A THEORETICAL DETAILS

A.1 FULL FUNCTIONAL DERIVATIVES

Full functional derivatives of our NFE are:

$$\mathcal{L}_{\rho,\gamma}(\hat{\mu},\hat{\Lambda}) \approx \int_{\mathcal{X}} p(x)\rho\left[\frac{1}{2}\hat{\Lambda}(x)\left(y(x)-\hat{\mu}(x)\right)^2 - \frac{1}{2}\log\hat{\Lambda}(x)\right] + (1-\rho)\left[\gamma||\nabla\hat{\mu}(x)||_2^2 + (1-\gamma)||\nabla\hat{\Lambda}(x)||_2^2\right]dx$$

$$\begin{cases} \frac{\delta\mathcal{L}}{\delta\hat{\mu}} &= p(x)\rho\hat{\Lambda}(x)(\hat{\mu}(x)-y(x)) - 2(1-\rho)\gamma\Delta\hat{\mu}(x) \\ \frac{\delta\mathcal{L}}{\delta\hat{\Lambda}} &= \frac{p(x)\rho}{2}\left[(y(x)-\hat{\mu}(x))^2 - \frac{1}{\hat{\Lambda}(x)}\right] - 2(1-\rho)(1-\gamma)\Delta\hat{\Lambda}(x) \end{cases} \tag{11}$$

After setting equal to zero we arrive at

$$\begin{cases} \hat{\Lambda}^*(x)(\hat{\mu}^*(x)-y(x)) = 2\left(\frac{1-\rho}{\rho}\right)\gamma\frac{\Delta\hat{\mu}^*(x)}{p(x)} \\ (y(x)-\hat{\mu}^*(x))^2 = \frac{1}{\hat{\Lambda}^*(x)} + 4\left(\frac{1-\rho}{\rho}\right)(1-\gamma)\frac{\Delta\hat{\Lambda}^*(x)}{p(x)} \end{cases} \tag{12}$$

A.2 PROOFS

Proposition 1. *Assuming there exists twice differentiable functions $\mu : \mathbb{R}^d \to \mathbb{R}, \Lambda : \mathbb{R}^d \to \mathbb{R}_{>0}$, the following properties hold*

- *i In the absence of regularization ($\rho = 1$), there are no solutions to the NFE.*
- *ii In the absence of data ($\rho = 0$), there is no unique solution to the NFE.*
- *iii There are no valid solutions to the NFE if $\rho \in (0,1), \gamma = 1$. Some regularization on precision function is needed for a solution to potentially exist.*

Proof. Without loss of generality, we consider a uniform $p(x)$ and drop it from the equations.

(i) When $\rho = 1$ the necessary conditions for an optima are

$$\begin{cases} \hat{\Lambda}^*(x)(\hat{\mu}^*(x)-y(x)) = 0 \\ (\hat{\mu}^*(x)-y(x))^2 = \frac{1}{\hat{\Lambda}^*(x)} \end{cases} \tag{13}$$

$$\implies \begin{cases} \hat{\Lambda}^*(x)(\hat{\mu}^*(x)-y(x)) = 0 \\ \hat{\Lambda}^*(x)(\hat{\mu}^*(x)-y(x))^2 = 1 \end{cases} \tag{14}$$

$$\implies \begin{cases} \hat{\Lambda}^*(x)(\hat{\mu}^*(x)-y(x)) = 0 \\ 0 \times (\hat{\mu}^*(x)-y(x)) = 1 \end{cases} \tag{15}$$

$$\implies 0 = 1 \tag{16}$$

which is a contradiction and there cannot exist μ, Λ that are solutions.

(ii) When $\rho = 0$ the integral we seek to maximize is:

$$\mathcal{L}_{\rho,\gamma}(\hat{\mu},\hat{\Lambda}) = \int_{\mathcal{X}} \rho \int_{\mathcal{Y}} p(y\,|\,x)\left[\frac{1}{2}\hat{\Lambda}(x)\left(y-\hat{\mu}(x)\right)^2 - \frac{1}{2}\log\hat{\Lambda}(x)\right]dy + (1-\rho)\left[\gamma||\nabla\hat{\mu}(x)||_2^2 + (1-\gamma)||\nabla\hat{\Lambda}(x)||_2^2\right]dx \tag{17}$$

$$= \int_{\mathcal{X}} \left[\gamma||\nabla\hat{\mu}(x)||_2^2 + (1-\gamma)||\nabla\hat{\Lambda}(x)||_2^2\right]dx. \tag{18}$$

Each term in this integral is non-negative, so the minimum value it could be is zero. Any pair of constant functions μ, Λ will minimize this integral, of which there are infinitely many.

