Decoupling Feature Extraction and Classification Layers for Calibrated Neural Networks

Mikkel Jordahn [1]   **Pablo M. Olmos** [2]

Abstract

Deep Neural Networks (DNN) have shown great promise in many classification applications, yet are widely known to have poorly calibrated predictions when they are over-parametrized. Improving DNN calibration without comprising on model accuracy is of extreme importance and interest in safety critical applications such as in the healthcare sector. In this work, we show that decoupling the training of feature extraction layers and classification layers in over-parametrized DNN architectures such as Wide Residual Networks (WRN) and Vision Transformers (ViT) significantly improves model calibration whilst retaining accuracy, and at a low training cost. In addition, we show that placing a Gaussian prior on the last hidden layer outputs of a DNN, and training the model variationally in the classification training stage, even further improves calibration. We illustrate these methods improve calibration across ViT and WRN architectures for several image classification benchmark datasets.

1. Introduction

Classification using machine learning is widely applied in many industries, ranging from application in medical diagnostics (Shen et al., 2019; Ahsan et al., 2022) to being a crucial part of self-driving car systems (Chen & Huang, 2017; Gupta et al., 2021). In applications with a safety or ethical aspect, it is not only the accuracy of the applied model that matters but also the calibration of the model. To give an example, if a machine learning model is applied in medical diagnostics, we simply do not just want to rely on the model's prediction, but rather the probability of this

prediction. With calibrated models, it is possible to refer to a doctor when model prediction uncertainty is high, which is not the case for poorly calibrated models.

Throughout this work, we refer to calibration in terms of the probabilities assigned to a prediction, i.e., if a perfectly calibrated model assigns probability p of class C in A events, we would expect p of those A events to be class C. In the case of deep neural networks (DNN) for classification, it is well known that over-parameterized models under cross-entropy training tend to be overconfident (Guo et al., 2017). Some argue that modern neural network architectures do not exhibit this behaviour (Minderer et al., 2021), but we find that they are still poorly calibrated when trained from scratch.

Several methods have been proposed to create well-calibrated DNNs. Some of these methods are particular model specifications such as Bayesian Neural Networks (BNNs) (MacKay, 1992; Gal & Ghahramani, 2016; Lakshminarayanan et al., 2017; Blei et al., 2018; Izmailov et al., 2018). Other methods have been proposed to *re-calibrate* models post-hoc, i.e., after the model has been regularly trained, promising to retain model accuracy (Guo et al., 2017; Ji et al., 2019). A class of methods also uses data augmentation to regularize, thus creating calibrated models directly during training (Müller et al., 2019; Thulasidasan et al., 2019).

In this work, we show empirical evidence that the common practice of jointly training the feature extraction layers (convolution or attention layers) and the classification layers (fully connected layers) can lead to uncalibrated models. We then demonstrate that two-stage training procedures that decouple the training of the feature extraction and classification layers remarkably improve calibration. Consequently, two training strategies are presented and analyzed: Two-Stage Training (TST) and Variational Two-Stage Training (V-TST). We provide the code for reproducing our main results at https://github.com/MJordahn/Decoupled-Layers-for-Calibrated-NNs.

In TST, we first train a neural network end-to-end with cross-entropy loss (CE) until convergence. In the second training stage, we freeze the feature extraction layers (convolution layers or attention layers) of the network, reinitialize the

[1]Cognitive Systems, Technical University of Denmark, Kongens Lyngby, Denmark [2]Signal Processing Group (GTS), Universidad Carlos III de Madrid, Madrid, Spain. Correspondence to: Mikkel Jordahn <mikkjo@dtu.dk>.
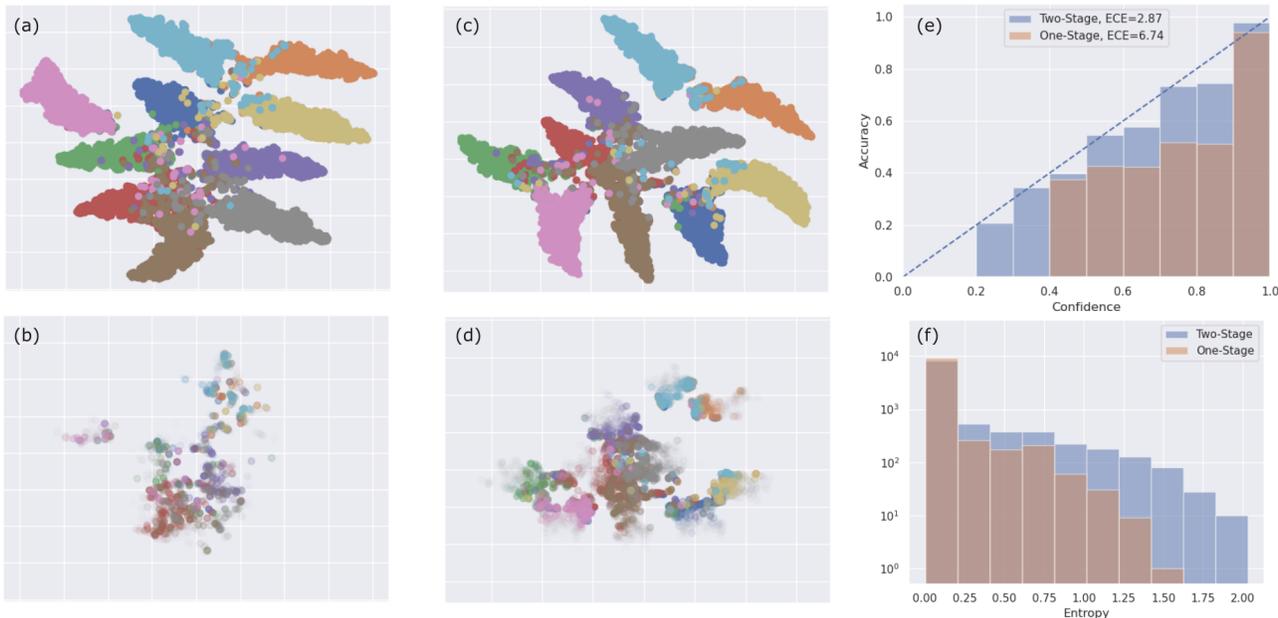
Figure 1. Last hidden layer outputs 2D t-SNE plots of test points from a WRN trained in **one stage** (left column), and **two stages** (center column) on CIFAR10 with accuracy 91.41% and 91.61% respectively. In the top row of t-SNE plots, (a) and (c), we plot all test points with equal color intensity, whilst in the second row, (b) and (d), higher color intensity indicates higher entropy of the prediction for a given point. Colors indicate which class the point truly belongs to. In (e) we show a calibration plot for the two models, and in (f) we show a histogram over the entropies of the predictions made by each model on the test set.

classifying fully connected (FC) layers, and re-train them on the same dataset. In V-TST, we additionally regularize the hidden feature space at the input of the last layer of the classifier with a Gaussian prior distribution and train the network with the evidence lower-bound (ELBO), enforcing better structure in the feature space and improving even further the model's calibration.

