DENN: Diverse Extrapolation in Neural Networks

In this work, we tackle the challenge of producing *diverse* functions from a common neural network (NN) architecture that, given the same training samples, fit well the training data while disagreeing on the value to predict for inputs that lie out-of-distribution (OOD). Existing approaches [5, 10, 2, 7] often under-estimate uncertainty, especially OOD, and struggle to carry over a prior uncertainty during training; instead, we force NNs to extrapolate more diversely, resulting in an ensemble with higher uncertainty for OOD inputs.

1 Proposed approach

Objective Let $\{f_k\}_{k\in 1\dots K}$ be an ensemble of K NNs. Since diversity in the weight space does not necessarily translates into diversity in the function space [1], we work directly in the latter by training predictors f_k which outputs are encouraged to differ in regions with lower density of training samples.

Training NNs with a repulsive constraint We increase the diversity of $\{f_k\}_{k\in 1\dots K}$ by penalizing the similarity between any f_k and a reference function g on OOD points stored in $\mathbb{X}$. We combine the desired low error on the training set $\mathcal{D}$ and a penalization on the diversity into our proposed training loss:

$$\mathcal{L}(f_k; g, \mathcal{D}, \mathbb{X}) = \underbrace{\frac{1}{n} \sum_{(x_i,y_i)\in\mathcal{D}} d(f_k(x_i), y_i))}_{\text{error loss } \mathcal{L}_{\text{err},\mathcal{D}}} + \underbrace{\frac{\lambda}{n_{\mathbb{X}}} \sum_{x\in\mathbb{X}} k(f_k(x), g(x))}_{\text{similarity loss } \mathcal{L}_{\text{sim},\mathbb{X}}} \tag{1}$$

where $\lambda \geq 0$ denotes a trade-off hyperparameter to be chosen. Note that the error loss and the similarity loss serve opposite objectives if $\mathbb{X} \cap \mathcal{D} \neq \emptyset$ which leads to a label smoothing effect [6]. Outside of $\mathcal{D}$, the similarity loss is the only one active: this induces a *repulsion* between f and g.

Choosing repulsive points Ideally, $\mathbb{X}$ should be sampled from the manifold where the observations lie, out of the training distribution. Since we don't have access to this exact distribution, we sample repulsive points either by adding noise to the training data to get access to the boundary of the training distribution [6], or by choosing training points from similar datasets, for instance to exploit the structure of images.

2 Case studies

For each experiment, g is a NN with the same architecture as the NNs in the DENN ensemble, trained beforehand using Eq. 1 with $\lambda = 0$. The NNs are trained with the MSE loss for experiments (1) and (3) and the cross-entropy loss for the classification task (2).

(1) Low-dimensional regression We evaluate in Fig. 1 the predictive uncertainty of different methods, that ideally should be able to cover the ground truth function. DENN uses a radial basis function between the outputs of f and g as k, evaluated at repulsive points generated at each training step by adding Gaussian noise to training points [6]. The DENN approach covers the second mode of the ground truth while concurrently maintaining a low uncertainty on the training set, unlike the other methods.

(2) Classification We compare the entropy of the prediction probabilities (averaged over the ensemble) for each input of different datasets in Fig. 2. DENN uses $\exp\left(-\frac{|H(g(x),f(x))-H(f(x),f(x))|^2}{2\sigma^2}\right)$ as k and samples the repulsive points from the FashionMNIST dataset [12]. The average prediction of DENN shows higher entropy (or uncertainty) on unrelated images from unseen datasets than a deep ensemble, without hurting our approach's generalization power on MNIST.

(3) Safe imitation learning An important challenge in imitation learning, where an agent learns to map states to actions using human experience and his own, is to detect which encountered states are outliers. In Fig. 3, we evaluate DENN's ability to detect such states on a pixel version of the MuJoCo Reacher task [11] (red sphere) and variants where the target is changed. DENN uses the same similarity loss as in (1) and samples

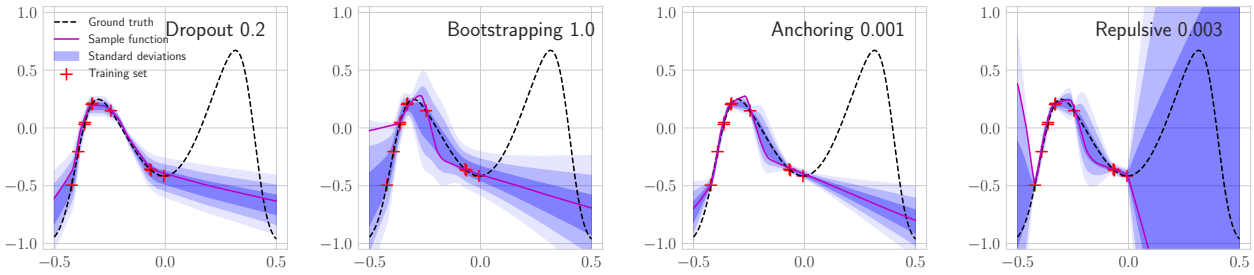


Figure 1: Comparison of the empirical (over 50 sampled functions) posterior predictive distribution for dropout at inference [5], input bootstrapping [4], anchoring [10] and DENN.

