Review of Fort et al. (2020)

Apostolos Psaros

1 Summary

In this paper the authors show empirically what has been conjectured before: Repeated optimization of a NN with different initialization leads to different weight vectors with similar training loss. Also, the NN evaluated at these different optima gives very different predictions. This is called diversity/disagreement of predictions.

On the other hand, recent techniques that use a single SGD to sample more weights for performing approximate Bayesian inference lead to very similar (not diverse) predictions. In this paper they study 4 such techniques which I will be calling "4 sampling techniques" for convenience.

We knew that the 4 sampling techniques give similar predictions, but here is more empirical evidence. Note, however, that the 4 sampling techniques come at a very small or no additional cost as compared to a single optimization. Note also that if we use a cyclical learning schedule we may be able to visit other modes as well: see empirical evidence in Section 4.5 in Fort and Jastrzebski (2019). However, Fort et al. (2020) have not tried to use cyclical learning schedule.

Note: These results are for classification but I believe similar results we can get for regression. I have not looked into it yet. See Figure 11 in Wilson and Izmailov (2020) for a starting point in function diversity for regression.

2 Two definitions

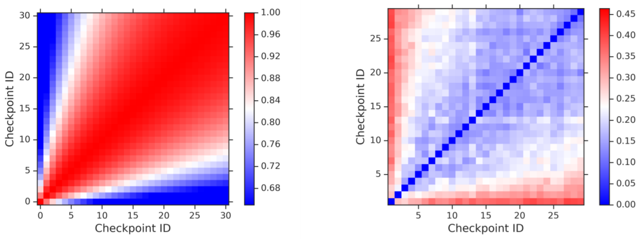
We will use cosine similarity in weight space (w-space) given as

$$cos(\theta_1, \theta_2) = \frac{\theta_1^T \theta_2}{\|\theta_1\| \|\theta_2\|} \tag{1}$$

and disagreement of 2 models in function space (f-space) given as

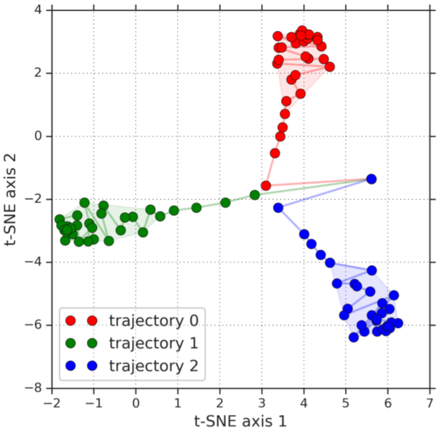
$$\frac{1}{N} \sum_{n=1}^{N} \mathbf{1}[f(x_n; \theta_1) \neq f(x_n; \theta_2)] \tag{2}$$

3 Discussion of figures



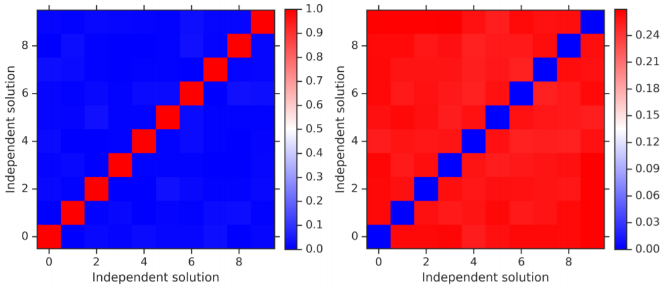
(a) Cosine similarity of weights (b) Disagreement of predictions

Fig. 1. Figure 2 a and b from the paper: a) Cosine similarity of weights along an optimization trajectory. After checkpoint 10, weights within a trajectory are very similar. b) Disagreement of functions obtained along an optimization trajectory. After checkpoint 10, models predict very similar labels. Predictions are not very diverse.