We demonstrate how these methods significantly improve calibration metrics on CIFAR10, CIFAR100 and SVHN and for different model architectures, in particular Wide Residual Networks (WRN) (Zagoruyko & Komodakis, 2016) and Vision Transformers (ViT) (Dosovitskiy et al., 2021).

In summary, our contributions are:

1. We show that calibration in DNNs can be significantly improved if feature extraction layers are not jointly learnt with classification layers.

2. We propose a two-stage training method that improves several calibration test metrics with respect to the same model trained in the usual end-to-end fashion.

3. We show that placing a probabilistic prior on the final hidden layer outputs and variationally training the classification layers further improves model calibration.

2. A Motivational Example

Before presenting TST and V-TST, we provide a brief proof-of-concept experiment. In Figure 1 we demonstrate the effect of decoupling feature extraction training from classification layer training for a WRN architecture. In (a) we show the 2D t-SNE (van der Maaten & Hinton, 2008) plot of the last hidden layer output $\mathbf{z}$ when the WRN is trained end-to-end using cross-entropy. In (b), the color intensity of the points depends on the classifier entropy $H(p(y|\mathbf{x}))$, so that zero entropy corresponds to zero color intensity (point not displayed). While the hidden space $\mathbf{z}$ is indeed structured by class (the overall classifier accuracy is 91.41%), this structure is mostly lost for those points with the largest uncertainty. For a calibrated classifier, we would expect that most points with the largest entropy correspond to cluster boundaries.

In (c) and (d), we show the same results for the WRN after re-training the classification FC layers from scratch in a second stage whilst the convolution layers are frozen. The overall test accuracy remains roughly the same (91.61%), but (d) suggests a clearer separation of high uncertainty points (and a larger presence of them). The points with higher entropy are generally located in areas of the feature space where the class clusters are close to each other, a

behaviour that suggests a better calibrated model. This is corroborated by the calibration plots (Guo et al., 2017) in (e). From the entropy histograms in (f), we can observe that simply re-training the classifier has re-calibrated its confidence, decreasing the fraction of near-zero entropy points.

3. Two-Stage Training

Based on these observations, we present two variations of our method for training calibrated DNNs: Two-Stage Training (TST) and Variational Two-Stage Training (V-TST). We show the algorithm for TST in Algorithm 1.

Algorithm 1 TST

1: Init. DNN M w. parameters $\{\beta, \phi\}$.
2: **Stage 1**: Train M with CE loss on D_{train} until convergence or early stopped.
3: Freeze parameters β of M.
4: Re-init, FC layers of M w. parameters $\{\theta, \nu\}$.
5: **Stage 2**: Train $\{\theta, \nu\}$ of M with CE loss on D_{train} until convergence.

Here β denotes the parameters of the feature extraction layers of the DNN model M and ϕ are the initial FC layer parameters. For convenience to later describe V-TST, the parameters of the FC layers in Stage 2 are split into θ (all FC layers except the final logit layer) and ν the parameters of the logit layer itself. Note that the training data in Stage 1 and 2 are identical; therefore, our method is not a variation of pretraining and fine-tuning procedures. We show a block diagram of the model and parameters in Figure 2 for clarification.
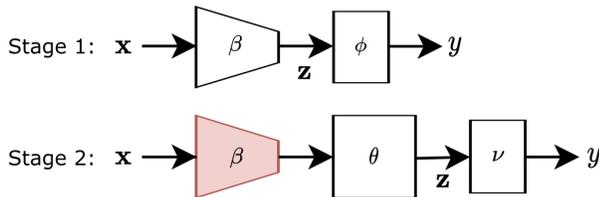

Figure 2. Block Diagram of Model M and belonging parameters. Red background denotes frozen parameters. $\mathbf{z}$ is the input to the logits layer.

The classification layers trained in Stage 1 and 2 can have the exact same structure, as we consider in the experiment in Figure 1. However, our experimental results suggest that even larger calibration improvements can be obtained by cross-validating the structure of the FC classification layers in Stage 2. In particular, appropriately choosing the dimensionality of the last hidden layer output $\mathbf{z}$ (the output

of the last layer parametrized by θ) can have a noticeable impact. In this regard, we hypothesize that further imposing structure on the feature space in the last hidden layer can be beneficial from the confidence calibration point of view. To verify such a hypothesis, we propose variational TST (V-TST).

In V-TST we place a probabilistic prior on $\mathbf{z}$, and the FC layers parametrized by θ and ν are trained using an ELBO-like loss function instead of CE loss (while the parameters β are fixed as in TST). More specifically, we consider the probabilistic model illustrated in Figure 3. We assume a Gaussian latent prior $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ and a label-reconstruction probabilistic model given by

$$p(y|\mathbf{z}) = \text{Cat}(\pi_\nu(\mathbf{z})) \tag{1}$$

where $\pi_\nu(\mathbf{z})$ is the logits layer in TST. To enforce larger regularization in the $\mathbf{z}$ space, the prior distribution for $\mathbf{z}$ is kept simple and is not dependent on the image $\mathbf{x}$.
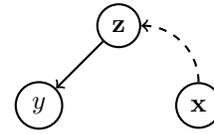

Figure 3. Latent Variable Model

Classification and training in the V-TST probabilistic model are attained using a conditional variational posterior that is dependent on the classifier input $\mathbf{x}$:

$$q(\mathbf{z}|\mathbf{x}) = \mathcal{N}(\mu_{\beta,\theta}(\mathbf{x}), \sigma_{\beta,\theta}(\mathbf{x})), \tag{2}$$

where $\mu_{\beta,\theta}(\mathbf{x})$ and $\sigma_{\beta,\theta}(\mathbf{x})$ are NNs that include the (frozen) feature extraction layers parametrized by β and the trainable FC layers parametrized by θ. Given the variational posterior in Equation (2), the FC layers parametrized by θ and ν are trained to maximize the following lower bound:

$$\begin{aligned} \log p(\mathbf{y}) \geq \text{ELBO}_{\theta,\nu} &= \int q(\mathbf{z}|\mathbf{x})\log\frac{p(y|\mathbf{z})p(\mathbf{z})}{q(\mathbf{z}|\mathbf{x})}d\mathbf{z} \\ &= \mathbb{E}_{q(\mathbf{z}|\mathbf{x})}\log[p(y|\mathbf{z})] - D_{KL}(q(\mathbf{z}|\mathbf{x})||p(\mathbf{z})) \end{aligned} \tag{3}$$

The ELBO in Equation (3) is optimized for all $(\mathbf{x}, y)$ training pairs by standard mini-batch optimization combined with Monte Carlo (MC) sampling and Gaussian re-parameterization for the expectation w.r.t. $q(\mathbf{z}|\mathbf{x})$ as done in Kingma & Welling (2014). At prediction time, we similarly estimate $p(y|\mathbf{x}) \approx \mathbb{E}_{q(\mathbf{z}|\mathbf{x})}[p(y|\mathbf{z})]$ using MC sampling with m samples.

