator computes the average similarity within clusters (solid red lines) minus the average similarity between clusters (dotted blue lines). Shorter lines indicate higher similarity and larger kernel values. **Right:** Estimator standard deviation for various sample sizes. Even though the estimator does not converge in theory with the inner sample size m, it may still be influenced significantly by it for small sample sizes.

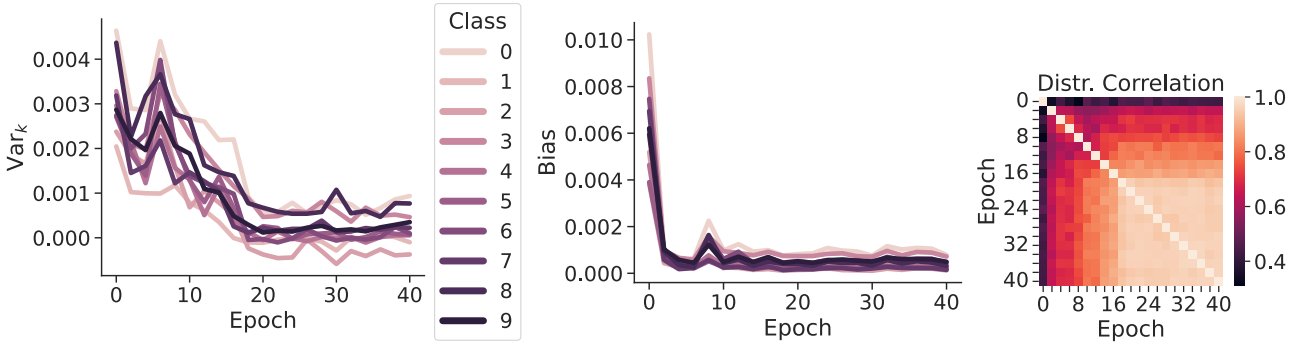


Figure 3. **Left:** The variance starts high and is reduced throughout training. From 20 epochs onwards, the variance stays stable for all classes and no overfitting can be observed. **Mid:** The bias is reduced a lot quicker than the variance, reaching its minimum at 5 epochs, and converges after 10 epochs. **Right:** The distributional correlation between training epochs shows similar to the variance that convergence happens around epoch 20. Remarkably, the 'square' of very high correlations indicates that the model is stable in its convergence and does not iterate through equally good solutions.

dataset. In all cases, kernel entropy shows strong performance as uncertainty measure. Last, we use kernel entropy to outperform other baselines in uncertainty estimation for natural language generation in Section 5.3. There, we evaluate single OPT models of different sizes on the question answering datasets CoQA and TriviaQA. The source code of all experiments is openly available at `https://github.com/MLO-lab/BVCD_generative_models`.

5.1. Image Generation

For image generation, we use conditional diffusion models (Ho et al., 2020; Ho & Salimans, 2021) trained on MNIST-like datasets. We use InfiMNIST to sample an infinite number of uniquely perturbed MNIST images (Loosli et al., 2007). By simulating the data generation process we can assess the ground truth generalization error of the diffusion model. We sample $n = 20$ distinct training sets from InfiMNIST each of size 60.000 (similar as MNIST). We then train a model on each training set. This is in correspondance to how generalization error, bias, and variance are evaluated for regression and classification tasks (Ueda & Nakano, 1996; Gruber & Buettner, 2023). We use $m = 20$ generated images per class and per model for all estimators. The predictive kernel entropy is only evaluated on a single model to stay as closely as possible to practical constraints. Our kernel choice is the commonly used RBF kernel $k_{\mathrm{rbf}}(x, y) = \exp(-\gamma \|x - y\|_2^2)$, where x and y are flattened images and γ a normalization factor based on the number of pixels (Schölkopf, 1997; Schölkopf & Smola, 2002; Liu et al., 2020). We first analyse the generalization behavior and then assess different approaches for uncertainty estimation. In Figure 3, we plot the distributional variance, bias, and distributional correlation throughout training

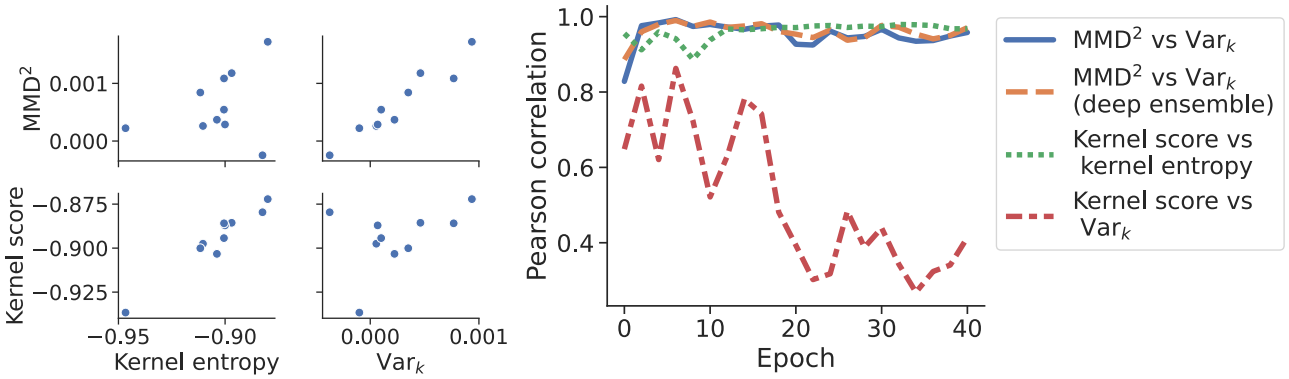


Figure 4. **Left:** Dependence between squared MMD and distributional variance. The distributional variance correlates linearly with the squared MMD. **Right:** Pearson correlation between squared MMD and distributional variance is very high (≈ 0.95) throughout training. Approximation via deep ensembles does not deteriorate this relation. Consequently, distributional variance and kernel entropy represent viable measure of uncertainty.

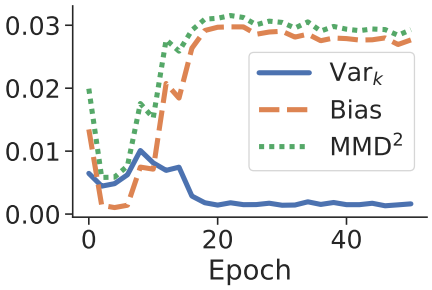


Figure 5. MMD^2, variance, and bias for class '0' throughout training with reduced training set of '0's. After 5 epochs, mode collapse occurs, which is only expressed in the increased bias. This indicates, that mode collapse is a contrary phenomenon to overfitting.

for every second epoch. As can be seen, the model converges quicker for the bias than the variance (around epochs 10 and 20). Further, no overfitting occurs since both variance and bias stay small. The correlation matrix also shows convergence around epoch 20, but, more interestingly, it shows a square of high correlations in the lower right corner. This is an indication that the diffusion model training is stable in its convergence and does not iterate through different minima in the optimization landscape.

In Figure 4, we compare the relations between kernel score and MMD^2 to distributional variance and predictive kernel entropy for each class. As can be seen, the predictive kernel entropy correlates strongly linearly with the generalization error (kernel score) but not so much with the generalization discrepancy (MMD^2), while the distributional variance correlates strongly linearly with the MMD^2 but not the kernel score. This correlation can be observed throughout the whole training and gives a very high Pearson correlation coefficient of around 0.95. Importantly for practical settings, the correlation between MMD^2 and distributional variance does not deteriorate when we use a deep ensemble trained on a single dataset.

In summary, these results demonstrate that both distributional variance as well as predictive kernel entropy are viable measure of uncertainty to predict the correctness of generated instances, either in terms of kernel score or MMD^2.

We next set out to use our estimators for bias and variance to elucidate the phenomenon of mode collapse. When generative models are used to learn the distribution of a given training set, there are often groups of different sizes present. In these cases, a common occurence is mode collapse towards the majority groups, i.e. the model catastrophically fails to model the minority groups. To simulate this scenario in our setting, we repeat our evaluation but reduce the frequency of images of digit '0' to $\approx 1\%$ in each training set. The bias-variance curves of digit '0' across training (Figure 5) reveals the expected mode collapse, and, most importantly, demonstrates that it is only expressed in terms of the bias. The variance term is further reduced throughout training as if no collapse occured. This suggests that mode collapse may be seen as a contrary phenomenon to overfitting. While in overfitting, the variance increases and the bias reduces, this is vice versa for mode collapse for prolonged training. In Appendix D, we conduct additional evaluations and confirm that our findings are robust with respect to common kernel choices.

5.2. Audio Generation

We next evaluate the fitting and generalization behavior of the generative flow model Glow-TTS (Kim et al., 2020) on the text-to-speech dataset LJSpeech (Ito & Johnson, 2017) throughout training (Appendix D). We train a deep ensemble via $n = 10$ different weight initializations on 90% of the available data. We evaluate the models every 2000 training

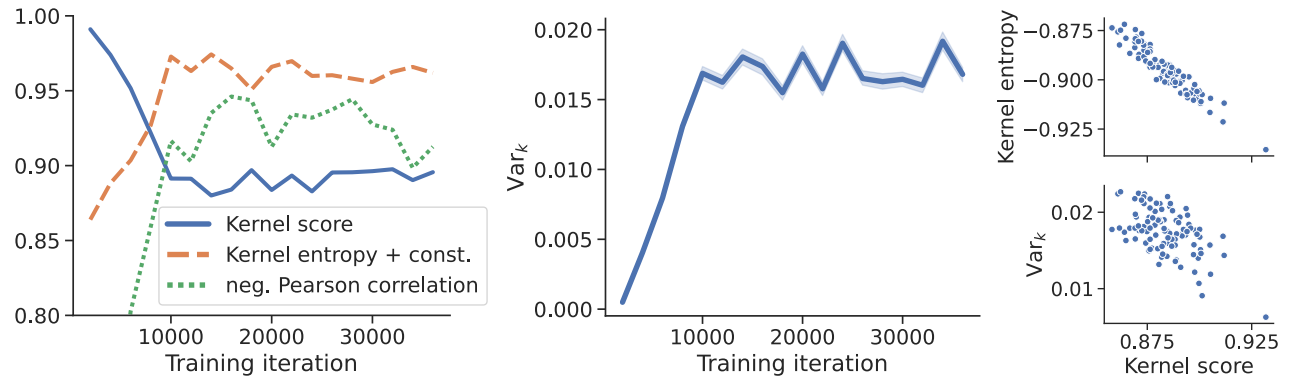


Figure 6. **Left:** Kernel score, predictive kernel entropy, and their Pearson correlation throughout training for Glow-TTS. The entropy indicates that the model initially predicts too narrow distributions, which widen until convergence at around 10000 iterations. After convergence, the correlation between predictive entropy and kernel score is very high. **Mid:** The variance of an Deep Ensemble is also initially very small and converges at the same time as kernel score and entropy. **Right:** Comparing kernel score and kernel entropy as well as variance for 100 test instances at 16000 training iterations. Again, the correlation shows strong linearity for kernel entropy.

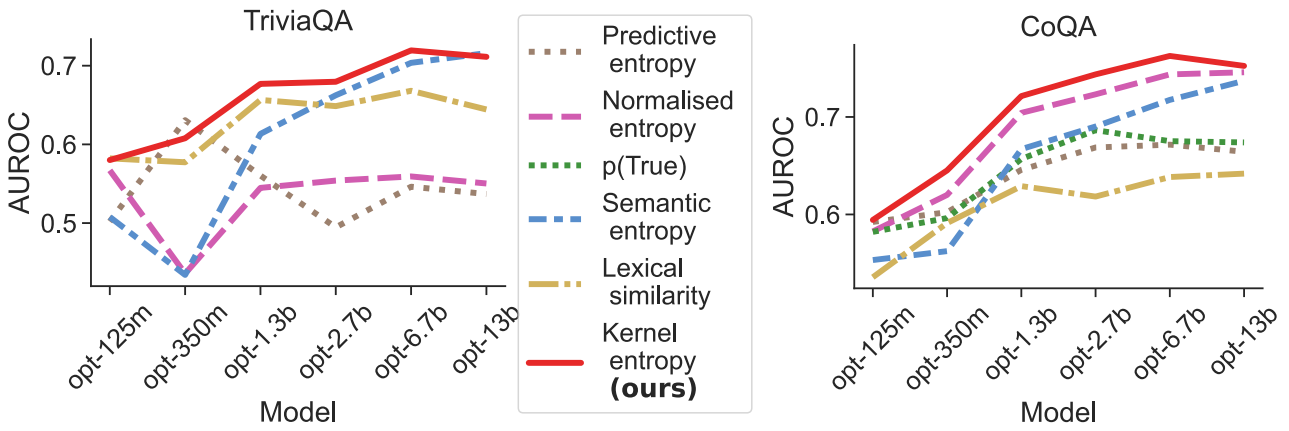


Figure 7. Area-Under-Curve of answer accuracy based on thresholds for different uncertainty measures. The kernel entropy outperforms other baselines in predicting the correctness of generated answers across a wide range of differently sized models.

iterations by generating $m = 10$ speech waveforms for each of 100 test instances. The evaluation includes the kernel score, kernel entropy and the distributional variance. Note that ensemble members are only required to compute the distributional variance – we compute kernel entropy for a single model, as before. Here, we use the Laplacian kernel $k_{\mathrm{lap}}(x, y) = \exp(-\gamma \|x - y\|_1)$ (Schölkopf & Smola, 2002), where x and y are waveforms represented by vectors and γ a normalization constant based on the waveform length. The results are depicted in Figure 6. Our analyses reveal that similar as for image generation, overfitting does not occur for prolonged training and the Pearson correlation between kernel entropy and kernel score is very high after convergence. But, the entropy is initially very small and increases until convergence. This indicates that a successful training requires to widen the predicted distribution. Consequently, the correlation between kernel entropy and kernel score is negative, since badly fitted instances have a more narrow predicted distribution. We also conducted the evaluations with the RBF kernel, which gives similar but slightly more erratic curves than the Laplacian kernel (Appendix D).

5.3. Natural Language Generation

An instance-level uncertainty measure is supposed to predict the correctness of an individual prediction and should therefore be highly correlated to the loss. In all experiments so far, we observed a very high correlation between kernel entropy and kernel score. This indicates that kernel entropy is an excellent measure of uncertainty. In the following, we examine kernel entropy to predict the correctness of LLMs on question answering datasets. Here, the setup differs from the previous experiments by two aspects.

First, we follow Kuhn et al. (2023) and do not use a kernel score but a binarized version of the RougeL similarity as loss. For two sequences s, t, it is defined as $\mathrm{RougeL}(s,t) = \frac{2}{\mathrm{length}(s)+\mathrm{length}(t)} \mathrm{LCS}(s,t)$, where LCS is the length of the **l**ongest **c**ommon **s**equence between its two inputs (Lin & Och, 2004). Kuhn et al. (2023) propose to use the binary loss $L(answer, target) = \mathbf{1}_{\mathrm{RougeL}(answer,target)>0.3}$. Specifically, they claim that this particular loss matches human-based evaluation 89% of the time for CoQA and 96% of the time for TriviaQA. This turns predicting the loss value into a binary classification problem. Consequently, the AUROC is more meaningful than Pearson correlation for evaluating the performance of uncertainty measures (Kadavath et al., 2022).

Second, we do not directly use the generated answers as inputs for a kernel but, instead, their vector embeddings. A well-trained vector embedder maps text into a semantically meaningful vector space in which a kernel then measures similarities (Camacho-Collados & Pilehvar, 2018).

The investigated uncertainty baselines are predictive entropy, normalised (predictive) entropy, p(True), semantic entropy, and lexical similarity (Kuhn et al., 2023). We consider uncertainty estimation for question answering predictions of the datasets CoQA (Reddy et al., 2019) with 7983 test instances and TriviaQA (Joshi et al., 2017) with 17943 test instances. We use OPT models (Zhang et al., 2022) of all available sizes except the 30 billion parameter version, which is computationally prohibitive. For our kernel entropy, we use the RBF kernel and text embeddings computed via a pretrained e5-small-v2 (Wang et al., 2022). The results are depicted in Figure 7. As can be seen, kernel entropy is the most robust approach and outperforms other baselines for uncertainty estimation in almost all cases. We can achieve further improvements in our approach when we use alternative embedders (c.f. Appendix D). The cosine similarity, which is used in natural language processing (Steinbach et al., 2000), and other kernels show similar results as the RBF kernel in Appendix D.

Our approach of combining an embedding model with a p.s.d. kernel outperforms all other uncertainty baselines. However, a good embedding model might not always be accessible for every language setup. In Appendix D, we additionally evaluate kernel entropy based on a sequence kernel, which does not require an embedding model. While the performance is worse compared to using an embedder, the sequence kernel is still competitive and clearly outperforms lexical similarity.

6. Limitations

Our work builds upon large bodies of literature, and, consequently, shares their limitations. We give an overview of these in the following.

Our bias-variance decomposition assumes that the MMD and kernel score are meaningful measures of generalization performance. The choice of kernel can be inspired by the large body of work on support vector machines (Schölkopf & Smola, 2002) and MMD-based hypothesis tests (Gretton et al., 2012a). Further, we share similar limitations as other bias-variance decompositions in the literature (Ueda & Nakano, 1996; Gruber & Buettner, 2023). Specifically, the intractability of simulating the exact training distribution is often limiting the bias-variance estimates in practice. Deep ensembles can give reasonable approximations, which we use for the audio experiments.

Our language experiments are adopted from past literature on uncertainty estimation (Malinin & Gales, 2020; Kadavath et al., 2022; Kuhn et al., 2023). Since this branch of research is mostly empirical, it is not possible to give guarantees of how each uncertainty baseline performs for new settings. This limitation also holds for kernel entropy, and is underlined by its strong but negative correlations with the error for the audio experiments. Recent theoretical work in the classification setting explored the link between uncertainty and accuracy via the class-aggregated calibration error (CACE) (Jiang et al., 2021; Kirsch & Gal, 2022). However, this relationship still needs to be generalized because CACE is not defined for non-classification models.

Last, our estimators require i.i.d. samples to converge strongly with sample size (Capiński & Kopp, 2004). This usually holds for samples from an individual model (i.e. when we estimate kernel entropy, kernel score, or MMD). However, it may not hold when estimating the variance, covariance, or bias of the model.

7. Conclusion

In this work we introduced the first bias-variance-covariance decomposition beyond the mean squared error for kernel scores. We proposed estimators for the variance and covariance terms which only require samples of the predictive distributions. This allows to evaluate all terms in the composition for arbitrary generative models and even in the closed-source setting. We studied empirically the fitting behavior of common models for image and audio generations, and demonstrated that kernel entropy and variance are viable measures of uncertainty. Finally, we showed that kernel entropy outperforms other baselines for predicting the correctness of LLMs in question answering tasks.

