

# Appendix for 'Learning Global Pairwise Interactions with Bayesian Neural Networks'

Tianyu Cui and Pekka Marttinen and Samuel Kaski<sup>1</sup>

## 1 Monte Carlo Estimation of Bayesian GEH

We estimate the mean and variance of the Bayesian M-GEH by Monte Carlo (MC) integration. Unbiased estimators for the mean and variance follow from Eq.6 of main text. To also account for posterior uncertainty in model parameters  $\mathbf{W}$  we approximate two nested expectations, one w.r.t. the posterior distribution,  $q_\theta(\mathbf{W})$ , and the other w.r.t. the conditional empirical distribution  $p(\mathbf{x}|\mathbf{x} \in A_m)$ , as in Eq.6 of main text.

We first draw one sample from  $\text{M-GEH}_g^{i,j}(\mathbf{W})$  as

$$s_k^{i,j} = \sum_{m=1}^M \frac{|A_m|}{\sum_{k=1}^M |A_k|} \left| \frac{1}{L_m} \sum_{l=1}^{L_m} \frac{\partial^2 g^{\mathbf{W}^{(k)}}(\mathbf{x}_m^{(l)})}{\partial x_i \partial x_j} \right|, \quad (1)$$

where  $\mathbf{x}_m^{(l)} \sim p(\mathbf{x}|\mathbf{x} \in A_m)$ ,  $|A_m|$  is the size of  $m$  group, and  $L_m$  is the number of samples that we use to estimation the expectation for group  $m$ . Then the unbiased estimators for the mean ( $\hat{m}^{i,j}$ ) and variance ( $\hat{v}^{i,j}$ ) can be obtained by calculating the sample mean and variance of  $s_k$  with sample size  $K$ . By making use of backward propagation, MC estimation requires  $O(K(\sum_{m=1}^M L_m)D)$  backward-passes ( $D$  is the input dimension, and  $K$  and  $L_m$  are the number of samples that we can choose based on the computational resources), since for each backward passing, we can get the gradient of **all** input features. Moreover, if we assume M-GEH to be approximately Gaussian, which follows from the CLT for large  $K$ , a 95% credible interval can be estimated as  $(\hat{m} - 2\sqrt{\hat{v}}, \hat{m} + 2\sqrt{\hat{v}})$ , which can then be used in statistical tests. To have a stable estimation,  $K$  do not need to be very large, and 400 samples is usually enough according to our experiments. We use a same  $L_m$  for all  $A_m$  in practice, which is normally the smallest size of group. Algorithm 1 shows the detail about how to implement MC Bayesian M-GEH.

### 1.1 Determine $K$ in Monte Carlo Estimator

In this section, we test how many Monte Carlo samples ( $K$  in Algorithm 1) we need to obtain a stable estimator of mean and standard deviation for simulation dataset and public dataset.

Figure 1 shows how the mean and standard deviation of estimated interaction effects change when we increase  $K$  on simulated dataset, and Figure 2 is results on California Housing price dataset. We can notice that the mean becomes stable when  $K$  is larger than 50, and std is stable when  $K$  is larger than 400. So  $K = 400$  should be a safe choice for practice.

**Input:** training data  $D = \{(y_n, \mathbf{x}_n)\}_{n=1}^N$

**Output:** Mean ( $\hat{m}^{i,j}$ ) and variance ( $\hat{v}^{i,j}$ ) of M-GEH for each pair of features  $(i, j)$

```

/* Modeling Interactions */
1 Fit a BNN  $g^{\mathbf{W}}(\mathbf{x})$  to the training data  $D$ ;
/* Detecting Interactions */
2 Cluster test data into  $M$  groups  $A_m$ , and calculate the size of
   each group  $|A_m|$ ;
3 Draw  $\{\mathbf{W}^{(k)}\}_{k=1}^K$  from  $q_\theta(\mathbf{W})$ ;
4 for each draw  $\mathbf{W}^{(k)}$  do
5   for each group  $A_m$  do
6     Sample  $L_m$  datapoints  $\{\mathbf{x}_m^{(l)}\}_{l=1}^{L_m}$  from each group
        $A_m$ ;
7     for each draw  $\mathbf{x}_m^{(l)}$  do
8       Calculate the input Hessian on  $\mathbf{x}_m^{(l)}$ ;
9     end
10  end
11  Compute  $s_k^{i,j}$  according to Eq.1;
12 end
13 Calculate the sample mean  $\hat{m}^{i,j}$  and variance  $\hat{v}^{i,j}$  from MC
   samples  $\{s_k^{i,j}\}_{k=1}^K$ .