(iii) We return to the α, β-parameterization for this proof. Suppose there is no mean regularization, that is $\alpha = 0$.

$$\implies \begin{cases} \hat{\Lambda}^*(x)(\hat{\mu}^*(x) - y(x)) = 0 \\ (\hat{\mu}^*(x) - y(x))^2 = \frac{1}{\hat{\Lambda}^*(x)} + 4\beta\Delta\hat{\Lambda}^*(x) \end{cases} \tag{19}$$

From the first condition we see that there must be perfect matching between $\hat{\mu}^*$ and y since $\hat{\Lambda}^* > 0$ in order to define a valid normal distribution.

$$\implies \begin{cases} (\hat{\mu}^*(x) - y(x)) = 0 \\ (\hat{\mu}^*(x) - y(x))^2 = \frac{1}{\hat{\Lambda}^*(x)} + 4\beta\Delta\hat{\Lambda}^*(x) \end{cases} \tag{20}$$

$$\implies \begin{cases} (\hat{\mu}^*(x) - y(x)) = 0 \\ 0 = \frac{1}{\hat{\Lambda}^*(x)} + 4\beta\Delta\hat{\Lambda}^*(x) \end{cases} \tag{21}$$

Now, note that as $\beta \to 0$,

$$\implies \begin{cases} (\hat{\mu}^*(x) - y(x)) = 0 \\ 0 = \frac{1}{\hat{\Lambda}^*(x)} \end{cases} \tag{22}$$

but $\hat{\Lambda}^* \in \mathbb{R}$, so the second condition can never be satisfied. Thus, in order for a solution to exist if $\alpha = 0 \implies \beta > 0$. These α, β values correspond to $\rho \in (0, 1), \gamma \neq 1$.

□

This proposition implies the existence, or rather the lack thereof of solutions to the NFE. Should there be no mean regularization, then there needs to be at least some present for the precision. The theory potentially suggests that the vice versa of this should also guarantee valid solutions (i.e., $\alpha > 0$ and $\beta = 0$); however, in practice this does not hold true. The reason lies in the stipulation that $y(x) \neq \hat{\mu}(x)$ a.e.

Typically, this condition can be satisfied while still allowing for countably many values of x in which $y(x) = \hat{\mu}(x)$. The problem is that we fit models using a finite amount of data. As mentioned previously, we typically minimize the objective function by approximating it using a MC estimate with $\mathcal{D}$ as samples. An alternative perspective of this decision is that we are actually calculating the expected values exactly with respect to an empirical distribution imposed by $\mathcal{D}$: $p(x, y) :\propto \sum_{(x_i, y_i) \in \mathcal{D}} \delta(||x - x_i||)\delta(y - y_i)$ where $\delta(\cdot)$ is the Dirac delta function.[4] Because of this, a single value of x can possess non-zero measure, thus it only takes a single instance of $y(x) = \hat{\mu}(x)$ for the statement $y(x) \neq \hat{\mu}(x)$ a.e. to be false. This, unfortunately, is very likely to happen while solving for $\hat{\mu}, \hat{\Lambda}$. **Thus, we can conclude that no matter what, $\left(\frac{1-\rho}{\rho}\right)(1 - \gamma) > 0$ for a valid solution to be guaranteed to exist.**

B Experimental Details

B.1 Datasets

We chose 64 datapoints in each of the simulated datasets. The generating processes for each simulated dataset is included in Table 2 and can be seen in Fig. 4. The homoskedastic data is simulated in the same way, but with $f(x) = 1$. For testing, we simulate a new dataset of 64 datapoints with the same process. Table 3 summarizes the UCI datasets.

B.2 Training Details

We take 22 values of γ, ρ that range from 10^{-10} up to $1 - 10^{-5}$ ($\rho, \gamma \in \{0.9999, 0.999, 0.99, 0.9, 0.8, 0.7, 0.6, 0.5, 0.4, 0.3, 0.2, 0.1, 0.01, 0.001, 0.0001, 0.00001, 0.000001, 0.0000001, 0.00000001, 0.000000001, 0.0000000001, 0.00000000001\}$)

[4] Not to be confused with the functional derivative operator δ.

