Supplemental Material: Adaptive, Distribution-Free Prediction Intervals for Deep Neural Networks

Danijel Kivaranovic
ISOR
University of Vienna

Kory D. Johnson
Institute for Statistics and Mathematics
Vienna University of Economics and Business

Hannes Leeb
ISOR, DS@UniVie
University of Vienna

In this supplement, we present further results on simulated data. Figure 1 shows how the algorithms behave as a function of the number of observations. We see that our two methods, *pav* and *conf-nn*, control average coverage for all n (left panel). Also both methods approximate the oracle as n grows (middle and right panel). We observed that the intervals of *pav* are more conservative and slightly longer than the intervals of *conf-nn*. The prediction intervals of *neg-ll* are very wide for smaller sample sizes but perform similarly in larger samples. The *bayes* method performs similarly well as our two methods. The *conf-fw* and *high-q* methods also control average coverage, however the intervals are slightly longer (middle panel) compared to the intervals of our two methods. The reason is that both methods do not adapt to the heteroskedasticity of the data. Only *qreg-un* does not control average coverage (left panel). This demonstrates that adjustment of the intervals is necessary to control average coverage. While our methods, *pav* and *conf-nn*, do not assume normality, we achieve results that are close to the optimal prediction intervals of the oracle.

In Figure 1 of the main paper, we show more detailed results for sample size equal to 100,000. In Figure 2 we show the same simulation results for sample size equal to 5,000 (top) and 47,222 (bottom). In the left panel, we see that even in small samples the methods *pav* and *conf-nn* provide coverage close to the the nominal level for a wide range of $\beta' X$. In the middle panel, we see that the relative length approaches 1 (i.e. it approaches the length of the oracle) as the sample sizes grows from 5,000 to 47,222. In the right panel, we see that the combined prediction error approaches 0 as the sample size increases, demonstrating that it is accurately estimating the oracle. On the other hand, the methods *conf-fw* and *high-q* undercover when $\beta' X$ is large and overcover when $\beta' X$ is small. The reason for this is that these two methods do not adapt to the heteroskedasticity of the data as seen in the middle and right panel. We see that the method *neg-ll* performs considerably worse when the sample size is small, but is comparable to *pav* and *conf-nn* when the sample size is larger. This suggests that this method needs more data to accurately estimate the distribution of the data. This is surprising, since *neg-ll* assumes that the data is Gaussian, which is the case in our artificial data example. We again see that *qreg-un* is consistently below the nominal level due to overfitting. The *bayes* method performs similary to *pav* and *conf-nn*, but we again note that *bayes* implicitly assumes that the data is Gaussian.

Figure 3 explores the effect of architectures and loss function on the performance of our procedure. We use the same simulated data with sample size fixed to 50,000. For simplicity we only consider the conformal intervals given elsewhere as *conf-nn*. We considered two architectures: a network with one hidden layer (dep1) and a network with two hidden layers (dep2). We also considered two different loss functions: the quantile loss function of the main paper (loss_l1) and the modified version of it where we square each term in the loss function (loss_l2). The squared version is supposed to estimate the conditional mean instead of the conditional median. However, note that in our Gaussian setting the conditional mean is equal to the conditional median. Therefore it is not surprising that there is little difference in performance as a result of switching to the $l2$-loss. On the other hand, we observed that the networks with two hidden layers perform worse than the networks with one hidden layer. In the left panel, we see that these prediction intervals adapt much worse to heteroskedasticity of the data (they overcover considerably when $\beta' X$ is small and undercover when $\beta' X$ is large). Therefore, the intervals are either much longer or much shorter (middle panel). This can be explained by the fairly simple structure of the data which does not require more complex neural networks.