```

**Algorithm 1:** MC estimation of Bayesian M-GEH

## 2 Proof of Accuracy Improvement Properties

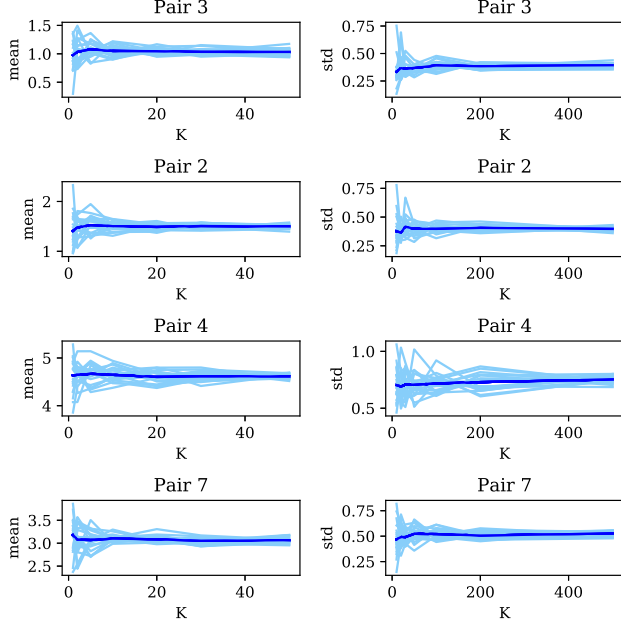
### 2.1 Proof of Property 1

The estimation error of interaction effect between feature  $i$  and  $j$ ,  $L^{i,j} = |\text{M-GEH}_g^{i,j} - \text{M-GEH}_f^{i,j}|$ , can be further derived through:

$$\begin{aligned}
 L^{i,j} &= \sum_{m=1}^M \frac{|A_m|}{\sum_{k=1}^M |A_k|} \left| \mathbb{E}_{p(\mathbf{x}|\mathbf{x} \in A_m)} \left[ \frac{\partial^2 g^{\mathbf{W}}(\mathbf{x})}{\partial x_i \partial x_j} \right] \right| \\
 &\quad - \sum_{m=1}^M \frac{|A_m|}{\sum_{k=1}^M |A_k|} \left| \mathbb{E}_{p(\mathbf{x}|\mathbf{x} \in A_m)} \left[ \frac{\partial^2 f(\mathbf{x})}{\partial x_i \partial x_j} \right] \right| \\
 &\leq \sum_{m=1}^M \frac{|A_m|}{\sum_{k=1}^M |A_k|} \left| \mathbb{E}_{p(\mathbf{x}|\mathbf{x} \in A_m)} \left[ \frac{\partial^2 (g^{\mathbf{W}}(\mathbf{x}) - f(\mathbf{x}))}{\partial x_i \partial x_j} \right] \right| \\
 &= \sum_{m=1}^M \frac{|A_m|}{\sum_{k=1}^M |A_k|} l_m^{i,j}
 \end{aligned} \quad (2)$$

where  $g$  is the learned neural network, and  $f$  is the underlining data generating process. We denote  $l_m^{i,j}$  as the estimation error from  $m$ th group.

<sup>1</sup> Department of Computer Science, Aalto University, Finland. Corresponding to Tianyu Cui. email: firstname.lastname@aalto.fi



**Figure 1.** Number of MC samples for simulated dataset. We repeat tests 20 times. Each light blue line is the result for each test, and blue line is their average. We can notice that the mean becomes stable for most interaction pairs when  $K$  is larger than 40, and std is stable when  $K$  is larger than 400.

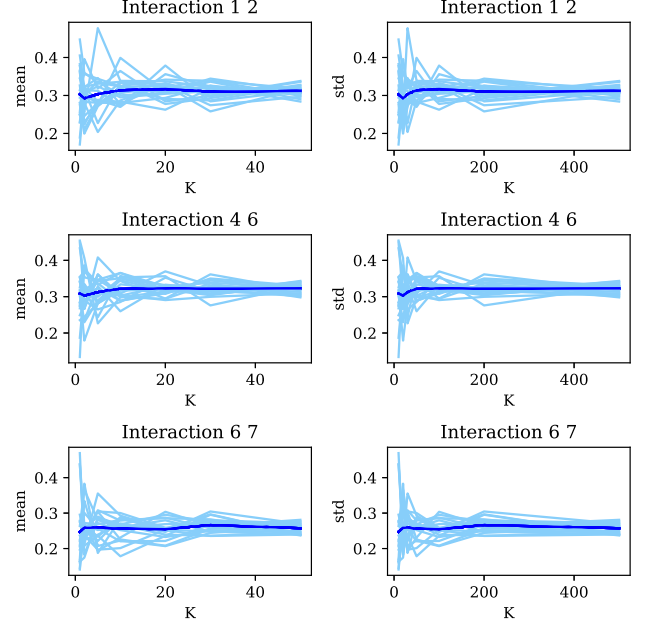
$$\begin{aligned}
 l_m^{i,j} &= \left| \iint_{A_m} \left[ \frac{\partial^2 (g^{\mathbf{W}}(\mathbf{x}) - f(\mathbf{x}))}{\partial x_i \partial x_j} \right] p(\mathbf{x} | \mathbf{x} \in A_m) dx_i dx_j \right| \\
 &\leq P_m \left| \iint_{A_m} \frac{\partial^2 (g^{\mathbf{W}}(\mathbf{x}) - f(\mathbf{x}))}{\partial x_i \partial x_j} dx_i dx_j \right| \\
 &= P_m \left| \iint_{A_{ms} + A_{m1} + A_{m2} + A_{m3} + A_{m4}} \frac{\partial^2 (g^{\mathbf{W}} - f)(\mathbf{x})}{\partial x_i \partial x_j} dx_i dx_j \right| \quad (3)
 \end{aligned}$$