4. Related Work

There exists a wide range of literature concerning designing and training well-calibrated neural networks with good uncertainty quantification. Firstly, the BNN literature holds in common that a prior is placed on all or some parameters of the neural network, and a posterior over these parameters is approximated. Several BNN methods have seen much interest in recent years, including Deep Ensembles (Lakshminarayanan et al., 2017), Laplace Approximation for Neural Networks (MacKay, 1992), Mean-Field BNNs (Graves, 2011; Blei et al., 2018), Node-Based BNNs (Dusenberry et al., 2020; Trinh et al., 2022) and MC-Dropout (Gal & Ghahramani, 2016).

TST, on the other hand, is a training method modification that falls into the implicit regularisation methods proposed for improving uncertainty quantification. There exist other implicit regularisation methods such as the Focal Loss (Lin et al., 2020; Mukhoti et al., 2020) or OKO (Muttenthaler et al., 2024), yet TST is not an alternative to these, but rather complementary. Another popular regularisation type is data augmentations for improved uncertainty quantification, such as Mix-Up (Thulasidasan et al., 2019) and Label-Smoothing (Müller et al., 2019). Finally, post-hoc calibration methods exist, aiming to calibrate the soft-max probabilities of an already-trained neural network, namely Temperature Scaling (TS) (Guo et al., 2017).

In the recently proposed Variational Classifier by Dhuliawala et al. (2024), the authors also propose using variational inference for classifiers. In the work, the authors assume a latent prior $\mathbf{z}$ dependent on y, and training requires the use of an auxiliary discriminator since the latent distribution $q(\mathbf{z}|y)$ is implicit. This is only accessible by sampling from $\mathbf{z} \sim q(\mathbf{z}|\mathbf{x})$ for class samples $\mathbf{x} \sim p(\mathbf{x}|y)$. This results in a mini-max optimization objective over the ELBO. Further, the models are always trained end-to-end. V-TST on the other hand, has much less complex inference. In fact, V-TST can be viewed as a discriminative classifier with a stochastic final layer that requires training using ELBO instead of CE. All the complex and heavy feature extraction layers are trained in a supervised fashion. The V-TST generative process models exclusively the marginal label distribution $p(y)$, not $p(y|\mathbf{x})$, resulting in a simple ELBO objective for which training convergence is very fast.

Wan et al. (2018) is another work that proposes imposing a Gaussian structure on the feature space of a classifier, but they use a likelihood regularizer to enforce this structure rather than the ELBO, nor do they train in two stages and do not report on calibration improvements.

In Kumar et al. (2022), the authors investigate the effect of linear-probing versus full model fine-tuning, which may seem similar in it's nature to TST. However, the work of Kumar et al. is specific to pretrained models, whereas ours is general to classifier training. TST and V-TST are general training methods for classifiers in applications or domains where it may not be possible to pretrain. Moreover, the theoretical justifications for why linear probing is a more robust fine-tuning method also do not hold for TST since we have non-linearities in the FC layers.

5. Experiments

We run a number of experiments to verify the benefits of TST and V-TST. As base models, we use over-parameterized image classifiers. We train WRN-28-10 (Zagoruyko & Komodakis, 2016) on CIFAR10, SVHN, and CIFAR100 and additionally show that these results hold for ViT (Dosovitskiy et al., 2021) models trained on CIFAR10. We use these models as the results of training Stage 1, initializing new fully connected layers, and freezing the feature extraction layers (Convolutional layers for WRN and Attention layers for ViT). We then perform Stage 2 of TST and V-TST, training with CE loss and the ELBO respectively. Once again, the training set is shared in Stages 1 and 2. We always train with Adam optimizer (Kingma & Ba, 2014). For comparison, we also use temperature scaling on the WRNs and include them in our evaluation for comparison to our method. We refer to Appendix A for details on the WRN/ViT training.

Once trained, we evaluate the models on metrics related to model fit and calibration. We show the accuracy, the expected calibration error (ECE) and the maximum calibration error (MCE) (Naeini et al., 2015) on the test set, and the negative log-likelihood (NLL) on both the training set and the test set. We evaluate ECE and MCE using 10 bins and report them in percentage.

We also evaluate the models on datasets with *distribution shifts*. For CIFAR10 and CIFAR100, we evaluate on CIFAR10-C and CIFAR100-C (Hendrycks & Dietterich, 2019) respectively, whilst for SVHN, we evaluate on rotated versions of the SVHN datasets. For details on these rotations we refer to Appendix C. Finally, we evaluate on *out-of-distribution* (OOD) datasets. For the models trained on SVHN we evaluate on CIFAR10 and CIFAR100, for CIFAR10 we evaluate on CIFAR100 and SVHN and for CIFAR100 we evaluate on CIFAR10 and SVHN. We evaluate OOD detection using AUROC (Bradley, 1997) and false-positive rate at 95% (FPR95). For the base models (WRN and ViT), we only run one seed, whilst we run ten seeds of parameter re-initialization for Stage 2 for TST and V-TST and report the mean and standard error of the mean.

For both TST and V-TST we re-initialize the fully-connected layers in Stage 2 with three FC layers. We do this to allow some flexibility in the classification layer, without over-

4

Table 1. Test Metrics on In-Distribution Data Test-sets. For V-TST, $m = 1$ and $m = 10$ indicates using 1 and 10 samples in the MC approximation, respectively. Temp. WRN is the temperature scaled WRN. Bolded fonts indicate best performing model for given dataset.