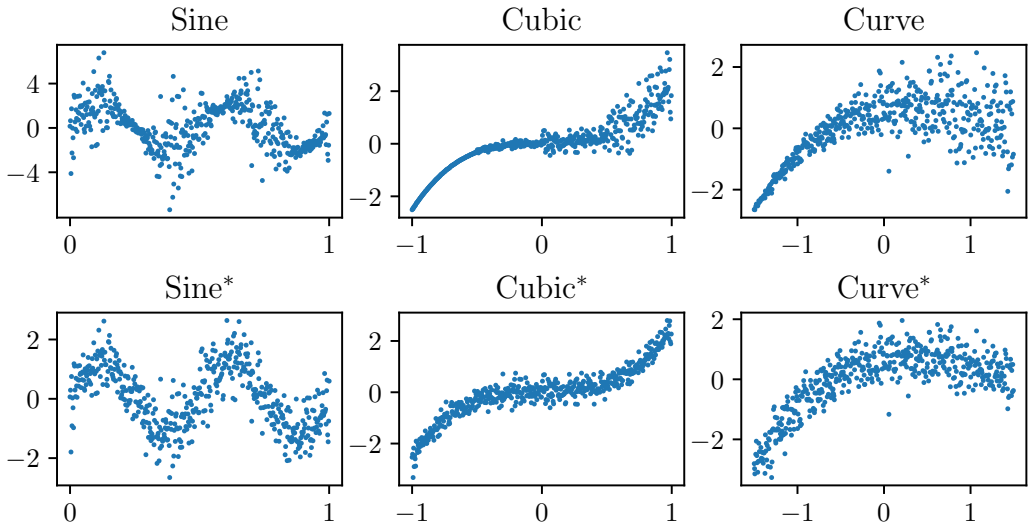

Figure 4: Visualization of heteroskedastic and homoskedastic versions of simulated datasets. Specific details for the functional form of these can be found in Table 2.

Table 2: Simulated datasets. Each dataset is defined by a true μ function and then a noise function f. All data is generated as $\mu(x) + \epsilon(x)$ where $\epsilon(x) \sim \mathcal{N}(0, f(x)^2)$. After the datasets were generated they were scaled to have mean zero and standard deviation one. The homoskedastic versions of each dataset fix $f(x) = 1$. The datasets are shown in Fig. 4.

Dataset	Mean (μ)	Noise Pattern (f)	Domain
Sine	$\mu(x) = 2\sin(4\pi x)$	$f(x) = \sin(6\pi x) + 1.25$	$x \in [0, 1]$
Cubic	$\mu(x) = x^3$	$f(x) = \begin{cases} 0.1 & \text{for } x < -0.5 \\ 1 & \text{for } x \in [-0.5, 0.0) \\ 3 & \text{for } x \in [0.0, 0.5) \\ 10 & \text{for } x \geq .5 \end{cases}$	$x \in [-1, 1]$
Curve	$\mu(x) = x - 2x^2 + 0.5x^3$	$f(x) = x + 1.5$	$x \in [-1.5, 1.5]$

Table 3: UCI dataset.

Dataset	Train Size	Test Size	Input Dimension
Concrete	687	343	8
Housing	337	168	13
Power	6379	3189	4
Yacht	204	102	6