where  $p_m$  is the highest density of the conditional probability distribution  $p(\mathbf{x} | \mathbf{x} \in A_m)$ . We divide the domain of  $A_m$  into finite subregions, which contains a rectangle subregion  $A_{ms}$ , and several non-rectangled subregions (for example  $\{A_{mi}\}_{i=1}^4$  in Figure 3), and the rectangle subregion does not have to touch the boundary of the group. This is generally true if the domain of each group is compact.

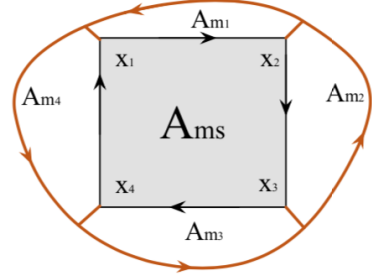
Use Figure 3 as an example, for subregion  $A_{ms}$ ,

$$\begin{aligned}
 &\left| \iint_{A_{ms}} \frac{\partial^2 (g^{\mathbf{W}}(\mathbf{x}) - f(\mathbf{x}))}{\partial x_i \partial x_j} dx_i dx_j \right| \\
 &= \left| (g^{\mathbf{W}}(\mathbf{x}_2) - f(\mathbf{x}_2)) + (g^{\mathbf{W}}(\mathbf{x}_4) - f(\mathbf{x}_4)) \right. \\
 &\quad \left. - (g^{\mathbf{W}}(\mathbf{x}_1) - f(\mathbf{x}_1)) + (g^{\mathbf{W}}(\mathbf{x}_3) - f(\mathbf{x}_3)) \right| \quad (4) \\
 &\leq 4\epsilon
 \end{aligned}$$

where  $\epsilon$  is the prediction error of  $g(\cdot)$ . For those non-rectangled sub-



**Figure 2.** Number of MC samples for California housing dataset. Mean becomes stable for most interaction pairs when  $KM$  is larger than 50, and std is stable when  $K$  is larger than 400.



**Figure 3.** Decomposition of each group

resions, such as  $A_{m1}$ , according to Green's theorem:

$$\begin{aligned}
 &\left| \iint_{A_{m1}} \frac{\partial^2 (g^{\mathbf{W}}(\mathbf{x}) - f(\mathbf{x}))}{\partial x_i \partial x_j} dx_i dx_j \right| \\
 &= \left| \oint_{A_{m1}} \frac{\partial (g^{\mathbf{W}}(\mathbf{x}) - f(\mathbf{x}))}{\partial x_j} dx_j \right| \quad (5) \\
 &\leq L \left| \oint_{A_{m1}} dx_j \right| = L |\Delta x_j|.
 \end{aligned}$$

Here we assume that  $g(\cdot)$  and  $f(\cdot)$  are both  $L$ -Lipschitz functions, and  $\Delta x_j$  is the maximum difference of feature  $x_j$  in subregion  $A_{m1}$ .

Based on the above reasoning, we can conclude that:

$$L^{i,j} = \sum_{m=1}^M \frac{|A_m|}{\sum_{k=1}^M |A_k|} l_m^{i,j} \leq \alpha\epsilon + \beta L, \quad (6)$$

thus we proved property 1.

From Eq.6, we can notice that the upper bound consists of two parts:  $\alpha\epsilon$  and  $\beta L$ . If the area of the rectangle subregion  $A_{ms}$  is large,  $\beta$  will

be small, and the bound will be tighter and also will be dominated by the prediction error. Moreover, if we want to make the bound even much tighter, instead of one rectangle we can use a combination of multiple rectangles inside the region.

Thus training a better BNN, can reduce the upper bound of interaction estimation error, thus can obtain a more stable and accurate estimated interaction effects.

## 2.2 Proof of Property 2

If we denote  $\pi(\mathbf{y}, \mathbf{x}) = \pi(\mathbf{y}|\mathbf{x})\pi(\mathbf{x})$  is the true model and  $p(g^{\mathbf{W}}(\mathbf{x})|\mathbf{x})$  is the posterior predictive distribution of model  $g^{\mathbf{W}}(\mathbf{x})$  given  $\mathbf{x}$ , we call the uncertainty of  $p(g^{\mathbf{W}}(\mathbf{x})|\mathbf{x})$  *perfectly calibrated* when  $\pi(\mathbf{y}|\mathbf{x}) = p(g^{\mathbf{W}}(\mathbf{x})|\mathbf{x})$ . Another way to define calibration is: if the  $\xi$  percentage Bayesian credible interval of the sample mean of  $p(g^{\mathbf{W}}(\mathbf{x})|\mathbf{x})$  is the same as the  $\xi$  frequentist confidence interval of the sample mean of  $\pi(\mathbf{y}|\mathbf{x})$  for all  $\xi \in [0, 1]$  with an infinite number of data, predictive distribution  $p(g^{\mathbf{W}}(\mathbf{x})|\mathbf{x})$  is *perfectly calibrated*. So if the credible interval is closer to the corresponding confidence interval,  $p(g^{\mathbf{W}}(\mathbf{x})|\mathbf{x})$  is *better calibrated*. Here we assume that the sample mean of both  $\pi(\mathbf{y}|\mathbf{x})$  and  $p(g^{\mathbf{W}}(\mathbf{x})|\mathbf{x})$  are Gaussian distributed, which is generally true according to CLT.

