TIC-TAC: A Framework for Improved Covariance Estimation in Deep Heteroscedastic Regression

Megh Shukla [1]   **Mathieu Salzmann** [1] [2]   **Alexandre Alahi** [1]

Abstract

Deep heteroscedastic regression involves jointly optimizing the mean and covariance of the predicted distribution using the negative log-likelihood. However, recent works show that this may result in sub-optimal convergence due to the challenges associated with covariance estimation. While the literature addresses this by proposing alternate formulations to mitigate the impact of the predicted covariance, we focus on improving the predicted covariance itself. We study two questions: (1) Does the predicted covariance truly capture the randomness of the predicted mean? (2) In the absence of supervision, how can we quantify the accuracy of covariance estimation? We address (1) with a *Taylor Induced Covariance (TIC)*, which captures the randomness of the predicted mean by incorporating its gradient and curvature through the second order Taylor polynomial. Furthermore, we tackle (2) by introducing a *Task Agnostic Correlations (TAC)* metric, which combines the notion of correlations and absolute error to evaluate the covariance. We evaluate TIC-TAC across multiple experiments spanning synthetic and real-world datasets. Our results show that not only does TIC accurately learn the covariance, it additionally facilitates an improved convergence of the negative log-likelihood. Our code is available at https://github.com/vita-epfl/TIC-TAC

1. Introduction

Modeling the target distribution is an important design choice in heteroscedastic regression. Typically, the target is assumed to follow a multivariate normal distribution, where the true mean and covariance are sample dependent and unknown. Deep heteroscedastic regression learns this distribution by predicting the mean and covariance through two neural networks, which are jointly optimized to minimize the negative log-likelihood. However, recent results in deep heteroscedastic regression show that this joint optimization leads to sub-optimal convergence.

This challenge is primarily attributed to covariance estimation in heteroscedastic regression (Skafte et al., 2019). Recent studies show that the gradient of incorrect variance predictions significantly hinders optimization, and address this by proposing alternate formulations to mitigate its impact during optimization (Skafte et al., 2019; Seitzer et al., 2022; Stirn et al., 2023; Immer et al., 2023). While these approaches aim at regularizing the covariance, this begets the question: Can we improve upon the predicted covariance? We argue that the current parameterization for the covariance may not truly explain the randomness of the predicted mean. Indeed, we observe in Figure 1 that in the absence of direct supervision, the predicted variance may take on arbitrary values leading to sub-optimal convergence. Moreover, evaluating the covariance is challenging without ground-truth labels. Optimization metrics such as the likelihood are not a direct measure for the covariance since they also incorporate the performance of the mean estimator.

Hence, this paper studies covariance estimation in deep heteroscedastic regression. We distill the challenges into two problems: (1) How do we model the covariance to explain the randomness of the prediction? (2) How do we evaluate the predicted covariance in the absence of annotations?

Our first contribution, the **Taylor Induced Covariance (TIC)**, explains the randomness of the prediction through its gradient and curvature. We develop a closed-form approximation for the covariance of the prediction through its second order Taylor polynomial. Modeling the covariance through the gradient and curvature quantifies the variation in the prediction within a small neighborhood of the input. TIC when learnt through the negative log-likelihood not only captures the underlying correlations but also improves the convergence of the negative log-likelihood.

Our second contribution, the **Task Agnostic Correlations (TAC)**, addresses the lack of a direct metric to evaluate the

---

[1]École Polytechnique Fédérale de Lausanne (EPFL)
[2]Swiss Data Science Center (SDSC).
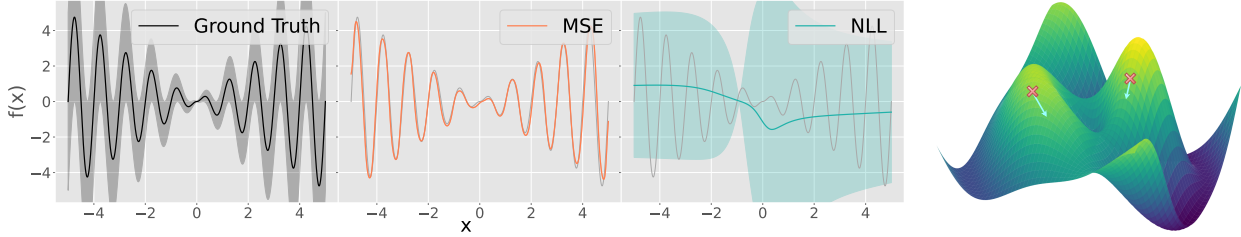Correspondence to: Megh Shukla <megh.shukla@epfl.ch>.

Figure 1: *Motivation.* (Left) We learn a varying amplitude sinusoidal with heteroscedastic variance (shaded region). We observe sub-optimal convergence since the predicted variance may be arbitrary and incorrectly minimizes the likelihood. We address this through a *Taylor Induced Covariance* by tying the randomness of the prediction to its gradient and curvature. (Right) *The gradient and curvature quantify the variation in the prediction within a small neighborhood of the input.*

covariance. By definition, an accurate covariance correctly estimates the underlying correlations. Hence, given a partial observation of the target, the covariance should accurately update the prediction towards the unobserved target through conditioning of the predicted distribution. Consequently, we quantify TAC as the mean absolute error between the updated prediction and the unobserved target. While the likelihood is a measure of optimization, TAC quantifies the accuracy of the learnt correlations.

We design and perform extensive experiments on synthetic (sinusoidal, multivariate) and real-world (UCI Regression and Human Pose - MPII, LSP) datasets using two metrics: TAC and the likelihood. Our experiments show that TIC outperforms the state-of-the-art baselines in learning correlations across all tasks, demonstrating improved covariance estimation in heteroscedastic regression. Additionally, we also observe that incorporating TIC into the negative log-likelihood improves convergence. Our code and environment are publicly available for reproducibility[1].

2. Deep Heteroscedastic Regression

The goal of heteroscedastic regression is to learn the unknown target distribution $p(Y|X = \boldsymbol{x})$, which is commonly assumed to be a multivariate normal distribution $\mathcal{N}(\boldsymbol{\mu}_{Y|X}, \boldsymbol{\Sigma}_{Y|X})$. Typically, deep heteroscedastic regression is performed through minimizing the negative log-likelihood of the predicted distribution $q(\hat{Y}|X = \boldsymbol{x}) = \mathcal{N}(\hat{\boldsymbol{y}}, \text{Cov}(\hat{Y}|X))$. This involves the joint optimization of estimators for the mean $\hat{\boldsymbol{y}} = f_\theta(\boldsymbol{x})$ and the covariance $\text{Cov}(\hat{Y}|X) = g_\Theta(\boldsymbol{x})$ over the dataset (Nix & Weigend, 1994; Kendall & Gal, 2017):

$$\mathbb{E}_{p(X,Y)}\left[\log \left| \text{Cov}(\hat{Y}|X) \right| + (\boldsymbol{y} - \hat{\boldsymbol{y}})^T \text{Cov}(\hat{Y}|X)^{-1} (\boldsymbol{y} - \hat{\boldsymbol{y}}) \right]. \quad (1)$$

The advantage of deep heteroscedastic regression over its non-parametric counterparts like the Gaussian Process (Le et al., 2005) is the ability to extract complex features from in-

puts such as images. This has lead to its adoption across various paradigms such as active learning (Houlsby et al., 2011; Gal et al., 2017), uncertainty estimation (Gal & Ghahramani, 2016; Kendall & Gal, 2017; Lakshminarayanan et al., 2017; Russell & Reale, 2021), image reconstruction (Dorta et al., 2018), human pose estimation (Gundavarapu et al., 2019; Nakka & Salzmann, 2023; Tekin et al., 2017), and other vision tasks (Lu & Koniusz, 2022; Simpson et al., 2022; Liu et al., 2018; Bertoni et al., 2019).

The challenge with deep heteroscedastic regression is that, while mean estimation is supervised, the covariance lacks direct supervision and needs to be inferred. This creates optimization challenges when the predicted covariance is incorrect. For instance, Skafte et al. (2019) highlights that an incorrectly predicted small variance effectively increases the learning rate, affecting optimization. Similarly, Seitzer et al. (2022) observes that poor convergence is often accompanied with a large predicted variance, which further affects convergence.