(c) t-SNE of predictions

Fig. 2. Figure 2 c from the paper: 2-D t-SNE projections of predictions obtained from weights along 3 different optimization trajectories. Projections corresponding to the same trajectory (color) are very close to each other but far from predictions from different trajectories. This shows that weights within a trajectory give rise to very similar predictive models but different trajectories give rise to very different models.



(a) Results using *SmallCNN*

Fig. 3. Figure 3 a from the paper: Left: Cosine similarity between optima (final weights in each trajectory) from different optimization trajectory. Different optima are almost orthogonal to each other. Right: Corresponding predictive models/functions obtained disagree on predictions.

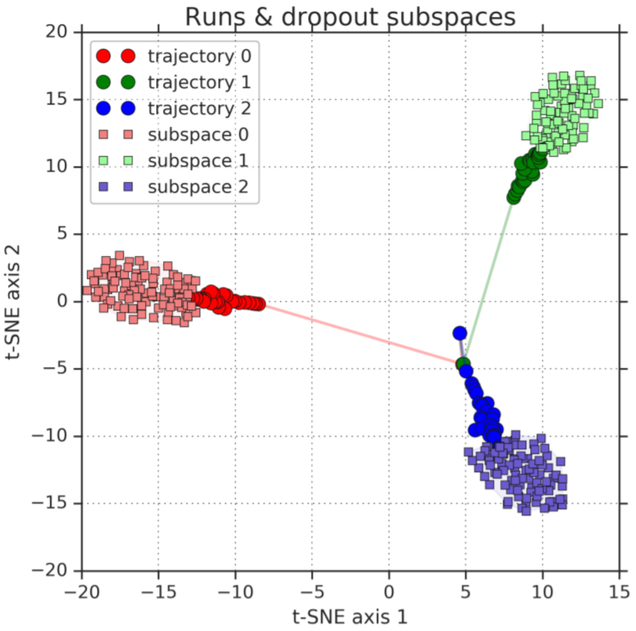


Fig. 4. Figure 4 from the paper: 2-D t-SNE projections of predictions obtained from 3 different optimization trajectories. Additional predictions (less opacity in figure) are added by using one of the 4 sampling techniques (here dropout). Remember these 4 sampling techniques at zero or low cost sample more weights along a single optimization trajectory. It is seen that the sampling techniques only help better explore a single mode and the predictions are not very diverse.