In the rest of this section, we first define the *closeness* between the  $\xi$  credible interval and the  $\xi$  confidence interval. Then we show that for two predictive model  $g^{\mathbf{W}_1}(\cdot)$  and  $g^{\mathbf{W}_2}(\cdot)$ , if  $p(g^{\mathbf{W}_1}(\mathbf{x})|\mathbf{x})$  is better calibrated than  $p(g^{\mathbf{W}_2}(\mathbf{x})|\mathbf{x})$ , then  $p(\sum_{i \in S} \phi_i g^{\mathbf{W}_1}(\mathbf{x}^{(i)})|\mathbf{x}^{(i \in S)})$  is better calibrated than  $p(\sum_{i \in S} \phi_i g^{\mathbf{W}_2}(\mathbf{x}^{(i)})|\mathbf{x}^{(i \in S)})$ , where  $\mathbf{x}^{(i \in S)}$  is the set of  $|S|$  data points sampled from data distribution. Thus we have proved property 2, because gradient (or Hessian) can be regarded as a linear combination with infinitesimal changes, thus Eq.1 can be written in the form  $\sum_i \phi_i g^{\mathbf{W}}(\mathbf{x}^{(i)})$  with properly chosen  $\phi_i$ .

### 2.2.1 Closeness between Two Intervals

We denote  $\text{CreI}_\xi$  to be the  $\xi$  credible interval of  $\hat{m}$ , the mean of  $p(g^{\mathbf{W}}(\mathbf{x})|\mathbf{x})$ , and  $\text{ConI}_\xi$  to be the  $\xi$  confidence interval of  $m$ , the mean of  $\pi(\mathbf{y}|\mathbf{x})$ . Then we define the closeness of  $\text{CreI}_\xi$  and  $\text{ConI}_\xi$  to be:

$$\delta(\xi) = \frac{|\text{CreI}_\xi \cap \text{ConI}_\xi|}{|\text{CreI}_\xi \cup \text{ConI}_\xi|}.$$

When  $\delta(\xi) = 1$ , two intervals perfectly match, and when  $\delta(\xi) = 0$ , two intervals have no intersection.

### 2.2.2 Calibration Improvement Preserved under Linear Combination

We first prove that if  $p(g^{\mathbf{W}_1}(\mathbf{x})|\mathbf{x})$  is better calibrated than  $p(g^{\mathbf{W}_2}(\mathbf{x})|\mathbf{x})$ , then  $p(g^{\mathbf{W}_1}(\mathbf{x}^{(i)}) + g^{\mathbf{W}_1}(\mathbf{x}^{(j)})|\mathbf{x}^{(i)}, \mathbf{x}^{(j)})$  is better calibrated than  $p(g^{\mathbf{W}_2}(\mathbf{x}^{(i)}) + g^{\mathbf{W}_2}(\mathbf{x}^{(j)})|\mathbf{x}^{(i)}, \mathbf{x}^{(j)})$ .

We denote that  $\hat{m}_i \sim N(\hat{\mu}_i, \hat{\sigma}_i^2)$ ,  $\hat{m}_j \sim N(\hat{\mu}_j, \hat{\sigma}_j^2)$  where  $\hat{m}_i$  and  $\hat{m}_j$  are the sample mean of  $p(g^{\mathbf{W}}(\mathbf{x}^{(i)}))$  and  $p(g^{\mathbf{W}}(\mathbf{x}^{(j)}))$  respectively. And  $m_i \sim N(\mu_i, \sigma_i^2)$ ,  $m_j \sim N(\mu_j, \sigma_j^2)$ , where  $m_i$  and  $m_j$  are the sample mean of  $\pi(\mathbf{y}^{(i)}|\mathbf{x}^{(i)})$  and  $\pi(\mathbf{y}^{(j)}|\mathbf{x}^{(j)})$ .