Several recent methods aim to alter the negative log-likelihood to mitigate the impact of the predicted covariance in optimization. Seitzer et al. (2022) addresses this by proposing β-NLL, which scales the negative log-likelihood objective (Eq. 1) with the predicted variance for optimization: $\mathcal{L}_{\beta-\text{NLL}} = \lfloor \text{Var}(\hat{Y}|X)^\beta \rfloor * \mathcal{L}_{\text{NLL}}$. This scaling aims to reduce the impact of the predicted variance in the training process. While simple and effective, β-NLL is not a result of a valid distribution, and the optimized values do not translate to the variance of a distribution.

The recent method of Stirn et al. (2023) proposes an alternative approach by scaling the gradients of the mean estimator with the predicted covariance. Effectively, the mean estimator is trained to minimize the mean squared error, and the covariance estimator is trained to minimize the negative log-likelihood. This involves conflicting assumptions; while the mean estimator assumes that the multivariate residual is uncorrelated, the covariance estimator is expected to recover correlations from this residual (Immer et al., 2023).

[1] https://github.com/vita-epfl/TIC-TAC

Unlike previous works which regularize the (co-)variance, Immer et al. (2023) uses the natural parameterization of the Gaussian: $n_1 = \frac{\mu}{\sigma^2}$ and $n_2 = \frac{-1}{2\sigma^2}$ for regression. Additionally, the method uses Bayesian techniques to regularize the network as well as obtain a posterior over the parameters. Similar to Seitzer et al. (2022), the method assumes a diagonal covariance matrix. However, this assumption diminishes the main advantages of learning the covariance, such as correlation analysis, sampling from the target distribution, and updating our predictions conditioned on partial observations of the target.

In contrast to previous works which focus on regularization as a means to improve optimization, this paper focuses on improving the predicted covariance *within the negative log-likelihood formulation*. The drawback of $\mathrm{Cov}(\hat{Y}|X) = g_\Theta(\boldsymbol{x})$ being an arbitrary mapping from $\boldsymbol{x}$ to a positive definite matrix is common to all the aforementioned approaches. This drawback is significant since, in the absence of supervision, $g_\Theta(\boldsymbol{x})$ can take on any value that minimizes the objective and does not necessarily represent the randomness of the prediction. Therefore, we propose a novel closed-form approximation for the predicted covariance and show that incorporating the gradient and curvature of the prediction better explains its randomness.

3. Taylor Induced Covariance (TIC)

Let us return to the prediction distribution $q(\hat{Y}|X=\boldsymbol{x})$ and ponder on a fundamental question: What is the randomness of a prediction $\hat{y}$ for a sample $\boldsymbol{x}$? Intuitively, we quantify the covariance as a function of how quickly the predicted mean changes within a small radius of $\boldsymbol{x}$. Larger derivatives imply a rapid change in $\hat{y}$, and as a result the model has a higher variance about its estimate.

We therefore proceed by introducing a heuristic interpretation of the neighborhood, which allows us to take principled steps towards a closed-form approximation.

3.1. ϵ - Neighborhood

For a continuously distributed random variable X, the probability of exactly observing $p(X = x)$ is zero. Instead, the standard approach (for example Sec. 2.4 in (Evans & Rosenthal, 2004)) is to observe X in the neighborhood of x: $X \in [x - \delta, x + \delta]$. The definition of this neighborhood is not rigid, allowing for a heuristic interpretation. For instance, we can represent this neighborhood stochastically: $X = x + \epsilon$. Here, x is the observation, and ϵ is a random variable, which we set to be a zero-mean isotropic Gaussian distribution $p(\epsilon) = \mathcal{N}(0, \sigma_\epsilon^2(\boldsymbol{x})\boldsymbol{I}_m)$ for future analysis.

The advantage of this heuristic is that it allows us to represent $\hat{y} = f_\theta(\boldsymbol{x} + \epsilon)$ stochastically. While the variance of ϵ is unknown (we will later show that it is learnt), we assume

heteroscedasticity, which allows us to represent neighborhoods of varying spatial extents for each $\boldsymbol{x}$. We therefore model $\mathrm{Cov}(f_\theta(x + \epsilon))$, and continue by taking the second order Taylor polynomial of $f_\theta(\boldsymbol{x} + \epsilon)$.

3.2. Second Order Taylor Polynomial

The second order Taylor polynomial introduces the notion of gradient and curvature in modeling the covariance, and quantifies the rate at which a function can change within a small neighborhood around $\boldsymbol{x}$. We have

$$f_\theta(\boldsymbol{x} + \epsilon) = f_\theta(\boldsymbol{x}) + \boldsymbol{J}(\boldsymbol{x})\epsilon^T + \frac{\boldsymbol{h}}{2} ,$$
$$\text{where } \boldsymbol{h}_i = \epsilon\, \mathsf{H}_i(\boldsymbol{x})\epsilon^T \ \ \forall i \in 1 \dots n . \tag{2}$$

Here, $\boldsymbol{x} \in \mathbb{R}^m$ is the input, $f_\theta(\boldsymbol{x}) \in \mathbb{R}^n$ represents the multivariate prediction, $\epsilon \in \mathbb{R}^m$ represents the neighborhood of $\boldsymbol{x}$, $\boldsymbol{J}(\boldsymbol{x}) \in \mathbb{R}^{n \times m}$ corresponds to the Jacobian matrix, and $\mathsf{H}(\boldsymbol{x}) \in \mathbb{R}^{n \times m \times m}$ represents the Hessian tensor. We note that all the individual terms in Eq. 2 are n-dimensional.

3.3. Covariance Estimation

The covariance of Eq. 2, $\mathrm{Cov} f_\theta(\boldsymbol{x} + \epsilon)$, *with respect to the random variable ϵ is*

$$\mathrm{Cov}(\boldsymbol{J}(\boldsymbol{x})\epsilon^T) + \mathrm{Cov}(\frac{\boldsymbol{h}}{2}) + 2\left[\mathrm{Cov}(\boldsymbol{J}(\boldsymbol{x})\epsilon^T, \frac{\boldsymbol{h}}{2})\right] . \tag{3}$$

We obtain this since $f_\theta(\boldsymbol{x})$ is a constant with respect to ϵ. Below, we evaluate the three terms individually.

3.3.1. ESTIMATING $\mathrm{Cov}(\boldsymbol{J}(\boldsymbol{x})\epsilon^T, \boldsymbol{h}/2)$

We begin by noting that $\boldsymbol{J}(\boldsymbol{x})\epsilon^T$ and $\boldsymbol{h}$ are n-dimensional vectors with elements $[\dots \boldsymbol{J}_i(\boldsymbol{x})\epsilon^T \dots]$ and $[\dots \epsilon\, \mathsf{H}_k(\boldsymbol{x})\epsilon^T \dots]$, respectively. The covariance between any two elements is given by

$$\mathrm{Cov}\big(\boldsymbol{J}_i(\boldsymbol{x})\epsilon^T, \epsilon\, \mathsf{H}_k(\boldsymbol{x})\epsilon^T\big)$$
$$= \mathbb{E}\big(\boldsymbol{J}_i(\boldsymbol{x})\epsilon^T \epsilon\, \mathsf{H}_k(\boldsymbol{x})\epsilon^T\big) - \mathbb{E}\big(\boldsymbol{J}_i(\boldsymbol{x})\epsilon^T\big)\mathbb{E}\big(\epsilon\, \mathsf{H}_k(\boldsymbol{x})\epsilon^T\big)$$
$$= 0 . \tag{4}$$

Odd and Even Functions. We use the property of odd-even functions (Shynk, 2012), which is based on symmetry and anti-symmetry of a function. Recall that an odd function is defined as $f(-t) = -f(t)$ and an even function as $f(-t) = f(t)$. Furthermore, the product of an odd and an even function is odd, and the product of two even functions is even. Finally, the integral of an odd function over its domain evaluates to zero.

We note that $\boldsymbol{J}_i(\boldsymbol{x})\epsilon^T = \sum_k \boldsymbol{J}_{i,k}(\boldsymbol{x})\epsilon_k^T$ is an odd function with respect to ϵ. Furthermore, our design choice of $p(\epsilon) = \mathcal{N}(0, \sigma_\epsilon^2(\boldsymbol{x})\boldsymbol{I}_m)$ implies that $p(\epsilon)$ is an even function. The

term $\mathbb{E}\big(\boldsymbol{J}_i(\boldsymbol{x})\epsilon^T\big)$ can be written as $\int_\epsilon \boldsymbol{J}_i(\boldsymbol{x})\epsilon^T p(\epsilon)\mathrm{d}\epsilon$. This term represents the integral of a product of an odd and an even function, which evaluates to zero.