on a logit scale for all of the experiments run on neural networks. For the NFEs we take 20 values from 10^{-6} up to $1 - 10^{-7}$ ($\rho, \gamma \in \{0.999999, 0.99999, 0.99999, 0.9999, 0.999, 0.99, 0.9, 0.8, 0.7, 0.6, 0.5, 0.4, 0.3, 0.2, 0.1, 0.01, 0.001, 0.0001, 0.00001, 0.000001\}$), also on a logit scale. This scaling increases the absolute density of points evaluated near the extreme cases of 0 and 1 where the theoretical analysis of the NFE focused. The ranges differ slightly due to numerical stability during the fitting. The limiting cases of $\gamma, \rho \in \{0, 1\}$ were omitted for numerical stability and the ranges of values for the NFEs vs neural networks vary slightly for the same reason. The values of ρ, γ that were taken along the $\rho = 1 - \gamma$ line were $\rho, \gamma \in \{0.0000e+00, 1.0000e-11, 1.0000e-10, 1.0000e-09, 1.0000e-08, 1.0000e-07, 1.0000e-06, 1.0000e-05, 1.0000e-04, 1.0000e-03, 1.0000e-02, 1.0000e-01, 1.1000e-01, 1.2000e-01, 1.3000e-01, 1.4000e-01, 1.5000e-01, 1.6000e-01, 1.7000e-01, 1.8000e-01, 1.9000e-01, 2.0000e-01, 2.1000e-01, 2.2000e-01, 2.3000e-01, 2.4000e-01, 2.5000e-01, 2.6000e-01, 2.7000e-01, 2.8000e-01, 2.9000e-01, 3.0000e-01, 3.1000e-01, 3.2000e-01, 3.3000e-01, 3.4000e-01, 3.5000e-01, 3.6000e-01, 3.7000e-01, 3.8000e-01, 3.9000e-01, 4.0000e-01, 4.1000e-01, 4.2000e-01, 4.3000e-01, 4.4000e-01, 4.5000e-01, 4.6000e-01, 4.7000e-01, 4.8000e-01, 4.9000e-01, 5.0000e-01, 5.1000e-01, 5.2000e-01, 5.3000e-01, 5.4000e-01, 5.5000e-01, 5.6000e-01, 5.7000e-01, 5.8000e-01, 5.9000e-01, 6.0000e-01, 6.1000e-01, 6.2000e-01, 6.3000e-01, 6.4000e-01, 6.5000e-01, 6.6000e-01, 6.7000e-01, 6.8000e-01, 6.9000e-01, 7.0000e-01, 7.1000e-01, 7.2000e-01, 7.3000e-01, 7.4000e-01, 7.5000e-01, 7.6000e-01, 7.7000e-01, 7.8000e-01, 7.9000e-01, 8.0000e-01, 8.1000e-01, 8.2000e-01, 8.3000e-01, 8.4000e-01, 8.5000e-01, 8.6000e-01, 8.7000e-01, 8.8000e-01, 8.9000e-01, 9.0000e-01, 9.9000e-01, 9.9900e-01, 9.9990e-01, 1.0000e+00]\}$. All experiments were run on Nvidia Quadro RTX 8000 GPUs. Approximately 400 total GPU hours were used across all experiments.

B.3 METRICS

- Sobolev norm: For the one-dimensional datasets the function is evaluated on a dense grid and then the gradients are approximated via finite differences and a trapezoidal approximation to the integral is taken. In the case of the NFE, we only assess the function on the solved for, discretized points while with the neural networks we interpolate between points. For the higher-dimensional UCI datasets the gradients are also numerically approximated in the same way but only at the points in the train/test sets.

- MSE: In the fully non-parametric, unconstrained setting, the maximum likelihood estimates at each x_i are $\hat{\mu}(x_i) = y(x_i)$ and $\hat{\Lambda}(x_i) = (y(x_i) - \mu(x_i))^{-2} \implies \hat{\Lambda}^{-1/2}(x_i) = |y(x_i) - \mu(x_i)|$, serving as motivation for checking these differences.

Variability over runs The experiments were each run six times with different seeds. The standard deviations over the metrics displayed in Fig. 2 are shown in Fig. 5. The Sobolev norms show that there is the most variability in the overfitting regions O_{I} and parts of O_{II}. This indicates that the functions themselves vary across runs. However, when turning to quality of fits, the MSEs show a different pattern of regions of instability, and O_{I} has low variability in terms of actual performance.

B.4 NFE

For the discretized field theory we take $n_{ft} = 4096$ evenly spaced points on the interval $[-1, 1]$. There are two datapoints placed beyond $[-1, 1]$ because the method we use to estimate the gradients requires the datapoints to have left and right neighbors. These datapoints were not included when computing our metrics. Of these 4096 datapoints 64 were randomly selected to be used for training neural networks $\hat{\mu}_\theta, \hat{\Lambda}_\phi$. The field theory results were consistent across choices of $n_{ft} \in \{256, 512, 1024, 2048, 4096\}$. We present results for $n_{ft} = 4096$ in the main paper. We train for 100000 epochs and use the Adam optimizer with a basic triangular cycle that scales initial amplitude by half each cycle on the learning rate. The minimum and maximum learning rates were 0.0005 and 0.01. The cycles were 5000 epochs long. We clip the gradients at 1000.