We only present the case<sup>2</sup> when  $\text{CreI}_\xi \cap \text{ConI}_\xi$  is different from  $\text{CreI}_\xi$  or  $\text{ConI}_\xi$ . We can calculate the intervals based on Gaussian:  $\text{CreI}_\alpha^{(i)} = [\hat{\mu}_i - \alpha\hat{\sigma}_i, \hat{\mu}_i + \alpha\hat{\sigma}_i]$  and  $\text{ConI}_\alpha^{(i)} = [\mu_i - \alpha\sigma_i, \mu_i + \alpha\sigma_i]$ , where  $\alpha$  is the value of percent point function for  $\xi$ . Then for data  $\mathbf{x}^{(i)}$ , the closeness of interval is

$$\begin{aligned} \delta_i(\alpha) &= \frac{\alpha(\sigma_i + \hat{\sigma}_i) + \mu_i - \hat{\mu}_i}{\alpha(\sigma_i + \hat{\sigma}_i) + \hat{\mu}_i - \mu_i} \\ &= 1 - 2 \frac{\hat{\mu}_i - \mu_i}{\alpha(\sigma_i + \hat{\sigma}_i) + \hat{\mu}_i - \mu_i}, \end{aligned} \quad (7)$$

where we assume  $\hat{\mu}_i + \alpha\hat{\sigma}_i$  is greater than  $\mu_i + \alpha\sigma_i$  (another case can be shown in the same way). And also for data  $\mathbf{x}^{(j)}$ :

$$\begin{aligned} \delta_j(\alpha) &= \frac{\alpha(\sigma_j + \hat{\sigma}_j) + \mu_j - \hat{\mu}_j}{\alpha(\sigma_j + \hat{\sigma}_j) + \hat{\mu}_j - \mu_j} \\ &= 1 - 2 \frac{\hat{\mu}_j - \mu_j}{\alpha(\sigma_j + \hat{\sigma}_j) + \hat{\mu}_j - \mu_j}. \end{aligned} \quad (8)$$

The sample mean of  $p(g^{\mathbf{W}_2}(\mathbf{x}^{(i)}) + g^{\mathbf{W}_2}(\mathbf{x}^{(j)})|\mathbf{x}^{(i)}, \mathbf{x}^{(j)})$  is equal to  $\hat{m}_i + \hat{m}_j$ , because  $\mathbf{x}^{(i)}$  and  $\mathbf{x}^{(j)}$  are independent. Thus  $\hat{m}_i + \hat{m}_j \sim N(\hat{\mu}_i + \hat{\mu}_j, \hat{\sigma}_i^2 + \hat{\sigma}_j^2)$ , and also  $m_i + m_j \sim N(\mu_i + \mu_j, \sigma_i^2 + \sigma_j^2)$ . Here we consider the intervals with percent point function equals to  $\sqrt{2}\alpha$ , then  $\text{CreI}_{\sqrt{2}\alpha}^{(i,j)} = [\hat{\mu}_i + \hat{\mu}_j - \sqrt{2}\alpha\sqrt{\hat{\sigma}_i^2 + \hat{\sigma}_j^2}, \hat{\mu}_i + \hat{\mu}_j + \sqrt{2}\alpha\sqrt{\hat{\sigma}_i^2 + \hat{\sigma}_j^2}]$  and  $\text{ConI}_{\sqrt{2}\alpha}^{(i,j)} = [\mu_i + \mu_j - \sqrt{2}\alpha\sqrt{\sigma_i^2 + \sigma_j^2}, \mu_i + \mu_j + \sqrt{2}\alpha\sqrt{\sigma_i^2 + \sigma_j^2}]$ . Thus the closeness of these two intervals is:

$$\begin{aligned} \delta_{i,j}(\sqrt{2}\alpha) &= \frac{\sqrt{2}\alpha(\sqrt{\sigma_i^2 + \sigma_j^2} + \sqrt{\hat{\sigma}_i^2 + \hat{\sigma}_j^2}) + (\mu_i + \mu_j) - (\hat{\mu}_i + \hat{\mu}_j)}{\sqrt{2}\alpha(\sqrt{\sigma_i^2 + \sigma_j^2} + \sqrt{\hat{\sigma}_i^2 + \hat{\sigma}_j^2}) + (\hat{\mu}_i + \hat{\mu}_j) - (\mu_i + \mu_j)} \\ &= 1 - 2 \frac{(\hat{\mu}_i + \hat{\mu}_j) - (\mu_i + \mu_j)}{\sqrt{2}\alpha(\sqrt{\sigma_i^2 + \sigma_j^2} + \sqrt{\hat{\sigma}_i^2 + \hat{\sigma}_j^2}) + (\hat{\mu}_i + \hat{\mu}_j) - (\mu_i + \mu_j)} \\ &\geq 1 - 2 \frac{(\hat{\mu}_i + \hat{\mu}_j) - (\mu_i + \mu_j)}{\alpha(\sigma_i + \sigma_j + \hat{\sigma}_i + \hat{\sigma}_j) + (\hat{\mu}_i + \hat{\mu}_j) - (\mu_i + \mu_j)} \\ &\geq \delta_i(\alpha) + \delta_j(\alpha) - 1 \end{aligned} \quad (9)$$