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning theoretically and empirically. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here. However, we emphasise that uncertainty estimation may improve safety and trustworthiness of generative models but without

guarantees. Practitioners should always verify any adaptations on an unseen test set and be wary of distribution shifts.

Acknowledgements

Co-funded by the European Union (ERC, TAIPO, 101088594 to FB). Views and opinions expressed are however those of the authors only and do not necessarily reflect those of the European Union or the European Research Council. Neither the European Union nor the granting authority can be held responsible for them.

References

Baum, J., Kanagawa, H., and Gretton, A. A kernel stein test of goodness of fit for sequential models. In *International Conference on Machine Learning*, pp. 1936–1953. PMLR, 2023.

Bińkowski, M., Sutherland, D. J., Arbel, M., and Gretton, A. Demystifying MMD GANs. In *International Conference on Learning Representations*, 2018.

Bishop, C. M. and Nasrabadi, N. M. *Pattern recognition and machine learning*, volume 4. Springer, 2006.

Breiman, L. Random forests. *Machine learning*, 45(1): 5–32, 2001.

Brier, G. W. Verification of forecasts expressed in terms of probability. *Monthly Weather Review*, 78(1):1 – 3, 1950. doi: 10.1175/1520-0493(1950)078⟨0001:VOFEIT⟩2.0.CO;2. URL https://journals.ametsoc.org/view/journals/mwre/78/1/1520-0493_1950_078_0001_vofeit_2_0_co_2.xml.

Brown, G. *Diversity in neural network ensembles*. PhD thesis, Citeseer, 2004.

Camacho-Collados, J. and Pilehvar, M. T. From word to sense embeddings: A survey on vector representations of meaning. *Journal of Artificial Intelligence Research*, 63: 743–788, 2018.

Capiński, M. and Kopp, P. E. *Measure, integral and probability*, volume 14. Springer, 2004.

Choi, J., Lee, J., Shin, C., Kim, S., Kim, H., and Yoon, S. Perception prioritized training of diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11472–11481, 2022.

Chwialkowski, K., Strathmann, H., and Gretton, A. A kernel test of goodness of fit. In *International conference on machine learning*, pp. 2606–2615. PMLR, 2016.

Dawid, A. P. The geometry of proper scoring rules. *Annals of the Institute of Statistical Mathematics*, 59(1):77–93, 2007.

Dziugaite, G. K., Roy, D. M., and Ghahramani, Z. Training generative neural networks via maximum mean discrepancy optimization. In *Proceedings of the Thirty-First Conference on Uncertainty in Artificial Intelligence*, pp. 258–267, 2015.

Eaton, M. A method for evaluating improper prior distributions. Technical report, University of Minnesota, 1981.

Eaton, M. L., Giovagnoli, A., and Sebastiani, P. A predictive approach to the bayesian design problem with application to normal regression models. *Biometrika*, 83(1):111–125, 1996.

Eren, G. and The Coqui TTS Team. Coqui TTS, January 2021. URL https://github.com/coqui-ai/TTS.

Fomicheva, M., Sun, S., Yankovskaya, L., Blain, F., Guzmán, F., Fishel, M., Aletras, N., Chaudhary, V., and Specia, L. Unsupervised quality estimation for neural machine translation. *Transactions of the Association for Computational Linguistics*, 8:539–555, 2020.

Friedman, J. H. Stochastic gradient boosting. *Computational statistics & data analysis*, 38(4):367–378, 2002.

Gneiting, T. and Raftery, A. E. Strictly proper scoring rules, prediction, and estimation. *Journal of the American Statistical Association*, 102(477):359–378, 2007. doi: 10.1198/016214506000001437. URL https://doi.org/10.1198/016214506000001437.

Gretton, A., Herbrich, R., and Smola, A. J. The kernel mutual information. In *2003 IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003. Proceedings.(ICASSP'03).*, volume 4, pp. IV–880. IEEE, 2003.

Gretton, A., Borgwardt, K. M., Rasch, M. J., Schölkopf, B., and Smola, A. A kernel two-sample test. *Journal of Machine Learning Research*, 13(25):723–773, 2012a. URL http://jmlr.org/papers/v13/gretton12a.html.

Gretton, A., Sejdinovic, D., Strathmann, H., Balakrishnan, S., Pontil, M., Fukumizu, K., and Sriperumbudur, B. K. Optimal kernel choice for large-scale two-sample tests. *Advances in neural information processing systems*, 25, 2012b.

Gruber, S. G. and Buettner, F. Uncertainty estimates of predictions via a general bias-variance decomposition. In *International Conference on Artificial Intelligence and Statistics*, pp. 11331–11354. PMLR, 2023.

Hastie, T., Tibshirani, R., Friedman, J. H., and Friedman, J. H. *The elements of statistical learning: data mining, inference, and prediction*, volume 2. Springer, 2009.

Hekler, A., Brinker, T. J., and Buettner, F. Test time augmentation meets post-hoc calibration: uncertainty quantification under real-world conditions. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 14856–14864, 2023.

Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., and Hochreiter, S. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017.

Ho, J. and Salimans, T. Classifier-free diffusion guidance. In *NeurIPS 2021 Workshop on Deep Generative Models and Downstream Applications*, 2021.

Ho, J., Jain, A., and Abbeel, P. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

Ito, K. and Johnson, L. The lj speech dataset. https://keithito.com/LJ-Speech-Dataset/, 2017.

Jiang, Y., Nagarajan, V., Baek, C., and Kolter, J. Z. Assessing generalization of sgd via disagreement. In *International Conference on Learning Representations*, 2021.

Joshi, M., Choi, E., Weld, D. S., and Zettlemoyer, L. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1601–1611, 2017.

Kadavath, S., Conerly, T., Askell, A., Henighan, T., Drain, D., Perez, E., Schiefer, N., Hatfield-Dodds, Z., DasSarma, N., Tran-Johnson, E., et al. Language models (mostly) know what they know. *arXiv preprint arXiv:2207.05221*, 2022.

Karras, T., Aittala, M., Hellsten, J., Laine, S., Lehtinen, J., and Aila, T. Training generative adversarial networks with limited data. *Advances in neural information processing systems*, 33:12104–12114, 2020.

Kasneci, E., Seßler, K., Küchemann, S., Bannert, M., Dementieva, D., Fischer, F., Gasser, U., Groh, G., Günnemann, S., Hüllermeier, E., et al. Chatgpt for good? on opportunities and challenges of large language models for education. *Learning and individual differences*, 103: 102274, 2023.

Kim, J., Kim, S., Kong, J., and Yoon, S. Glow-tts: A generative flow for text-to-speech via monotonic alignment search. *Advances in Neural Information Processing Systems*, 33:8067–8077, 2020.

Király, F. J. and Oberhauser, H. Kernels for sequentially ordered data. *Journal of Machine Learning Research*, 20 (31):1–45, 2019.

Kirsch, A. and Gal, Y. A note on" assessing generalization of sgd via disagreement". *Transactions on Machine Learning Research*, 2022.

Kübler, J., Jitkrittum, W., Schölkopf, B., and Muandet, K. Learning kernel tests without data splitting. *Advances in Neural Information Processing Systems*, 33:6245–6255, 2020.

Kuhn, L., Gal, Y., and Farquhar, S. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. In *The Eleventh International Conference on Learning Representations*, 2023.

Lakshminarayanan, B., Pritzel, A., and Blundell, C. Simple and scalable predictive uncertainty estimation using deep ensembles. *Advances in neural information processing systems*, 30, 2017.

Li, C.-L., Chang, W.-C., Cheng, Y., Yang, Y., and Póczos, B. Mmd gan: Towards deeper understanding of moment matching network. *Advances in neural information processing systems*, 30, 2017.

Li, Y., Swersky, K., and Zemel, R. Generative moment matching networks. In *International conference on machine learning*, pp. 1718–1727. PMLR, 2015.

Li, Z., Zhang, X., Zhang, Y., Long, D., Xie, P., and Zhang, M. Towards general text embeddings with multi-stage contrastive learning. *arXiv preprint arXiv:2308.03281*, 2023.

Lin, C.-Y. and Och, F. J. Automatic evaluation of machine translation quality using longest common subsequence and skip-bigram statistics. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL-04)*, pp. 605–612, 2004.

Liu, F., Xu, W., Lu, J., Zhang, G., Gretton, A., and Sutherland, D. J. Learning deep kernels for non-parametric two-sample tests. In *International conference on machine learning*, pp. 6316–6326. PMLR, 2020.

Liu, Y. and Yao, X. Ensemble learning via negative correlation. *Neural networks*, 12(10):1399–1404, 1999a.

Liu, Y. and Yao, X. Simultaneous training of negatively correlated neural networks in an ensemble. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 29(6):716–725, 1999b.

Liu, Z., Luo, P., Wang, X., and Tang, X. Deep learning face attributes in the wild. In *Proceedings of the IEEE*

international conference on computer vision, pp. 3730–3738, 2015.

Loosli, G., Canu, S., and Bottou, L. Training invariant support vector machines using selective sampling. In Bottou, L., Chapelle, O., DeCoste, D., and Weston, J. (eds.), *Large Scale Kernel Machines*, pp. 301–320. MIT Press, Cambridge, MA., 2007. URL http://leon.bottou.org/papers/loosli-canu-bottou-2006.

Malinin, A. and Gales, M. Uncertainty estimation in autoregressive structured prediction. In *International Conference on Learning Representations*, 2020.

Meskó, B. and Topol, E. J. The imperative for regulatory oversight of large language models (or generative ai) in healthcare. *npj Digital Medicine*, 6(1):120, 2023.

Murphy, K. P. *Probabilistic Machine Learning: An introduction*. MIT Press, 2022. URL probml.ai.

Ning, Y., He, S., Wu, Z., Xing, C., and Zhang, L.-J. A review of deep learning based speech synthesis. *Applied Sciences*, 9(19):4050, 2019.

OpenAI. Gpt-4 technical report, 2023.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., and Chintala, S. Pytorch: An imperative style, high-performance deep learning library. In Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 32*, 2019.

Paul, D., Sanap, G., Shenoy, S., Kalyane, D., Kalia, K., and Tekade, R. K. Artificial intelligence in drug discovery and development. *Drug discovery today*, 26(1):80, 2021.

Radford, A., Metz, L., and Chintala, S. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, 2015.

Rame, A., Kirchmeyer, M., Rahier, T., Rakotomamonjy, A., Gallinari, P., and Cord, M. Diverse weight averaging for out-of-distribution generalization. *Advances in Neural Information Processing Systems*, 35:10821–10836, 2022.

Ramesh, A., Pavlov, M., Goh, G., Gray, S., Voss, C., Radford, A., Chen, M., and Sutskever, I. Zero-shot text-to-image generation. In *International Conference on Machine Learning*, pp. 8821–8831. PMLR, 2021.

Reddy, S., Chen, D., and Manning, C. D. Coqa: A conversational question answering challenge. *Transactions of the Association for Computational Linguistics*, 7:249–266, 2019.

Ren, Y., Zhu, J., Li, J., and Luo, Y. Conditional generative moment-matching networks. *Advances in Neural Information Processing Systems*, 29, 2016.

Ren, Y., Luo, Y., and Zhu, J. Improving generative moment matching networks with distribution partition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 9403–9410, 2021.

Särndal, C.-E., Swensson, B., and Wretman, J. *Model assisted survey sampling*. Springer Science & Business Media, 2003.

Schölkopf, B. *Support vector learning*. PhD thesis, Oldenbourg München, Germany, 1997.

Schölkopf, B. and Smola, A. J. *Learning with kernels: support vector machines, regularization, optimization, and beyond*. MIT press, 2002.

Schrab, A., Kim, I., Guedj, B., and Gretton, A. Efficient aggregated kernel tests using incomplete u-statistics. *Advances in Neural Information Processing Systems*, 35: 18793–18807, 2022.

Schrab, A., Kim, I., Albert, M., Laurent, B., Guedj, B., and Gretton, A. Mmd aggregated two-sample test. *Journal of Machine Learning Research (JMLR)*, 24, 2023.

Shao, J. *Mathematical statistics*. Springer Science & Business Media, 2003.

Shekhar, S., Kim, I., and Ramdas, A. A permutation-free kernel two-sample test. *Advances in Neural Information Processing Systems*, 35:18168–18180, 2022.

Song, K., Tan, X., Qin, T., Lu, J., and Liu, T.-Y. Mpnet: Masked and permuted pre-training for language understanding. *Advances in Neural Information Processing Systems*, 33:16857–16867, 2020.

Steinbach, M., Karypis, G., and Kumar, V. A comparison of document clustering techniques. Technical report, Department of Computer Science and Engineering, University of Minnesota, 2000.

Steinwart, I. and Ziegel, J. F. Strictly proper kernel scores and characteristic kernels on compact spaces. *Applied and Computational Harmonic Analysis*, 51:510–542, 2021.

Tseng, H.-Y., Li, Q., Kim, C., Alsisan, S., Huang, J.-B., and Kopf, J. Consistent view synthesis with pose-guided diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16773–16783, 2023.

Ueda, N. and Nakano, R. Generalization error of ensemble estimators. In *Proceedings of International Conference on Neural Networks (ICNN'96)*, volume 1, pp. 90–95. IEEE, 1996.

Wang, L., Yang, N., Huang, X., Jiao, B., Yang, L., Jiang, D., Majumder, R., and Wei, F. Text embeddings by weakly-supervised contrastive pre-training. *arXiv preprint arXiv:2212.03533*, 2022.

Wang, W., Wei, F., Dong, L., Bao, H., Yang, N., and Zhou, M. Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers. *Advances in Neural Information Processing Systems*, 33: 5776–5788, 2020.

Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., et al. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 conference on empirical methods in natural language processing: system demonstrations*, pp. 38–45, 2020.

Wu, J. and Shang, S. Managing uncertainty in ai-enabled decision making and achieving sustainability. *Sustainability*, 12(21):8758, 2020.

Zhang, S., Roller, S., Goyal, N., Artetxe, M., Chen, M., Chen, S., Dewan, C., Diab, M., Li, X., Lin, X. V., et al. Opt: Open pre-trained transformer language models. *arXiv preprint arXiv:2205.01068*, 2022.

A. Overview

In the following, we offer an extended discussion of kernel scores and MMD literature in Appendix B, further theoretical results in Appendix C, more experimental details and additional experiments in Appendix D, and we provide all missing proofs in Appendix E.

B. Related Work on Kernel Scores and Maximum Mean Discrepancy

The kernel score and the MMD^2 only differ by a target dependent constant (Steinwart & Ziegel, 2021). Consequently, the gradient of the kernel score and the MMD^2 are equal, and, thus, optimizing kernel score or MMD^2 gives similar results. Importantly, Bińkowski et al. (2018) demonstrated that a specific setup of the MMD^2, referred to as kernel inception distance (KID), is a better metric for image generation than the commonly used Fréchet inception distance (FID) (Heusel et al., 2017). This directly motivates the use of the MMD^2 (and thereby the kernel score) as loss/metric to evaluate generative models. The MMD^2 can be intuitively explained via the kernel trick (Schölkopf, 2002): A kernel can be decomposed into an inner product, which turns the MMD^2 into the squared euclidean distance within this inner product space (referred to as reproducing kernel Hilbert space, c.f. Equation (15)). The specific form of the inner product space depends on the chosen kernel. Originally, Gretton et al. (2012a) introduced the MMD for non-parametric hypothesis tests. Li et al. (2015) and Dziugaite et al. (2015) simultaneously demonstrated that the MMD can be used to train state-of-the-art (w.r.t. 2015) image generators. These models are usually referred to as moment matching networks and successive improvements have been proposed (Ren et al., 2016; Li et al., 2017; Ren et al., 2021). Even though contemporary state-of-the-art image generation training optimizes a different loss, the MMD is still relevant for image generation model selection via the KID (Karras et al., 2020; Choi et al., 2022; Tseng et al., 2023).

C. Extended Theoretical Results

In this section, we provide more minor theoretical results, which are very close to our main contribution in Theorem 3.2.

C.1. Decomposition in Reproducing Kernel Hilbert Spaces

The literature on MMD expanded to a significant size in the last decade (Gretton et al., 2012a;b; Chwialkowski et al., 2016; Liu et al., 2020; Kübler et al., 2020; Shekhar et al., 2022; Schrab et al., 2022; 2023). The MMD is usually used in the context of a reproducing kernel Hilbert spaces (RKHS) (Schölkopf & Smola, 2002). In the following, we express Theorem 3.2 according to a RKHS $\mathscr{H}$ to offer an alternative perspective on our result. Assume the kernel $k\colon \mathscr{Y} \times \mathscr{Y} \to \mathbb{R}$ is associated with an RKHS $\mathscr{H}$ with inner product $\langle .,. \rangle_{\mathscr{H}}$ and norm $\|.\|_{\mathscr{H}}$ such that $k(x,y) = \langle k(x,.), k(y,.) \rangle_{\mathscr{H}}$. The norm based on k in the distribution space relates to the RKHS norm via $\|Q\|_k = \|\mu_Q\|_{\mathscr{H}}$ with mean embedding $\mu_Q \coloneqq \mathbb{E}[k(Y,.)] \in \mathscr{H}$ for a $Y \sim Q \in \mathscr{P}$. Consequently, given a prediction $\hat{P}$ we have

$$\underbrace{\mathbb{E}\left[S_k\left(\hat{P}, Y\right)\right]}_{\text{Generalization Error}} = \underbrace{-\|\mu_Q\|^2_{\mathscr{H}}}_{\text{Noise}} + \underbrace{\left\|\mathbb{E}\left[\mu_{\hat{P}}\right] - \mu_Q\right\|^2_{\mathscr{H}}}_{\text{Bias}} + \underbrace{\mathbb{E}\left[\left\|\mu_{\hat{P}} - \mathbb{E}\left[\mu_{\hat{P}}\right]\right\|^2_{\mathscr{H}}\right]}_{\text{Variance}}. \tag{13}$$

The covariance decomposition can be expressed similarly since $\langle P \mid k \mid Q \rangle = \langle \mu_P, \mu_Q \rangle_{\mathscr{H}}$. Note that the bias and variance terms in Theorem 3.2 and Equation (13) are equal. As a further note, we can also express the kernel score and the MMD for $P, Q \in \mathscr{P}$ and $y \in \mathscr{Y}$ in terms of the RKHS since it holds

$$S_k(P, y) = \|\mu_P\|^2_{\mathscr{H}} - 2\langle \mu_P, k(y,.) \rangle_{\mathscr{H}} \tag{14}$$

and

$$\mathrm{MMD}^2_k(P, Q) = \|\mu_P - \mu_Q\|^2_{\mathscr{H}} \tag{15}$$

(c.f. (Steinwart & Ziegel, 2021) for a more detailed discussion).