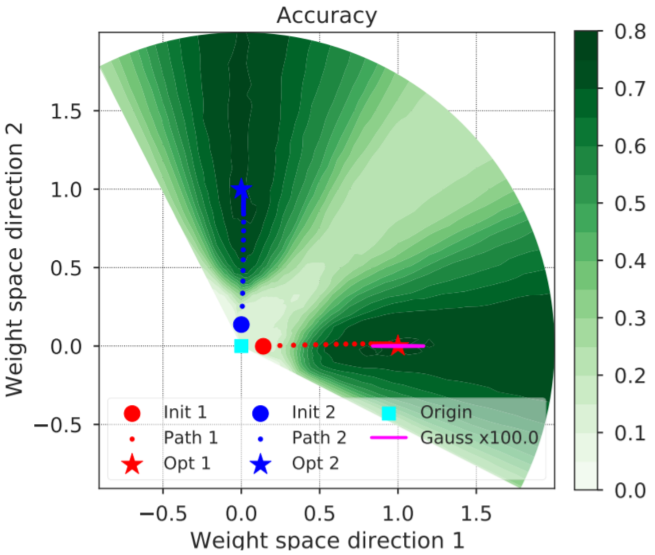


Fig. 5. Figure 5 left from the paper: Having 3 points in the w-space we can create a unique 2-D plane passing through these 3 points. See for example Li et al. (2018) and *losslandscape.com* where they create a plane connecting one weight plus 2 random directions from it. In this figure the 3 points for creating a plane are the origin and the 2 optima (stars) found by 2 different optimization trajectories. The dots show the 2 trajectories as projected on this plane (as far as I understand it). It seems that because of high dimensionality the points along the optimization trajectories fall very close to the lines connecting the origin and the optima. It is also seen that the 2 optima achieve similar training loss while if we tried to connect the 2 optima with a straight line we would have to pass through a region of high loss. Connecting "tunnels" between 2 optima can be found with the techniques of Garipov et al. (2018) and Draxler et al. (2018). See also Fort and Jastrzebski (2019) for tunnels connecting M modes. These tunnels are useful because we can use the different weights in the tunnel for ensembling. In order to produce this plot we need to go at each pixel in the plot (corresponding to a weight vector) and compute its training loss. Finally, with pink you can see the different weights obtained by using one of the 4 sampling techniques. They are very close to the optimum of the 1st trajectory and they are orthogonal to the optimum of the 2nd trajectory.

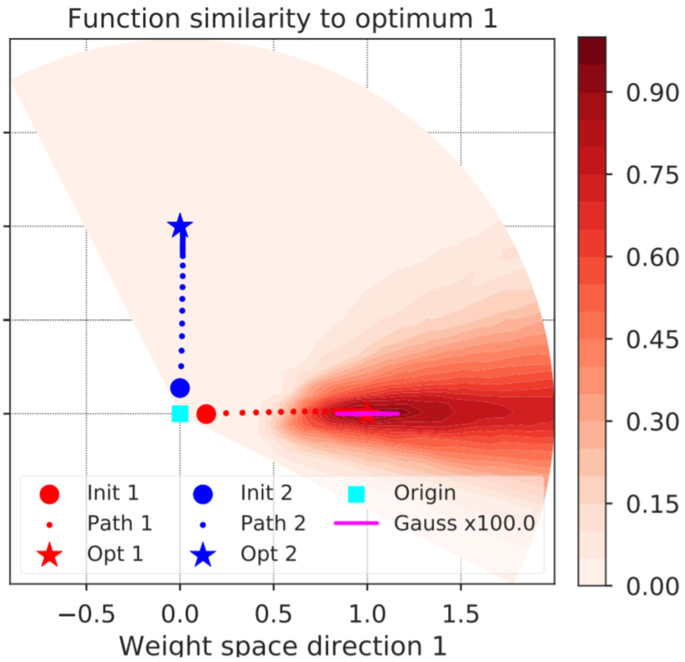


Fig. 6. Figure 5 middle from the paper: Function similarity of each model in the plane as compared to the 1st optimum. This shows that when we leave the low loss valley around the weight the predictions are very different. Finally, with pink you can see the function similarities of the weights obtained by using one of the 4 sampling techniques. The functions obtained are very similar to the function obtained using the optimum of the 1st trajectory.

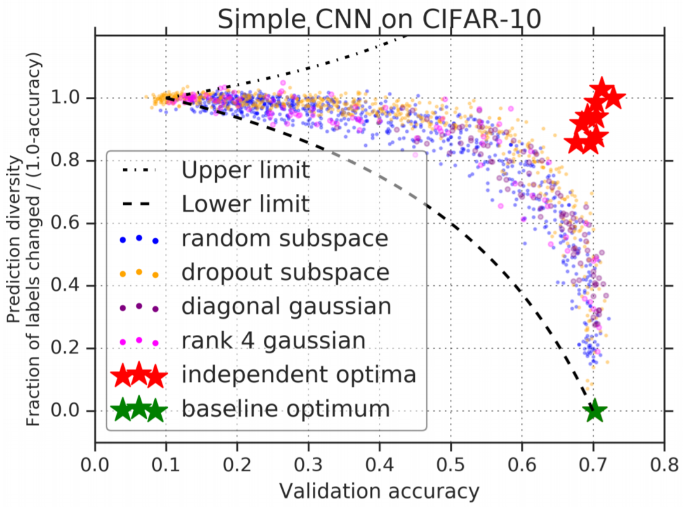


Fig. 7. Figure 6 left from the paper: Y-axis is function disagreement of each sample with baseline optimum. Green star shows the single optimum: zero diversity because it is compared with itself. Dots show diversity (function disagreement) vs accuracy for the samples obtained via the 4 sampling techniques. For example, you can get more diverse predictions using the 4 sampling techniques by increasing the noise. But this comes at the cost of accuracy because you leave the loss valley shown in figure 5 of the paper and you visit a high-loss region. With red the predictions by different trajectories are shown. We can see that there is no trade off for different trajectories. We have similar accuracy and very different predictions.