DATASET	MODEL	ACCURACY	ECE	MCE	TRAIN NLL	TEST NLL
CIFAR10	V-TST, M=10	**92.58±0.02**	**1.27±0.101**	28.48±9.006	0.0671±0.0004	0.2617±0.0013
	V-TST, M=1	91.09±0.06	1.52±0.144	**14.77±1.781**	0.1125±0.001	0.3732±0.0043
	TST	**92.59±0.02**	2.18±0.353	**14.89±2.249**	**0.0573±0.0002**	**0.2482±0.004**
	TEMP. WRN	92.53	4.4	26.77	0.063	0.2965
	WRN	92.53	5.88	35.84	0.1038	0.4944
CIFAR100	V-TST, M=10	**71.93±0.04**	**5.83±0.143**	**14.03±0.59**	0.078±0.001	1.202±0.0052
	V-TST, M=1	69.18±0.09	7.34±0.291	16.85±0.694	0.1094±0.0009	1.4315±0.0104
	TST	71.26±0.06	7.07±0.084	16.74±1.942	0.0887±0.0003	**1.1331±0.002**
	TEMP. WRN	71.55	15.24	33.21	**0.0739**	1.3708
	WRN	71.56	21.74	82.08	0.1161	2.2588
SVHN	V-TST, M=10	94.76±0.01	0.47±0.066	**12.67±1.958**	0.0587±0.0003	0.2195±0.0012
	V-TST, M=1	94.73±0.02	0.48±0.067	22.63±6.889	0.059±0.0003	0.2193±0.0011
	TST	**95.1±0.01**	**0.43±0.016**	25.04±9.617	0.0439±0.0001	**0.1837±0.0004**
	TEMP. WRN	95.07	1.88	19.51	**0.0415**	0.1926
	WRN	95.07	3.48	26.61	0.0605	0.2869

parametrizing. We test 6 dimensions over the final hidden layer output $\mathbf{z}$ in the range $[2, 512]$. We refer to Appendix B for details on the models used in TST and V-TST and training details. We also note that in Stage 2 of both TST and V-TST, we only train for 40 additional epochs, but in most of our experiments, much less are required to converge. For all models trained we use early stopping based on the validation loss.

Regarding V-TST, we only use the 1 sample in the MC approximation during training and m samples in the testing phase. For stability reasons, we choose a diagonal covariance matrix in the variational family and upper-bound the variance to 1. Additionally, we note that we downscale the KL loss by the dimension of the final hidden layer. We do this because the KL otherwise dominates the CE loss during training.

5.1. Benchmark Results

In Table 1, we show the evaluation metrics on the in-distribution data. Throughout this section, across all tables, the models listed with the same name have the same hyperparameters for a specific dataset but only show the best-performing configurations. For the full results, we refer to Appendix E. First, note that the test accuracy in all cases is comparable, except for a slight degradation in the case of V-TST when $m = 1$. More importantly, we show that both V-TST and TST improve ECE across all datasets and, in many cases, improve the MCE. V-TST improves base WRN model ECE by 78.40%, 73.18%, and 87.64% for CIFAR10, CIFAR100, and SVHN, respectively. In Figure 4 and Figure 5, we compare the calibration plots for the V-

TST and base WRN on SVHN and CIFAR100 respectively to illustrate the improvements further. Here, it can be seen how V-TST improves the ECE across all bins, illustrating that improved calibration is not intrinsic to specific bins but rather across all predictions.
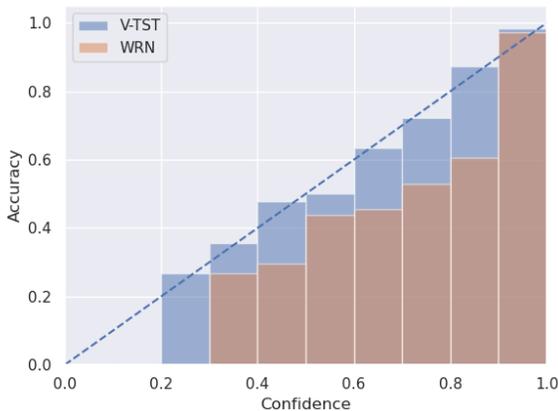
Figure 4. ECE plot of V-TST trained model and base WRN trained on SVHN.

Interestingly, we additionally note how both TST and V-TST decrease NLL on both the training set and the test set, which seems to indicate that training with TST is not simply a regularizer against overfitting. Had this been the case, one would expect to see a decrease in the generalization gap, i.e. an increase in the training NLL, but a decrease in the test NLL. Figure 6 illustrates the TST and V-TST effect in the

Table 2. Evaluation Metrics on Shifted Data and OOD Data

DATASET	MODEL	SHIFT ECE	SHIFT MCE	OOD AUROC	OOD FPR95
CIFAR10	V-TST, M=10	**10.41±0.22**	22.92±0.57	0.821±0.003	0.751±0.002
	V-TST, M=1	12.89±0.3	**21.01±0.58**	0.722±0.005	0.824±0.006
	TST	11.62±0.75	25.39±1.74	0.874±0.002	0.699±0.009
	TEMP. WRN	16.45	36.77	0.872	0.746
	WRN	20.27	45.06	**0.891**	**0.653**
CIFAR100	V-TST, M=10	**14.3±0.23**	**26.96±0.46**	0.791±0.002	0.809±0.004
	V-TST, M=1	16.8±0.4	30.72±0.63	0.783±0.004	0.822±0.006
	TST	17.44±0.08	28.51±0.14	**0.823±0.002**	**0.764±0.006**
	TEMP. WRN	29.58	47.67	0.778	0.816
	WRN	40.47	60.38	0.785	0.892
SVHN	V-TST, M=10	35.72±0.58	52.36±0.59	0.872±0.004	0.537±0.003
	V-TST, M=1	35.67±0.59	52.33±0.59	0.859±0.004	0.549±0.006
	TST	**30.97±0.07**	**48.39±0.09**	0.935±0.001	0.415±0.006
	TEMP. WRN	37.53	55.09	0.934	0.461
	WRN	47.06	80.65	**0.945**	**0.344**

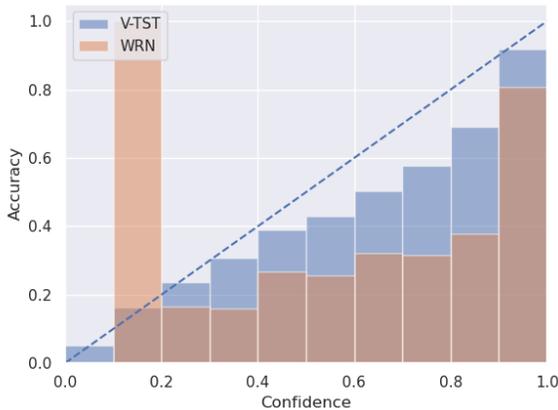*Figure 5.* ECE plot of V-TST trained model and base WRN trained on CIFAR100.

structure of the hidden space **z**. We show t-SNE plots of the WRN in (a) and (b), temperature-scaled WRN in (c) and (d), for TST in (e) and (f) and V-TST in (g) and (h) on SVHN. In the top row we plot all test points, whilst in the bottom row we again scale color intensity by $H(p(y|\mathbf{x}))$. We note how the entropy of most test points increases significantly as we go from left to right in the columns and how, in (g) and (h), we manage to get Gaussian distributions within each class.