C.2. Covariance decomposition without Identical Distribution Assumption

Let k be a p.s.d. kernel and $\hat{P}$ a predicted distribution for a target $Y \sim Q$ similar as in Theorem 3.2.

Algorithm 1 Estimating predictive kernel entropy

Required: Generated outputs $a_1, \ldots, a_n$ for a given input, p.s.d. kernel k, (optional) embedder ϕ
if ϕ is given **then**
  for $i = 1$ **to** n **do**
    $a_i \leftarrow \phi(a_i)$
  end for
end if
Return: $-\frac{1}{n(n-1)} \sum_{i=1}^{n} \sum_{j \neq i}^{n} k(a_i, a_j)$

If we have an ensemble prediction $\hat{P}^{(n)} := \frac{1}{n} \sum_{i=1}^{n} \hat{P}_i$ with members $\hat{P}_1, \ldots, \hat{P}_n$, then

$$\mathrm{Var}_k\left(\hat{P}^{(n)}\right) = \frac{1}{n^2} \sum_{i=1}^{n} \mathrm{Var}_k\left(\hat{P}_i\right) + \frac{1}{n^2} \sum_{i=1}^{n} \sum_{\substack{j=1 \\ j \neq i}}^{n} \mathrm{Cov}_k\left(\hat{P}_i, \hat{P}_j\right). \tag{16}$$

This results is part of the proof in Equation (20) of Theorem 3.2.

D. Extended Experiments

In this section, we give more details on the experiments and show further results.

D.1. Experimental Details

We give some additional details on the experimental setup. First, we formalize computing the kernel entropy in Algorithm 1. Second, we describe in more detail the image, audio, and natural language generation experiments.

D.1.1. Image Generation

We use the following procedure for the simulation in Figure 2. First, we sample 32 distinct training sets from InfiMNIST of size 60.000. Then, we train a conditional diffusion model on each training set for 20 epochs. We adopt implementation and hyperparameters from open source PyTorch code of a conditional diffusion model trained on normal MNIST (Paszke et al., 2019). This includes an initial learning rate of 1e-4, a batch size of 256, 400 diffusion steps, and a feature dimension of 128. We then generate 100 images of class '0' of each model after training. Finally, to get the approximate standard deviation of each tick and each line in Figure 2, we estimate the distributional variance 1000 times on randomly drawn samples without replacement of all generated images. The normalization constant in the RBF kernel is set to $\gamma = \frac{1}{728}$ since each image has 728 pixels. We also repeated the whole procedure for other classes with similar results.

The results in Figure 3 and 4 are produced in a similar manner. Only difference here is that we train on only 20 training sets for 40 epochs and generate only 20 images per class. We chose these numbers based on the insights gained by the previous simulation experiment. The correlation matrix is the average of all class-wise correlation matrices. The values of the corresponding average covariance matrix of epochs 20 to 40 is given by (all values rounded)

$$\left(\mathrm{Cov}_k\left(P_i, P_j\right)\right)_{i,j=20\ldots40} \approx 10^3 \cdot \begin{pmatrix} 5.1 & 4.9 & 4.9 & 4.9 & 4.9 & 4.9 & 4.8 & 4.9 & 4.9 & 4.8 & 4.9 \\ 4.9 & 5.1 & 4.9 & 4.8 & 4.9 & 4.9 & 4.9 & 4.9 & 4.9 & 4.9 & 4.9 \\ 4.9 & 4.9 & 5.1 & 4.8 & 4.9 & 4.9 & 4.8 & 4.9 & 4.9 & 4.8 & 4.9 \\ 4.9 & 4.8 & 4.8 & 5.1 & 4.9 & 4.8 & 4.9 & 4.9 & 4.9 & 4.9 & 4.9 \\ 4.9 & 4.9 & 4.9 & 4.9 & 5.2 & 4.9 & 4.9 & 4.9 & 4.9 & 4.9 & 4.9 \\ 4.9 & 4.9 & 4.9 & 4.8 & 4.9 & 5.2 & 4.9 & 4.9 & 4.9 & 4.9 & 5.0 \\ 4.8 & 4.9 & 4.8 & 4.9 & 4.9 & 4.9 & 5.1 & 4.9 & 4.9 & 4.9 & 4.9 \\ 4.9 & 4.9 & 4.9 & 4.9 & 4.9 & 4.9 & 4.9 & 5.1 & 4.9 & 4.9 & 4.9 \\ 4.9 & 4.9 & 4.9 & 4.9 & 4.9 & 4.9 & 4.9 & 4.9 & 5.1 & 4.9 & 5.0 \\ 4.8 & 4.9 & 4.8 & 4.9 & 4.9 & 4.9 & 4.9 & 4.9 & 4.9 & 5.2 & 5.0 \\ 4.9 & 4.9 & 4.9 & 4.9 & 4.9 & 5.0 & 4.9 & 4.9 & 5.0 & 5.0 & 5.2 \end{pmatrix}. \tag{17}$$

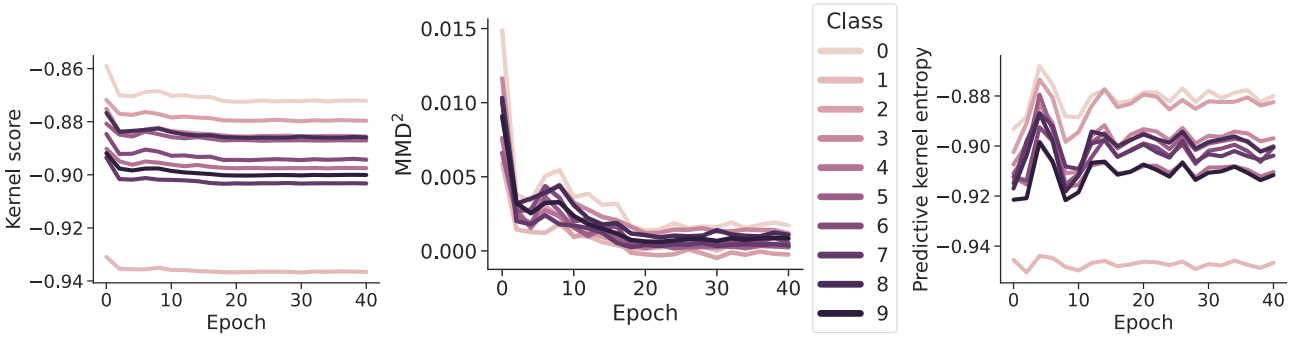


Figure 8. **Left:** Kernel score throughout training. The kernel score cannot be compared meaningfully between classes since each optimum depends on a constant specific to each target class. **Mid:** MMD^2 throughout training. Here, it is easier to compare the errors since the MMD^2 is zero for the optimal prediction. **Right:** The predictive kernel entropy of each class fluctuates throughout training but stays constant upon convergence.

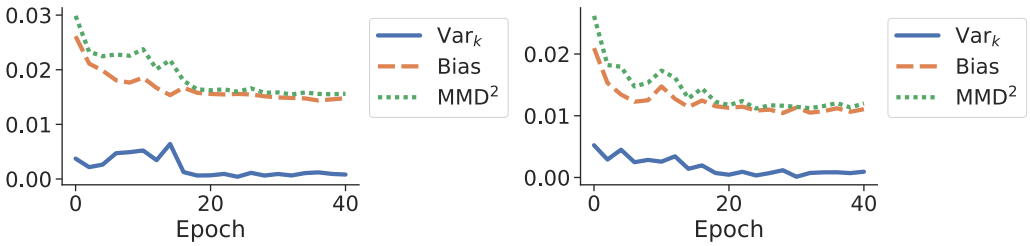


Figure 9. Generalization curves of the digit, which we reduce in frequency during training of DDPMs on InfiMNIST. The reduced digit has only approx. 60 training instances while the other digits have approx. 6000 each. **Left:** We reduce and evaluate exclusively digit '2'. **Right:** We reduce and evaluate exclusively digit '3'. We cannot observe a mode collapse as with digit '0' in Figure 5. The bias is still elevated compared to normal digit frequencies (c.f. Figure 3).

This covariance matrix confirms that $\text{Var}_k(P_i)$ and $\text{Cov}_k(P_i, P_j)$ are approximately equal for all $i, j \in \{20, \ldots, 40\}$ with $i \neq j$. We made use of this assumption in Example 3.4.
All training was done on Nvidia RTX5000 GPUs.

D.1.2. Audio Generation

For the audio experiments, we use an implementation of Glow-TTS given in the TTS library (Eren & The Coqui TTS Team, 2021). The LJSpeech dataset consists of 13.100 instances of text-speech pairs. Each speech is of a single woman reading out loud the corresponding text. We use a random 90% of the data for training and a batch size of 32. On this single training set, we train 12 randomly initialized models for 100 epochs with an initial learning rate of 1e-2. The evaluation happens every 2.000 gradient descent iterations for each model. For a single model in a single evaluation step, we generate 10 waveforms (speeches) for each of 100 test instances. The normalization constant in the Laplacian and RBF kernel is set to $\gamma = \frac{1}{\lambda_{\text{iter,inst}}}$, where $\lambda_{\text{iter,inst}}$ is the longest generated waveform in each evaluation step iter and each test instance inst. The generated audio instances are of various length, so we pad them with zeros to match their length.
Further, all training was done on Nvidia RTX5000 GPUs. But, noteworthy to this experiment, storing the model iterations and generated waveforms required up to 400 GB of hard disk storage.

D.1.3. Natural Language Generation

For the natural language experiments, we adopted the experimental setup of Kuhn et al. (2023). We used their provided code implementations including the hyperparameters. This includes a temperature of $T = 0.5$ for generating the answers used for uncertainty estimation. Similarly, we use 10 answer generations for each prompt for TriviaQA and 20 answer generations for CoQA. All natural language models are pretrained and downloaded from HuggingFace (Wolf et al., 2020). We used a single Nvidia A6000 GPU for the natural language experiments.

D.2. Additional Results

In the following, we give some additional results for image, audio, and language generation.

D.2.1. Image Generation

We start with the image experiments. In Figure 8, we show the corresponding generalization error (kernel score), MMD^2, and predictive kernel entropy values throughout training of the same setup as in Figure 3. As can be seen, MMD^2 is more interpretative for comparing different classes. But, the MMD can also not be evaluated in a lot of practical cases including the audio and natural language settings in this work. Further, the kernel entropy does not show a trend throughout training contrary to the audio setting seen in Figure 6. To test the robustness of our findings with respect to the kernel choice, we repeat all evaluations with the Laplacian kernel with $\gamma = \frac{1}{28^2}$ in Figure 10, and the polynomial kernel $k_{\mathrm{pol}}(x, y) = \left(\frac{\langle x,y\rangle+1}{28^2}\right)^3$ in Figure 11. We also repeat the mode collapse evaluation of Figure 5 with these kernels in Figure 12. As can be seen, all reported findings still hold with only minor differences. This suggests that the choice of kernel is fairly robust towards spotting major trends during model training.

Further, we perform similar experiments with the RBF kernel as in Figure 5, where we observed a mode collapse. In Figure 9, we repeat the evaluation but use digit '2' and digit '3' instead of digit '0'. As can be seen via our bias-variance decomposition, mode collapse does not occur, but the bias term is still larger than in Figure 3.

To confirm that our evaluations also give meaningful results for larger setups, we train the DCGAN architecture (Radford et al., 2015) for image generation on the CelebA dataset (Liu et al., 2015), which consists of colored 64x64 images. We train an ensemble of 12 models to approximate the bias-variance decomposition. For each evaluation step, we sample 20 images per model, and we use the Laplacian kernel. We evaluate the MMD^2 of the individual models and of the ensemble of all models, and, as can be seen in Figure 13, the variance indicates the regions in which the ensemble outperforms the individual models in error. This is predicted by Theorem 3.2 since the ensemble size reduces the variance term in the generalisation error.

This underlines that our decomposition is a useful tool to investigate and analyze the fitting behaviour of generative models.

D.2.2. Audio Generation

We repeat the evaluations of the audio generations in Section 5.2, where we used the Laplacian kernel, with the RBF kernel. It is known that the RBF kernel does not scale well to higher dimensions (Bińkowski et al., 2018). We represent the generated audio instances via vectors of various lengths, often exceeding 100,000 dimensions (c.f. Figure 14).

Consequently, we expect the RBF kernel to behave more erratic than the Laplacian kernel in the main paper. The results are shown in Figure 15.

D.2.3. Natural Language Generation

Next, we continue with the natural language experiments.
In Figure 17, we confirm that the correlation between the predictive kernel entropy and the RougeL (which is supposed to be maximized) has the same sign as in the image experiments in Figure 4.

We also evaluate different embedders. For this, we compare different ones, which have been pretrained on a variety of different training sets and with different embedding dimensions available on HuggingFace. These are e5-small-v2 (384 dimensions; used in the main paper) (Wang et al., 2022), gte-large (1024 dimensions) (Li et al., 2023), all-mpnet-base-v2 (768 dimensions) (Song et al., 2020), as well as all-MiniLM-l6-v2 and all-MiniLM-L12-v2 (both 384 dimensions) (Wang et al., 2020). The models all-mpnet-base-v2, all-MiniLM-l6-v2, and all-MiniLM-L12-v2 included the training set of TriviaQA in their training data, while e5-small-v2 and gte-large did not. Consequently, these three models are an unfair comparison for TriviaQA to other baselines not using its training set. In Figure 18, we compare the ability of each embedder in combination with the RBF kernel to predict the answer accuracy for the same settings as in Figure 7 in the main paper. As we can see for CoQA, all embedders provide approximately similar performance. This indicates that kernel entropy is a robust approach as long as the embedder is meaningful. The results for TriviaQA are similar, but all-MiniLM-L6-v2 and all-MiniLM-L12-v2 perform comparably better. We hypothesis that this is due to them being trained on the training set of TriviaQA. This suggests that we can achieve even better uncertainty estimates by using a task specific training set.

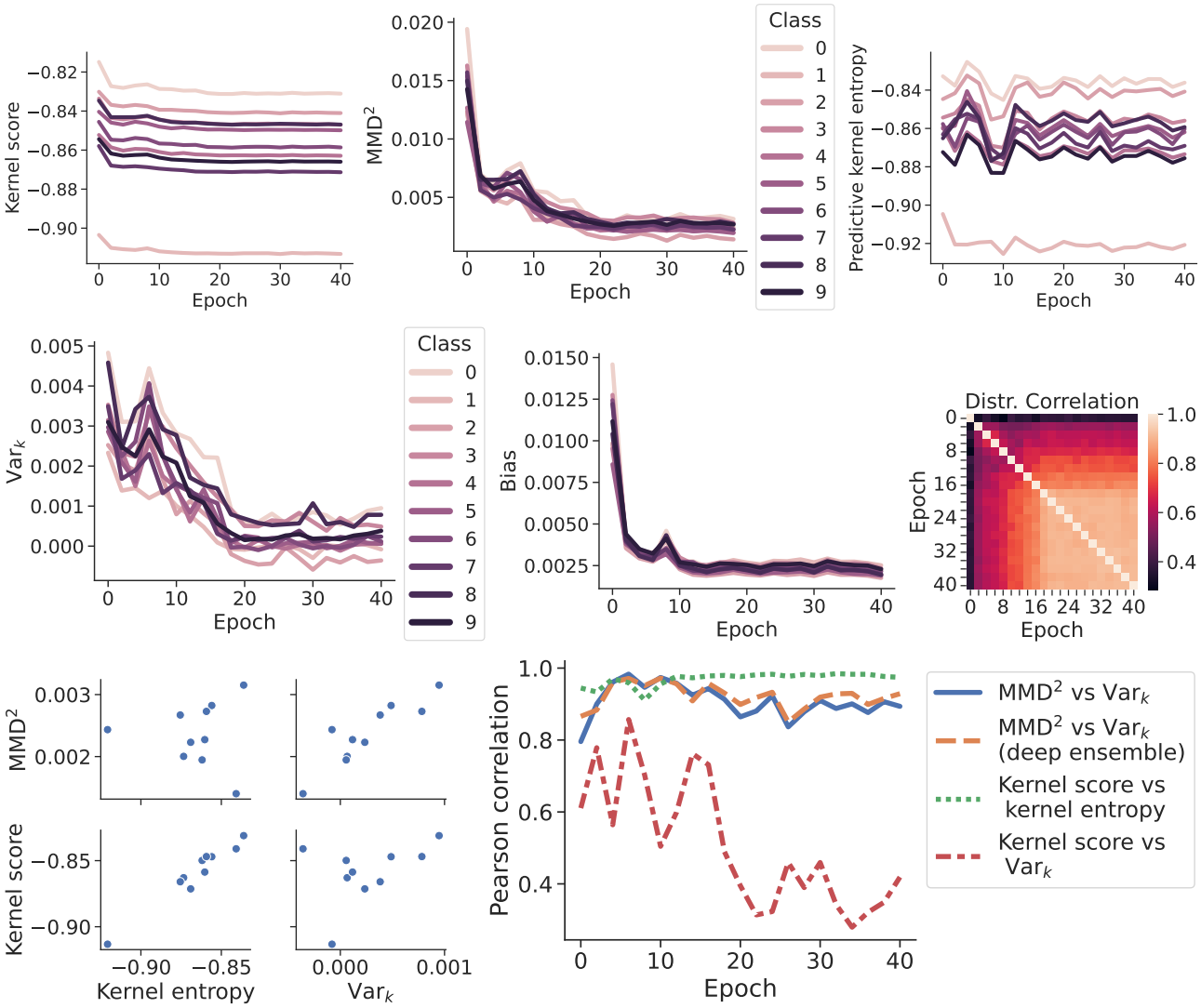
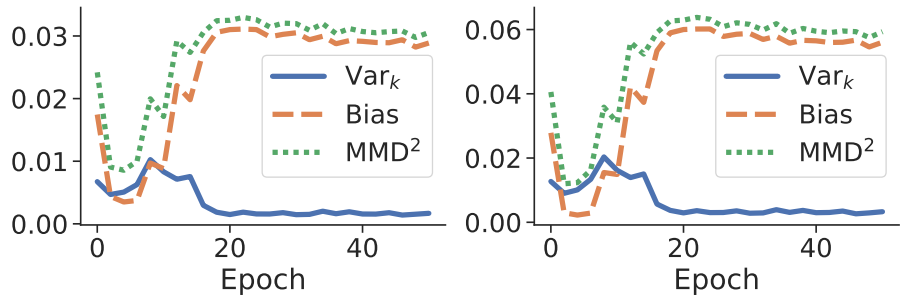


Figure 10. Evaluations in Figures 3, 4, and 8 repeated with Laplacian kernel. The reported findings based on the RBF kernel also hold for the Laplacian kernel and only minor differences can be spotted.