The quadratic term $\epsilon\, \mathsf{H}_k(\boldsymbol{x})\epsilon^T$ can be written as $\sum_i \sum_j \mathsf{H}_{i,j}^{(k)}\epsilon_i\epsilon_j$, which is an even function. Subsequently, $\boldsymbol{J}_i(\boldsymbol{x})\epsilon^T\epsilon\,\mathsf{H}_k(\boldsymbol{x})\epsilon^T$ is a product of odd $\boldsymbol{J}_i(\boldsymbol{x})\epsilon^T$ and even $\epsilon\,\mathsf{H}_k(\boldsymbol{x})\epsilon^T$ terms. Finally, we can write $\mathbb{E}\big(\boldsymbol{J}_i(\boldsymbol{x})\epsilon^T\epsilon\,\mathsf{H}_k(\boldsymbol{x})\epsilon^T\big)$ as $\int_\epsilon \boldsymbol{J}_i(\boldsymbol{x})\epsilon^T\epsilon\,\mathsf{H}_k(\boldsymbol{x})\epsilon^T p(\epsilon)\mathrm{d}\epsilon$, which represents the integral of a product of odd, even, and even functions, which also evaluates to zero.

As a result, we get $\mathrm{Cov}\big(\boldsymbol{J}_i(\boldsymbol{x})\epsilon^T, \epsilon\,\mathsf{H}_k(\boldsymbol{x})\epsilon^T\big) = 0\ \forall i,k$, implying that $\mathrm{Cov}(\boldsymbol{J}(\boldsymbol{x})\epsilon^T, \boldsymbol{h}/2) = 0$.

3.3.2. ESTIMATING $\mathrm{Cov}(\boldsymbol{J}(\boldsymbol{x})\epsilon^T)$ AND $\mathrm{Cov}(\boldsymbol{h}/2)$

Estimating $\mathrm{Cov}(\boldsymbol{J}(\boldsymbol{x})\epsilon^T)$ and $\mathrm{Cov}(\boldsymbol{h}/2)$ in Eq. 3 is easier since they follow a linear and quadratic form, respectively, with known solutions for isotropic Gaussian random variables (Eq. 375, 379 in (Petersen & Pedersen, 2012)). Specifically, we have

$$\mathrm{Cov}(\boldsymbol{J}(\boldsymbol{x})\ \epsilon^T) = k_1(x)\boldsymbol{J}(\boldsymbol{x})\boldsymbol{J}(\boldsymbol{x})^T$$
$$\mathrm{Cov}(\boldsymbol{h}/2)_{i,j} = k_2(x)\ \mathtt{Trace}\left(\mathsf{H}_{i,:,:}(\boldsymbol{x})\ \mathsf{H}_{j,:,:}(\boldsymbol{x})\right). \quad (5)$$

Since we do not know the variance of the ϵ and its transformation for each $\boldsymbol{x}$, we define them through positive quantities $k_1(\boldsymbol{x})$ and $k_2(\boldsymbol{x})$, which are optimized by the covariance estimator $g_\Theta(\boldsymbol{x})$. We also note that both $\mathrm{Cov}(\boldsymbol{J}(\boldsymbol{x})\ \epsilon^T)$ and $\mathrm{Cov}(\boldsymbol{h}/2)$ have dimensions $n \times n$.

Finally, we obtain the solution for Eq. 3 by substituting Eq. 5 and Eq. 4 into it, which yields

$$\mathrm{Cov}f_\theta(\boldsymbol{x}+\epsilon) = k_1(\boldsymbol{x})\boldsymbol{J}(\boldsymbol{x})\boldsymbol{J}(\boldsymbol{x})^T + \mathcal{H}$$
$$\text{where } \mathcal{H}_{i,j} = k_2(\boldsymbol{x})\ \mathtt{Trace}\left(\mathsf{H}_{i,:,:}(\boldsymbol{x})\ \mathsf{H}_{j,:,:}(\boldsymbol{x})\right). \quad (6)$$

3.4. Formulation

We defined the covariance through ϵ, the neighborhood random variable which allows us to capture the gradient and curvature of $f_\theta(x+\epsilon)$. However, the target y could have stochasticity that does not depend upon the neighborhood. We take as an example the function $y = c + \mathcal{N}(0, \Sigma(x))$. Here, the stochasticity of y is independent of the neighborhood ϵ. To address scenarios such as these, we introduce a new random variable $\varepsilon \sim \mathcal{N}(0, \Sigma(x))$, which is heteroscedastic and is independent of $f_\theta(x+\epsilon)$. Indeed, ε does not depend upon the gradient or curvature of f_θ. Subsequently, we can write $\hat{y} = f_\theta(x+\epsilon) + \varepsilon$. Since ε is independent of $f_\theta(x+\epsilon)$, we can write the covariance as the sum of $\mathrm{Cov}f_\theta(x+\epsilon)$ and $\Sigma(x)$. This is possible because the sum of two independent Gaussians results in a Gaussian with the means and covariances summed. We therefore model the covariance through the gradient and curvature as well as account for the inherent stochasticity of the samples.

Algorithm 1 *Taylor Induced Covariance*

Input: $\boldsymbol{x}$: Input sample
Input: f_θ: Mean estimator
Output: $\mathrm{Cov}(\hat{Y}|X)$: Covariance prediction

```
// Parallelized using vmap
```
$\boldsymbol{J}(\boldsymbol{x}) = \mathtt{get\_jacobian\_wrt\_x}(f_\theta(\boldsymbol{x}))$
$\mathsf{H}(\boldsymbol{x}) = \mathtt{get\_hessian\_wrt\_x}(f_\theta(\boldsymbol{x}))$
$k_1(\boldsymbol{x}), k_2(\boldsymbol{x}), k_3(\boldsymbol{x}) = g_\Theta(\boldsymbol{x})$

```
// Jacobian term
// J.shape = (out_dims × in_dims)
```
jacobian $= k_1(\boldsymbol{x})\boldsymbol{J}(\boldsymbol{x})\boldsymbol{J}(\boldsymbol{x})^T$

```
// Hessian term
// H.shape = (out_dims × in_dims × in_dims)
```
$\mathcal{H}_{i,j} = k_2(\boldsymbol{x})\,\mathtt{Trace}\left(\mathsf{H}_{i,:,:}(\boldsymbol{x})\,\mathsf{H}_{j,:,:}(\boldsymbol{x})\right)$
hessian $= \mathcal{H}$

```
// Independent term
```
independent $= k_3(\boldsymbol{x})$

```
// Taylor Induced Covariance
```
TIC $=$ jacobian $+$ hessian $+$ independent

```
// Train using negative loglikelihood
```
return TIC

We learn the covariance of ε through $k_3(\boldsymbol{x}) \in \mathbb{R}^{n \times n}$, a learnable positive definite matrix which is optimized via the covariance estimator $g_\Theta(x)$. The final expression for TIC is

$$\mathrm{Cov}(\hat{Y}|X=x) \approx k_1(\boldsymbol{x})\boldsymbol{J}(\boldsymbol{x})\boldsymbol{J}(\boldsymbol{x})^T + \mathcal{H} + k_3(\boldsymbol{x}) . \quad (7)$$

The covariance estimator $g_\Theta(x)$ predicts $k_1(\boldsymbol{x}), k_2(\boldsymbol{x})$ and $k_3(\boldsymbol{x})$, where $k_1(\boldsymbol{x}), k_2(\boldsymbol{x})$ are positive scalars. We enforce $k_3(x)$ to be positive definite by predicting an unconstrained matrix and multiplying it with its transpose, similarly to previous work. The covariance estimator is jointly optimized with the mean and is trained to minimize the negative log-likelihood by substituting Eq. 7 into Eq. 1.

3.5. Discussion

At first, the use of the Hessian in TIC resembles its use in optimization (Gilmer et al., 2022; Kingma & Ba, 2015). The Cramer-Rao bound (Ly et al., 2017) links the variance of a parametric estimator with its inverse Fisher information. However, the Fisher information computes the Hessian with respect to the *parameters*, measuring its sensitivity over all samples. By contrast, the Hessian in our formulation is computed with respect to the input, allowing us to model heteroscedasticity.