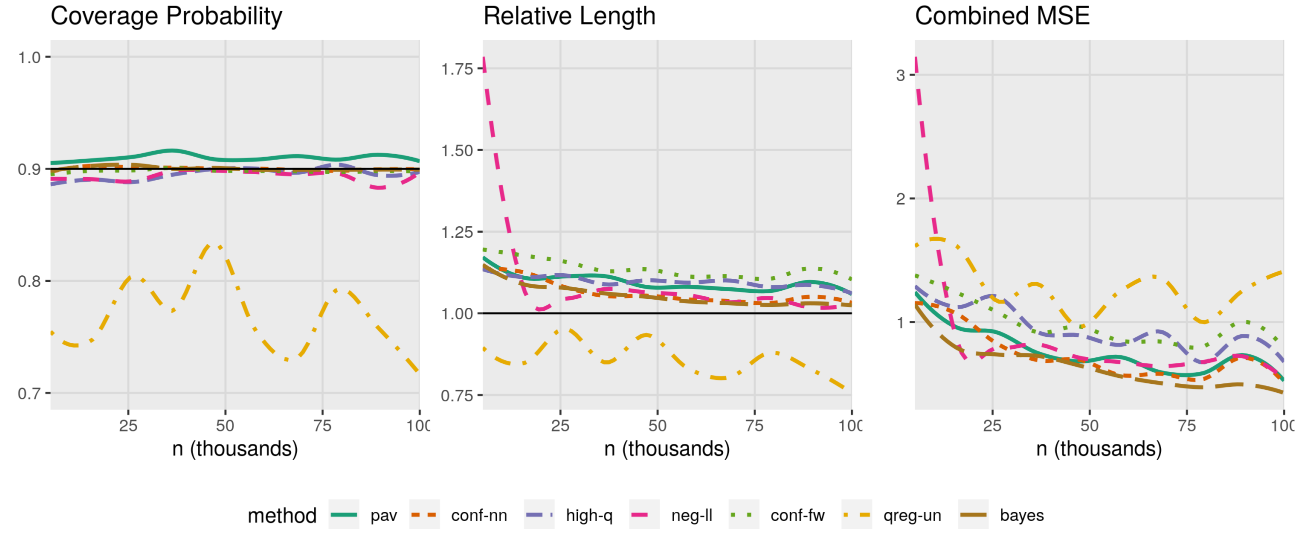

Figure 1: Asymptotic performance of procedures. Both relative length and combined MSE are computed relative to the oracle predictions and prediction intervals.