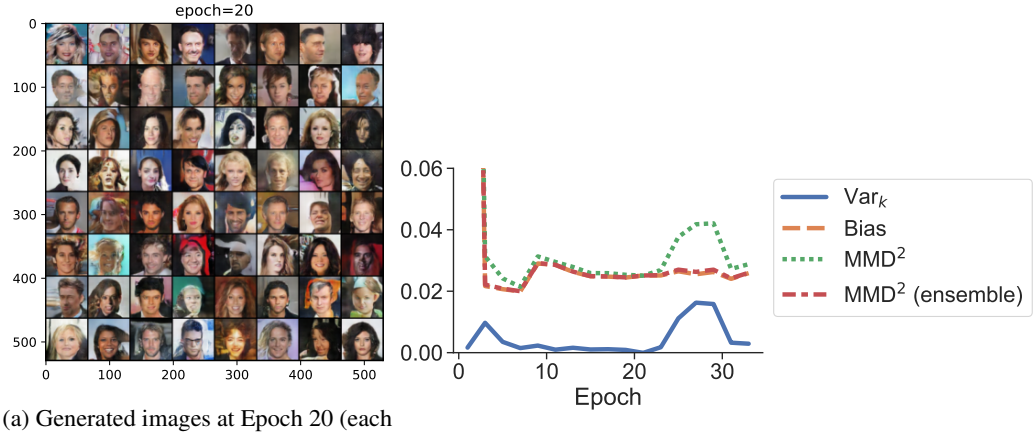
Figure 11. Evaluations in Figures 3, 4, and 8 repeated with polynomial kernel. The reported findings based on the RBF kernel also hold for the polynomial kernel and only minor differences can be spotted. One such difference is that the distributional variance shows a strong negative correlation with the kernel score, which cannot be seen for the other kernels.



(a) Laplacian kernel.

(b) Polynomial kernel.

Figure 12. Evaluation of Figure 5 repeated with other kernels. The mode collapse result is observable independent of kernel choice.



(a) Generated images at Epoch 20 (each row is a different model).

(b) Evaluations throughout training.

Figure 13. We train an ensemble of 12 DCGAN models on the CelebA dataset and evaluate the MMD^2, bias, and variance. The bias and variance are approximated via the ensemble and do not necessarily reflect the ground truth bias and variance. The variance curve indicates the section of improvement in error by using an ensemble instead of a single model as suggested by Theorem 3.2.

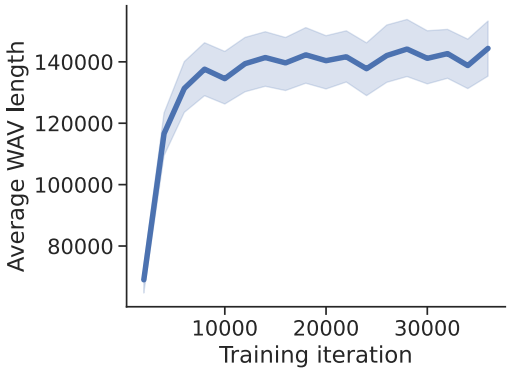


Figure 14. Average WAV length of generated audio instances by Glow-TTS. Initially, the model generates only very short instances, which explains the low initial kernel entropy.

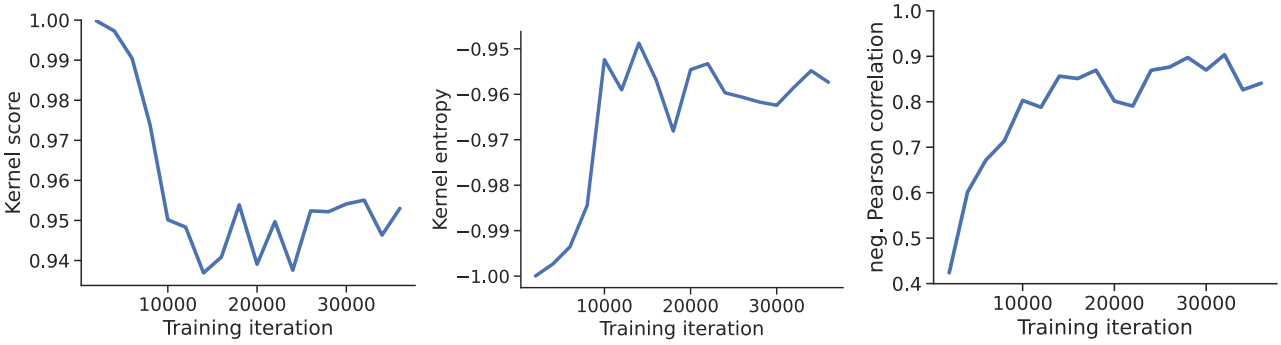


Figure 15. Audio generation results for kernel score, kernel entropy, and negative Pearson correlation between them with the RBF kernel. The trends are similar as in Figure 6 but more erratic and the absolute correlation is slightly less.

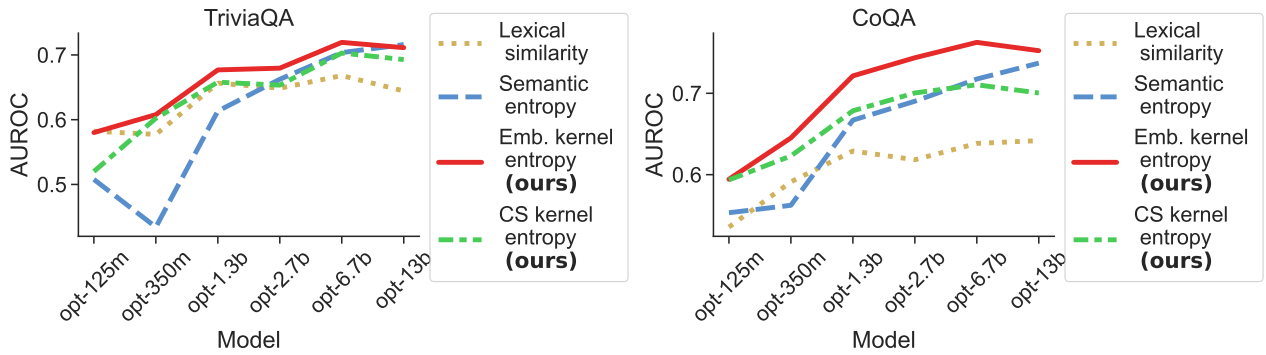


Figure 16. Area-Under-Curve (AUROC) of answer accuracy based on thresholds for best-performing uncertainty measures (embedding-based kernel entropy, semantic entropy) and model-free closed-source uncertainty measures (cs kernel entropy, lexical similarity). Even when no embedding model is available, the kernel entropy is a very strong baseline in predicting the correctness of generated answers across a wide range of differently sized models. In our evaluations, cs kernel entropy strongly outperforms lexical similarity.

Table 1. AUROC between answer correctness and uncertainty estimates based on CS kernel entropy (ours) and lexical similarity. These are the only uncertainty measures in our experiments which do not require a pretrained semantic model and are applicable to any closed-source LLM.

DATASET	MODEL	CS KERNEL ENT.	LEX. SIM.
TRIVIAQA	OPT-125M	0.520	**0.582**
	OPT-350M	**0.602**	0.577
	OPT-1.3B	**0.657**	0.656
	OPT-2.7B	**0.653**	0.648
	OPT-6.7B	**0.702**	0.668
	OPT-13B	**0.692**	0.644
COQA	OPT-125M	**0.593**	0.535
	OPT-350M	**0.623**	0.591
	OPT-1.3B	**0.678**	0.629
	OPT-2.7B	**0.700**	0.618
	OPT-6.7B	**0.710**	0.638
	OPT-13B	**0.700**	0.641

We also compare the impact of different kernel choices in Figure 19. The differences are marginal for cosine similarity, and RBF and polynomial kernel. Contrary, the Laplacian kernel performs often worse. The lack of a difference between RBF or polynomial kernel and cosine similarity is surprising considering that e5-small-v2 has been trained using the cosine similarity (Wang et al., 2022). Our results suggests that the embedder has substantially more influence than the kernel, and that resources should be spent on optimizing the former and not the latter.

Further, we compare different number of generated answers to estimate the kernel entropy in the case of Opt-13b on CoQA in Figure 20. As can be seen, more samples are continuously better. Consequently, we expect to get even better results in Figure 7 for larger numbers of generations.

Last, we also evaluate kernel entropy with a sequence kernel, which does not require an embedding model. This makes kernel entropy as an uncertainty estimate applicable to scenarios, where no embedding model exists. We choose the contiguous subsequence (cs) kernel defined by

$$k_{\text{CS}}(x, y) = \frac{k_t(x, y)}{\sqrt{k_t(x, x)}\sqrt{k_t(y, y)}} \tag{18}$$

for a hyperparameter $t \in \mathbb{N}$ with

$$k_t\left(x_{(1:l)}, y_{(1:l')}\right) = \sum_{i=1}^{l-t+1} \sum_{j=1}^{l'-t+1} \mathbf{1}_{x_{(i:i+t-1)} = y_{(j:j+t-1)}} \tag{19}$$

where $x_{(i:i+t-1)} = (x_i, x_{i+1}, \ldots, x_{i+t-1})^\intercal$ and $\mathbf{1}_{x=y} = 1$ if $x = y$ else 0 (Baum et al., 2023). The kernel k_t is p.s.d. (c.f.

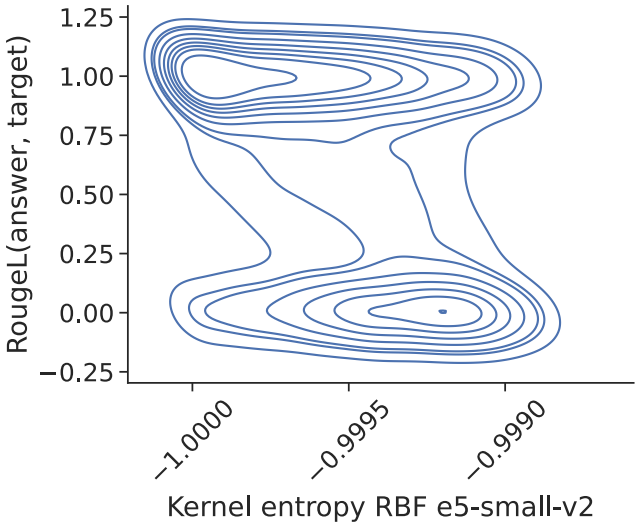


Figure 17. Kernel density estimation of kernel entropy and RougeL between answer and target for opt-1.3b model on CoQA. A large RougeL corresponds to a high accuracy and a low error. Consequently, low kernel entropy indicates a high likelihood of answer correctness.

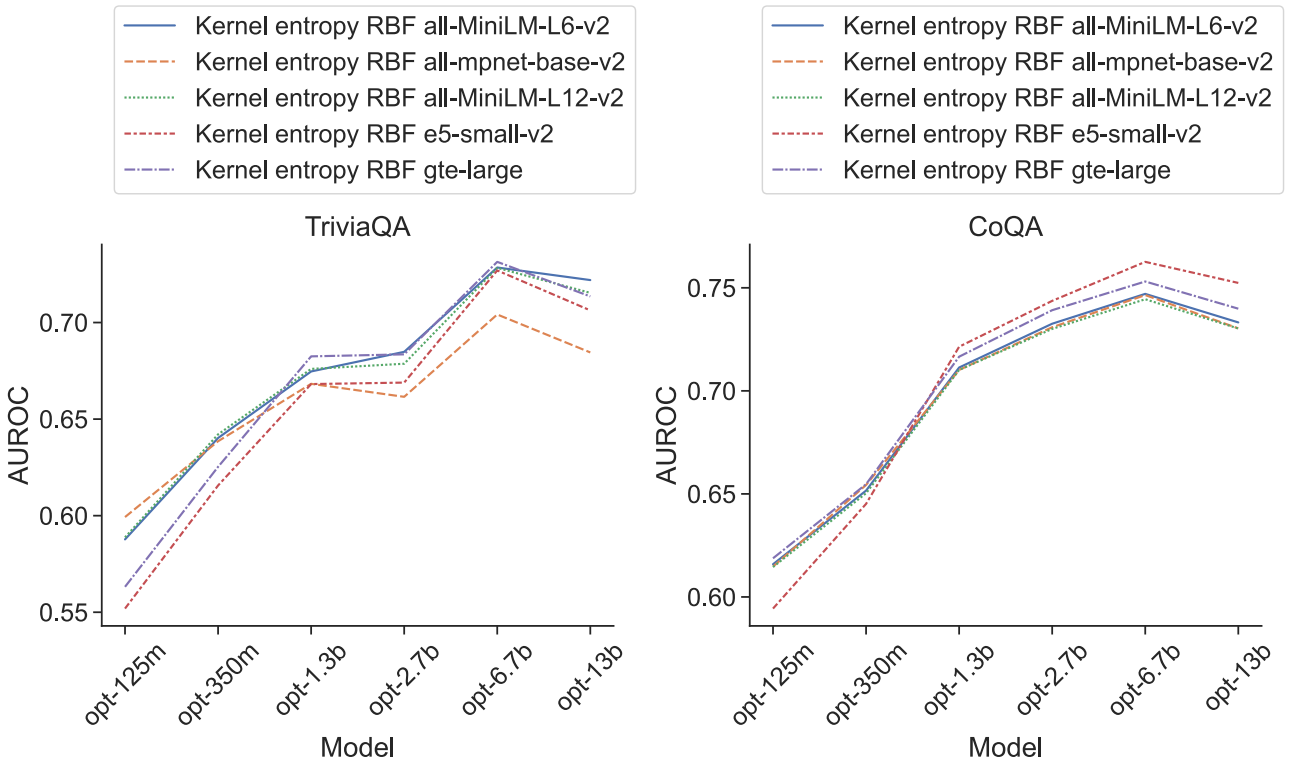


Figure 18. Comparing uncertainty estimates of kernel entropy for different embedders and RBF kernel. In both cases, the differences are not substantial. But, all-MiniLM-L6-v2 and all-MiniLM-L12-v2 perform comparably better on TriviaQA, likely due to them being trained on the TriviaQA training set.

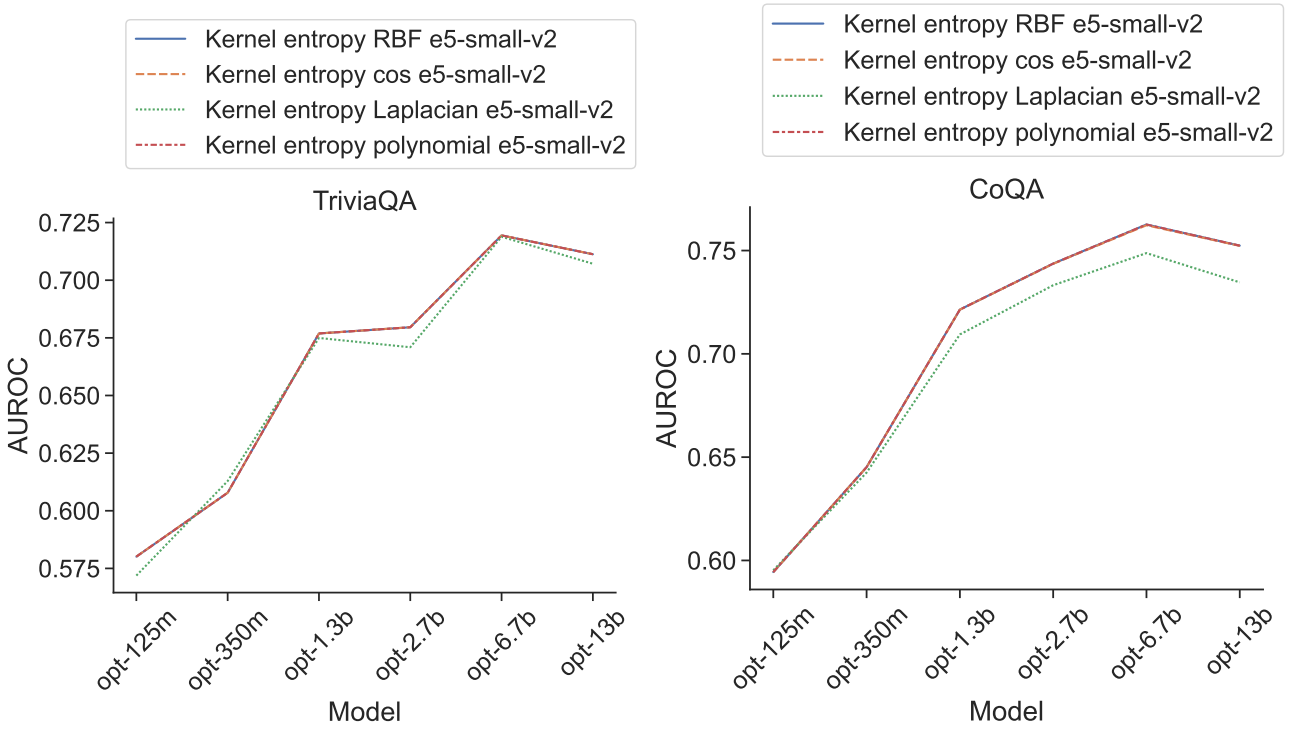


Figure 19. Comparing uncertainty estimates of kernel entropy for cosine similarity and RBF, polynomial, and Laplacian kernel. Even though the embedder e5-small-v2 is trained via cosine similarity, the performance differences are marginal.

Király & Oberhauser (2019)) and counts the number of contiguous subsequences of length t, which appear in x and y. The p.s.d. kernel k_{CS} is the normalised version bounded between 0 and 1. Since our evaluations do not include a validation set, we pick $t = 2$ based on (Baum et al., 2023). In general, better AUROC performance may be achieved by optimizing kernel hyperparameters via a validation set. In Figure 16 and Table 1 we compare the AUROC performance of the cs kernel with other uncertainty estimates from the main paper. As can be seen, the cs kernel entropy is a competitive baseline and strongly outperforms lexical similarity. We specifically highlight the comparison with the lexical similarity, since it is the only other approach within our evaluations which is applicable to any setting with generated sequences. This includes the closed-source setting and settings, where the model has no understanding of natural language (the baseline p(True) requires Q&A queries, which strongly depend on the capabilities of the underlying model).

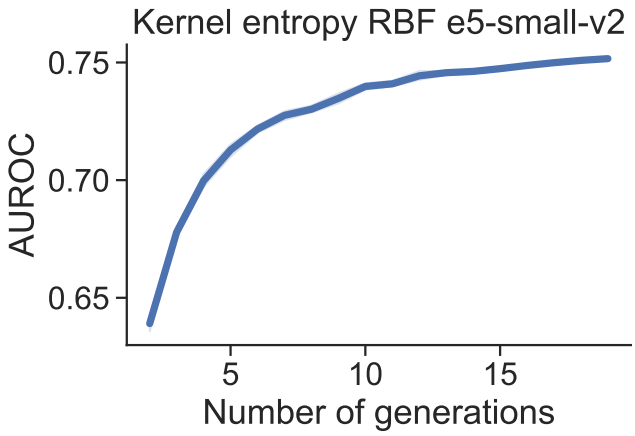


Figure 20. Number of generated answers to estimate kernel entropy compared to the AUROC for Opt-13b on CoQA. More generations monotonically improve kernel entropy as an uncertainty estimate.

