To all reviewers. We would like to sincerely thank you for your important ideas and constructive comments. First, we would like to clarify that B-RAI [24] is a recently proposed algorithm for estimating the posterior *probability* of causal relations among observed variables. It is not related to the deep learning domain. The B2N algorithm [25], introduces principles for converting *fixed* causal relations into a deep neural network (NN) structure. Simply using B-RAI in B2N is not trivial, and we introduced three important algorithmic contributions: a principled method for weight sharing among ensemble-networks (Algorithm 1), a scheduling algorithm for parameter learning in Sec. 3.2.2 (a method for jointly sampling parameters for update), and an anytime stochastic inference & uncertainty estimation (e.g., expected entropy) in Sec. 3.2.3. In addition, a theoretical contribution is the suggestion of a lurking confounder (Fig. 1), and an approach for integrating it out (Eq. 6). Thus, making the prior distribution over parameters of the discriminative NN dependant on the underlying generative mechanisms of unlabeled data. Importantly, this confounder is *missing* (and untreated) from traditional Bayesian deep learning approaches, and to the best of our knowledge, our work is the first to address it. We will clearly state these contributions in the paper. In our experiments we distilled the contribution of the BRAINet structure and training & inference algorithms. Distilling was done by an ablation study, and the use of a standard loss function for OOD and calibration experiments (we also demonstrated training BRAINet with OOD-detection loss function). We added all required answers and clarifications, which will improve the quality of the final version, and appreciate if you will up your scores accordingly.

To reviewer #1. As you suggest, we will define B2N, RAI, and GGT in the paper. BRAINet demonstrates a clear advantage over Deep-Ensembles in Fig. 5, and over MC-dropout & Bayes-by-Backprop in Fig. 6. A clear advantage in both classification accuracy and calibration. Shaded areas represent 2σ. An ensemble of 15 (last point on the curve, Fig. 5), having a total of 3.6M parameters, is equivalent or worse than BRAINet with 3X fewer parameters (6th point on the graph). We will use a different scale for the axes in Fig. 5 to improve clarity. Compared to MC-dropout and Bayes-by-Backprop, for the same model size, BRAINet with 2 forward passes significantly outperforms 15 forward passes in MC-dropout and Bayes-by-backprop. Furthermore, a forward pass in MC-dropout uses most of the network weights ($\geq$50%), whereas BRAINet uses a fraction of it ($<$10%). OOD Detection: other baselines, such as [3][18], rely on defining a loss function, specifically crafted for OOD detection, often using some OOD data during training [18]. Optimizing for a specific loss hinders other objectives, e.g., accuracy and calibration. In all our experiments we used the common cross-entropy loss, except for the results in Table 2 where we used [3] to demonstrate that BRAINet can be coupled with other methods to further improve their performance. BRAINet is not targeted specifically at OOD detection, and in fact may even improve accuracy and calibration in addition to being able to detect OOD (Table 1). In this manner, BRAINet is similar to other ensemble and Bayesian NN methods against which we compared. Thank you for your comments, we will improve the clarity accordingly.

To reviewer #2. Clarity section: **(1)** Our method adds predictive uncertainty to existing topologies by using their feature extractors (embeddings) and replacing their heads. Rohekar et al., (2018) demonstrated that the B2N algorithm can replace the last several layers (convolutional and MLP) of existing networks while maintaining accuracy. Relying on their findings, we followed the same practice but added a new capability: (anytime) uncertainty estimation. We will clarify this in the experiments section. **(2)** You are correct. As you proposed, we will include the proposed high-level outline. **(3)** In section 3.2.2, sc is the score produced by B-RAI. It is the log of the posterior probability of the structure. This allows marginalizing out θ (approximating Eq. 6). Quality section: **(1)** Yes. The model size of a single network is fixed $240K$, with 200 neurons in each layer. The difference along the X-axis (Fig. 5) is due to a varying ensemble size. A sampled network from BRAINet is $\geq 240K$, and the max number of neurons in a layer is ≤ 200. We will include these details in the appendix. **(2)** ECE and calibration diagrams for section 4.2 are under work (currently a clear advantage of BRAINet) and will be included in the appendix. **(3)** No. We have not tried applying dropout before every layer. This can further improve the results for both BRAINet (adding dropout to the feature extraction layers) and MC-dropout. We preferred a setup in which we can distill the effect of BRAINet without other factors. We will clarify in the paper that further improvements can be made using you suggestion. **(4)** As requested, here is the table. Thank you for identifying typos.