So when  $p(g^{\mathbf{W}_1}(\mathbf{x})|\mathbf{x})$  is better calibrated than  $p(g^{\mathbf{W}_2}(\mathbf{x})|\mathbf{x})$ , we have  $\delta_i^1(\alpha) > \delta_i^2(\alpha)$  and  $\delta_j^1(\alpha) > \delta_j^2(\alpha)$ . Then the lower bound of  $\delta_{i,j}^1(\sqrt{2}\alpha)$  will be greater than  $\delta_{i,j}^2(\sqrt{2}\alpha)$ , and this applies for all  $\alpha$ , so  $p(g^{\mathbf{W}_1}(\mathbf{x}^{(i)}) + g^{\mathbf{W}_1}(\mathbf{x}^{(j)})|\mathbf{x}^{(i)}, \mathbf{x}^{(j)})$  is better calibrated than  $p(g^{\mathbf{W}_2}(\mathbf{x}^{(i)}) + g^{\mathbf{W}_2}(\mathbf{x}^{(j)})|\mathbf{x}^{(i)}, \mathbf{x}^{(j)})$ . When  $p(g^{\mathbf{W}_1}(\mathbf{x})|\mathbf{x})$  is perfectly calibrated, we have  $\delta_{i,j}^1(\sqrt{2}\alpha) \geq \delta_i^1(\alpha) + \delta_j^1(\alpha) - 1 = 1$ , thus  $p(g^{\mathbf{W}_1}(\mathbf{x}^{(i)}) + g^{\mathbf{W}_1}(\mathbf{x}^{(j)})|\mathbf{x}^{(i)}, \mathbf{x}^{(j)})$  is also perfectly calibrated.

It can be generalized to the distribution of all possible linear combinations of predictions trivially, since the linear combination of independent Gaussian distributions are also Gaussian distribution with linearly combined mean and standard deviation. So  $p(\sum_{i \in S} \phi_i g^{\mathbf{W}_1}(\mathbf{x}^{(i)})|\mathbf{x}^{(i \in S)})$  is also better calibrated than  $p(\sum_{i \in S} \phi_i g^{\mathbf{W}_2}(\mathbf{x}^{(i)})|\mathbf{x}^{(i \in S)})$  if  $p(g^{\mathbf{W}_1}(\mathbf{x})|\mathbf{x})$  is better calibrated than  $p(g^{\mathbf{W}_2}(\mathbf{x})|\mathbf{x})$ .

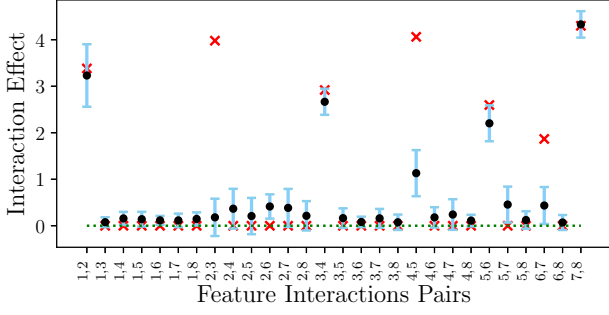


Figure 4. 95% CIs of AEH on simulated dataset.

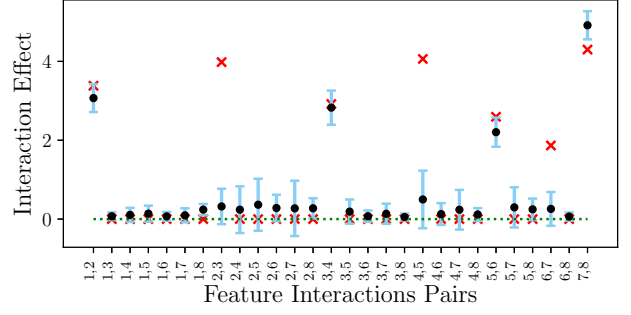


Figure 6. 95% CIs of AEH on simulated dataset with mean-field BNN.

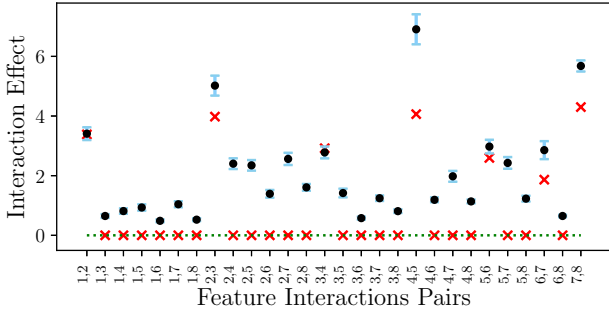


Figure 5. 95% CIs of EAH on simulated dataset.

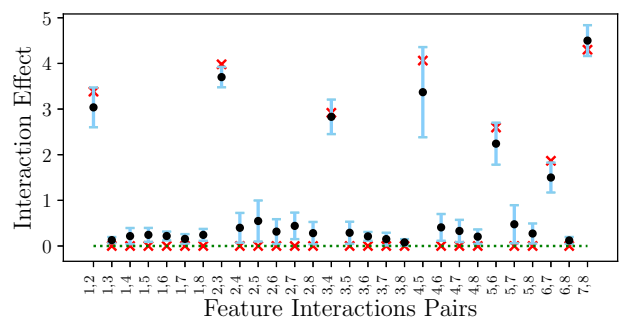


Figure 7. 95% CIs of 5-GEH on simulated dataset with mean-field BNN.