We incorporate TIC within the negative log-likelihood formulation and do not employ covariance specific regularization. Similar to previous works (Kendall & Gal, 2017; Skafte et al., 2019; Seitzer et al., 2022; Stirn et al., 2023; Immer et al., 2023), our method is an approximation without theoretical guarantees. This approximation results from a heuristic interpretation of the neigborhood, as well as the
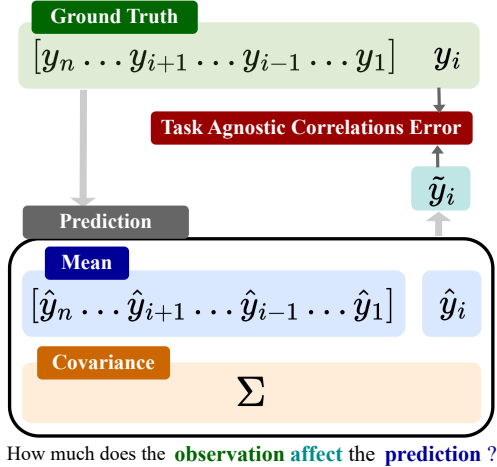
How much does the **observation affect** the **prediction** ?

Figure 2: *Task Agnostic Correlations (TAC).* We propose the TAC metric for covariance evaluation. Given the ground truth $\boldsymbol{y}$, predicted mean $\hat{\boldsymbol{y}}$ and covariance $\boldsymbol{\Sigma}$, TAC quantifies the improvement in the predicted mean given partial observations of the ground truth. TAC uses conditioning of the normal distribution to directly assess the covariance.

use of the second order taylor polynomial. However, our experimental evaluations show that TIC provides accurate covariance estimates and works well in practice.

Limitations. The computational complexity in TIC arises from computing the Hessian. While determining this for a general network architecture is non-trivial, computing the Hessian for a function that maps $x \in \mathbb{R}^m$ to $y \in \mathbb{R}^n$ has a complexity of $O(nm^3)$ (Yao et al., 2020), which is large. There are multiple possible ways to mitigate this in practice. The simplest approach would be to use parallelization, which we provide in our code. The second approach would be to use a smaller, proxy model in place of a large model (which could be retained for mean estimation). This smaller model could be trained through a student-teacher setup using techniques from knowledge distillation (Gou et al., 2021). The reduced parameter count would decrease the computational requirements of the Hessian. An interesting direction for future research would be to find useful approximations of the Hessian with respect to the input, similarly to research in optimization which approximates the Hessian with respect to the parameters.

4. Task Agnostic Correlations (TAC)

How can we evaluate covariance estimation in the absence of ground-truth annotation? Existing methods (Kendall & Gal, 2017; Seitzer et al., 2022; Stirn et al., 2023) use metrics such as likelihood scores and the mean squared error for evaluation. However, these methods are skewed towards learning the mean; a perfect mean estimator $f_\theta(x)$ would re-

---

Algorithm 2 *Task Agnostic Correlations*

Input: $\boldsymbol{y}$: Ground truth, $\hat{\boldsymbol{y}}$: Target prediction
Input: $\text{Cov}(\hat{Y}|X)$: Covariance prediction
Output: TAC error

dimensions = get_dimensions($\hat{\boldsymbol{y}}$)
TAC = zeros(shape=dimensions)

for *i in dimensions* **do**
    // Observe all but one dimension
    obs_dim = set(dimensions) - set(i)
    hidden_dim = i
    // Conditioning the normal distribution
    $\boldsymbol{\Sigma}_{22} = \text{Cov}(\hat{Y}|X)$[obs_dim, obs_dim]
    $\boldsymbol{\Sigma}_{12} = \text{Cov}(\hat{Y}|X)$[hidden_dim, obs_dim]
    $\tilde{\boldsymbol{y}} = \hat{\boldsymbol{y}}$[hidden_dim] $+ (\boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1}\,(\boldsymbol{y}$[obs_dim] $- \hat{\boldsymbol{y}}$[obs_dim]$))$
    // Error between updated and true value
    TAC [i] $= |\,\tilde{\boldsymbol{y}} - \boldsymbol{y}$[hidden_dim]$\,|$

return TAC.mean()

---

sult in zero mean squared error, while log-likelihood scores put greater emphasis on the determinant of the covariance and do not directly assess correlations. Other metrics such as the Conditional Marginal Likelihood (CML) (Lotfi et al., 2022) are a measure of generalization. Therefore, we argue for the use of a much more direct method to assess the covariance. Specifically, we reason that the goal of estimating the covariance is to encode the relation between the target variables. *Therefore, partially observing a set of correlated targets should improve the prediction of the hidden targets since by definition the covariance encodes this correlation.* As an example, if P and Q are correlated, then observing P should improve our estimate of Q. Hence, we propose a new metric that evaluates the accuracy of correlations, which we call the *Task Agnostic Correlations*, (Figure 2).

4.1. Algorithm

Formally, given an n-dimensional target prediction $\hat{\boldsymbol{y}}$, the ground truth $\boldsymbol{y}$, and the predicted covariance $\text{Cov}(\hat{Y}|X{=}\boldsymbol{x})$, we define the TAC error as $\sum_i |y_i - \tilde{y}_i|/n$, where $\tilde{y}_i$ is the updated mean obtained after conditioning $\mathcal{N}(\tilde{y}_i, \text{Cov}(\hat{Y}|X) \mid \boldsymbol{y}_{j\neq i}, \boldsymbol{x})$. For each prediction $\hat{y}_i$, we obtain a revised estimate $\tilde{y}_i$ by conditioning it over the ground truth of the remaining variables $\boldsymbol{y}_{i\neq j}$. We measure the absolute error of this revised estimate against the ground truth of the unobserved variable and repeat for all i. An accurate estimate of $\text{Cov}(\hat{Y}|X{=}\boldsymbol{x})$ will decrease the error whereas an incorrect estimate will cause an increase. We describe this in Algorithm 2.

4.2. Discussion

This evaluation bears resemblance with *leave-one-out*, where we observe one $\tilde{y}_i$ given other observations $\boldsymbol{y}_{j\neq i}$.

(a) Sinusoidal with $y = 5 \sin (2\pi x)$ and $\sigma(x) = |x|$

(b) Sinusoidal with $y = |x| \sin (2\pi x)$ and $\sigma(x) = |x|$

(c) Sinusoidal with $y = (5 - |x|) \sin (2\pi x)$ and $\sigma(x) = |x|$
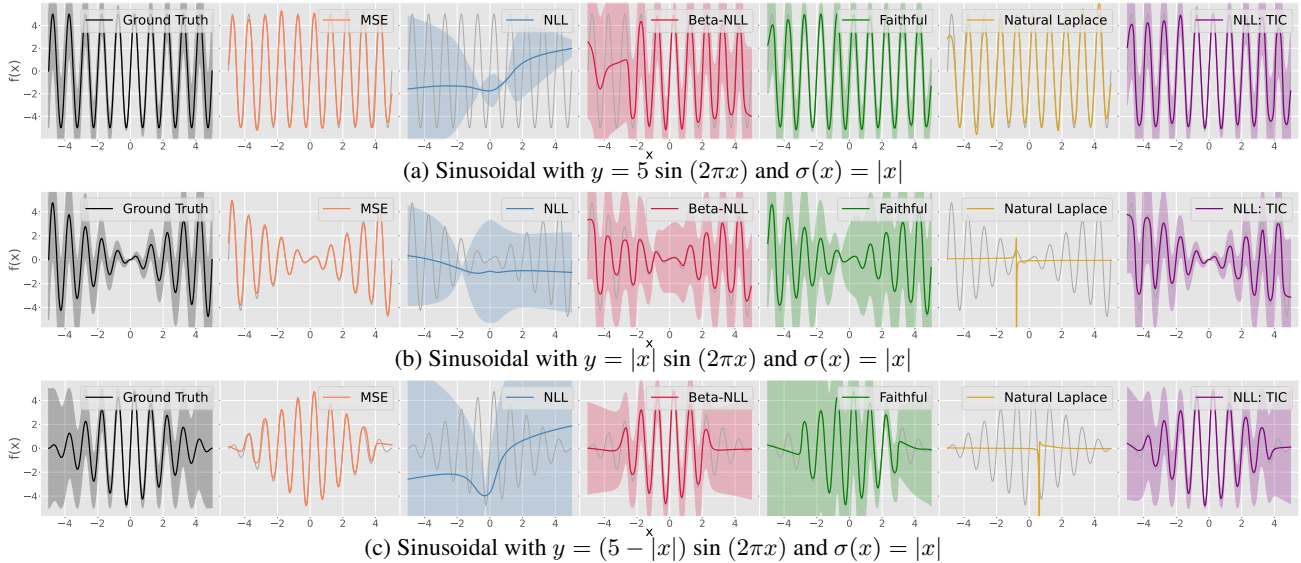
Figure 3: *Univariate.* We perform experiments on three different sinusoidals, showing that incorporating the gradient and curvature of the predicted mean results in accurate variance estimation. The TIC parameterization also results in an improved convergence of the negative log-likelihood.