E. Missing Proofs

In this section, we give all missing proofs for Theorem 3.2 and for the statements in Section 4. We first start with the proof for the bias-variance-covariance decomposition in Section E.1. Then in Section E.2, we solve the expectation and the variance of the distributional variance estimator proposed in Equation (9). Last, we do the same for the distributional covariance estimator of Equation (10) in Section E.3.

E.1. Bias-Variance-Covariance Decomposition

In Theorem 3.2, the covariance decomposition is introduced after stating the more simpler bias-variance decomposition. Here, we prove both in one go. Assume we have target $Y \sim Q \in \mathscr{P}$ and ensemble prediction $\hat{P}^{(n)} := \frac{1}{n}\sum_{i=1}^{n} \hat{P}_i$ of $n \in \mathbb{N}$ identically distributed predictions $\hat{P}_1, \ldots, \hat{P}_n$ with outcomes in $\mathscr{P}$. Further, assume that all expectations in the following are finite. We will use $\hat{P} := \hat{P}_1$ and $\hat{P}' := \hat{P}_2$. Then, the decomposition can be constructed via

$$
\begin{aligned}
&\mathbb{E}\left[-S_k\left(\hat{P}^{(n)}, Y\right)\right] \\
&= \mathbb{E}\left[\left\|\hat{P}^{(n)}\right\|_k^2 - 2\left\langle \hat{P}^{(n)} \,\middle|\, k \,\middle|\, \delta_Y \right\rangle\right] \\
&\overset{\text{(i)}}{=} \mathbb{E}\left[\left\|\hat{P}^{(n)}\right\|_k^2 - 2\left\langle \hat{P}^{(n)} \,\middle|\, k \,\middle|\, Q \right\rangle\right] \\
&= \mathbb{E}\left[\left\|\hat{P}^{(n)}\right\|_k^2\right] - 2\left\langle \mathbb{E}\left[\hat{P}^{(n)}\right] \,\middle|\, k \,\middle|\, Q \right\rangle \\
&= \mathbb{E}\left[\left\|\hat{P}^{(n)}\right\|_k^2\right] - 2\left\langle \mathbb{E}\left[\hat{P}^{(n)}\right] \,\middle|\, k \,\middle|\, Q \right\rangle + 2\left\|\mathbb{E}\left[\hat{P}^{(n)}\right]\right\|_k^2 - 2\mathbb{E}\left[\left\langle \mathbb{E}\left[\hat{P}^{(n)}\right] \,\middle|\, k \,\middle|\, \hat{P}^{(n)} \right\rangle\right] \\
&= \mathbb{E}\left[\left\|\hat{P}^{(n)} - \mathbb{E}\left[\hat{P}^{(n)}\right]\right\|_k^2\right] - 2\left\langle \mathbb{E}\left[\hat{P}^{(n)}\right] \,\middle|\, k \,\middle|\, Q \right\rangle + \left\|\mathbb{E}\left[\hat{P}^{(n)}\right]\right\|_k^2 \\
&= \mathbb{E}\left[\left\|\hat{P}^{(n)} - \mathbb{E}\left[\hat{P}^{(n)}\right]\right\|_k^2\right] - 2\left\langle \mathbb{E}\left[\hat{P}^{(n)}\right] \,\middle|\, k \,\middle|\, Q \right\rangle + \left\|\mathbb{E}\left[\hat{P}^{(n)}\right]\right\|_k^2 + \|Q\|_k^2 - \|Q\|_k^2 \\
&= -\|Q\|_k^2 + \left\|\mathbb{E}\left[\hat{P}^{(n)}\right] - Q\right\|_k^2 + \mathbb{E}\left[\left\|\hat{P}^{(n)} - \mathbb{E}\left[\hat{P}^{(n)}\right]\right\|_k^2\right] \\
&\overset{\text{(ii)}}{=} -\|Q\|_k^2 + \left\|\mathbb{E}\left[\hat{P}\right] - Q\right\|_k^2 + \mathbb{E}\left[\left\|\hat{P}^{(n)} - \mathbb{E}\left[\hat{P}\right]\right\|_k^2\right] \\
&= -\|Q\|_k^2 + \left\|\mathbb{E}\left[\hat{P}\right] - Q\right\|_k^2 + \mathbb{E}\left[\left\|\frac{1}{n}\sum_{i=1}^n \hat{P}_i - \mathbb{E}\left[\hat{P}\right]\right\|_k^2\right] \\
&= -\|Q\|_k^2 + \left\|\mathbb{E}\left[\hat{P}\right] - Q\right\|_k^2 + \frac{1}{n^2}\sum_{i,j=1}^n \mathbb{E}\left[\left\langle \hat{P}_i - \mathbb{E}\left[\hat{P}\right] \,\middle|\, k \,\middle|\, \hat{P}_j - \mathbb{E}\left[\hat{P}\right] \right\rangle\right] \\
&= -\|Q\|_k^2 + \left\|\mathbb{E}\left[\hat{P}\right] - Q\right\|_k^2 + \frac{1}{n^2}\sum_{i,j=1}^n \mathrm{Cov}_k\left(\hat{P}_i, \hat{P}_j\right) \\
&= -\|Q\|_k^2 + \frac{1}{n^2}\sum_{i=1}^n \mathrm{Var}_k\left(\hat{P}_i\right) + \frac{1}{n^2}\sum_{\substack{i,j=1 \\ i\neq j}}^n \mathrm{Cov}_k\left(\hat{P}_i, \hat{P}_j\right) + \left\|\mathbb{E}\left[\hat{P}\right] - Q\right\|_k^2 \\
&\overset{\text{(ii)}}{=} -\|Q\|_k^2 + \left\|\mathbb{E}\left[\hat{P}\right] - Q\right\|_k^2 + \frac{1}{n^2} n\, \mathrm{Var}_k\left(\hat{P}\right) + \frac{1}{n^2} n\left(n-1\right) \mathrm{Cov}_k\left(\hat{P}, \hat{P}'\right) \\
&= \underbrace{-\|Q\|_k^2}_{\text{Noise}} + \underbrace{\left\|\mathbb{E}\left[\hat{P}\right] - Q\right\|_k^2}_{\text{Bias}} + \underbrace{\frac{1}{n}\mathrm{Var}_k\left(\hat{P}\right)}_{\text{Variance}} + \underbrace{\frac{n-1}{n}\mathrm{Cov}_k\left(\hat{P}, \hat{P}'\right)}_{\text{Covariance}}
\end{aligned}
\tag{20}
$$

(i) Y and $\hat{P}^{(n)}$ independently distributed
(ii) $\hat{P}_1, \ldots, \hat{P}_n$ identically distributed

E.2. Distributional Variance Estimator

In general, the notation of Eaton (1981) allows us to write for random variables P and Q with outcomes in a distribution space, and for independent $X \sim P$ and $Y \sim Q$ that

$$
\mathbb{E}\left[k\left(X, Y\right) \mid P, Q\right] = \int k\left(x, y\right) \mathrm{d}P \otimes Q\left(x, y\right) = \int\int k\left(x, y\right) \mathrm{d}P\left(x\right) \mathrm{d}Q\left(y\right) = \left\langle P \,|\, k \,|\, Q \right\rangle, \tag{21}
$$

where we used Tonelli's Theorem to split the integral.

Assume the variables are defined as in Section 4. As a reminder: $P, P_1, \ldots, P_n$ i.i.d. and $X_{i1}, \ldots, X_{im} \sim P_i$ i.i.d. for a given $i = 1, \ldots, n$. Note we have for $j \neq t$ that X_{ij} and X_{it} are independent given P_i for all $i = 1, \ldots, n$. From this follows

$$\begin{aligned}
\mathbb{E}\left[k\left(X_{ij}, X_{it}\right)\right] &= \mathbb{E}\left[\mathbb{E}\left[k\left(X_{ij}, X_{it}\right) \mid P_i\right]\right] \\
&\overset{\text{iid}}{=} \mathbb{E}\left[\mathbb{E}\left[k\left(X_{i1}, X_{i2}\right) \mid P_i\right]\right] \\
&\overset{\text{iid}}{=} \mathbb{E}\left[\langle P_i \,|\, k \,|\, P_i \rangle\right] \\
&\overset{\text{iid}}{=} \mathbb{E}\left[\langle P \,|\, k \,|\, P \rangle\right]
\end{aligned} \tag{22}$$

as well as for $i \neq s$

$$\begin{aligned}
\mathbb{E}\left[k\left(X_{ij}, X_{st}\right)\right] &= \mathbb{E}\left[\mathbb{E}\left[k\left(X_{ij}, X_{st}\right) \mid P_i, P_s\right]\right] \\
&\overset{\text{iid}}{=} \mathbb{E}\left[\mathbb{E}\left[k\left(X_{i1}, X_{s1}\right) \mid P_i, P_s\right]\right] \\
&\overset{\text{iid}}{=} \mathbb{E}\left[\langle P_i \,|\, k \,|\, P_s \rangle\right] \\
&\overset{\text{iid}}{=} \langle \mathbb{E}\left[P\right] \,|\, k \,|\, \mathbb{E}\left[P\right] \rangle .
\end{aligned} \tag{23}$$

E.2.1. Expectation of the Estimator

Using these equations gives

$$\begin{aligned}
&\mathbb{E}\left[\widehat{\mathrm{Var}}_k^{(n,m)}\right] \\
&= \mathbb{E}\left[\frac{1}{nm}\sum_{i=1}^{n}\sum_{j=1}^{m}\left(\frac{1}{m-1}\sum_{\substack{t=1\\t\neq j}}^{m} k\left(X_{ij}, X_{it}\right) - \frac{1}{(n-1)\,m}\sum_{\substack{s=1\\s\neq i}}^{n}\sum_{t=1}^{m} k\left(X_{ij}, X_{st}\right)\right)\right] \\
&= \frac{1}{nm}\sum_{i=1}^{n}\sum_{j=1}^{m}\left(\frac{1}{m-1}\sum_{\substack{t=1\\t\neq j}}^{m} \mathbb{E}\left[k\left(X_{ij}, X_{it}\right)\right] - \frac{1}{(n-1)\,m}\sum_{\substack{s=1\\s\neq i}}^{n}\sum_{t=1}^{m} \mathbb{E}\left[k\left(X_{ij}, X_{st}\right)\right]\right) \\
&\overset{\text{Eq. (22) \& (23)}}{=} \frac{1}{nm}\sum_{i=1}^{n}\sum_{j=1}^{m}\left(\frac{1}{m-1}\sum_{\substack{t=1\\t\neq j}}^{m} \mathbb{E}\left[\langle P \,|\, k \,|\, P \rangle\right] - \frac{1}{(n-1)\,m}\sum_{\substack{s=1\\s\neq i}}^{n}\sum_{t=1}^{m} \langle \mathbb{E}\left[P\right] \,|\, k \,|\, \mathbb{E}\left[P\right] \rangle\right) \\
&= \mathbb{E}\left[\langle P \,|\, k \,|\, P \rangle\right] - \langle \mathbb{E}\left[P\right] \,|\, k \,|\, \mathbb{E}\left[P\right] \rangle \\
&= \mathbb{E}\left[\langle P - \mathbb{E}\left[P\right] \,|\, k \,|\, P - \mathbb{E}\left[P\right] \rangle\right] \\
&= \mathrm{Var}_k\left(P\right).
\end{aligned} \tag{24}$$

E.2.2. Variance of the Estimator

Note that for a U-statistic $\hat{U}_n = \frac{1}{n(n-1)}\sum_{i=1}^{n}\sum_{\substack{j=1\\j\neq i}}^{n} h\left(X_i, X_j\right)$ based on i.i.d. samples $X_1, \ldots, X_n$ and symmetric function h, the estimator variance is given by

$$\mathbb{V}\left(\hat{U}_n\right) = \frac{2}{n\,(n-1)}\mathbb{V}\left(h\left(X_1, X_2\right)\right) + \frac{4\,(n-2)}{n\,(n-1)}\mathbb{V}\left(\mathbb{E}\left[h\left(X_1, X_2\right) \mid X_2\right]\right) \tag{25}$$

(Shao, 2003).

We will use the law of total variance (TV) several times to create independence between the summands, since for dependent random variables X and Y with scalars a, b we have $\mathbb{V}(aX + bY) = a^2\mathbb{V}(X) + b^2\mathbb{V}(Y) + 2ab\,\mathrm{Cov}(X, Y)$ (Shao, 2003).

From this also follows that

$$
\begin{aligned}
&\mathbb{V}\left(\widehat{\mathrm{Var}}_k^{(n,m)}\right) \\
&= \mathbb{V}\left(\frac{1}{nm}\sum_{i=1}^{n}\sum_{j=1}^{m}\left(\frac{1}{m-1}\sum_{\substack{t=1\\t\neq j}}^{m} k\left(X_{ij}, X_{it}\right) - \frac{1}{(n-1)\,m}\sum_{\substack{s=1\\s\neq i}}^{n}\sum_{t=1}^{m} k\left(X_{ij}, X_{st}\right)\right)\right) \\
&= \mathbb{V}\left(\frac{1}{nm\,(m-1)}\sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{\substack{t=1\\t\neq j}}^{m} k\left(X_{ij}, X_{it}\right)\right) + \mathbb{V}\left(\frac{1}{n\,(n-1)\,m^2}\sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{\substack{s=1\\s\neq i}}^{n}\sum_{t=1}^{m} k\left(X_{ij}, X_{st}\right)\right) \\
&\quad - 2\,\mathrm{Cov}\left(\frac{1}{nm\,(m-1)}\sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{\substack{t=1\\t\neq j}}^{m} k\left(X_{ij}, X_{it}\right), \frac{1}{n\,(n-1)\,m^2}\sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{\substack{s=1\\s\neq i}}^{n}\sum_{t=1}^{m} k\left(X_{ij}, X_{st}\right)\right)
\end{aligned}
\tag{26}
$$

We will analyse each term successively and then combine the results further down in (35).

$$
\begin{aligned}
&\mathbb{V}\left(\frac{1}{nm(m-1)}\sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{\substack{t=1\\t\neq j}}^{m}k\left(X_{ij},X_{it}\right)\right)\\
&\overset{\text{TV}}{=}\mathbb{V}\left(\mathbb{E}\left[\frac{1}{nm(m-1)}\sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{\substack{t=1\\t\neq j}}^{m}k\left(X_{ij},X_{it}\right)\mid P_1\dots P_n\right]\right)\\
&\quad+\mathbb{E}\left[\mathbb{V}\left(\frac{1}{nm(m-1)}\sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{\substack{t=1\\t\neq j}}^{m}k\left(X_{ij},X_{it}\right)\mid P_1\dots P_n\right)\right]\\
&\overset{\text{iid}}{=}\mathbb{V}\left(\frac{1}{nm(m-1)}\sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{\substack{t=1\\t\neq j}}^{m}\mathbb{E}\left[k\left(X_{ij},X_{it}\right)\mid P_i\right]\right)\\
&\quad+\mathbb{E}\left[\frac{1}{n^2}\sum_{i=1}^{n}\mathbb{V}\left(\frac{1}{m(m-1)}\sum_{j=1}^{m}\sum_{\substack{t=1\\t\neq j}}^{m}k\left(X_{ij},X_{it}\right)\mid P_i\right)\right]\\
&\overset{\text{iid}}{=}\mathbb{V}\left(\frac{1}{n}\sum_{i=1}^{n}\mathbb{E}\left[k\left(X_{i1},X_{i2}\right)\mid P_i\right]\right)+\frac{1}{n}\mathbb{E}\left[\mathbb{V}\left(\frac{1}{m(m-1)}\sum_{j=1}^{m}\sum_{\substack{t=1\\t\neq j}}^{m}k\left(X_{1j},X_{1t}\right)\mid P_1\right)\right]\\
&\overset{\text{iid}}{=}\frac{1}{n}\mathbb{V}\left(\langle P\,|\,k\,|\,P\rangle\right)+\frac{1}{n}\mathbb{E}\left[\mathbb{V}\left(\frac{1}{m(m-1)}\sum_{j=1}^{m}\sum_{\substack{t=1\\t\neq j}}^{m}k\left(X_{1j},X_{1t}\right)\mid P_1\right)\right]\\
&\overset{\text{(i)}}{=}\frac{1}{n}\mathbb{V}\left(\langle P\,|\,k\,|\,P\rangle\right)+\frac{4(m-2)}{nm(m-1)}\zeta_1+\frac{2}{nm(m-1)}\zeta_2
\end{aligned}
\tag{27}
$$

(i) with $\zeta_1=\mathbb{E}\left[\mathbb{V}\left(\mathbb{E}\left[k\left(X_{11},X_{12}\right)\mid X_{12},P_1\right]\mid P_1\right)\right]$ and $\zeta_2=\mathbb{E}\left[\mathbb{V}\left(k\left(X_{11},X_{12}\right)\mid P_1\right)\right]$ based on (25).

$$
\begin{aligned}
&\mathbb{V}\left(\frac{1}{n(n-1)m^2}\sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{\substack{s=1\\s\neq i}}^{n}\sum_{t=1}^{m}k\left(X_{ij},X_{st}\right)\right)\\
&\overset{\text{TV}}{=}\underbrace{\mathbb{V}\left(\mathbb{E}\left[\frac{1}{n(n-1)m^2}\sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{\substack{s=1\\s\neq i}}^{n}\sum_{t=1}^{m}k\left(X_{ij},X_{st}\right)\mid P_1\dots P_n\right]\right)}_{\text{(I)}:=}\\
&\quad+\underbrace{\mathbb{E}\left[\mathbb{V}\left(\frac{1}{n(n-1)m^2}\sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{\substack{s=1\\s\neq i}}^{n}\sum_{t=1}^{m}k\left(X_{ij},X_{st}\right)\mid P_1\dots P_n\right)\right]}_{\text{(II)}:=}.
\end{aligned}
\tag{28}
$$

Due to the length of the expression, we first solve (I) and then (II).

$$
\begin{aligned}
\text{(I)} &\overset{\text{iid}}{=} \mathbb{V}\left(\frac{1}{n\left(n-1\right)m^2}\sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{\substack{s=1\\s\neq i}}^{n}\sum_{t=1}^{m}\mathbb{E}\left[k\left(X_{ij},X_{st}\right)\mid P_i,P_s\right]\right)\\
&\overset{\text{iid}}{=} \mathbb{V}\left(\frac{1}{n\left(n-1\right)}\sum_{i=1}^{n}\sum_{\substack{s=1\\s\neq i}}^{n}\left\langle P_i\,|\,k\,|\,P_s\right\rangle\right)\\
&\overset{\text{(i)}}{=} \frac{4\left(n-2\right)}{n\left(n-1\right)}\zeta_3+\frac{2}{n\left(n-1\right)}\zeta_4
\end{aligned}
\tag{29}
$$

(i) with $\zeta_3 = \mathbb{V}\left(\mathbb{E}\left[\left\langle P_1\,|\,k\,|\,P_2\right\rangle \mid P_1\right]\right)$ and $\zeta_4 = \mathbb{V}\left(\left\langle P_1\,|\,k\,|\,P_2\right\rangle\right)$ based on (25).