In Table 2, we show the calibration metrics of the models when evaluated on shifted and OOD datasets. In the case of distribution shifts, we see that in all cases, TST and V-TST significantly improve upon the calibration of the WRN and outperform temperature scaling. When it comes to OOD

detection tasks, TST and V-TST generally perform worse than baselines. We discuss potential reasons for this in the limitations section.

5.2. MC Samples Effect on V-TST Performance

Based on the previous results, a natural question that arises is how many samples are required in the MC approximation of $\mathbb{E}_{q(\mathbf{z}|\mathbf{x})}[p(y|\mathbf{z})]$ at V-TST prediction time. In order to verify this, we run an experiment using V-TST models trained with varying latent variable dimension on CIFAR10, and track the accuracy and ECE as we increase the number of samples used in the MC approximation. For each m, we run 10 seeds and report the mean and the standard error of the mean which is indicated by error bars. The results can be seen in Figure 7. Across all models, we see that as we increase the number of samples in approximation the accuracy improves. Additionally, for all models except for dimension of 32, we see that the ECE also improves as we increase the number of samples. In this regard, the results reported in Table 1 and Table 2 can be potentially improved by tuning parameter m.
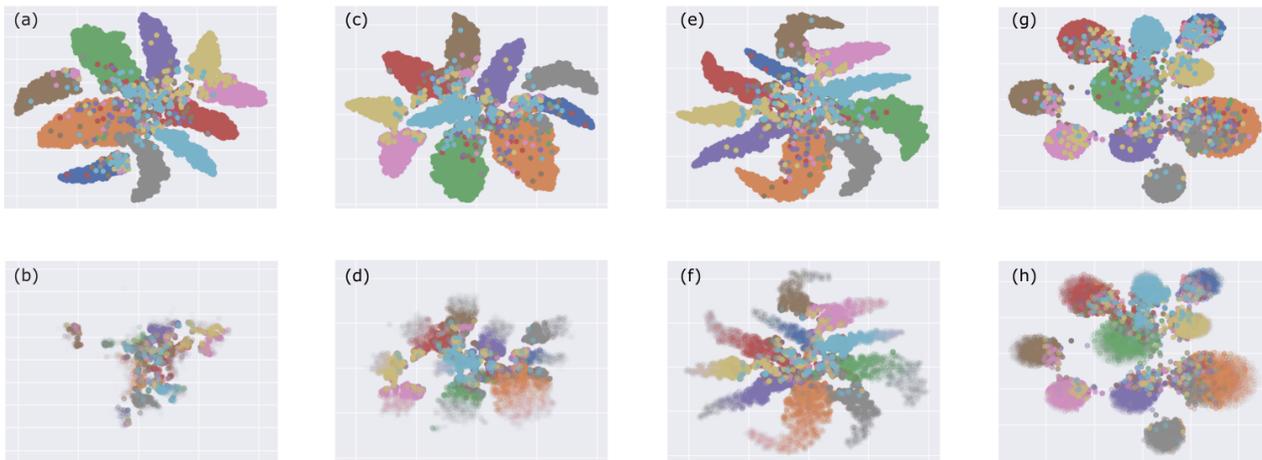
Figure 6. Last hidden layer outputs 2D t-SNE plots of test points from SVHN. (a) and (b) is from a WRN trained in **one stage**, (c) and (d) is from a **temperature scaled** version of the WRN from the first column, (e) and (f) from **TST**, and (g) and (h) is **V-TST**. In the top row we plot all test points with equal color intensity, whilst in the second row, higher color intensity indicates higher entropy of the prediction for a given point. Colors indicate which class the point truly belongs to. All these features are behind the improvements reported for TST and V-TST.
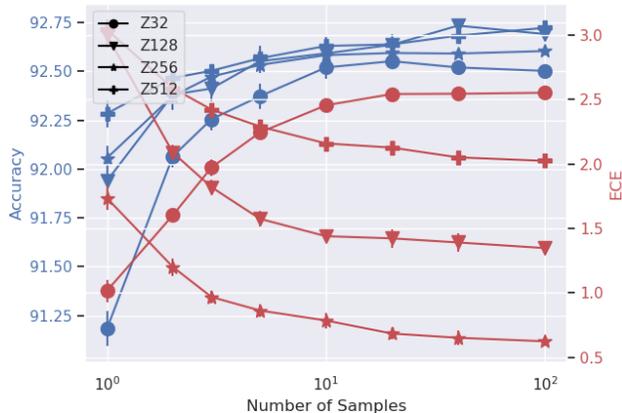

Figure 7. Effect of number of samples in Monte Carlo approximation during test evaluation of $\mathbb{E}_{q(\mathbf{z}|\mathbf{x})}[p(y|\mathbf{z})]$ in V-TST trained models on CIFAR10.

5.3. Ablation Study: Under-parametrized Architecture

To investigate our belief that TST and V-TST are beneficial when we have over-parametrized models, we run an experiment in which we define a much simpler CNN architecture than the WRN. We know that this model is not over-parametrized because it does not overfit the training data - the test and training accuracy are almost identical. We refer to Appendix D for details on this architecture. The test results are in Table 3. Here, we see that the two versions of

V-TST worsen the ECE, although they improve the MCE, whilst TST improves both ECE and MCE, although at the cost of accuracy. We refer to Appendix F for the full test results.

Table 3. CNN Ablation Study Test Metrics on CIFAR10

MODEL	ACCURACY	ECE	MCE
V-TST, M=10	**75.95**±**0.2**	3.27±0.05	**10.65**±**4.35**
V-TST, M=1	73.22±0.1	1.61±0.3	**9.08**±**5.47**
TST	70.77±0.2	**0.76**±**0.17**	7.79±4.67
CNN	72.48	1.04	19.89

5.4. Ablation Study: Training Models End-to-End

As part of our ablation studies, we also test whether the MLP we add at the end of the feature extraction layers and before the logits layer is the reason for the improved calibration metrics rather than TST and V-TST. Therefore, we train the best-performing versions of TST and V-TST models end-to-end on CIFAR10 and evaluate them similarly to how we have previously done. We only do this for one seed and, therefore, report no standard error of the mean. The results can be seen in Table 4 where we include TST and V-TST for comparison but do not report standard error of the mean. Var. E2E indicates that the adapted WRN architecture has been trained end-to-end in one step with the ELBO, while

E2E is the adapted architecture trained with CE only. Here we clearly show that training in one step with the adapted architectures does not result in equivalent calibration metrics, but training variationally (Var. E2E) still improves upon E2E. This suggests that the VAE-like stochastic layer we presented for V-TST has itself regularization properties that improve model's calibration and uncertainty quantification.