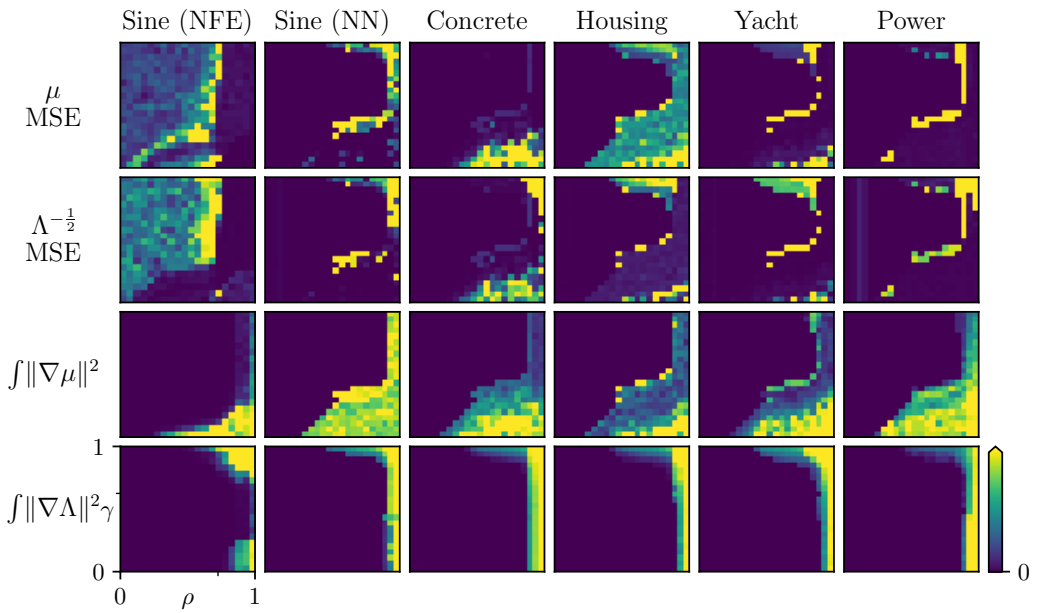

Figure 5: The standard deviation over the six runs of each metric shown in Fig. 2

B.5 Simulated Data with Neural Networks

For all of the simulated datasets except for *sine* we train for 600000 epochs and use the Adam optimizer with a basic triangular cycle that scales initial amplitude by half each cycle on the learning rate. The minimum and maximum learning rates were 0.0001 and 0.01. The cycles were 50000 epochs. The first 250000 epochs are only spend on training $\hat{\mu}_\theta$ while the remaining 350000 epochs are spent training both $\hat{\mu}_\theta, \hat{\Lambda}_\phi$. We clip the gradients at 1000. The training for the *sine* dataset was the same, except trained for 2500000 epochs.

B.6 UCI Data with Neural Networks

For the *concrete, housing* and *yacht* datasets we train for 500000 epochs and use the Adam optimizer with a basic triangular cycle that scales initial amplitude by half each cycle on the learning rate. The minimum and maximum learning rates were 0.0001 and 0.01. The cycles were 50000 epochs. The first 250000 epochs are only spend on training $\hat{\mu}_\theta$ while the remaining 250000 epochs are spent training both $\hat{\mu}_\theta, \hat{\Lambda}_\phi$. Meanwhile on the *power* dataset, we had to use minibatching due to the size of the dataset. We used minibatches of 1000 and trained for 50000 total epochs with the first 25000 dedicated solely to $\hat{\mu}_\theta$ and the remainder training both $\hat{\mu}_\theta, \hat{\Lambda}_\phi$. The same cyclic learning rate was used but with cycle length 5000. We clip the gradients at 1000.

B.7 Practical Suggestion

We can also view the $\rho = 1 - \gamma$ line that we search from the perspective of the α, β parameterization of the regularizers. Let $\rho, \gamma \in (0, 1)$ such that

$$\rho = 1 - \gamma.$$

Furthermore, we know that $\alpha = \frac{1-\rho}{\rho}\gamma$ and that $\beta = \frac{1-\rho}{\rho}(1-\gamma)$. If we are interested in the model settings for $(\rho(t) = t, \gamma(t) = 1 - t)$ for $t \in (0, 1)$, it then follows that we are equivalently interested