### 3 CIs of AEH and EAH on Simulated Data

Figure 4 and Figure 5 shows the uncertainty of AEH and EAH respectively on the simulated dataset of first experiment. We can find that AEH (Figure 4) can reject most false interactions properly, but it also rejected some true interactions such as interaction 2 3 and interaction 6 7. Compared with AEH, EAH won't reject any true interactions (Figure 5), but it failed to reject all false interactions. So AEH has the highest FNR and EAH has the highest FPR, and our method M-GEH provides a good balance between this two types of error.

### 4 Model Interactions with Mean-field BNN

We also implemented mean-field BNN to model interactions on simulation dataset, and in this case the optimal number of clusters  $M$  is 5. We use a mean-field Bayesian neural network with 3 hidden layers of sizes 100, 100, and 100 nodes. During training, we set the length-scale of prior distribution  $l$  to  $2 \times 10^{-5}$ , and learning rate of Adam to  $10^{-3}$ . Figure 6 to Figure 8 show CIs for different interaction measures on mean-field BNN.

Compared with concrete dropout BNN, mean-field BNN has a better calibrated prediction uncertainty, because it has a more flexible variables to model uncertainty. Thus according to property 2 in the main text, the uncertainty of M-GEH from mean-field BNN is also

<sup>2</sup> It is easy to prove when one interval contains another interval.

better calibrated. We can see that the CIs cover true values (Figure 6 to Figure 8 in appendix) more often than concrete dropout (Figure 4-5 in appendix and Figure 2 in main paper), especially for EAH (according to Figure 8 and Figure 5).

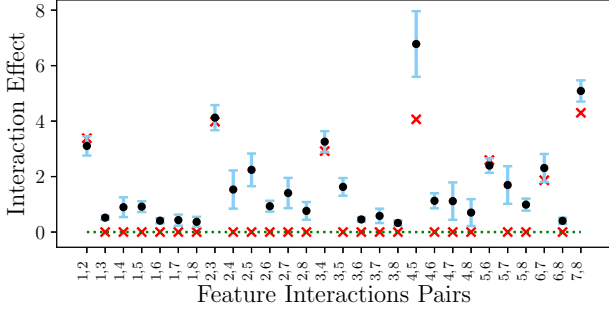
But one drawback of mean-field BNN is that it will not be able to select important features as mentioned in the main text.

### 5 Visualization of Top 3 Interactions

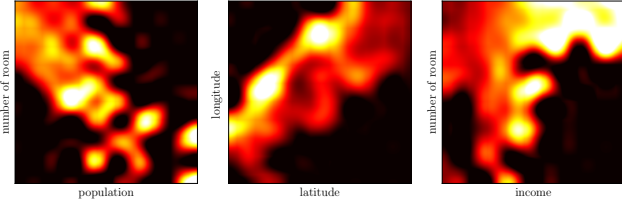
Figure 9 to Figure 11 show top 3 interactions for each dataset (California housing dataset, Bike Sharing dataset, and energy efficiency dataset) in our experiments, while in the main text we only select one interaction for each. We can see that all detected interactions are meaningful.

### 6 Determine the optimal number of clusters

Figure 12 shows  $\Delta_M^2$  as a function of  $M$  on each dataset, thus it determines the optimal number of clusters in each case. We can find that for the most cases,  $\Delta_M^2$  becomes stable before  $M = 20$ .



**Figure 8.** 95% CIs of EAH on simulated dataset with mean-field BNN.

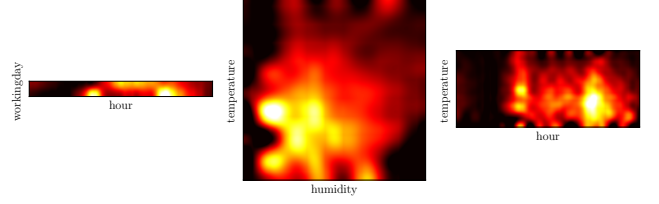


**Figure 9.** Top 3 interactions for California housing dataset.

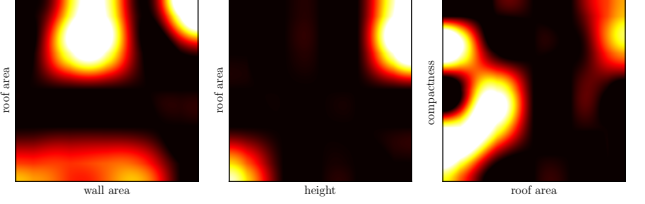
## 7 Permutation Distribution of Maximum Interaction effects

In this section, we show permutation distribution of the maximum interaction effect for each dataset in Figure 13.