Table 4. Test Metrics for End-to-End Training on CIFAR10

MODEL	ACCURACY	ECE	MCE
V-TST, M=10	**92.58**	**1.27**	28.48
V-TST, M=1	91.09	1.52	**14.77**
TST	92.59	2.18	14.89
VAR. E2E, M=10	92.16	4.72	22.81
VAR. E2E, M=1	91.65	5.85	26.07
E2E	91.41	6.74	37.22

5.5. Ablation Study: Vision Transformer

We also show that TST and V-TST does not just improve calibration in convolution architectures, but also in Vision Transformers (ViT) (Dosovitskiy et al., 2021). We train a ViT architecture on CIFAR10, and perform Stage 2 of TST and V-TST as we did with the WRN. The results can be seen in Table 5. Once again, V-TST maintains the accuracy of the ViT model, but significantly improves both the ECE and the MCE again. We refer to Appendix G for shift data and OOD data results.

Table 5. ViT Ablation Study Test Metrics on CIFAR10

MODEL	ACCURACY	ECE	MCE
V-TST, M=10	**82.84**±0.04	**1.23**±0.18	**14.93**±2.54
V-TST, M=1	81.14±0.06	4.2±0.3	17.67±3.62
TST	82.3±0.06	1.78±0.37	17.71±2.2
VIT	82.56	13.71	34.27

5.6. Two-Stage Training for Pretrained Models

Finally, we investigate if TST and V-TST also can be used on models that have first been pretrained and later fine-tuned for a classification task. We do this by taking the pretrained ViT-B/16 from (Dosovitskiy et al., 2021), unfreeze all of the layers and fine-tune it for a classification task on Tiny ImageNet (Le & Yang, 2015), and using this as input to Stage 2 of TST and V-TST respectively. The results can be seen in Table 6 where FT indicates fine-tuning. Interestingly here we note that TST actually worsens the ECE compared to the baseline model, whilst V-TST improves accuracy,

ECE and MCE. Once again, this suggests that the V-TST training has good regularization properties.

Table 6. Test Metrics on Tiny ImageNet

MODEL	ACCURACY	ECE	MCE
V-TST, M=10	**86.7**±0.02	**1.96**±0.07	9.76±0.56
V-TST, M=1	85.63±0.05	2.3±0.27	**8.62**±0.55
TST	85.99±0.07	3.47±0.44	14.77±1.69
FT VIT-B/16	85.46	2.19	12.54

6. Limitations of TST and V-TST

In this section we discuss concerns that may arise with TST and V-TST. Firstly, there is the computation cost of having to train in two stages. However, we find that in Stage 2, the models converge extremely quickly. In the case of TST used for the WRN architecture, the best validation loss is obtained after 1.167 epochs on average across the three datasets. V-TST is slightly more expensive with the best validation loss for V-TST for the same datasets and models being obtained after 10.33. The training for V-TST is also slower due to having to sample $\mathbf{z}$ in every training step, but minimally so, as we only use $m = 1$ during training. At prediction time, it is necessary to increase the number of MC samples to significantly improve the performance of the V-TST models. Making predictions on the CIFAR10 test-set takes 0.932 seconds for TST and even using $m = 100$ in V-TST takes only 1.406 seconds, a cost that may be worth it in applications where model calibration is critical.

A potential indication of our experiment with the pretrained ViT-B/16 model on Tiny ImageNet is that the beneficial effects of TST are mainly observed when feature extraction layers and classification layers are trained jointly from scratch. This would explain why TST does not provide any ECE or MCE gains in the pretrained model setting because the feature extraction layers are much closer to the optimal values at the beginning of Stage 1. We leave further investigations of this for future work.

An unexpected downside to TST and V-TST is that they at times reduce the performance on OOD data. To us this indicates that fitting training data well (even in terms of calibration), and training stochastically with respect to the inputs does not teach the model anything about modalities it has not seen before. An interesting future investigation would be to use BNN methods to improve OOD performance. There is much literature that lends itself readily to this for TST and V-TST, as there is much BNN research that concerns itself with only learning distributions over subsets of layer parameters, such as last layer Laplace Approximations (Kristiadi et al., 2020). Another future investigation

could be to use the likelihood of a new test point under the prior $p(\mathbf{z})$ to perform OOD detection, although in generative models, specifically VAEs, the success of doing this has been varied (Nalisnick et al., 2019; Eduardo et al., 2020).

7. Conclusion

The consensus in the DNN literature is that CE training over-parametrized models leads to poor model calibration, an effect we observe in WRN and ViT architectures. What our results indicate is that jointly training feature extraction layers and classification layers is one of the main drivers of this poor calibration. By freezing the feature extraction layers in TST and V-TST, we reduce the overall flexibility of the model. We hypothesise that this means that the classifier we train in Stage 2 cannot move data points arbitrarily far from decision boundaries to increase label likelihood artificially, as can happen when feature space and classification layers are jointly trained.

We showed that enforcing larger regularization in the feature space by adding a probabilistic prior in the $\mathbf{z}$ space significantly improved calibration even further. This is because learning a distribution over $\mathbf{z}$, $q(\mathbf{z}|\mathbf{x})$ and sampling from this at training time when predicting y, is an explicit augmentation of the features that is input to the logits layer. By training this way, the model learns to map several points in $\mathbf{z}$ to the same y, more so than it would without the variational training. Our results in the end-to-end ablation study also corroborated that variational training leads to better calibration as we saw that the V-TST architecture trained end-to-end provided increased calibration metrics in comparison to the regular end-to-end trained model. We therefore believe that this line of work of variational training for discriminative classifiers will be of interest to the community interested in calibrated classifiers.

Acknowledgments

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here.

References

Ahsan, M. M., Luna, S. A., and Siddique, Z. Machine-learning-based disease diagnosis: A comprehensive review. *Healthcare*, 10(3), 2022. ISSN 2227-9032. doi: 10.3390/healthcare10030541. URL https://www.mdpi.com/2227-9032/10/3/541.

Blei, D. M., Kucukelbir, A., and McAuliffe, J. D. Variational inference: A review for statisticians. 2018.

Bradley, A. P. The use of the area under the roc curve in the evaluation of machine learning algorithms. *Pattern Recognition*, 30(7):1145–1159, 1997. ISSN 0031-3203. doi: https://doi.org/10.1016/S0031-3203(96)00142-2. URL https://www.sciencedirect.com/science/article/pii/S0031320396001422.