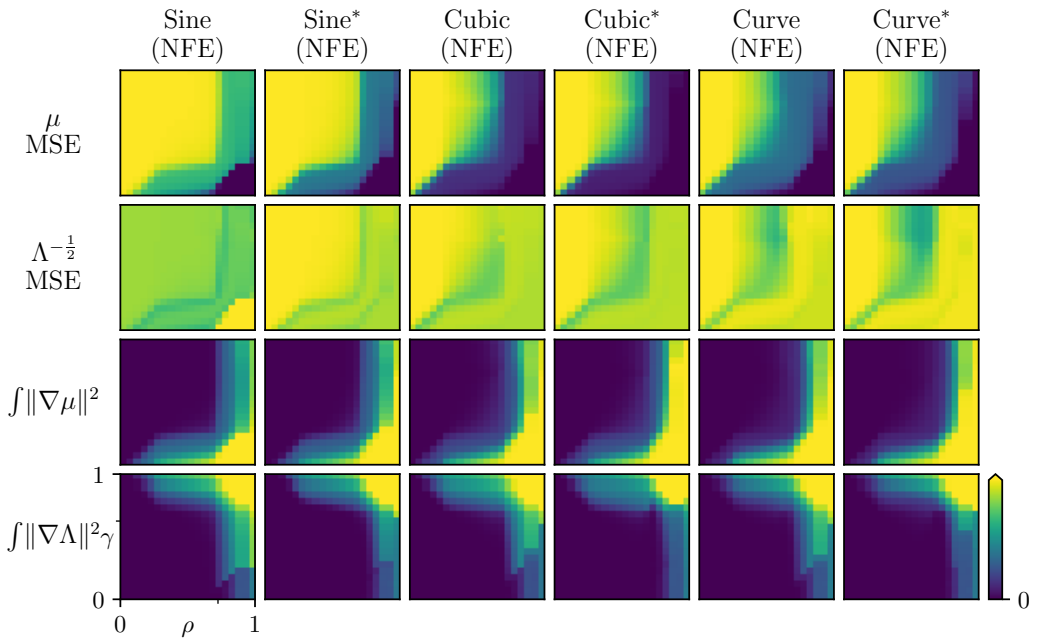

Figure 6: Same configuration as Fig. 2, except all results here pertain to minimizing the NFE on six different synthetic datasets described in Table 2. Dataset names with an $*$ are the homoskedastic counterparts.

in

$$
\begin{aligned}
(\alpha(t), \beta(t)) &= \left(\frac{1-\rho(t)}{\rho(t)} \gamma(t), \frac{1-\rho(t)}{\rho(t)} (1-\gamma(t)) \right) \\
&= \left(\frac{1-t}{t}(1-t), \frac{1-t}{t} t \right) \\
&= \left(\frac{(1-t)^2}{t}, 1-t \right) \\
\implies & \begin{cases} t & = 1-\beta(t) \\ \alpha(t) & = \frac{\beta(t)^2}{t} \end{cases} \text{ or } \sqrt{t\alpha(t)} = \beta(t)
\end{aligned}
$$

C Additional Results

C.1 All Synthetic Dataset Results

Both NFE and neural networks were fit to the heteroskedastic and homoskedastic synthetic datasets described in Table 2. The main results for these displayed as phase diagrams of various metrics can be seen in Fig. 6 and Fig. 7 respectively. We largely see the same trends as were exhibited by the real-world datasets seen in Fig. 2.

C.2 Effect of Neural Network Size

We used the same training methods to fit models with one and two hidden layers and fit them to the *concrete* dataset. The results in the phase diagram were consistent with the other experiments, as can be seen in Fig. 8.