For the next term note that $\frac{1}{m^2}\sum_{j=1}^{m}\sum_{t=1}^{m}k\left(X_{ij},X_{st}\right)$ and $\frac{1}{m^2}\sum_{j=1}^{m}\sum_{t=1}^{m}k\left(X_{aj},X_{bt}\right)$ are independent given P_i, P_s, P_a, P_b for $i\neq s\neq a\neq b$ from which follows

$$
\operatorname{Cov}\left(\frac{1}{m^2}\sum_{j=1}^{m}\sum_{t=1}^{m}k\left(X_{ij},X_{st}\right),\frac{1}{m^2}\sum_{c=1}^{m}\sum_{d=1}^{m}k\left(X_{ac},X_{bd}\right)\right)=0. \tag{30}
$$

Consequently, we have

$$
\begin{aligned}
\text{(II)} = \mathbb{E}&\left[\frac{1}{n^2\left(n-1\right)^2}\sum_{i=1}^{n}\sum_{\substack{s=1\\s\neq i}}^{n}\mathbb{V}\left(\frac{1}{m^2}\sum_{j=1}^{m}\sum_{t=1}^{m}k\left(X_{ij},X_{st}\right)\mid P_i,P_s\right)\right]\\
&+\mathbb{E}\left[\frac{1}{n^2\left(n-1\right)^2}\sum_{i=1}^{n}\sum_{\substack{s=1\\s\neq i}}^{n}\sum_{\substack{b=1\\b\neq i\\b\neq s}}^{n}\operatorname{Cov}\left(\frac{1}{m^2}\sum_{j=1}^{m}\sum_{t=1}^{m}k\left(X_{ij},X_{st}\right),\frac{1}{m^2}\sum_{c=1}^{m}\sum_{d=1}^{m}k\left(X_{ic},X_{bd}\right)\mid P_i,P_s,P_b\right)\right]\\
&+\mathbb{E}\left[\frac{1}{n^2\left(n-1\right)^2}\sum_{i=1}^{n}\sum_{\substack{s=1\\s\neq i}}^{n}\sum_{\substack{a=1\\a\neq s\\a\neq i}}^{n}\operatorname{Cov}\left(\frac{1}{m^2}\sum_{j=1}^{m}\sum_{t=1}^{m}k\left(X_{ij},X_{st}\right),\frac{1}{m^2}\sum_{c=1}^{m}\sum_{d=1}^{m}k\left(X_{ac},X_{sd}\right)\mid P_i,P_s,P_a\right)\right]\\
\overset{\text{sym}}{=}&\underbrace{\mathbb{E}\left[\frac{1}{n^2\left(n-1\right)^2}\sum_{i=1}^{n}\sum_{\substack{s=1\\s\neq i}}^{n}\mathbb{V}\left(\frac{1}{m^2}\sum_{c=1}^{m}\sum_{d=1}^{m}k\left(X_{ic},X_{sd}\right)\mid P_i,P_s\right)\right]}_{\text{(IIa)}:=}\\
&+\underbrace{\mathbb{E}\left[\frac{2}{n^2\left(n-1\right)^2}\sum_{i=1}^{n}\sum_{\substack{s=1\\s\neq i}}^{n}\sum_{\substack{b=1\\b\neq i\\b\neq s}}^{n}\operatorname{Cov}\left(\frac{1}{m^2}\sum_{j=1}^{m}\sum_{t=1}^{m}k\left(X_{ij},X_{st}\right),\frac{1}{m^2}\sum_{c=1}^{m}\sum_{d=1}^{m}k\left(X_{ic},X_{bd}\right)\mid P_i,P_s,P_b\right)\right]}_{\text{(IIb)}:=}.
\end{aligned}
\tag{31}
$$

Due to the length of the expressions, we again first look at (IIa) and then (IIb).

$$
\begin{aligned}
\text{(IIa)} &\overset{\text{iid}}{=} \mathbb{E}\left[\frac{1}{n^2(n-1)^2}\sum_{i=1}^{n}\sum_{\substack{s=1\\ s\neq i}}^{n}\mathbb{V}\left(\frac{1}{m^2}\sum_{j=1}^{m}\sum_{t=1}^{m}k\left(X_{ij},X_{st}\right)\mid P_i,P_s\right)\right]\\
&\overset{\text{iid}}{=} \frac{1}{n(n-1)}\mathbb{E}\left[\mathbb{V}\left(\frac{1}{m^2}\sum_{j=1}^{m}\sum_{t=1}^{m}k\left(X_{1j},X_{2t}\right)\mid P_1,P_2\right)\right]\\
&= \frac{1}{n(n-1)}\mathbb{E}\left[\operatorname{Cov}\left(\frac{1}{m^2}\sum_{j=1}^{m}\sum_{t=1}^{m}k\left(X_{1j},X_{2t}\right),\frac{1}{m^2}\sum_{i=1}^{m}\sum_{s=1}^{m}k\left(X_{1i},X_{2s}\right)\mid P_1,P_2\right)\right]\\
&= \frac{1}{n(n-1)m^4}\sum_{j=1}^{m}\sum_{t=1}^{m}\sum_{i=1}^{m}\sum_{s=1}^{m}\mathbb{E}\left[\operatorname{Cov}\left(k\left(X_{1j},X_{2t}\right),k\left(X_{1i},X_{2s}\right)\mid P_1,P_2\right)\right]\\
&\overset{\text{iid}}{=} \frac{1}{n(n-1)m^4}\sum_{j=1}^{m}\sum_{t=1}^{m}\mathbb{E}\left[\operatorname{Cov}\left(k\left(X_{1j},X_{2t}\right),k\left(X_{1j},X_{2t}\right)\mid P_1,P_2\right)\right]\\
&\quad+\frac{1}{n(n-1)m^4}\sum_{j=1}^{m}\sum_{t=1}^{m}\sum_{\substack{i=1\\ i\neq j}}^{m}\mathbb{E}\left[\operatorname{Cov}\left(k\left(X_{1j},X_{2t}\right),k\left(X_{1i},X_{2t}\right)\mid P_1,P_2\right)\right]\\
&\quad+\frac{1}{n(n-1)m^4}\sum_{j=1}^{m}\sum_{t=1}^{m}\sum_{\substack{s=1\\ s\neq t}}^{m}\mathbb{E}\left[\operatorname{Cov}\left(k\left(X_{1j},X_{2t}\right),k\left(X_{1j},X_{2s}\right)\mid P_1,P_2\right)\right]\\
&\overset{\text{iid}}{=} \frac{1}{n(n-1)m^2}\mathbb{E}\left[\operatorname{Cov}\left(k\left(X_{11},X_{21}\right),k\left(X_{11},X_{21}\right)\mid P_1,P_2\right)\right]\\
&\quad+\frac{m-1}{n(n-1)m^2}\mathbb{E}\left[\operatorname{Cov}\left(k\left(X_{11},X_{21}\right),k\left(X_{12},X_{21}\right)\mid P_1,P_2\right)\right]\\
&\quad+\frac{m-1}{n(n-1)m^2}\mathbb{E}\left[\operatorname{Cov}\left(k\left(X_{11},X_{21}\right),k\left(X_{11},X_{22}\right)\mid P_1,P_2\right)\right]\\
&\overset{\text{(i)}}{=} \frac{1}{n(n-1)m^2}\underbrace{\mathbb{E}\left[\mathbb{V}\left(k\left(X_{11},X_{21}\right)\mid P_1,P_2\right)\right]}_{\zeta_6:=}\\
&\quad+\frac{2(m-1)}{n(n-1)m^2}\underbrace{\mathbb{E}\left[\operatorname{Cov}\left(k\left(X_{11},X_{21}\right),k\left(X_{12},X_{21}\right)\mid P_1,P_2\right)\right]}_{\zeta_5:=}
\end{aligned}
\tag{32}
$$

(i) follows from symmetry of k and assumption of identical distributions.

Further, we have $\operatorname{Cov}\left(k\left(X_{ij},X_{st}\right),k\left(X_{ic},X_{bd}\right)\mid P_i,P_s,P_b\right)=0$ whenever $c\neq j$ (since $s\neq b$ and independence assumption), giving

$$
\begin{aligned}
(\mathrm{IIb}) &= \mathbb{E}\left[\frac{2}{n^2(n-1)^2}\sum_{i=1}^{n}\sum_{\substack{s=1\\s\neq i}}^{n}\sum_{\substack{b=1\\b\neq i\\b\neq s}}^{n}\mathrm{Cov}\left(\frac{1}{m^2}\sum_{j=1}^{m}\sum_{t=1}^{m}k\left(X_{ij},X_{st}\right),\frac{1}{m^2}\sum_{c=1}^{m}\sum_{d=1}^{m}k\left(X_{ic},X_{bd}\right)\mid P_i,P_s,P_b\right)\right]\\
&= \mathbb{E}\left[\frac{2}{n^2(n-1)^2m^4}\sum_{i=1}^{n}\sum_{\substack{s=1\\s\neq i}}^{n}\sum_{\substack{b=1\\b\neq i\\b\neq s}}^{n}\sum_{j=1}^{m}\sum_{t=1}^{m}\sum_{d=1}^{m}\mathrm{Cov}\left(k\left(X_{ij},X_{st}\right),k\left(X_{ij},X_{bd}\right)\mid P_i,P_s,P_b\right)\right]\\
&\overset{\text{iid}}{=} \mathbb{E}\left[\frac{2}{n^2(n-1)^2m^4}\sum_{i=1}^{n}\sum_{\substack{s=1\\s\neq i}}^{n}\sum_{\substack{b=1\\b\neq i\\b\neq s}}^{n}\sum_{j=1}^{m}\sum_{t=1}^{m}\sum_{d=1}^{m}\mathrm{Cov}\left(k\left(X_{11},X_{21}\right),k\left(X_{11},X_{31}\right)\mid P_1,P_2,P_3\right)\right]\\
&= \frac{2(n-2)}{n(n-1)m}\underbrace{\mathbb{E}\left[\mathrm{Cov}\left(k\left(X_{11},X_{21}\right),k\left(X_{11},X_{31}\right)\mid P_1,P_2,P_3\right)\right]}_{\zeta_9:=}
\end{aligned}
\tag{33}
$$

The only term left is

$$
\begin{aligned}
&\operatorname{Cov}\left(\frac{1}{nm(m-1)}\sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{\substack{t=1\\t\neq j}}^{m}k\left(X_{ij},X_{it}\right),\frac{1}{n(n-1)m^2}\sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{\substack{s=1\\s\neq i}}^{n}\sum_{t=1}^{m}k\left(X_{ij},X_{st}\right)\right)\\
&=\frac{1}{n^2(n-1)m^3(m-1)}\sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{\substack{t=1\\t\neq j}}^{m}\sum_{o=1}^{n}\sum_{p=1}^{m}\sum_{\substack{s=1\\s\neq o}}^{n}\sum_{r=1}^{m}\operatorname{Cov}\left(k\left(X_{ij},X_{it}\right),k\left(X_{op},X_{sr}\right)\right)\\
&\overset{\text{(i)}}{=}\frac{1}{n^2(n-1)m^3(m-1)}\sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{\substack{t=1\\t\neq j}}^{m}\sum_{p=1}^{m}\sum_{\substack{s=1\\s\neq i}}^{n}\sum_{r=1}^{m}\\
&\qquad\left(\operatorname{Cov}\left(k\left(X_{ij},X_{it}\right),k\left(X_{ip},X_{sr}\right)\right)+\operatorname{Cov}\left(k\left(X_{ij},X_{it}\right),k\left(X_{sp},X_{ir}\right)\right)\right)\\
&\overset{\text{(ii)}}{=}\frac{2}{n^2(n-1)m^3(m-1)}\sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{\substack{t=1\\t\neq j}}^{m}\sum_{p=1}^{m}\sum_{\substack{s=1\\s\neq i}}^{n}\sum_{r=1}^{m}\operatorname{Cov}\left(k\left(X_{ij},X_{it}\right),k\left(X_{ip},X_{sr}\right)\right)\\
&\overset{\text{iid}}{=}\frac{2mn(n-1)}{n^2(n-1)m^3(m-1)}\sum_{j=1}^{m}\sum_{\substack{t=1\\t\neq j}}^{m}\sum_{p=1}^{m}\operatorname{Cov}\left(k\left(X_{1j},X_{1t}\right),k\left(X_{1p},X_{21}\right)\right)\\
&=\frac{2}{nm^2(m-1)}\left(\sum_{j=1}^{m}\sum_{\substack{t=1\\t\neq j}}^{m}\sum_{p\in\{j,t\}}\operatorname{Cov}\left(k\left(X_{1j},X_{1t}\right),k\left(X_{1p},X_{21}\right)\right)+\sum_{j=1}^{m}\sum_{\substack{t=1\\t\neq j}}^{m}\sum_{\substack{p=1\\p\neq j\\p\neq t}}^{m}\operatorname{Cov}\left(k\left(X_{1j},X_{1t}\right),k\left(X_{1p},X_{21}\right)\right)\right)\\
&\overset{\text{(iii)}}{=}\frac{2}{nm^2(m-1)}\left(\underbrace{\sum_{j=1}^{m}\sum_{\substack{t=1\\t\neq j}}^{m}\sum_{p\in\{j,t\}}\operatorname{Cov}\left(k\left(X_{11},X_{12}\right),k\left(X_{11},X_{21}\right)\right)}_{2m(m-1)\text{ summands}}+\underbrace{\sum_{j=1}^{m}\sum_{\substack{t=1\\t\neq j}}^{m}\sum_{\substack{p=1\\p\neq j\\p\neq t}}^{m}\operatorname{Cov}\left(k\left(X_{11},X_{12}\right),k\left(X_{13},X_{21}\right)\right)}_{m(m-1)(m-2)\text{ summands}}\right)\\
&=\frac{4m(m-1)}{nm^2(m-1)}\operatorname{Cov}\left(k\left(X_{11},X_{12}\right),k\left(X_{11},X_{21}\right)\right)+\frac{2m(m-1)(m-2)}{nm^2(m-1)}\operatorname{Cov}\left(k\left(X_{11},X_{12}\right),k\left(X_{13},X_{21}\right)\right)\\
&=\frac{4}{nm}\underbrace{\operatorname{Cov}\left(k\left(X_{11},X_{12}\right),k\left(X_{11},X_{21}\right)\right)}_{\zeta_7:=}+\frac{2(m-2)}{nm}\underbrace{\operatorname{Cov}\left(k\left(X_{11},X_{12}\right),k\left(X_{13},X_{21}\right)\right)}_{\zeta_8:=}.
\end{aligned}
\tag{34}
$$

(i) when $i \neq o \neq s$, then $\operatorname{Cov}\left(k\left(X_{ij},X_{it}\right),k\left(X_{op},X_{sr}\right)\right)=0$.
(ii) symmetry of k and iid property result in $\operatorname{Cov}\left(k\left(X_{ij},X_{it}\right),k\left(X_{ip},X_{sr}\right)\right)=\operatorname{Cov}\left(k\left(X_{ij},X_{it}\right),k\left(X_{sp},X_{ir}\right)\right)$ as long as $i \neq s$
(iii) iid and symmetry of k.

It follows from inserting (27), (29), (33), and (34), into (26) that

$$
\begin{aligned}
\mathbb{V}\left(\widehat{\mathrm{Var}}_k^{(n,m)}\right) = & \underbrace{\frac{1}{n}\mathbb{V}\left(\langle P \,|\, k \,|\, P\rangle\right) + \frac{4\,(n-2)}{n\,(n-1)}\zeta_3 - \frac{4\,(m-2)}{nm}\zeta_8}_{\mathscr{O}\left(\frac{1}{n}\right)} + \underbrace{\frac{2}{n\,(n-1)}\zeta_4}_{\mathscr{O}\left(\frac{1}{n^2}\right)} \\
& + \underbrace{\frac{4\,(m-2)}{nm\,(m-1)}\zeta_1 - \frac{8}{nm}\zeta_7 + \frac{2\,(n-2)}{n\,(n-1)\,m}\zeta_9}_{\mathscr{O}\left(\frac{1}{nm}\right)} \\
& + \underbrace{\frac{2}{nm\,(m-1)}\zeta_2}_{\mathscr{O}\left(\frac{1}{nm^2}\right)} + \underbrace{\frac{2\,(m-1)}{n\,(n-1)\,m^2}\zeta_5}_{\mathscr{O}\left(\frac{1}{n^2 m}\right)} + \underbrace{\frac{1}{n\,(n-1)\,m^2}\zeta_6}_{\mathscr{O}\left(\frac{1}{n^2 m^2}\right)} .
\end{aligned}
\tag{35}
$$

In summary, our estimator is in $\mathscr{O}\left(\frac{1}{n}\left(1+\frac{1}{m}\right)\right)$ and consequently consistent w.r.t. n but not m. However, in Equation (35) appear several terms linearly, or even quadratically, shrinking with m. Consequently, it depends on the given setting how important m is, which is also demonstrated in Figure 2 on the right. There, we used our estimator with the RBF kernel for InfiMNIST generations as described in Appendix D.1.1.

E.2.3. Illustrations of the Estimator

The estimator can also be visualized in different ways as the following. We can interpret it as an operator on clusters in Figure 2. Alternatively, we can also understand it in the context of Gram-like block matrices like: For $i, s \in \{1, \ldots, n\}$ define the quadratic matrices $\mathbf{K}_{is} = \left(k_{is_{jt}}\right)_{j,t=1\ldots m} \in \mathbb{R}^{m\times m}$ with entries $k_{is_{jt}} = k\left(X_{ij}, X_{st}\right)$. Then we have the colored block matrix

$$
\begin{pmatrix}
\mathbf{K}_{11} & \cdots & \cdots & \mathbf{K}_{n1} \\
\vdots & \ddots & \mathbf{K}_{is} & \vdots \\
\vdots & \mathbf{K}_{si} & \ddots & \vdots \\
\mathbf{K}_{1n} & \cdots & \cdots & \mathbf{K}_{nn}
\end{pmatrix}
=
\left(\begin{array}{cccc|cc}
k_{11_{11}} & \cdots & \cdots & k_{11_{1m}} & k_{12_{11}} & \cdots \\
\vdots & \ddots & k_{11_{jt}} & \vdots & \vdots & \ddots \\
\vdots & k_{11_{tj}} & \ddots & \vdots & \vdots & \ddots \\
k_{11_{m1}} & \cdots & \cdots & k_{11_{mm}} & k_{12_{m1}} & \cdots \\
\hline
k_{21_{11}} & \cdots & \cdots & k_{21_{1m}} & k_{22_{11}} & \cdots \\
\vdots & \ddots & \ddots & \vdots & \vdots & \ddots
\end{array}\right) .
\tag{36}
$$

The proposed distributional variance estimator is then the average of all cyan entries without the red ones (blocks on diagonal without diagonal entries) minus the average of all black entries (off-diagonal blocks).