Chen, Z. and Huang, X. End-to-end learning for lane keeping of self-driving cars. In *2017 IEEE Intelligent Vehicles Symposium (IV)*, pp. 1856–1860, 2017. doi: 10.1109/IVS.2017.7995975.

Dhuliawala, S. Z., Sachan, M., and Allen, C. Variational classification: A probabilistic generalization of the softmax classifier. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856. URL https://openreview.net/forum?id=EWv9XGOpB3.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., and Houlsby, N. An image is worth 16x16 words: Transformers for image recognition at scale. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021. URL https://openreview.net/forum?id=YicbFdNTTy.

Dusenberry, M., Jerfel, G., Wen, Y., Ma, Y., Snoek, J., Heller, K., Lakshminarayanan, B., and Tran, D. Efficient and scalable Bayesian neural nets with rank-1 factors. In III, H. D. and Singh, A. (eds.), *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 2782–2792. PMLR, 13–18 Jul 2020. URL https://proceedings.mlr.press/v119/dusenberry20a.html.

Eduardo, S., Nazabal, A., Williams, C. K. I., and Sutton, C. Robust variational autoencoders for outlier detection and repair of mixed-type data. In Chiappa, S. and Calandra, R. (eds.), *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, volume 108 of *Proceedings of Machine Learning Research*, pp. 4056–4066. PMLR, 26–28 Aug 2020. URL https://proceedings.mlr.press/v108/eduardo20a.html.

Gal, Y. and Ghahramani, Z. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In Balcan, M. F. and Weinberger, K. Q. (eds.), *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pp. 1050–1059, New York, New York, USA, 20–22 Jun 2016. PMLR. URL https://proceedings.mlr.press/v48/gal16.html.

Graves, A. Practical variational inference for neural networks. In Shawe-Taylor, J., Zemel, R., Bartlett, P., Pereira, F., and Weinberger, K. (eds.), *Advances in Neural Information Processing Systems*, volume 24. Curran Associates, Inc., 2011. URL https://proceedings.neurips.cc/paper_files/paper/2011/file/7eb3c8be3d411e8ebfab08eba5f49632-Paper.pdf.

Guo, C., Pleiss, G., Sun, Y., and Weinberger, K. Q. On calibration of modern neural networks. In *International Conference on Machine Learning*, pp. 1321–1330, 2017.

Gupta, A., Anpalagan, A., Guan, L., and Khwaja, A. S. Deep learning for object detection and scene perception in self-driving cars: Survey, challenges, and open issues. *Array*, 10:100057, 2021. ISSN 2590-0056. doi: https://doi.org/10.1016/j.array.2021.100057. URL https://www.sciencedirect.com/science/article/pii/S2590005621000059.

Hendrycks, D. and Dietterich, T. Benchmarking neural network robustness to common corruptions and perturbations. *Proceedings of the International Conference on Learning Representations*, 2019.

Izmailov, P., Podoprikhin, D., Garipov, T., Vetrov, D. P., and Wilson, A. G. Averaging weights leads to wider optima and better generalization. In *Conference on Uncertainty in Artificial Intelligence*, 2018. URL https://api.semanticscholar.org/CorpusID:3833416.

Ji, B., Jung, H., Yoon, J., Kim, K., and Shin, Y. Binwise temperature scaling (bts): Improvement in confidence calibration performance through simple scaling techniques. *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*, pp. 4190–4196, 2019. URL https://api.semanticscholar.org/CorpusID:201698107.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2014. URL https://api.semanticscholar.org/CorpusID:6628106.

Kingma, D. P. and Welling, M. Auto-Encoding Variational Bayes. In *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*, 2014.

Kristiadi, A., Hein, M., and Hennig, P. Being bayesian, even just a bit, fixes overconfidence in relu networks. In *Proceedings of the 37th International Conference on Machine Learning*, ICML'20. JMLR.org, 2020.

Kumar, A., Raghunathan, A., Jones, R. M., Ma, T., and Liang, P. Fine-tuning can distort pretrained features and underperform out-of-distribution. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=UYneFzXSJWh.

Lakshminarayanan, B., Pritzel, A., and Blundell, C. Simple and scalable predictive uncertainty estimation using deep ensembles. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, pp. 6405–6416, Red Hook, NY, USA, 2017. Curran Associates Inc. ISBN 9781510860964.

Le, Y. and Yang, X. S. Tiny imagenet visual recognition challenge. 2015. URL https://api.semanticscholar.org/CorpusID:16664790.

Lecun, Y., Chopra, S., Hadsell, R., Ranzato, M., and Huang, F. *A tutorial on energy-based learning*. MIT Press, 2006.

Lin, T.-Y., Goyal, P., Girshick, R., He, K., and Dollár, P. Focal loss for dense object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(2):318–327, 2020. doi: 10.1109/TPAMI.2018.2858826.

MacKay, D. J. C. A practical bayesian framework for back-propagation networks. *Neural Computation*, 4(3):448–472, 1992. doi: 10.1162/neco.1992.4.3.448.

Minderer, M., Djolonga, J., Romijnders, R., Hubis, F. A., Zhai, X., Houlsby, N., Tran, D., and Lucic, M. Revisiting the calibration of modern neural networks. In Beygelzimer, A., Dauphin, Y., Liang, P., and Vaughan, J. W. (eds.), *Advances in Neural Information Processing Systems*, 2021. URL https://openreview.net/forum?id=QRBvLayFXI.

Mukhoti, J., Kulharia, V., Sanyal, A., Golodetz, S., Torr, P., and Dokania, P. Calibrating deep neural networks using focal loss. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H. (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 15288–15299. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/aeb7b30ef1d024a76f21a1d40e30c302-Paper.pdf.

Müller, R., Kornblith, S., and Hinton, G. *When does label smoothing help?* Curran Associates Inc., Red Hook, NY, USA, 2019.

Muttenthaler, L., Vandermeulen, R. A., Zhang, Q., Unterthiner, T., and Müller, K.-R. Set learning for accurate and calibrated models, 2024.

Naeini, M. P., Cooper, G. F., and Hauskrecht, M. Obtaining well calibrated probabilities using bayesian binning. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*, AAAI'15, pp. 2901–2907. AAAI Press, 2015. ISBN 0262511290.

Nalisnick, E. T., Matsukawa, A., Teh, Y. W., Görür, D., and Lakshminarayanan, B. Do deep generative models know what they don't know? In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019. URL https://openreview.net/forum?id=H1xwNhCcYm.