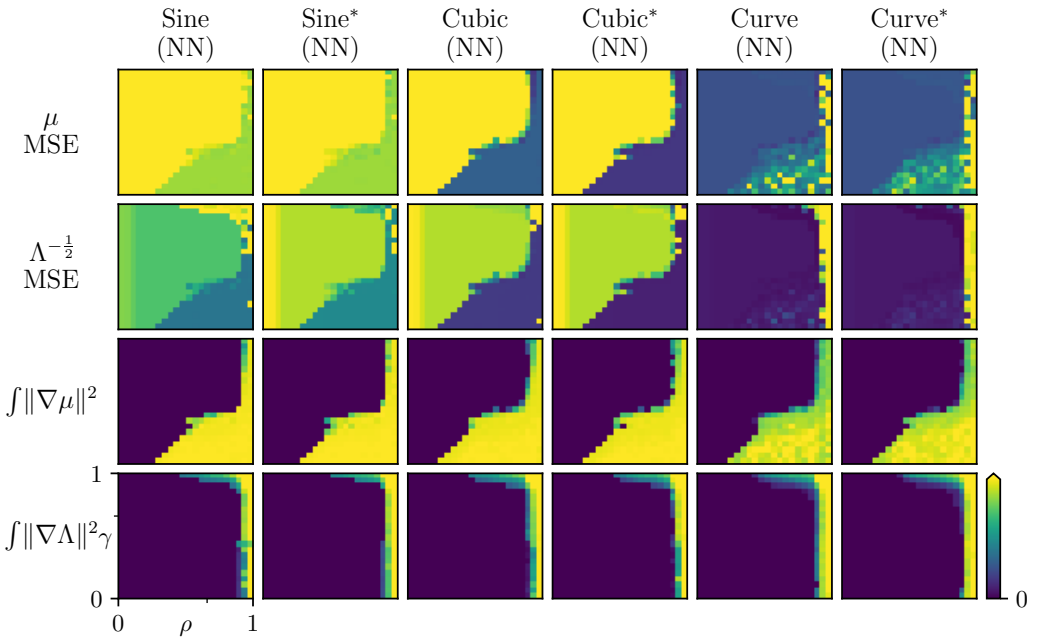

Figure 7: Same configuration as Fig. 2 and Fig. 6, except all results here pertain to training a neural network on six different synthetic datasets described in Table 2. Dataset names with an $*$ are the homoskedastic counterparts.

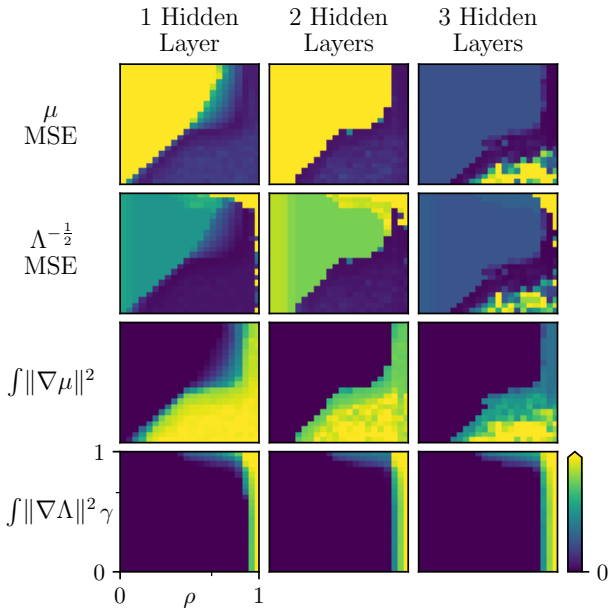

Figure 8: Same configuration as Fig. 2, however, these results pertain all to fitting a neural network of various sizes on the *concrete* dataset.

D COMPARISON TO BASELINES

We compare the performance of our diagonal $\rho + \gamma = 1$ search against two baselines, β-NLL (Seitzer et al., 2022) and an ensemble of MLE-fit heteroskedastic regression models (Lakshminarayanan et al., 2017). We use μ MSE, $\Lambda^{-\frac{1}{2}}$ MSE, and expected calibration error (ECE) to evaluate the models. In all cases lower values are better.

D.1 MODEL ARCHITECTURE

All (individual) models have the same architecture: fully connected neural networks with three hidden layers of 128 nodes and leaky ReLU activations for the synthetic and UCI datasets and fully connected neural networks with three hidden layers of 256 nodes for the ClimSim data (Yu et al., 2023). Note that both baselines model the variance while our approach models the precision (inverse-variance). In all cases we use a softplus on the final layer of the variance/precision networks to ensure the output is positive.

For the $\beta-$NLL implementation we take $\beta = 0.5$ as suggested in Seitzer et al. (2022). The ensemble method we use fits 6 individual heteroskedastic neural networks and combines their outputs into a mixture distribution that is approximated with a normal distribution. We do not add in adversarial noise as the authors state it does not make a significant difference. We fit six $\beta-$NLL models and six MLE-ensembles.

D.2 DIAGONAL SELECTION CRITERIA

After conducting our diagonal search we found the model that minimized μ MSE and the model that minimized $\Lambda^{-\frac{1}{2}}$ MSE on the *training* data. In some cases these models coincided. We then used the model that was on the midpoint (on a logit scale) of the $\rho + \gamma = 1$ line between these two models to compare. The results are reported in Table 4. In all cases our method is competitive with or exceeds the performance of these two baselines–particularly on real-world data.

D.3 TRAINING DETAILS

Training for our method was conducted as described in sections B.2 and B.6 of the appendix.

For the baselines, on all of the simulated datasets we train for 600000 epochs and use the Adam optimizer with a basic triangular cycle that scales initial amplitude by half each cycle on the learning rate. The minimum and maximum learning rates were 0.0001 and 0.01. The cycles were 50000 epochs. We clip the gradients at 1000. The same optimization scheme is performed for the UCI datasets but for 500000 epochs for the *Housing*, *Concrete*, and *Yacht* datasets. The *Power* dataset was trained for 50000 epochs with batches of 1000.