While *leave-one-out* can be generalized to *leave-k-out*, we do not observe any change in the evaluation trend. A method having lower *leave-one-out* also has a lower *leave-k-out* error. Moreover, *leave-k-out* requires taking $\binom{n}{k}$ combinations, which is significantly higher than taking n combinations in *leave-one-out*. This motivates the use of the *leave-one-out*.

We highlight that this metric is agnostic of downstream tasks involving covariance estimation. *We also note that TAC and the log-likelihood are complementary: while log-likelihood is a measure of optimization, TAC is a measure of accuracy of the learnt correlations.* Hence, we use TAC as an additional metric for all multivariate experiments.

5. Experiments

The goal of this paper is to improve covariance estimation in deep heteroscedastic regression. Therefore, we specifically focus on multivariate *outputs*, and readdress several existing experimental designs. Our synthetic experiments consist of learning a univariate sinusoidal and a multivariate distribution. We conduct our real-world experiments on the UCI regression repository and the MPII, LSP 2D human pose estimation datasets.

Our baselines consist of different (co-)variance models in deep heteroscedastic regression. These include the negative log-likelihood (Dorta et al., 2018), and its variants: β-NLL (Seitzer et al., 2022), Faithful Heteroscedastic Regression (Stirn et al., 2023), and Natural Laplace (Immer et al., 2023) (univariate). We refer to the diagonal covariance (Kendall & Gal, 2017) and the TIC formulation as *NLL-Diagonal*

and *NLL-TIC* since they are optimized using the negative log-likelihood. We take care to provide a fair comparison; all methods are randomly initialized with the same mean and covariance estimators, with each method having its own learning rate scheduler. Furthermore, the batching and ordering of samples is the same for all methods. We train our method and all baselines for 100 epochs using a learning rate scheduler which reduces the learning rate on plateau. Unless specified, we use simple fully connected layers with batch normalization as our network architecture.

Since covariance estimation lacks direct supervision, we do not make training and evaluation splits of the dataset to increase the number of samples. While this may seem questionable, we reason that the covariance is a measure of correlation as well as variance. If too few samples are provided for training then the resulting covariance is nearly singular. Moreover, existing work (Skafte et al., 2019; Seitzer et al., 2022; Stirn et al., 2023) show that the negative log-likelihood is prone to sub-optimal convergence and does not overfit the training samples. Finally, our evaluation remains fair since our experimental methodology is the same for all.

5.1. Synthetic Data

Univariate. We repeat the experiments of Seitzer et al. (2022) with a major revision. First, we introduce heteroscedasticity and substantially increase the variance of the samples. Second, we simulate different sinusoidals having constant and varying amplitudes. We draw 50,000 samples and train a fully-connected network with batch normalization (Ioffe & Szegedy, 2015) for 100 epochs. Our results
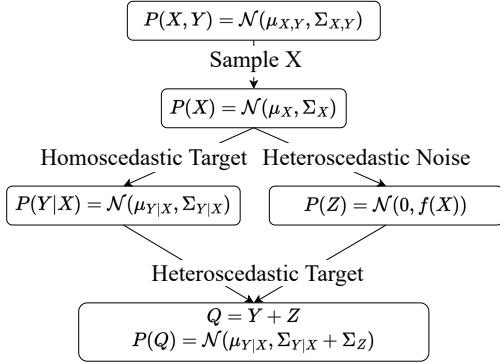
$$P(X, Y) = \mathcal{N}(\mu_{X,Y}, \Sigma_{X,Y})$$

Sample X

$$P(X) = \mathcal{N}(\mu_X, \Sigma_X)$$

Homoscedastic Target   Heteroscedastic Noise

$$P(Y|X) = \mathcal{N}(\mu_{Y|X}, \Sigma_{Y|X})$$ $$P(Z) = \mathcal{N}(0, f(X))$$

Heteroscedastic Target

$$Q = Y + Z$$
$$P(Q) = \mathcal{N}(\mu_{Y|X}, \Sigma_{Y|X} + \Sigma_Z)$$

Figure 4: *Multivariate Schematic.* We present a simple method to simulate heteroscedastic data. We first randomly sample the input x, which in turn is used to sample an initial target y. We then add sample-dependent noise z, giving us the target q, which the network is required to learn.

Table 1: *Multivariate Results.* We compare the log-likelihood value for all methods. We skip NLL-Diagonal and β-NLL, both of which have very low likelihoods since they assume a diagonal covariance.

Method	Dim: 4	6	8	10	12	14	16	18	20
MSE	-10.1	-16.4	-23.1	-30.9	-36.1	-41.6	-49.2	-53.2	-66.6
NLL	-8.2	-14.9	-19.9	-26.6	-34.2	-42.7	-46.6	-60.9	-67.2
Faithful	-8.7	-14.7	-20.3	-27.3	-32.4	-40.2	-48.6	-55.4	-69.0
NLL-TIC	**-7.6**	**-11.7**	**-15.8**	**-19.9**	**-23.3**	**-26.8**	**-30.3**	**-34.2**	**-39.7**

are shown in Figure 3. We observe that in the absence of direct supervision, the negative log-likelihood incorrectly overestimates the variance since it does not represent the randomness of the predicted mean. Furthermore, both β-NLL and Faithful are susceptible to incorrect variance predictions because the methods regularize the variance, which compromises on variance fits. While Natural Laplace fits the constant amplitude sinusoidal, the method results in unstable optimization for sinusoidals of varying amplitude.

Multivariate. We propose a new experiment for multivariate analysis to study heteroscedastic covariance. We let X, Y be jointly distributed and sample x from this distribution. Subsequently, we sample y conditioned on x. To simulate heteroscedasticity, we draw samples from Z, a zero mean random variable whose covariance $\Sigma_Z = \text{diag}(\sqrt{|x|})$ depends on x. Since Y and Z are independent given X, their sum also satisfies the normal distribution $Q|X \sim \mathcal{N}(\mu_{Y|X}, \Sigma_{Y|X} + \Sigma_{Z|X})$. Therefore, the goal of this experiment is to model the mean and the heteroscedastic covariance of $Q|X$ given samples (x, q). The schematic for our experimental design is shown in Fig. 4.

We vary the dimensionality of x and q from 4 to 20 in steps of 2, and report the mean and standard deviation over ten
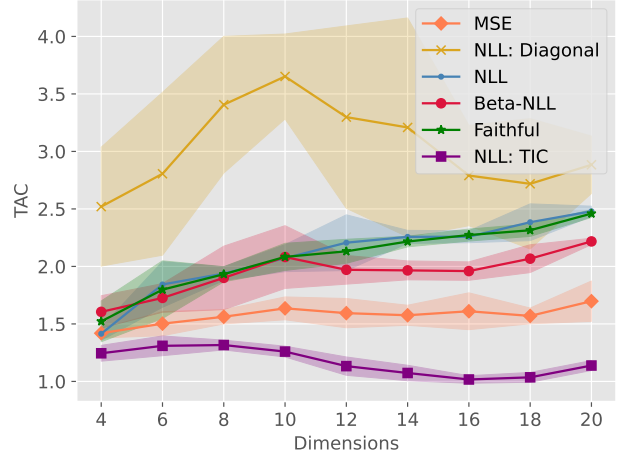

Figure 5: *Multivariate Results.* We plot the Task Agnostic Correlations (TAC) metric mean and standard deviation for all methods from dimensions 4 to 20. The gap between TIC and the baselines widens as the dimensionality increases.

trials for each dimension. Depending on the dimensionality, we draw from 4000 up to 20000 samples and report our results using TAC (Fig. 5) and the log-likelihood (Table 1). We observe two trends in Fig. 5: First, as the dimensionality of the samples increases, the gap between TIC and the other methods widens. This is because, with increasing dimensionality, the number of free parameters to estimate in the covariance matrix grows quadratically. An increase in parameters typically requires a non-linear growth in the number of samples for robust fitting. As a result, the difficulty of mapping the input to a positive definite matrix increases with dimensionality. Second, we note that TIC allows for better convergence in comparison to the naive parameterization of the covariance in NLL.

5.2. UCI Regression

We perform our analysis on twelve multivariate UCI regression (Dua & Graff, 2017) datasets, which have been used in previous work on negative log-likelihood (Stirn et al., 2023; Seitzer et al., 2022). Nevertheless, our goal of studying covariance estimation in deep heteroscedastic regression requires us to use a different pre-processing, as many of the datasets have univariate or low-dimensional targets.