DATASET	MODEL	SGD NLL/ACC	SWA NLL/ACC	SWAG-DIAG NLL/ACC	SWAG NLL/ACC	KFAC-LAPLACE NLL/ACC	SWA-DROPOUT NLL/ACC	SWA-TEMP NLL/ACC	BRAINET NLL/ACC
CIFAR-10	VGG-16	0.3285/93.17	0.2621/93.61	0.22/93.66	0.2016/93.60	0.2252/92.65	0.2328/93.23	0.2481/93.61	**0.2011/93.81**
CIFAR-10	PRERESNET-164	0.1814/95.49	0.145/96.09	0.1251/96.03	**0.1232**/96.03	0.1471/95.49	0.127/**96.18**	0.1347/96.09	0.1245/95.90
CIFAR-10	WRN28X10	0.1294/96.41	0.1075/96.46	0.1077/96.41	0.1122/96.32	0.121/96.17	0.1094/96.39	0.1064/96.46	**0.1044/96.48**
CIFAR-100	VGG-16	1.7308/73.15	1.278/74.30	1.0163/74.68	0.948/74.77	1.1915/72.38	1.1872/72.50	1.0386/74.30	**0.0935/74.96**

To reviewer #3. Answers to the clarifying questions: **(1)** Yes. For in-distribution, ϕ is conditionally independent of X given θ, i.e., $p(\phi|\theta, X_{\text{in}}) = p(\phi|\theta)$. For OOD this does not hold, $p(\phi|\theta, X) = p(\phi|\theta, X_{\text{OOD}})$. (**1a**) Yes, for OOD $p(\phi|\theta, X_{\text{OOD}})$ is expected to spread the probability mass across ϕ due to the direct dependency of ϕ on arbitrary X_{OOD} (which is also the case for in-distribution in common methods, e.g., MC-dropout, missing the θ confounder). (**1b**) In practice, this results in inconsistent responses by the sampled networks, i.e., $p(Y|X_{\text{OOD}})$ is expected to be spread. **(2)** Yes. That is accurate with one addition. Our method estimates the (posterior) probability of each structure (a score) given unlabeled data (X). This leads to Bayesian model averaging, where each structure is weighted differently. (**2a**) In the space of structures, the posterior distribution (given X, a prior for the parameters) may have multiple modes. We wanted to sample structures from this distribution; however we further wanted to exploit any structural similarities (e.g., distinct structures sampled from the same mode), and couple them into a hierarchy. This led to parameter sharing. In our method, considering the deeper layer is equivalent to looking at this space with less resolution, blurring the modes thereby merging close modes. Therefore, deeper layers share parameters. Layers closer to the input have distinct structures accounting for the higher resolution view of this space. Thus, as you commented, BRAINet captures this space efficiently, in addition to it being the only method that samples structures from this space ($p(\theta|X)$). Answers to the Improvements section: 1-**(I)** will be clarified based on you clarifying questions in the previous section. And your suggestions for specific lines in the text. 1-**(II)** We will clarify that the head of existing networks is learned while using their feature extraction layers. **(2)** As suggested, we will improve the related work section. NN structure learning from the previous wave (late 80's early 90's) were mostly heuristic, greedy search algorithms. Importantly, researcher of that wave didn't enjoy the advances in causal discovery (90's) and probabilistic machine learning on which our work relies. **(3)** Shortcomings of our method: currently it is suitable for learning structures for features rather than pixels. **(4)** In general, in all experiments we used default hyper parameters, with a fixed learning rate. We will provide all the required details.