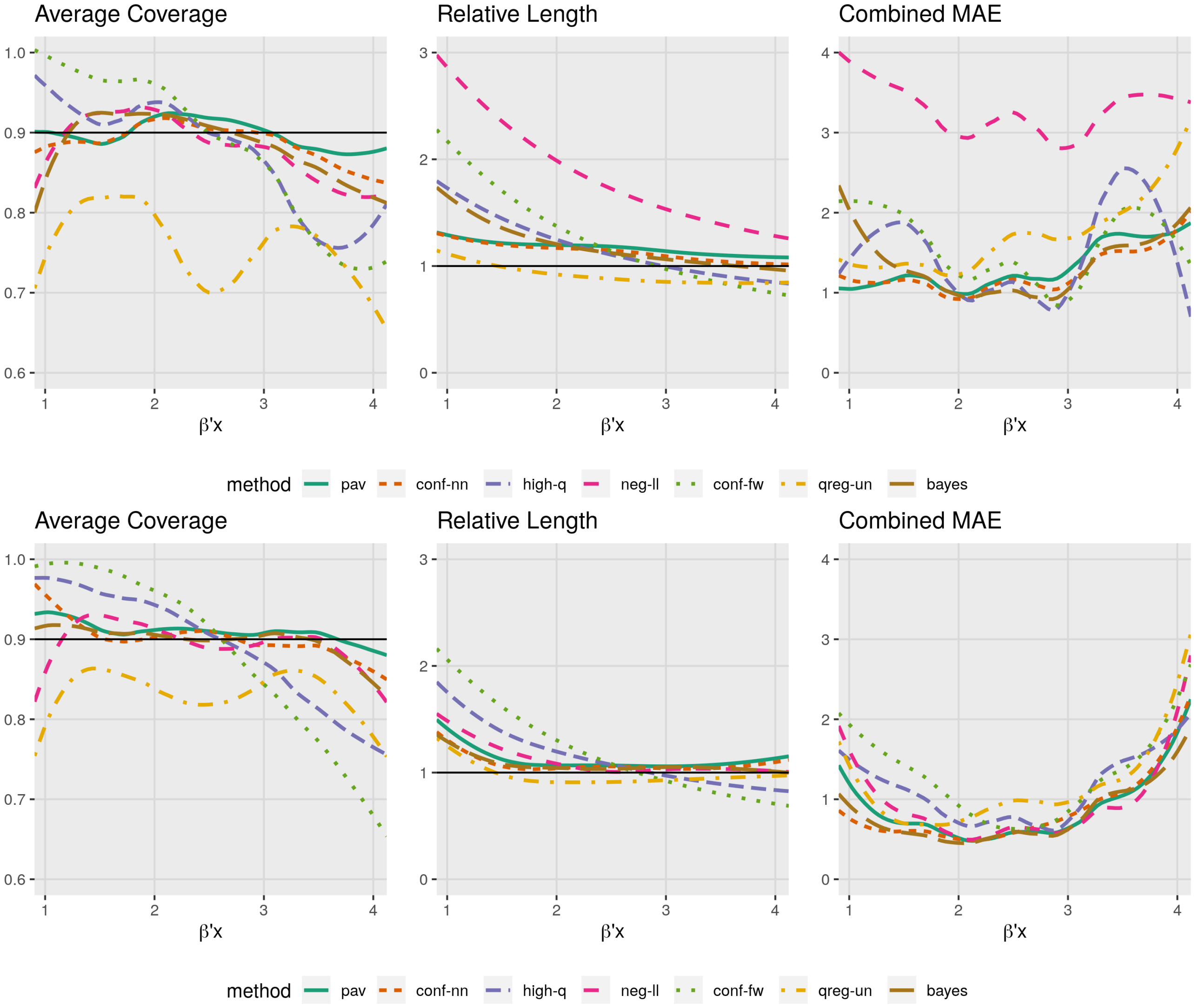

Figure 2: Simulation summaries for $n = 5,000$ (top) and $n = 47,222$ (bottom).

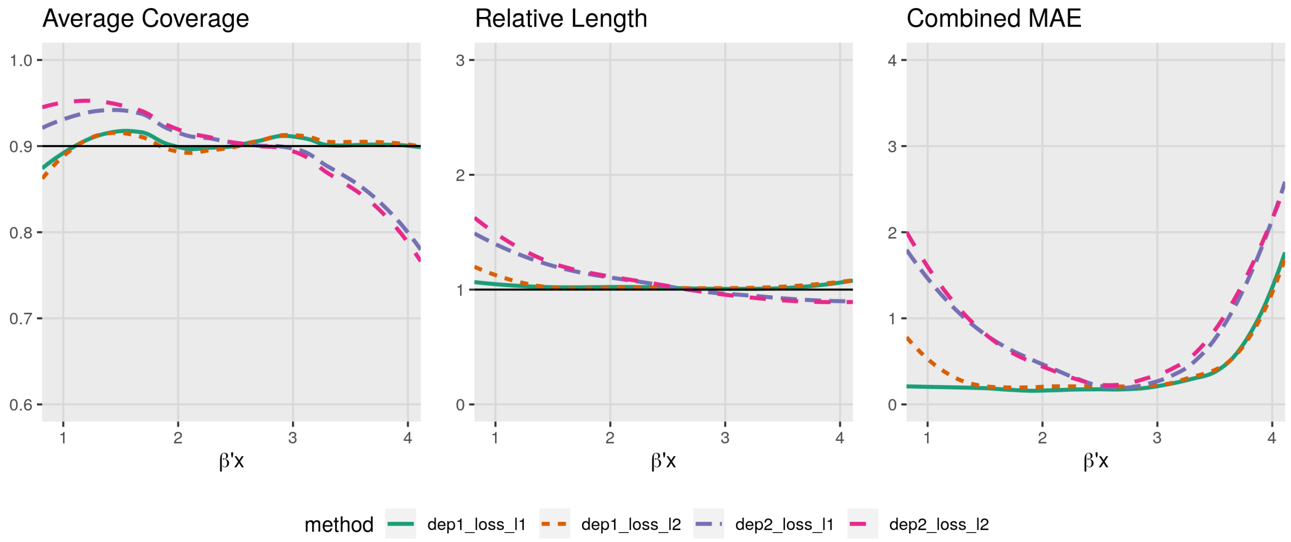

Figure 3: Results for different architectures and loss functions. The networks with one and two hidden layers are denoted by dep1 and dep2, respectively. The loss functions with the $l1$-type and $l2$-type loss are denoted by loss_l1 and loss_l2, respectively. Each measure is plotted as a function of the true linear component, $\beta' X$. Both length and combined mean absolute error (MAE) are computed relative to the oracle predictions and prediction intervals.