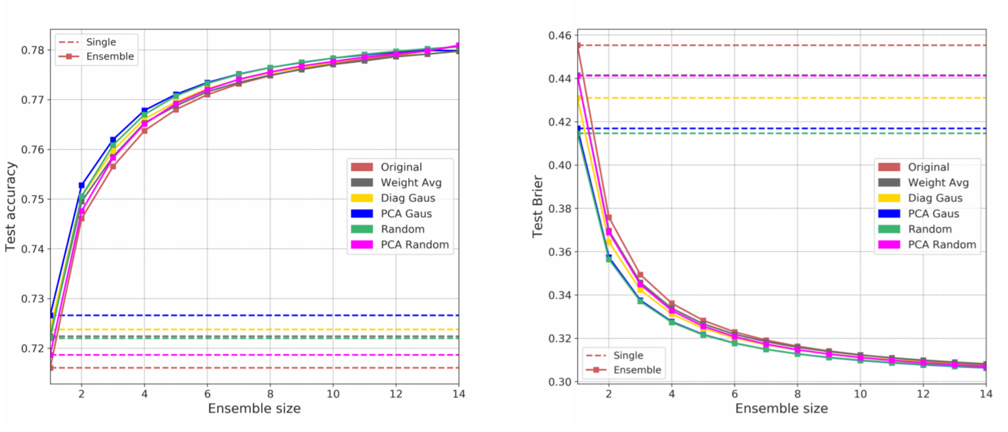


Fig. 8. Figure 8: Left: Here the authors use deep ensembles plus additional samples via the 4 sampling techniques. Remember each optimum in the ensemble has an optimization trajectory that we can use to sample more weights using one of the 4 sampling techniques. What they do here is similar to Multi-SWAG proposed in Wilson and Izmailov (2020). It shows that ensembling plus sampling can be a little better than ensembling. Ensembling is for sure better than just sampling however in terms of generalization. Right: Same for calibration (uncertainty quality). Small Brier score means better calibrated uncertainty.

References

Draxler, F., Veschgini, K., Salmhofer, M., and Hamprecht, F. A. (2018). “Essentially No Barriers in Neural Network Energy Landscape”. *arXiv preprint arXiv:1803.00885.*

Fort, S., Hu, H., and Lakshminarayanan, B. (2020). “Deep Ensembles: A Loss Landscape Perspective”. *arXiv:1912.02757 [cs, stat]*. arXiv: `1912.02757 [cs, stat]`.

Fort, S. and Jastrzebski, S. (2019). “Large Scale Structure of Neural Network Loss Landscapes”. *Advances in Neural Information Processing Systems 32*. Ed. by H. Wallach et al. Curran Associates, Inc., pp. 6709–6717.

Garipov, T., Izmailov, P., Podoprikhin, D., Vetrov, D. P., and Wilson, A. G. (2018). “Loss Surfaces, Mode Connectivity, and Fast Ensembling of Dnns”. *Advances in Neural Information Processing Systems*, pp. 8789–8798.

Li, H., Xu, Z., Taylor, G., Studer, C., and Goldstein, T. (2018). “Visualizing the Loss Landscape of Neural Nets”. *Advances in Neural Information Processing Systems*, pp. 6389–6399.

Wilson, A. G. and Izmailov, P. (2020). “Bayesian Deep Learning and a Probabilistic Perspective of Generalization”. arXiv: 2002.08791.