Specifically, for each dataset we randomly allocate 25% of the features as input and the remaining 75% features as multivariate targets at run-time. Some combinations of input variables may fare poorly at predicting the target variables. However, this is an interesting challenge for the covariance estimator, which needs to learn the underlying correlations even in unfavorable circumstances. Moreover, random splitting also allows our experiments to remain unbiased as we

Table 2: *UCI Regression.* We perform ten trials over all the datasets and report the TAC error and the log-likelihood. TIC outperforms all the baselines on ten out of twelve datasets in terms of TAC error, and outperforming on all but one dataset in terms of the likelihood error. Additionally, the TIC parameterization (NLL-TIC) results in improved convergence of the negative log-likelihood.

(a) Task Agnostic Correlations (TAC) Metric

Method	Abalone	Air	Appliances	Concrete	Electrical	Energy	Turbine	Naval	Parkinson	Power	Red Wine	White Wine
MSE	2.54	4.31	1.79	6.15	7.91	4.40	4.74	0.56	2.32	6.01	5.97	6.32
NLL-Diagonal	5.49	8.03	11.71	7.86	10.06	7.12	7.07	5.01	8.56	8.16	7.96	8.44
NLL	3.28	3.42	2.41	4.16	7.14	5.10	3.40	0.25	1.86	6.22	5.81	7.26
β-NLL	2.85	5.67	4.89	7.21	8.41	6.17	5.03	1.06	5.48	6.73	6.96	7.08
Faithful	2.96	3.27	1.79	3.93	7.36	2.90	3.29	**0.20**	**1.68**	5.81	5.74	6.89
NLL-TIC	**1.83**	**2.27**	**1.39**	**2.82**	**4.89**	**2.34**	**2.40**	0.28	2.54	**3.87**	**4.05**	**4.60**

(b) Log Likelihood Metric. We skip NLL-Diagonal and β-NLL which have very low likelihoods since the methods assume diagonal covariance. We remind the reader that the datasets are adapted for covariance estimation

Method	Abalone	Air	Appliances	Concrete	Electrical	Energy	Turbine	Naval	Parkinson	Power	Red Wine	White Wine
MSE	-60.7	-231.5	-99.6	-238.3	-494.6	-169.6	-230.8	-20.9	-154.0	-295.6	-305.8	-338.15
NLL	-8.5×10^3	-53.32	-84.5	-83.6	-57.9	-55.8	-27.1	4.1	-1.5×10^3	-34.2	-236.0	-206.0
Faithful	-9.4×10^3	-52.1	-55.4	-80.6	-57.3	-30.8	-26.1	**7.5**	-1.2×10^3	-33.9	-434.4	-250.9
NLL-TIC	**-13.4**	**-29.3**	**-42.45**	**-22.2**	**-35.8**	**-19.1**	**-22.9**	-10.3	**-63.0**	**-27.1**	**-30.63**	**-30.1**

do not control the split of variables at any instant. For all datasets, we standardize our variables with zero mean and a variance of ten (which yields better convergence for all methods). We perform 10 trials for each dataset and report the TAC error and likelihood in Table 2.

While TIC outperforms the baselines on a majority of the datasets, we particularly focus on the Naval dataset which highlights a limitation of the TIC parameterization. We observe that TIC may not be suitable if all samples have a low degree of variance. A low degree of variance (as indicated by the likelihood) results in accurate mean fits, which implies that small gradients are being backpropagated, and in turn affecting the TIC parameterization. However, we argue that datasets with a small degree of variance may not benefit from heteroscedastic modelling.

5.3. 2D Human Pose Estimation

We introduce experiments on human pose estimation since the human pose is an organised collection of points and is naturally suited for correlation analysis (Shukla et al., 2022). Moreover, popular human pose architectures (Newell et al., 2016; Sun et al., 2019; Kreiss et al., 2019; 2021; Xu et al., 2022) are either convolutional or transformer based, presenting a viable challenge to modelling the Taylor Induced Covariance. This is because TIC assumes vector inputs $x \in \mathbb{R}^m$ and predictions $\hat{y} \in \mathbb{R}^n$, whereas popular architectures rely on input images $\mathbf{X} \in \mathbb{R}^{C \times H \times W}$ and output heatmaps $\hat{\mathbf{Y}} \in \mathbb{R}^{\#\text{joints} \times 64 \times 64}$.

We therefore perform experiments on two architectures: the Stacked Hourglass (Newell et al., 2016) and ViTPose (Xu

et al., 2022). The Stacked Hourglass is a popular method which extends the convolutional U-Net (Ronneberger et al., 2015) architecture to predict heatmaps for human pose estimation. ViTPose is a recent state-of-the-art architecture which extends vision transformers (Dosovitskiy et al., 2021) to the task of human pose estimation.

For both architectures, we use soft-argmax (Li et al., 2021b;a) to reduce the heatmap of shape $\hat{\mathbf{Y}} \in \mathbb{R}^{\#\text{joints} \times 64 \times 64}$ to a vector of shape $\mathbb{R}^{\#\text{joints} * 2}$. Next, we recursively call the hourglass module until we obtain a one-dimensional vector encoding (Shukla, 2022) for the image, which serves as the input to the covariance estimator. For ViTPose, we obtain vector embeddings from a simple residual connection involving a one-dimensional downscaling and upscaling of the features predicted by the backbone network.

We use popular single person datasets: MPII (Andriluka et al., 2014) and LSP/LSPET (Johnson & Everingham, 2010; 2011), with the latter emphasizing on poses involving sports. We perform our analysis by merging the MPII and LSP-LSPET datasets to increase the number of samples. We train the pose estimator using the Adam optimizer with a 'ReduceLROnPlateau' learning rate scheduler for 100 epochs with the learning rate set to 1e-3. We use two augmentations: Shift+Scale+Rotate and horizontal flip. We refer the reader to the code for implementation details.

In addition to the likelihood, we continue to use TAC as our metric since for single person estimation, the scale of the person is fixed. Hence, TAC is highly correlated with PCKh/PCK, the preferred metric for multi-person multi-scale pose estimation. We perform five trials and report
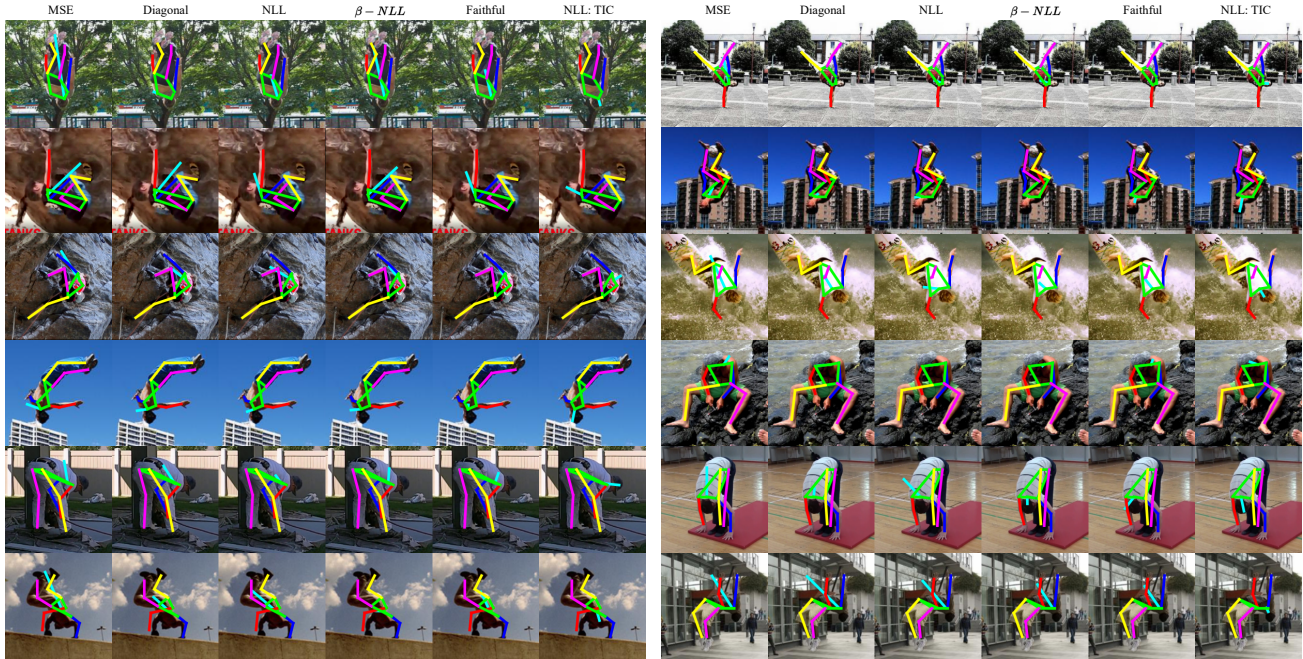
Figure 6: *Human Pose Visualization*. We show that the Taylor Induced Covariance (TIC) parameterization results in a more accurate pose estimation for complex poses. As an example, we visualize the updated prediction for the head conditioned on observing the ground truth for the remaining joints. We show that TIC accurately predicts the location for the head for complex poses in comparison to all other methods.