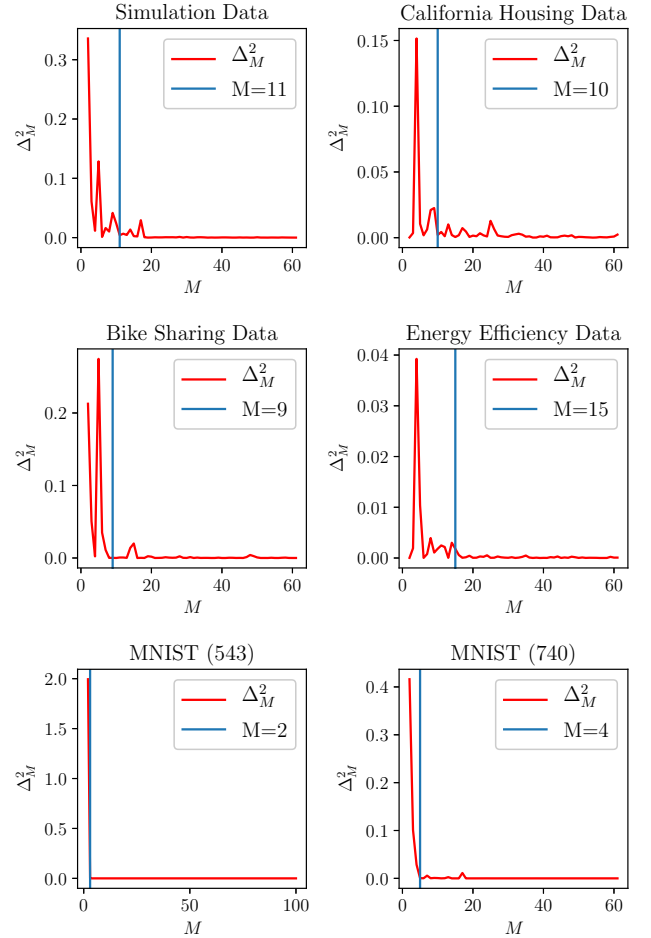
For each dataset, we permute the target and compute the maximum interaction effect on the permuted dataset, then we can obtain the permutation distribution of the maximum interaction effect. Then we calculate the p-value of top-3 estimated interaction effects for each dataset under its corresponding permutation distribution. KDE stands for the Gaussian kernel density estimation of the empirical permutation distribution.



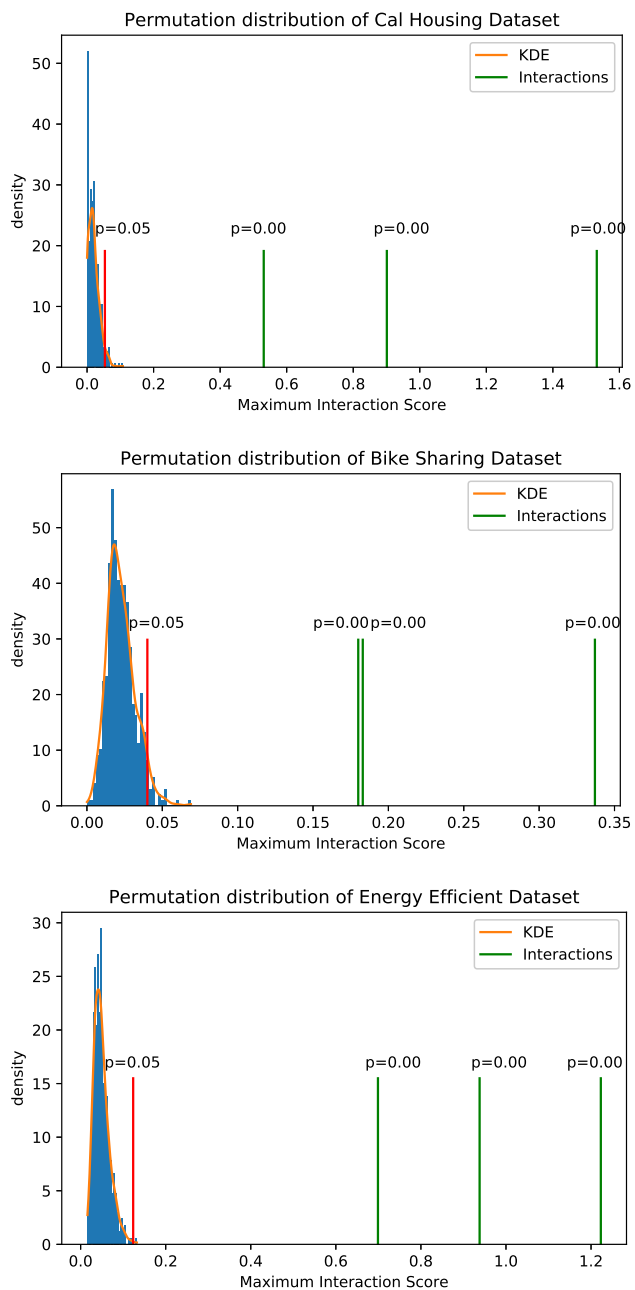
**Figure 10.** Top 3 interactions for bike sharing dataset.



**Figure 11.** Top 3 interactions for energy efficiency dataset



**Figure 12.** Optimal numbers of clusters based on weighted rank distance.



**Figure 13.** Permutation distributions of the maximum interaction effect on three real-world datasets.