D.4 CLIMSIM DATASET

The ClimSim dataset (Yu et al., 2023) is a largescale climate dataset. Its input dimension is 124 and output dimension is 128. We use all 124 inputs to model a single output, *solar flux*. We train on 10,091,520 of the approximately 100 million points for training and we use a randomly selected 7,209 points to evaluate our models.

D.5 DEFICIENCY OF ECE

Shortcomings of ECE (in isolation) are well documented (Kuleshov et al., 2018; Levi et al., 2022; Chung & Neiswanger, 2021). The main issue with ECE is it measures *average* calibration, while *individual* calibration is more desirable. On our diagonal search we found that often times the models that achieved the best ECE were those that were severely underfit and belonged to region U_I. In Table 4 we see that the MLE-ensemble is able to achieve low scores while being uncompetitive with respect to the two MSE metrics. The MLE-ensembles were unstable on several of the datasets with respect to the variance network which is consistent with Proposition 1. In particular this can be seen for the synthetic datasets the $\Lambda^{-\frac{1}{2}}$ MSE diverges to infinity.

Table 4: Comparison of our method against two baselines. We report the average and standard deviations of expected calibration error (ECE), μ MSE and $\Lambda^{-\frac{1}{2}}$ on test data. Lowest mean value for each metric is bolded.

Dataset / Metric	**Ours**	β-NLL Seitzer et al. (2022)	MLE Ensemble Lakshminarayanan et al. (2017)
Cubic			
ECE	**0.2380** ± 0.03	0.2385 ± 0.02	0.2411 ± 0.02
μ MSE	0.2339 ± 0.01	**0.1500** ± 0.01	1.1809 ± 1.88
$\Lambda^{-\frac{1}{2}}$ MSE	0.2397 ± 0.02	**0.1397** ± 0.01	inf ± nan
Curve			
ECE	0.1804 ± 0.02	**0.1754** ± 0.02	0.2432 ± 0.00
μ MSE	**0.4318** ± 0.12	0.4877 ± 0.16	1.0067 ± 0.19
$\Lambda^{-\frac{1}{2}}$ MSE	0.4655 ± 0.09	**0.4187** ± 0.20	inf ± nan
Sine			
ECE	0.2499 ± 0.00	**0.2082** ± 0.03	0.2313 ± 0.05
μ MSE	**0.7968** ± 0.00	4.4107 ± 6.90	0.9716 ± 0.06
$\Lambda^{-\frac{1}{2}}$ MSE	**0.7968** ± 0.00	4.3524 ± 6.89	inf ± nan
Concrete			
ECE	0.2471 ± 0.01	0.2552 ± 0.00	**0.0655** ± 0.01
μ MSE	**0.1055** ± 0.02	14.9882 ± 28.75	2.2454 ± 1.74
$\Lambda^{-\frac{1}{2}}$ MSE	**0.3028** ± 0.51	$1.3 \times 10^5 \pm 2.7 \times 10^5$	$1.3 \times 10^5 \pm 1.2 \times 10^5$
Housing			
ECE	**0.0653** ± 0.00	0.2631 ± 0.01	0.1332 ± 0.02
μ MSE	**1.2236** ± 0.00	851.8968 ± 1985.56	155.4494 ± 128.27
$\Lambda^{-\frac{1}{2}}$ MSE	**0.7610** ± 0.00	851.8959 ± 1985.56	218.8269 ± 195.38
Power			
ECE	0.2233 ± 0.01	0.2370 ± 0.00	**0.0285** ± 0.01
μ MSE	0.0350 ± 0.01	0.0313 ± 0.01	**0.0177** ± 0.00
$\Lambda^{-\frac{1}{2}}$ MSE	0.0343 ± 0.01	0.0360 ± 0.01	**0.0091** ± 0.00
Yacht			
ECE	0.3038 ± 0.04	0.2882 ± 0.02	**0.0463** ± 0.02
μ MSE	**0.0077** ± 0.01	34.1239 ± 194.87	6.2670 ± 13.96
$\Lambda^{-\frac{1}{2}}$ MSE	**0.0076** ± 0.01	34.1237 ± 194.87	8.0599 ± 19.18
Solar Flux			
ECE	**0.1503** ± 0.00	0.3007 ± 0.00	0.1924 ± 0.04
μ MSE	**0.2887** ± 0.00	0.4877 ± 0.16	1.0067 ± 0.19
$\Lambda^{-\frac{1}{2}}$ MSE	**0.1175** ± 0.00	0.2881 ± 0.01	$4.6 \times 10^9 \pm 9.9 \times 10^9$