Table 3: *Human Pose Results - ViTPose architecture*. We report the TAC error for each joint along with the average across all joints. Additionally, we report the likelihood score for all methods. We show that NLL-TIC outperforms baselines across all joints and successfully scales to convolutional and transformer based architectures.

Method	head	neck	lsho	lelb	lwri	rsho	relb	rwri	lhip	lknee	lankl	rhip	rknee	rankl	**Avg: TAC**	**Avg: LL**
MSE	6.14	7.12	7.05	8.60	10.56	6.78	8.33	10.35	7.67	7.90	9.69	7.40	7.82	9.72	8.22 ± 0.05	-973.7 ± 8.6
NLL-Diagonal	14.88	12.33	12.38	12.25	13.87	11.36	11.39	13.54	10.42	11.49	17.84	9.84	11.46	18.28	12.95 ± 1.36	-204.5 ± 177.0
NLL	4.97	5.76	4.86	4.58	6.62	4.48	4.36	6.59	5.97	5.80	7.88	5.78	5.68	7.81	5.80 ± 0.07	-91.61 ± 1.26
β-NLL	12.63	11.22	11.63	12.06	13.95	10.45	11.21	13.63	10.84	11.45	16.23	10.02	11.09	15.97	12.31 ± 0.31	-4.2e3 ± 1.6e3
Faithful	5.25	5.86	4.97	4.68	6.77	4.60	4.45	6.75	6.10	5.98	7.90	5.94	5.82	7.89	5.93 ± 0.03	-91.77 ± 0.11
NLL-TIC	**3.97**	**5.38**	**4.47**	**4.29**	**6.06**	**4.12**	**4.08**	**5.89**	**5.45**	**5.24**	**7.03**	**5.25**	**5.09**	**6.97**	**5.23 ± 0.03**	**-80.31 ± 0.39**

our results for the *ViTPose* backbone in Table 3. We report results on the Stacked Hourglass backbone in the appendix. Our experiments show that TIC outperforms all baselines, especially on challenging joints.

6. Conclusion

We improved covariance estimation in deep heteroscedastic regression through two contributions. With the Taylor Induced Covariance (TIC), we parameterize the predicted covariance to capture the randomness of the predicted mean through its gradient and curvature. With the Task Agnostic Correlations (TAC) metric, we have proposed a novel metric for covariance evaluation by leveraging conditioning of the normal distribution to quantify the accuracy of learnt correlations. Our extensive experiments across multiple tasks

have shown that, not only does TIC outperform the state of the art in learning the covariance, it also facilitates an improved convergence of the negative log-likelihood.

Acknowledgements

Impact Statement

This paper presents work whose goal is to advance the field of machine learning and its applications. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here.

References

Andriluka, M., Pishchulin, L., Gehler, P., and Schiele, B. 2d human pose estimation: New benchmark and state of the art analysis. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2014.

Bertoni, L., Kreiss, S., and Alahi, A. Monoloco: Monocular 3d pedestrian localization and uncertainty estimation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 6861–6871, 2019.

Dorta, G., Vicente, S., Agapito, L., Campbell, N. D., and Simpson, I. Structured uncertainty prediction networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5477–5485, 2018.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., and Houlsby, N. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=YicbFdNTTy.

Dua, D. and Graff, C. UCI machine learning repository, 2017. URL http://archive.ics.uci.edu/ml.

Evans, M. J. and Rosenthal, J. S. *Probability and statistics: The science of uncertainty*. Macmillan, 2004. URL https://www.utstat.toronto.edu/mikevans/jeffrosenthal/book.pdf.

Gal, Y. and Ghahramani, Z. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning*, pp. 1050–1059. PMLR, 2016.

Gal, Y., Islam, R., and Ghahramani, Z. Deep bayesian active learning with image data. *arXiv preprint arXiv:1703.02910*, 2017.

Gilmer, J., Ghorbani, B., Garg, A., Kudugunta, S., Neyshabur, B., Cardoze, D., Dahl, G. E., Nado, Z., and Firat, O. A loss curvature perspective on training instabilities of deep learning models. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=OcKMT-36vUs.

Gou, J., Yu, B., Maybank, S. J., and Tao, D. Knowledge distillation: A survey. *International Journal of Computer Vision*, 129(6):1789–1819, 2021.

Gundavarapu, N. B., Srivastava, D., Mitra, R., Sharma, A., and Jain, A. Structured aleatoric uncertainty in human pose estimation. In *CVPR Workshops*, volume 2, 2019.

Houlsby, N., Huszár, F., Ghahramani, Z., and Lengyel, M. Bayesian active learning for classification and preference learning. *stat*, 1050:24, 2011.

Immer, A., Palumbo, E., Marx, A., and Vogt, J. E. Effective bayesian heteroscedastic regression with deep neural networks. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=A6EquH0enk.

Ioffe, S. and Szegedy, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. pp. 448–456, 2015. URL http://jmlr.org/proceedings/papers/v37/ioffe15.pdf.

Johnson, S. and Everingham, M. Clustered pose and nonlinear appearance models for human pose estimation. In *Proceedings of the British Machine Vision Conference*, 2010. doi:10.5244/C.24.12.

Johnson, S. and Everingham, M. Learning effective human pose estimation from inaccurate annotation. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2011.

Kendall, A. and Gal, Y. What uncertainties do we need in bayesian deep learning for computer vision? In *Advances in neural information processing systems*, pp. 5574–5584, 2017.

Kingma, D. P. and Ba, J. Adam: a method for stochastic optimization. In *International Conference on Learning Representations*, 2015.

Kreiss, S., Bertoni, L., and Alahi, A. Pifpaf: Composite fields for human pose estimation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 11977–11986, 2019.

Kreiss, S., Bertoni, L., and Alahi, A. Openpifpaf: Composite fields for semantic keypoint detection and spatio-temporal association. *IEEE Transactions on Intelligent Transportation Systems*, 23(8):13498–13511, 2021.

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

Le, Q. V., Smola, A. J., and Canu, S. Heteroscedastic gaussian process regression. In *Proceedings of the 22nd international conference on Machine learning*, pp. 489–496, 2005.

Li, J., Chen, T., Shi, R., Lou, Y., Li, Y.-L., and Lu, C. Localization with sampling-argmax. *Advances in Neural Information Processing Systems*, 34:27236–27248, 2021a.

Li, J., Xu, C., Chen, Z., Bian, S., Yang, L., and Lu, C. Hybrik: A hybrid analytical-neural inverse kinematics solution for 3d human pose and shape estimation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 3383–3393, 2021b.

Liu, K., Ok, K., Vega-Brown, W., and Roy, N. Deep inference for covariance estimation: Learning gaussian noise models for state estimation. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1436–1443, 2018. doi: 10.1109/ICRA.2018.8461047.

Lotfi, S., Izmailov, P., Benton, G., Goldblum, M., and Wilson, A. G. Bayesian model selection, the marginal likelihood, and generalization. In *International Conference on Machine Learning*, pp. 14223–14247. PMLR, 2022.

Lu, C. and Koniusz, P. Few-shot keypoint detection with uncertainty learning for unseen species. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 19416–19426, 2022.

Ly, A., Marsman, M., Verhagen, J., Grasman, R. P., and Wagenmakers, E.-J. A tutorial on fisher information. *Journal of Mathematical Psychology*, 80:40–55, 2017.

Nakka, K. K. and Salzmann, M. Understanding pose and appearance disentanglement in 3d human pose estimation. *arXiv preprint arXiv:2309.11667*, 2023.

Newell, A., Yang, K., and Deng, J. Stacked hourglass networks for human pose estimation. In Leibe, B., Matas, J., Sebe, N., and Welling, M. (eds.), *Computer Vision – ECCV 2016*, pp. 483–499, Cham, 2016. Springer International Publishing. ISBN 978-3-319-46484-8.