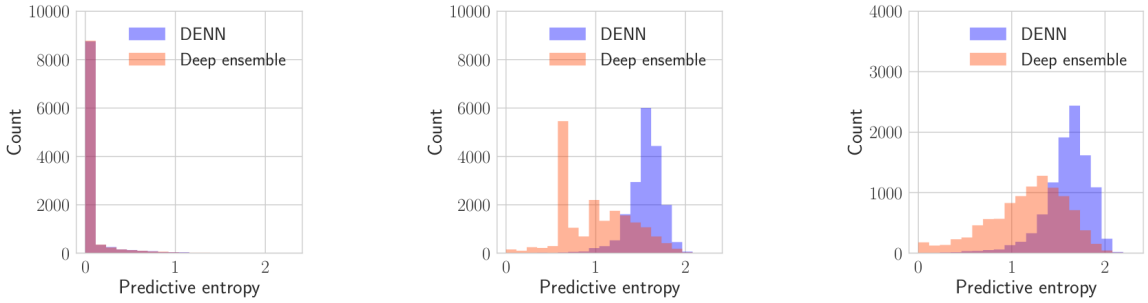
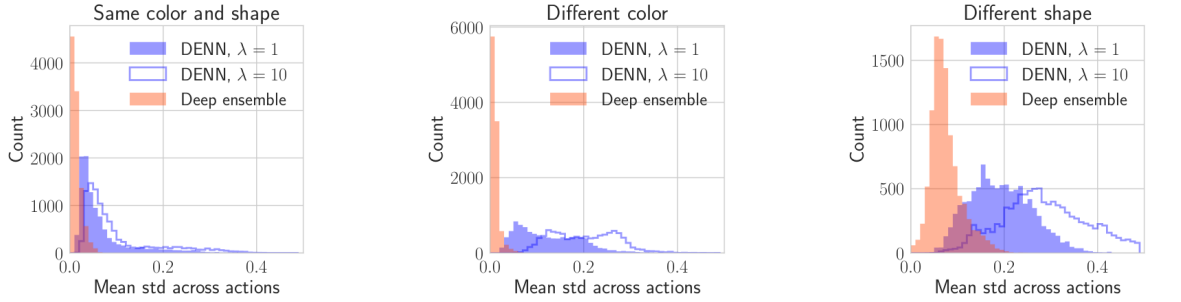


Figure 2: Histograms of per data point predictive entropy for DENN and a deep ensemble [7] on MNIST [8] (left, in distribution) and notMNIST [9] and KuzushijiMNIST [3] (resp. middle and right, OOD). More mass on the right means higher uncertainty on the inputs.



(a) Eval. on red sphere dataset (b) Eval. on blue sphere dataset (c) Eval. on red rectangle dataset

Figure 3: Histograms of the predictive uncertainty over actions. Higher values indicate higher uncertainty over the action to take.

repulsives points from a green target dataset. It shows lower (a) but sufficient generalization to separate the red dataset from the blue dataset, unlike the baseline, and is more uncertain than the deep ensemble on OOD inputs of different color (b) or shape (c), hinting that setting a repulsive constraint on one attribute – the color – of the target affected the whole representation of the target, including its shape.

3 Conclusion

In this work, we described a method aimed at obtaining more diverse NNs by using a modified loss function enacting directly in the function space. Future work includes working on more principled ways of sampling repulsive points and extending the approach to reinforcement learning.

References

[1] Blei, D.M., Kucukelbir, A., McAuliffe, J.D.: Variational inference: A review for statisticians. Journal of the American Statistical Association pp. 859–877 (2017)

[2] Blundell, C., Cornebise, J., Kavukcuoglu, K., Wierstra, D.: Weight uncertainty in neural network. In: International Conference on Machine Learning. pp. 1613–1622 (2015)

[3] Clanuwat, T., Bober-Irizar, M., Kitamoto, A., Lamb, A., Yamamoto, K., Ha, D.: Deep learning for classical japanese literature. arXiv:1812.01718 (2018)

[4] Efron, B.: The jackknife, the bootstrap, and other resampling plans. Siam (1982)

[5] Gal, Y., Ghahramani, Z.: Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In: International Conference on Machine Learning. pp. 1050–1059 (2016)

[6] Hafner, D., Tran, D., Irpan, A., Lillicrap, T., Davidson, J.: Reliable uncertainty estimates in deep neural networks using noise contrastive priors. In: Submitted to International Conference on Learning Representations (2019), under review

[7] Lakshminarayanan, B., Pritzel, A., Blundell, C.: Simple and scalable predictive uncertainty estimation using deep ensembles. In: Advances in Neural Information Processing Systems. pp. 6402–6413 (2017)

[8] LeCun, Y.: The mnist database of handwritten digits (1998)

[9] notMNIST link: https://yaroslavvb.blogspot.com/2011/09/notmnist-dataset.html. Dataset (2011)

[10] Pearce, T., Anastassacos, N., Zaki, M., Neely, A.: Bayesian inference with anchored ensembles of neural networks, and application to reinforcement learning. In: International Conference on Machine Learning Workshop on Exploration in Reinforcement Learning (2018)

[11] Todorov, E., Erez, T., Tassa, Y.: Mujoco: A physics engine for model-based control. In: 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems. pp. 5026–5033. IEEE (2012)

[12] Xiao, H., Rasul, K., Vollgraf, R.: Fashion-MNIST: A Novel Image Dataset for Benchmarking Machine Learning Algorithms. Dataset (2017)