E.3. Distributional Covariance Estimator

Figure 21 shows an illustration of the distributional covariance estimator for two distribution (=cluster) samples of joint (P, Q) and two per-distribution samples.

Note that we have under the given i.i.d. assumptions that

$$
\begin{aligned}
\mathbb{E}\left[k\left(X_{ij}, Y_{it}\right)\right] &= \mathbb{E}\left[\mathbb{E}\left[k\left(X_{ij}, Y_{it}\right) \mid P_i, Q_i\right]\right] \\
&\overset{\text{iid}}{=} \mathbb{E}\left[\mathbb{E}\left[k\left(X_{i1}, Y_{i1}\right) \mid P_i, Q_i\right]\right] \\
&\overset{\text{iid}}{=} \mathbb{E}\left[\langle P_i \,|\, k \,|\, Q_i\rangle\right] \\
&\overset{\text{iid}}{=} \mathbb{E}\left[\langle P \,|\, k \,|\, Q\rangle\right]
\end{aligned}
\tag{37}
$$

as well as for $i \neq s$

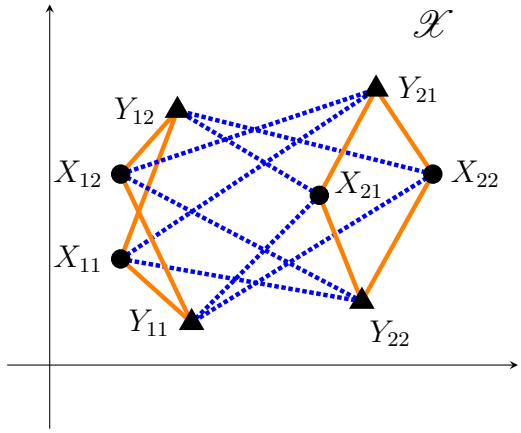


Figure 21. Illustration of the estimator $\widehat{\mathrm{Cov}}_k^{(n,m)}$ in the sample space $\mathscr{X}$ for $n = 2$ outer samples and $m = 2$ inner samples. The estimator computes the average similarity within clusters (solid orange lines) minus the average similarity between clusters (dotted blue lines). Shorter lines indicate higher similarity and larger kernel values.

$$
\begin{aligned}
\mathbb{E}\left[k\left(X_{ij}, Y_{st}\right)\right] &= \mathbb{E}\left[\mathbb{E}\left[k\left(X_{ij}, Y_{st}\right) \mid P_i, Q_s\right]\right] \\
&\overset{\text{iid}}{=} \mathbb{E}\left[\mathbb{E}\left[k\left(X_{i1}, Y_{s1}\right) \mid P_i, Q_s\right]\right] \\
&\overset{\text{iid}}{=} \mathbb{E}\left[\langle P_i \,|\, k \,|\, Q_s \rangle\right] \\
&\overset{\text{iid}}{=} \langle \mathbb{E}\left[P\right] \,|\, k \,|\, \mathbb{E}\left[Q\right] \rangle .
\end{aligned} \tag{38}
$$

E.3.1. Expectation of the Estimator

Now, we can prove that the covariance estimator is unbiased, i.e.

$$
\begin{aligned}
&\mathbb{E}\left[\widehat{\mathrm{Cov}}_k^{(n,m)}\left(\mathbf{X}, \mathbf{Y}\right)\right] \\
&= \mathbb{E}\left[\frac{1}{nm^2}\sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{t=1}^{m}\left(k\left(X_{ij}, Y_{it}\right) - \frac{1}{n-1}\sum_{\substack{s=1\\ s\neq i}}^{n} k\left(X_{ij}, Y_{st}\right)\right)\right] \\
&= \frac{1}{nm^2}\sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{t=1}^{m}\left(\mathbb{E}\left[k\left(X_{ij}, Y_{it}\right)\right] - \frac{1}{n-1}\sum_{\substack{s=1\\ s\neq i}}^{n} \mathbb{E}\left[k\left(X_{ij}, Y_{st}\right)\right]\right) \\
&\overset{\text{Eq. (37) \& (38)}}{=} \frac{1}{nm^2}\sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{t=1}^{m}\left(\mathbb{E}\left[\langle P \,|\, k \,|\, Q\rangle\right] - \frac{1}{n-1}\sum_{\substack{s=1\\ s\neq i}}^{n} \langle \mathbb{E}\left[P\right] \,|\, k \,|\, \mathbb{E}\left[Q\right]\rangle\right) \\
&= \mathbb{E}\left[\langle P \,|\, k \,|\, Q\rangle\right] - \langle \mathbb{E}\left[P\right] \,|\, k \,|\, \mathbb{E}\left[Q\right]\rangle \\
&= \mathbb{E}\left[\langle P - \mathbb{E}\left[P\right] \,|\, k \,|\, Q - \mathbb{E}\left[Q\right]\rangle\right] \\
&= \mathrm{Cov}_k\left(P, Q\right).
\end{aligned} \tag{39}
$$

E.3.2. Variance of the Estimator

Similar to the variance case, we also analyse its convergence rate:

$$
\begin{aligned}
&\mathbb{V}\left(\widehat{\mathrm{Cov}}_k^{(n,m)}(\mathbf{X},\mathbf{Y})\right)\\
&= \mathbb{V}\left(\frac{1}{nm^2}\sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{t=1}^{m}\left(k\left(X_{ij},Y_{it}\right)-\frac{1}{n-1}\sum_{\substack{s=1\\ s\neq i}}^{n}k\left(X_{ij},Y_{st}\right)\right)\right)\\
&= \mathbb{V}\left(\frac{1}{nm^2}\sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{t=1}^{m}k\left(X_{ij},Y_{it}\right)\right)+\mathbb{V}\left(\frac{1}{n\left(n-1\right)m^2}\sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{t=1}^{m}\sum_{\substack{s=1\\ s\neq i}}^{n}k\left(X_{ij},Y_{st}\right)\right)\\
&\quad -2\,\mathrm{Cov}\left(\frac{1}{nm^2}\sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{t=1}^{m}k\left(X_{ij},Y_{it}\right),\frac{1}{n\left(n-1\right)m^2}\sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{t=1}^{m}\sum_{\substack{s=1\\ s\neq i}}^{n}k\left(X_{ij},Y_{st}\right)\right)
\end{aligned}
\tag{40}
$$

$$
\begin{aligned}
&\mathbb{V}\left(\frac{1}{nm^2}\sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{t=1}^{m}k\left(X_{ij},Y_{it}\right)\right)\\
&\overset{\text{TV}}{=} \mathbb{V}\left(\mathbb{E}\left[\frac{1}{nm^2}\sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{t=1}^{m}k\left(X_{ij},Y_{it}\right)\mid P_1\ldots P_n,Q_1\ldots Q_n\right]\right)\\
&\quad +\mathbb{E}\left[\mathbb{V}\left(\frac{1}{nm^2}\sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{t=1}^{m}k\left(X_{ij},Y_{it}\right)\mid P_1\ldots P_n,Q_1\ldots Q_n\right)\right]\\
&\overset{\text{iid}}{=} \mathbb{V}\left(\frac{1}{nm^2}\sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{t=1}^{m}\mathbb{E}\left[k\left(X_{ij},Y_{it}\right)\mid P_i,Q_i\right]\right)\\
&\quad +\mathbb{E}\left[\frac{1}{n^2}\sum_{i=1}^{n}\mathbb{V}\left(\frac{1}{m^2}\sum_{j=1}^{m}\sum_{t=1}^{m}k\left(X_{ij},Y_{it}\right)\mid P_i,Q_i\right)\right]\\
&\overset{\text{iid}}{=} \mathbb{V}\left(\frac{1}{n}\sum_{i=1}^{n}\mathbb{E}\left[k\left(X_{i1},Y_{i1}\right)\mid P_i,Q_i\right]\right)+\frac{1}{n}\mathbb{E}\left[\mathbb{V}\left(\frac{1}{m^2}\sum_{j=1}^{m}\sum_{t=1}^{m}k\left(X_{1j},Y_{1t}\right)\mid P_1,Q_1\right)\right]\\
&\overset{\text{iid}}{=} \frac{1}{n}\mathbb{V}\left[\langle P\,|\,k\,|\,Q\rangle\right]+\frac{1}{n}\mathbb{E}\left[\mathbb{V}\left(\frac{1}{m^2}\sum_{j=1}^{m}\sum_{t=1}^{m}k\left(X_{1j},Y_{1t}\right)\mid P_1,Q_1\right)\right]\\
&\overset{\text{iid}}{=} \frac{1}{n}\underbrace{\mathbb{V}\left(\langle P\,|\,k\,|\,Q\rangle\right)}_{\eta_1:=}+\frac{1}{nm^2}\underbrace{\mathbb{E}\left[\mathbb{V}\left(k\left(X_{11},Y_{11}\right)\mid P_1,Q_1\right)\right]}_{\eta_2:=}
\end{aligned}
\tag{41}
$$

and for the second term we have

$$
\begin{aligned}
&\mathbb{V}\left(\frac{1}{n\left(n-1\right)m^2}\sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{t=1}^{m}\sum_{\substack{s=1\\s\neq i}}^{n}k\left(X_{ij},Y_{st}\right)\right)\\
&\overset{\text{TV}}{=}\underbrace{\mathbb{V}\left(\mathbb{E}\left[\frac{1}{n\left(n-1\right)m^2}\sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{\substack{s=1\\s\neq i}}^{n}\sum_{t=1}^{m}k\left(X_{ij},Y_{st}\right)\mid P_1\ldots P_n,Q_1\ldots Q_n\right]\right)}_{\text{(III)}:=}\\
&\quad+\underbrace{\mathbb{E}\left[\mathbb{V}\left(\frac{1}{n\left(n-1\right)m^2}\sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{\substack{s=1\\s\neq i}}^{n}\sum_{t=1}^{m}k\left(X_{ij},Y_{st}\right)\mid P_1\ldots P_n,Q_1\ldots Q_n\right)\right]}_{\text{(IV)}:=}.
\end{aligned}
\tag{42}
$$

Due to the length of the expression, we first solve (III) and then (IV).

$$
\begin{aligned}
\text{(III)} &\overset{\text{iid}}{=} \mathbb{V}\left(\frac{1}{n\left(n-1\right)m^2}\sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{\substack{s=1\\s\neq i}}^{n}\sum_{t=1}^{m}\mathbb{E}\left[k\left(X_{ij},Y_{st}\right)\mid P_i,Q_s\right]\right)\\
&\overset{\text{iid}}{=} \mathbb{V}\left(\frac{1}{n\left(n-1\right)m^2}\sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{\substack{s=1\\s\neq i}}^{n}\sum_{t=1}^{m}\mathbb{E}\left[k\left(X_{i1},Y_{s1}\right)\mid P_i,Q_s\right]\right)\\
&= \mathbb{V}\left(\frac{1}{n\left(n-1\right)}\sum_{i=1}^{n}\sum_{\substack{s=1\\s\neq i}}^{n}\left\langle P_i\,|\,k\,|\,Q_s\right\rangle\right)\\
&= \mathbb{E}\left[\left(\frac{1}{n\left(n-1\right)}\sum_{i=1}^{n}\sum_{\substack{s=1\\s\neq i}}^{n}\left\langle P_i\,|\,k\,|\,Q_s\right\rangle\right)^2\right]-\left(\mathbb{E}\left[\frac{1}{n\left(n-1\right)}\sum_{i=1}^{n}\sum_{\substack{s=1\\s\neq i}}^{n}\left\langle P_i\,|\,k\,|\,Q_s\right\rangle\right]\right)^2\\
&\overset{\text{iid}}{=} \frac{1}{n^2\left(n-1\right)^2}\sum_{i=1}^{n}\sum_{\substack{s=1\\s\neq i}}^{n}\sum_{j=1}^{n}\sum_{\substack{t=1\\t\neq j}}^{n}\mathbb{E}\left[\left\langle P_i\,|\,k\,|\,Q_s\right\rangle\left\langle P_j\,|\,k\,|\,Q_t\right\rangle\right]-\left\langle\mathbb{E}\left[P\right]\,|\,k\,|\,\mathbb{E}\left[Q\right]\right\rangle^2\\
&= \frac{1}{n^2\left(n-1\right)^2}\sum_{i=1}^{n}\sum_{\substack{s=1\\s\neq i}}^{n}\mathbb{E}\left[\left\langle P_i\,|\,k\,|\,Q_s\right\rangle^2\right]+\frac{1}{n^2\left(n-1\right)^2}\sum_{i=1}^{n}\sum_{\substack{s=1\\s\neq i}}^{n}\sum_{\substack{t=1\\t\neq j\\t\neq s}}^{n}\mathbb{E}\left[\left\langle P_i\,|\,k\,|\,Q_s\right\rangle\left\langle P_i\,|\,k\,|\,Q_t\right\rangle\right]\\
&\quad+\frac{1}{n^2\left(n-1\right)^2}\sum_{i=1}^{n}\sum_{\substack{s=1\\s\neq i}}^{n}\sum_{\substack{j=1\\j\neq i\\j\neq s}}^{n}\mathbb{E}\left[\left\langle P_i\,|\,k\,|\,Q_s\right\rangle\left\langle P_j\,|\,k\,|\,Q_s\right\rangle\right]\\
&\quad+\frac{1}{n^2\left(n-1\right)^2}\sum_{i=1}^{n}\sum_{\substack{s=1\\s\neq i}}^{n}\sum_{\substack{j=1\\j\neq i\\j\neq s}}^{n}\sum_{\substack{t=1\\t\neq i\\t\neq s\\t\neq j}}^{n}\mathbb{E}\left[\left\langle P_i\,|\,k\,|\,Q_s\right\rangle\left\langle P_j\,|\,k\,|\,Q_t\right\rangle\right]-\left\langle\mathbb{E}\left[P\right]\,|\,k\,|\,\mathbb{E}\left[Q\right]\right\rangle^2
\end{aligned}
\tag{43}
$$

$$
\begin{aligned}
&\overset{\text{iid}}{=} \frac{1}{n\left(n-1\right)}\mathbb{E}\left[\left\langle P_1\,|\,k\,|\,Q_2\right\rangle^2\right]+\frac{n-2}{n\left(n-1\right)}\mathbb{E}\left[\left\langle P_1\,|\,k\,|\,Q_2\right\rangle\left\langle P_1\,|\,k\,|\,Q_3\right\rangle\right]\\
&\quad+\frac{n-2}{n\left(n-1\right)}\mathbb{E}\left[\left\langle P_2\,|\,k\,|\,Q_1\right\rangle\left\langle P_3\,|\,k\,|\,Q_1\right\rangle\right]+\frac{\left(n-2\right)\left(n-3\right)}{n\left(n-1\right)}\left\langle\mathbb{E}\left[P\right]\,|\,k\,|\,\mathbb{E}\left[Q\right]\right\rangle^2-\left\langle\mathbb{E}\left[P\right]\,|\,k\,|\,\mathbb{E}\left[Q\right]\right\rangle^2\\
&= \frac{1}{n\left(n-1\right)}\mathbb{E}\left[\left\langle P_1\,|\,k\,|\,Q_2\right\rangle^2\right]+\frac{n-2}{n\left(n-1\right)}\mathbb{E}\left[\left\langle P_1\,|\,k\,|\,Q_2\right\rangle\left\langle P_1\,|\,k\,|\,Q_3\right\rangle\right]\\
&\quad+\frac{n-2}{n\left(n-1\right)}\mathbb{E}\left[\left\langle P_2\,|\,k\,|\,Q_1\right\rangle\left\langle P_3\,|\,k\,|\,Q_1\right\rangle\right]+\frac{\left(n-2\right)\left(n-3\right)-n\left(n-1\right)}{n\left(n-1\right)}\left\langle\mathbb{E}\left[P\right]\,|\,k\,|\,\mathbb{E}\left[Q\right]\right\rangle^2\\
&= \frac{1}{n\left(n-1\right)}\mathbb{E}\left[\left\langle P_1\,|\,k\,|\,Q_2\right\rangle^2\right]+\frac{n-2}{n\left(n-1\right)}\mathbb{E}\left[\left\langle P_1\,|\,k\,|\,Q_2\right\rangle\left\langle P_1\,|\,k\,|\,Q_3\right\rangle\right]\\
&\quad+\frac{n-2}{n\left(n-1\right)}\mathbb{E}\left[\left\langle P_2\,|\,k\,|\,Q_1\right\rangle\left\langle P_3\,|\,k\,|\,Q_1\right\rangle\right]+\frac{\left(n-2\right)\left(n-3\right)-n\left(n-1\right)}{n\left(n-1\right)}\left\langle\mathbb{E}\left[P\right]\,|\,k\,|\,\mathbb{E}\left[Q\right]\right\rangle^2\\
&= \frac{1}{n\left(n-1\right)}\mathbb{E}\left[\left\langle P_1\,|\,k\,|\,Q_2\right\rangle^2\right]+\frac{n-2}{n\left(n-1\right)}\mathbb{E}\left[\left\langle P_1\,|\,k\,|\,Q_2\right\rangle\left\langle P_1\,|\,k\,|\,Q_3\right\rangle\right]\\
&\quad+\frac{n-2}{n\left(n-1\right)}\mathbb{E}\left[\left\langle P_2\,|\,k\,|\,Q_1\right\rangle\left\langle P_3\,|\,k\,|\,Q_1\right\rangle\right]-\frac{4n-6}{n\left(n-1\right)}\left\langle\mathbb{E}\left[P\right]\,|\,k\,|\,\mathbb{E}\left[Q\right]\right\rangle^2\\
&\overset{\text{(i)}}{=} \frac{1}{n\left(n-1\right)}\eta_3+\frac{n-2}{n\left(n-1\right)}\eta_4
\end{aligned}
$$

(i) with $\eta_3 := \mathbb{E}\left[\langle P_1 \,|\, k \,|\, Q_2\rangle^2\right] - 2\,\langle \mathbb{E}\,[P] \,|\, k \,|\, \mathbb{E}\,[Q]\rangle^2$ and $\eta_4 := \mathbb{E}\,[\langle P_1 \,|\, k \,|\, Q_2\rangle\,\langle P_1 \,|\, k \,|\, Q_3\rangle] + \mathbb{E}\,[\langle P_2 \,|\, k \,|\, Q_1\rangle\,\langle P_3 \,|\, k \,|\, Q_1\rangle] - 4\,\langle \mathbb{E}\,[P] \,|\, k \,|\, \mathbb{E}\,[Q]\rangle^2$.