Nix, D. A. and Weigend, A. S. Estimating the mean and variance of the target probability distribution. In *Proceedings of 1994 ieee international conference on neural networks (ICNN'94)*, volume 1, pp. 55–60. IEEE, 1994.

Petersen, K. B. and Pedersen, M. S. The matrix cookbook, nov 2012. URL http://www2.compute.dtu.dk/pubdb/pubs/3274-full.html. Version 20121115.

Ronneberger, O., Fischer, P., and Brox, T. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pp. 234–241. Springer, 2015.

Russell, R. L. and Reale, C. Multivariate uncertainty in deep learning. *IEEE Transactions on Neural Networks and Learning Systems*, 33(12):7937–7943, 2021.

Seitzer, M., Tavakoli, A., Antic, D., and Martius, G. On the pitfalls of heteroscedastic uncertainty estimation with probabilistic neural networks. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=aPOpXlnV1T.

Shukla, M. Bayesian uncertainty and expected gradient length-regression: Two sides of the same coin? In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 2367–2376, 2022.

Shukla, M., Roy, R., Singh, P., Ahmed, S., and Alahi, A. Vl4pose: Active learning through out-of-distribution detection for pose estimation. In *Proceedings of the 33rd British Machine Vision Conference*, number CONF. BMVA Press, 2022.

Shynk, J. J. *Probability, random variables, and random processes: theory and signal processing applications*. John Wiley & Sons, 2012.

Simpson, I. J., Vicente, S., and Campbell, N. D. Learning structured gaussians to approximate deep ensembles. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 366–374, 2022.

Skafte, N., Jørgensen, M., and Hauberg, S. Reliable training and estimation of variance networks. *Advances in Neural Information Processing Systems*, 32, 2019.

Stirn, A., Wessels, H., Schertzer, M., Pereira, L., Sanjana, N., and Knowles, D. Faithful heteroscedastic regression with neural networks. In *International Conference on Artificial Intelligence and Statistics*, pp. 5593–5613. PMLR, 2023.

Sun, K., Xiao, B., Liu, D., and Wang, J. Deep high-resolution representation learning for human pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

Tekin, B., Marquez-Neila, P., Salzmann, M., and Fua, P. Learning to fuse 2d and 3d image cues for monocular body pose estimation. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

Xu, Y., Zhang, J., Zhang, Q., and Tao, D. Vitpose: Simple vision transformer baselines for human pose estimation. *Advances in Neural Information Processing Systems*, 35: 38571–38584, 2022.

Yao, Z., Gholami, A., Keutzer, K., and Mahoney, M. W. Py-hessian: Neural networks through the lens of the hessian. In *2020 IEEE international conference on big data (Big data)*, pp. 581–590. IEEE, 2020.
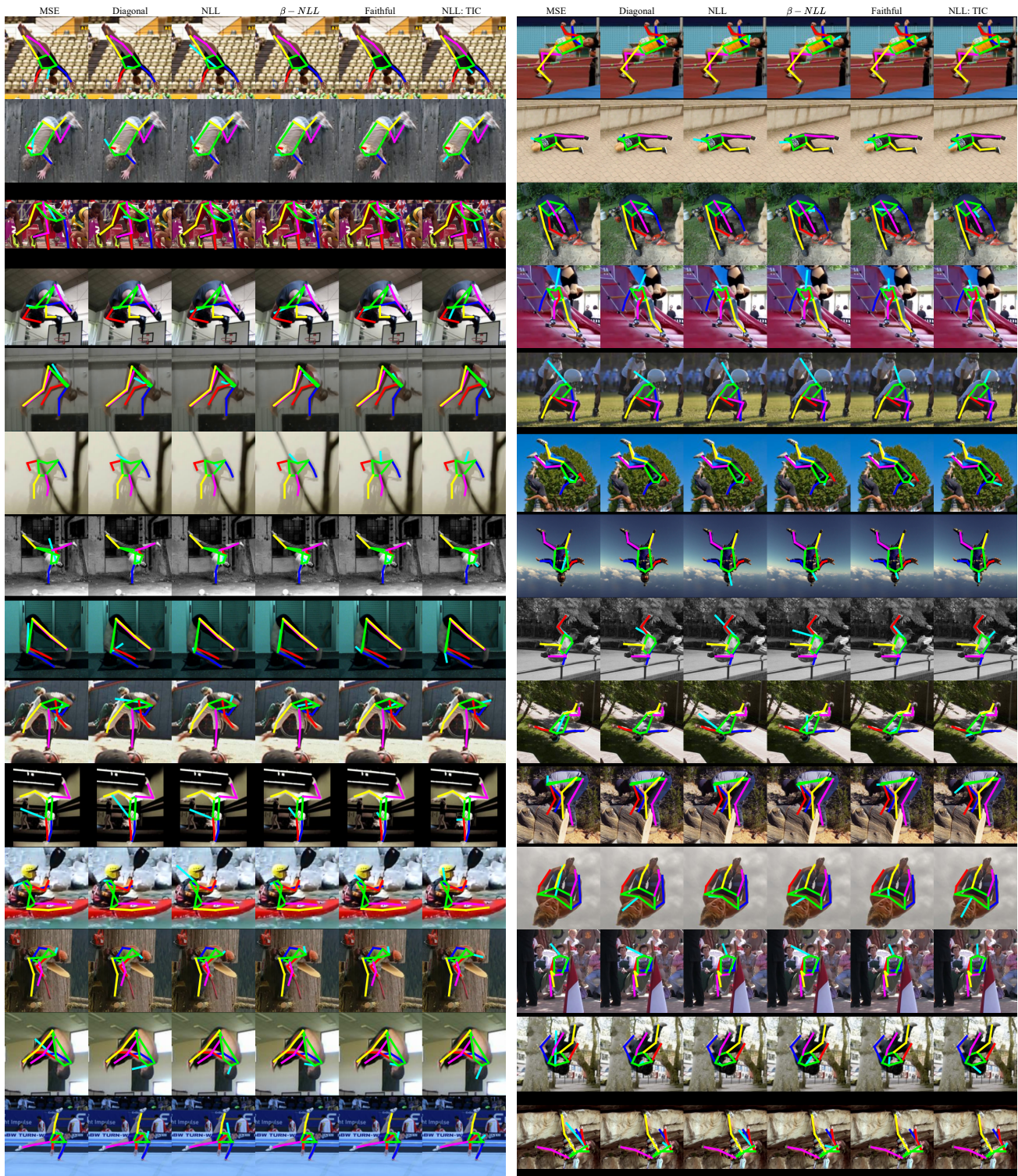
A. Additional Visualizations

Figure 7: We show additional visualizations to highlight the updated prediction for the head conditioned on observing the ground truth for the remaining joints. TIC accurately updates the location for the head based on successfully learning the correlations underlying the joints.

13

B. Human Pose Estimation - Hourglass

Table 4: *Human Pose Results - Stacked Hourglass architecture*. We report the TAC error for each joint along with the average across all joints. Additionally, we report the likelihood score for all methods. We show that NLL-TIC outperforms baselines across all joints and successfully scales to convolutional and transformer based architectures.

Method	head	neck	lsho	lelb	lwri	rsho	relb	rwri	lhip	lknee	lankl	rhip	rknee	rankl	**Avg: TAC**	**Avg: LL**
MSE	5.53	7.88	7.31	8.73	10.52	7.01	8.41	10.19	8.43	8.53	10.53	8.13	8.37	10.58	8.58 ± 0.21	-1018.6 ± 31.2
NLL-Diagonal	5.36	7.23	6.95	8.17	10.01	6.48	7.79	9.73	8.11	8.30	11.12	7.75	8.17	11.20	8.32 ± 3.19	-96.3 ± 127.2
NLL	4.48	6.81	5.38	5.19	7.13	5.11	4.86	6.89	6.62	6.35	8.45	6.43	6.17	8.40	6.31 ± 0.21	-93.0 ± 1.76
β-NLL	4.63	7.14	6.74	8.23	9.98	6.43	7.92	9.65	8.01	8.13	10.12	7.71	7.93	10.19	8.06 ± 0.17	-97.5 ± 0.25
Faithful	5.13	6.36	5.32	4.94	7.18	4.96	4.72	6.85	6.67	6.29	8.39	6.36	6.22	8.37	6.27 ± 0.06	-91.8 ± 0.22
NLL-TIC	**3.76**	**5.98**	**4.80**	**4.64**	**6.34**	**4.46**	**4.41**	**6.12**	**6.09**	**5.82**	**7.59**	**5.79**	**5.63**	**7.55**	**5.64** ± 0.03	**-88.6** ± 0.08