Due to analogous reasons as in the distributional variance case, we have

$$
\begin{aligned}
(\mathrm{IV}) = {} & \underbrace{\mathbb{E}\left[\frac{1}{n^2\,(n-1)^2}\sum_{i=1}^{n}\sum_{\substack{s=1\\ s\neq i}}^{n}\mathbb{V}\left(\frac{1}{m^2}\sum_{j=1}^{m}\sum_{t=1}^{m}k\left(X_{ij},Y_{st}\right) \mid P_i, Q_s\right)\right]}_{(\mathrm{IVa}):=} \\
& + \underbrace{\mathbb{E}\left[\frac{1}{n^2\,(n-1)^2}\sum_{i=1}^{n}\sum_{\substack{s=1\\ s\neq i}}^{n}\sum_{\substack{b=1\\ b\neq i\\ b\neq s}}^{n}\mathrm{Cov}\left(\frac{1}{m^2}\sum_{j=1}^{m}\sum_{t=1}^{m}k\left(X_{ij},Y_{st}\right), \frac{1}{m^2}\sum_{c=1}^{m}\sum_{d=1}^{m}k\left(X_{ic},Y_{bd}\right) \mid P_i, Q_b, Q_s\right)\right]}_{(\mathrm{IVb}):=} \\
& + \underbrace{\mathbb{E}\left[\frac{1}{n^2\,(n-1)^2}\sum_{i=1}^{n}\sum_{\substack{s=1\\ s\neq i}}^{n}\sum_{\substack{a=1\\ a\neq s\\ a\neq i}}^{n}\mathrm{Cov}\left(\frac{1}{m^2}\sum_{j=1}^{m}\sum_{t=1}^{m}k\left(X_{ij},Y_{st}\right), \frac{1}{m^2}\sum_{c=1}^{m}\sum_{d=1}^{m}k\left(X_{ac},Y_{sd}\right) \mid P_i, P_a, Q_s\right)\right]}_{(\mathrm{IVc}):=}.
\end{aligned}
\tag{44}
$$

Due to the length of the expressions, we again first look at (IVa), then (IVb), and then (IVc).

In the following equation, we use almost the identical steps as in Equation (32):

$$
\begin{aligned}
(\mathrm{IVa}) &= \mathbb{E}\left[\frac{1}{n^2(n-1)^2}\sum_{i=1}^{n}\sum_{\substack{s=1\\s\neq i}}^{n}\mathbb{V}\left(\frac{1}{m^2}\sum_{j=1}^{m}\sum_{t=1}^{m}k\left(X_{ij},Y_{st}\right)\mid P_i,Q_s\right)\right]\\
&\overset{\text{iid}}{=}\frac{1}{n(n-1)}\mathbb{E}\left[\mathbb{V}\left(\frac{1}{m^2}\sum_{j=1}^{m}\sum_{t=1}^{m}k\left(X_{1j},Y_{2t}\right)\mid P_1,Q_2\right)\right]\\
&=\frac{1}{n(n-1)}\mathbb{E}\left[\mathrm{Cov}\left(\frac{1}{m^2}\sum_{j=1}^{m}\sum_{t=1}^{m}k\left(X_{1j},Y_{2t}\right),\frac{1}{m^2}\sum_{i=1}^{m}\sum_{s=1}^{m}k\left(X_{1i},Y_{2s}\right)\mid P_1,Q_2\right)\right]\\
&=\frac{1}{n(n-1)m^4}\sum_{j=1}^{m}\sum_{t=1}^{m}\sum_{i=1}^{m}\sum_{s=1}^{m}\mathbb{E}\left[\mathrm{Cov}\left(k\left(X_{1j},Y_{2t}\right),k\left(X_{1i},Y_{2s}\right)\mid P_1,Q_2\right)\right]\\
&\overset{\text{iid}}{=}\frac{1}{n(n-1)m^4}\sum_{j=1}^{m}\sum_{t=1}^{m}\mathbb{E}\left[\mathrm{Cov}\left(k\left(X_{1j},Y_{2t}\right),k\left(X_{1j},Y_{2t}\right)\mid P_1,Q_2\right)\right]\\
&\quad+\frac{1}{n(n-1)m^4}\sum_{j=1}^{m}\sum_{t=1}^{m}\sum_{\substack{i=1\\i\neq j}}^{m}\mathbb{E}\left[\mathrm{Cov}\left(k\left(X_{1j},Y_{2t}\right),k\left(X_{1i},Y_{2t}\right)\mid P_1,Q_2\right)\right]\\
&\quad+\frac{1}{n(n-1)m^4}\sum_{j=1}^{m}\sum_{t=1}^{m}\sum_{\substack{s=1\\s\neq t}}^{m}\mathbb{E}\left[\mathrm{Cov}\left(k\left(X_{1j},Y_{2t}\right),k\left(X_{1j},Y_{2s}\right)\mid P_1,Q_2\right)\right]\\
&\overset{\text{iid}}{=}\frac{1}{n(n-1)m^2}\underbrace{\mathbb{E}\left[\mathrm{Cov}\left(k\left(X_{11},Y_{21}\right),k\left(X_{11},Y_{21}\right)\mid P_1,Q_2\right)\right]}_{\eta_5:=}\\
&\quad+\frac{m-1}{n(n-1)m^2}\underbrace{\mathbb{E}\left[\mathrm{Cov}\left(k\left(X_{11},Y_{21}\right),k\left(X_{12},Y_{21}\right)\mid P_1,Q_2\right)\right]}_{\eta_6:=}\\
&\quad+\frac{m-1}{n(n-1)m^2}\underbrace{\mathbb{E}\left[\mathrm{Cov}\left(k\left(X_{11},Y_{21}\right),k\left(X_{11},Y_{22}\right)\mid P_1,Q_2\right)\right]}_{\eta_7:=}.
\end{aligned}
\tag{45}
$$

Next, we have analogous to Equation (33)

$$
\begin{aligned}
(\mathrm{IVb}) &= \mathbb{E}\left[\frac{1}{n^2(n-1)^2}\sum_{i=1}^{n}\sum_{\substack{s=1\\s\neq i}}^{n}\sum_{\substack{b=1\\b\neq i\\b\neq s}}^{n}\mathrm{Cov}\left(\frac{1}{m^2}\sum_{j=1}^{m}\sum_{t=1}^{m}k\left(X_{ij},Y_{st}\right),\frac{1}{m^2}\sum_{c=1}^{m}\sum_{d=1}^{m}k\left(X_{ic},Y_{bd}\right)\mid P_i,Q_s,Q_b\right)\right]\\
&= \mathbb{E}\left[\frac{1}{n^2(n-1)^2m^4}\sum_{i=1}^{n}\sum_{\substack{s=1\\s\neq i}}^{n}\sum_{\substack{b=1\\b\neq i\\b\neq s}}^{n}\sum_{j=1}^{m}\sum_{t=1}^{m}\sum_{d=1}^{m}\mathrm{Cov}\left(k\left(X_{ij},Y_{st}\right),k\left(X_{ij},Y_{bd}\right)\mid P_i,Q_s,Q_b\right)\right]\\
&\overset{\text{iid}}{=} \mathbb{E}\left[\frac{1}{n^2(n-1)^2m^4}\sum_{i=1}^{n}\sum_{\substack{s=1\\s\neq i}}^{n}\sum_{\substack{b=1\\b\neq i\\b\neq s}}^{n}\sum_{j=1}^{m}\sum_{t=1}^{m}\sum_{d=1}^{m}\mathrm{Cov}\left(k\left(X_{11},Y_{21}\right),k\left(X_{11},Y_{31}\right)\mid P_1,Q_2,Q_3\right)\right]\\
&= \frac{n-2}{n(n-1)m}\underbrace{\mathbb{E}\left[\mathrm{Cov}\left(k\left(X_{11},Y_{21}\right),k\left(X_{11},Y_{31}\right)\mid P_1,Q_2,Q_3\right)\right]}_{\eta_9:=}
\end{aligned}
\tag{46}
$$

and in an almost identical manner

$$
\begin{aligned}
(\mathrm{IVc}) &= \mathbb{E}\left[\frac{1}{n^2(n-1)^2}\sum_{i=1}^{n}\sum_{\substack{s=1\\s\neq i}}^{n}\sum_{\substack{a=1\\a\neq s\\a\neq i}}^{n}\mathrm{Cov}\left(\frac{1}{m^2}\sum_{j=1}^{m}\sum_{t=1}^{m}k\left(X_{ij},Y_{st}\right),\frac{1}{m^2}\sum_{c=1}^{m}\sum_{d=1}^{m}k\left(X_{ac},Y_{sd}\right)\mid P_i,P_a,Q_s\right)\right]\\
&= \mathbb{E}\left[\frac{1}{n^2(n-1)^2m^4}\sum_{i=1}^{n}\sum_{\substack{s=1\\s\neq i}}^{n}\sum_{\substack{b=1\\b\neq i\\b\neq s}}^{n}\sum_{j=1}^{m}\sum_{t=1}^{m}\sum_{d=1}^{m}\mathrm{Cov}\left(k\left(X_{ij},Y_{st}\right),k\left(X_{ac},Y_{sd}\right)\mid P_i,P_a,Q_s\right)\right]\\
&\overset{\text{iid}}{=} \mathbb{E}\left[\frac{1}{n^2(n-1)^2m^4}\sum_{i=1}^{n}\sum_{\substack{s=1\\s\neq i}}^{n}\sum_{\substack{b=1\\b\neq i\\b\neq s}}^{n}\sum_{j=1}^{m}\sum_{t=1}^{m}\sum_{d=1}^{m}\mathrm{Cov}\left(k\left(X_{11},Y_{31}\right),k\left(X_{21},Y_{31}\right)\mid P_1,P_2,Q_3\right)\right]\\
&= \frac{n-2}{n(n-1)m}\underbrace{\mathbb{E}\left[\mathrm{Cov}\left(k\left(X_{11},Y_{31}\right),k\left(X_{21},Y_{31}\right)\mid P_1,P_2,Q_3\right)\right]}_{\eta_{10}:=}.
\end{aligned}
\tag{47}
$$

The only term left is

$$
\begin{aligned}
&\operatorname{Cov}\left(\frac{1}{nm^2}\sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{t=1}^{m}k\left(X_{ij},Y_{it}\right),\frac{1}{n\left(n-1\right)m^2}\sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{t=1}^{m}\sum_{\substack{s=1\\s\neq i}}^{n}k\left(X_{ij},Y_{st}\right)\right)\\
&=\frac{1}{n^2\left(n-1\right)m^4}\sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{t=1}^{m}\sum_{o=1}^{n}\sum_{p=1}^{m}\sum_{\substack{s=1\\s\neq o}}^{n}\sum_{r=1}^{m}\operatorname{Cov}\left(k\left(X_{ij},Y_{it}\right),k\left(X_{op},Y_{sr}\right)\right)\\
&\overset{\text{(i)}}{=}\frac{1}{n^2\left(n-1\right)m^4}\sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{\substack{t=1\\t\neq j}}^{m}\sum_{p=1}^{m}\sum_{\substack{s=1\\s\neq i}}^{n}\sum_{r=1}^{m}\operatorname{Cov}\left(k\left(X_{ij},Y_{it}\right),k\left(X_{ip},Y_{sr}\right)\right)\\
&\quad+\frac{1}{n^2\left(n-1\right)m^4}\sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{\substack{t=1\\t\neq j}}^{m}\sum_{p=1}^{m}\sum_{\substack{s=1\\s\neq i}}^{n}\sum_{r=1}^{m}\operatorname{Cov}\left(k\left(X_{ij},Y_{it}\right),k\left(X_{sp},Y_{ir}\right)\right)\\
&\overset{\text{iid}}{=}\frac{1}{nm^3}\sum_{j=1}^{m}\sum_{\substack{t=1\\t\neq j}}^{m}\sum_{p=1}^{m}\operatorname{Cov}\left(k\left(X_{1j},Y_{1t}\right),k\left(X_{1p},Y_{21}\right)\right)+\frac{1}{nm^3}\sum_{j=1}^{m}\sum_{\substack{t=1\\t\neq j}}^{m}\sum_{r=1}^{m}\operatorname{Cov}\left(k\left(X_{1j},Y_{1t}\right),k\left(X_{21},Y_{1r}\right)\right)\\
&=\frac{1}{nm^3}\sum_{j=1}^{m}\sum_{\substack{t=1\\t\neq j}}^{m}\operatorname{Cov}\left(k\left(X_{1j},Y_{1t}\right),k\left(X_{1j},Y_{21}\right)\right)+\frac{1}{nm^3}\sum_{j=1}^{m}\sum_{\substack{t=1\\t\neq j}}^{m}\operatorname{Cov}\left(k\left(X_{1j},Y_{1t}\right),k\left(X_{1t},Y_{21}\right)\right)\\
&\quad+\frac{1}{nm^3}\sum_{j=1}^{m}\sum_{\substack{t=1\\t\neq j}}^{m}\sum_{\substack{p=1\\p\neq j\\p\neq t}}^{m}\operatorname{Cov}\left(k\left(X_{1j},Y_{1t}\right),k\left(X_{1p},Y_{21}\right)\right)+\frac{1}{nm^3}\sum_{j=1}^{m}\sum_{\substack{t=1\\t\neq j}}^{m}\operatorname{Cov}\left(k\left(X_{1j},Y_{1t}\right),k\left(X_{21},Y_{1j}\right)\right)\\
&\quad+\frac{1}{nm^3}\sum_{j=1}^{m}\sum_{\substack{t=1\\t\neq j}}^{m}\operatorname{Cov}\left(k\left(X_{1j},Y_{1t}\right),k\left(X_{21},Y_{1t}\right)\right)+\frac{1}{nm^3}\sum_{j=1}^{m}\sum_{\substack{t=1\\t\neq j}}^{m}\sum_{\substack{p=1\\p\neq j\\p\neq t}}^{m}\operatorname{Cov}\left(k\left(X_{1j},Y_{1t}\right),k\left(X_{21},Y_{1p}\right)\right)\\
&\overset{\text{iid}}{=}\frac{1}{nm^3}\sum_{j=1}^{m}\sum_{\substack{t=1\\t\neq j}}^{m}\operatorname{Cov}\left(k\left(X_{11},Y_{12}\right),k\left(X_{11},Y_{21}\right)\right)+\frac{1}{nm^3}\sum_{j=1}^{m}\sum_{\substack{t=1\\t\neq j}}^{m}\operatorname{Cov}\left(k\left(X_{11},Y_{12}\right),k\left(X_{12},Y_{21}\right)\right)\\
&\quad+\frac{1}{nm^3}\sum_{j=1}^{m}\sum_{\substack{t=1\\t\neq j}}^{m}\sum_{\substack{p=1\\p\neq j\\p\neq t}}^{m}\operatorname{Cov}\left(k\left(X_{11},Y_{12}\right),k\left(X_{13},Y_{21}\right)\right)+\frac{1}{nm^3}\sum_{j=1}^{m}\sum_{\substack{t=1\\t\neq j}}^{m}\operatorname{Cov}\left(k\left(X_{11},Y_{12}\right),k\left(X_{21},Y_{11}\right)\right)\\
&\quad+\frac{1}{nm^3}\sum_{j=1}^{m}\sum_{\substack{t=1\\t\neq j}}^{m}\operatorname{Cov}\left(k\left(X_{11},Y_{12}\right),k\left(X_{21},Y_{12}\right)\right)+\frac{1}{nm^3}\sum_{j=1}^{m}\sum_{\substack{t=1\\t\neq j}}^{m}\sum_{\substack{p=1\\p\neq j\\p\neq t}}^{m}\operatorname{Cov}\left(k\left(X_{11},Y_{12}\right),k\left(X_{21},Y_{13}\right)\right)\\
&=\frac{m-1}{nm^2}\underbrace{\left(\operatorname{Cov}\left(k\left(X_{11},Y_{12}\right),k\left(X_{11},Y_{21}\right)\right)+\operatorname{Cov}\left(k\left(X_{11},Y_{12}\right),k\left(X_{21},Y_{11}\right)\right)\right)}_{\eta_{11}:=}\\
&\quad+\frac{m-1}{nm^2}\underbrace{\left(\operatorname{Cov}\left(k\left(X_{11},Y_{12}\right),k\left(X_{21},Y_{12}\right)\right)+\operatorname{Cov}\left(k\left(X_{11},Y_{12}\right),k\left(X_{12},Y_{21}\right)\right)\right)}_{\eta_{12}:=}\\
&\quad+\frac{\left(m-1\right)\left(m-2\right)}{nm^2}\underbrace{\left(\operatorname{Cov}\left(k\left(X_{11},Y_{12}\right),k\left(X_{13},Y_{21}\right)\right)+\operatorname{Cov}\left(k\left(X_{11},Y_{12}\right),k\left(X_{21},Y_{13}\right)\right)\right)}_{\eta_{13}:=}.
\end{aligned}
\tag{48}
$$

By combining all previous equations, we get

$$
\begin{aligned}
&\mathbb{V}\left(\widehat{\mathrm{Cov}}_k^{(n,m)}(\mathbf{X},\mathbf{Y})\right) \\
&= \underbrace{\frac{1}{n}\eta_1 + \frac{n-2}{n(n-1)}\eta_4 + \frac{-2(m-1)(m-2)}{nm^2}\eta_{13}}_{\mathscr{O}\left(\frac{1}{n}\right)} + \underbrace{\frac{1}{n(n-1)}\eta_3}_{\mathscr{O}\left(\frac{1}{n^2}\right)} \\
&\quad + \underbrace{\frac{-2(m-1)}{nm^2}(\eta_{11}+\eta_{12}) + \frac{n-2}{n(n-1)m}(\eta_9+\eta_{10})}_{\mathscr{O}\left(\frac{1}{nm}\right)} \\
&\quad + \underbrace{\frac{1}{nm^2}\eta_2}_{\mathscr{O}\left(\frac{1}{nm^2}\right)} + \underbrace{\frac{m-1}{n(n-1)m^2}(\eta_6+\eta_7)}_{\mathscr{O}\left(\frac{1}{n^2m}\right)} + \underbrace{\frac{1}{n(n-1)m^2}\eta_5}_{\mathscr{O}\left(\frac{1}{n^2m^2}\right)}.
\end{aligned}
\tag{49}
$$