Shen, L., Margolies, L. R., Rothstein, J. H., Fluder, E., McBride, R., and Sieh, W. Deep learning to improve breast cancer detection on screening mammography. *Scientific Reports*, 9(1):12495, 2019. doi: 10.1038/s41598-019-48995-4. URL https://doi.org/10.1038/s41598-019-48995-4.

Thulasidasan, S., Chennupati, G., Bilmes, J. A., Bhattacharya, T., and Michalak, S. E. On mixup training: Improved calibration and predictive uncertainty for deep neural networks. In *Neural Information Processing Systems*, 2019. URL https://api.semanticscholar.org/CorpusID:166228660.

Trinh, T. Q., Heinonen, M., Acerbi, L., and Kaski, S. Tackling covariate shift with node-based Bayesian neural networks. In Chaudhuri, K., Jegelka, S., Song, L., Szepesvari, C., Niu, G., and Sabato, S. (eds.), *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 21751–21775. PMLR, 17–23 Jul 2022. URL https://proceedings.mlr.press/v162/trinh22a.html.

van der Maaten, L. and Hinton, G. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(86):2579–2605, 2008. URL http://jmlr.org/papers/v9/vandermaaten08a.html.

Wan, W., Zhong, Y., Li, T., and Chen, J. Rethinking feature distribution for loss functions in image classification. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9117–9126, 2018. URL https://api.semanticscholar.org/CorpusID:3775537.

Zagoruyko, S. and Komodakis, N. Wide residual networks. In Wilson, R. C., Hancock, E. R., and Smith, W. A. P. (eds.), *Proceedings of the British Machine Vision Conference 2016, BMVC 2016, York, UK, September 19-22, 2016*. BMVA Press, 2016. URL http://www.bmva.org/bmvc/2016/papers/paper087/index.html.

A. Training Details for WRN and ViT

For all of the datasets we train a WRN 28-10 as it is specified in Zagoruyko and Komodakis (2016). We train it for 600 epochs using Adam optimizer with learning rate 10^{-4}, but employ early stopping based on the validation loss. We compute the validation loss based on a validation set, which is data we split from the training set. For CIFAR10 and SVHN we use 15% of the training set for validation, whilst for CIFAR100 we use only 5% of the data for validation. We use data augmentation during training, but refer to the submitted code for specifics on this, and also to be able to reproduce the metrics in Table 1 and Table 2.

The ViT model is based on the implementation in Dosovitskiy et al. (2021). We train this model for 1000 epochs, but similarly employ early stopping based on the validation loss. Here, we also use Adam optimizer with learning rate 10^{-4}. We use a patch size of 4, a token dim of size 512, depth of size 6, 8 heads, MLP dim of size 512 and head dimension of size 64. We use dropout in both the Transformer and the embeddings with $p = 0.1$. This model has approximately 9.5 million parameters, where the models in ()ViT have at least 80 million parameters explaining the lower performance of our base ViT models.

B. Training details for TST and V-TST

In Stage 2 of training with TST and V-TST we re-initialize the layers parametrized by θ and ν with the same architectures for both, and we do a hyperparameter search over the dimensions of ν. For the WRN architectures we re-initialize the FC layers parametrized by θ with an MLP with two layers of [input, output] size $[640, 3Z]$ and $[3Z, Z]$, where Z is the size of the latent dimension, which is a hyperparameter we test varying sizes of. For all models we test Z dimensions $[2, 8, 32, 128, 256, 512]$. We only use ReLU at the output of the first layer of the MLP, and do not use activation functions anywhere else. The parameters of ν is always $[Z, \#\text{classes}]$. The output of ν are logits. We only train on the same data that has been trained on in Stage 1 and still use Adam optimizer with learning rate 10^{-4}. We still employ early stopping based on the validation loss. In the case of V-TST, we use two MLPs of the same size parametrized by θ, one to output μ of the variational distribution, and one to output σ of the variational distribution. We use a log-sigmoid activation function in the output of the MLP parametrizing σ for stability reasons. For the ViT model, we have the same number of layers in the MLP parametrized by θ, but with [input, output] sizes $[512, 3Z]$ and $[3Z, Z]$.

C. SVHN Rotations

When evaluating shifted data metrics for SVHN, we evaluate on the full test-set with varying severities of rotations. We rotate with degrees $[10., 45., 90., 135., 180.]$.

D. Small CNN Architecture

In the experiment in which we show that TST and V-TST do not improve under-parametrized networks, we specify a CNN with the following architecture where we use max pooling with kernel (2,2) and relu activations after all layers except the last.

- Conv2D(3, 6, 5)

- Conv2D(6, 16, 5)

- Linear(400, 120)

- Linear(120, 84)

- Linear(84, 10)

E. Additional WRN Experiment Results

In all of the tables in this section, Z followed by a number in the model column indicates the dimensions of the final hidden layer output. For TST and V-TST, we report the mean and standard-error of the mean over 10 seeds. m is the number of

12

samples used to approximate $\mathbb{E}_{q(\mathbf{z}|\mathbf{x})}[p(y|\mathbf{z})]$. E-to-E indicates the TST architecture trained in one stage end-to-end and Var. E-to-E indicates the V-TST model trained end-to-end in one stage.

E.1. CIFAR10 WRN Two-Stage Results

Table 7. CIFAR10 WRN In Distribution Test Results

MODEL	ACCURACY	ECE	MCE	TRAIN NLL	TEST NLL
E-TO-E Z128	91.29	6.87	37.14	0.1036	0.5462
E-TO-E Z32	91.41	6.74	37.22	0.1141	0.5587
TST Z128	**92.590±0.02**	2.18±0.353	14.89±2.249	0.0573±0.0002	**0.248±0.004**
TST Z2	77.64±0.85	7.43±0.832	25.1±4.514	0.4522±0.02	0.9124±0.0199
TST Z256	**92.590±0.01**	2.93±0.015	15.86±2.857	**0.056±0.0001**	0.2532±0.0004
TST Z32	92.51±0.02	2.51±0.221	22.8±6.522	0.0586±0.0003	0.2592±0.0033
TST Z512	**92.590±0.01**	4.04±0.015	24.41±1.156	0.0601±0.0002	0.283±0.0008
TST Z8	91.79±0.09	2.11±0.168	17.1±2.025	0.0739±0.0029	0.3083±0.0097
TEMP. WRN 28-10	92.53	4.4	26.77	0.063	0.2965
V-TST Z128, M=1	91.9±0.04	2.82±0.104	22.27±6.658	0.081±0.0004	0.3262±0.0022
V-TST Z128, M=10	92.58±0.02	**1.270±0.101**	28.48±9.006	0.0671±0.0004	0.2617±0.0013
V-TST Z2, M=1	33.76±0.77	5.83±0.517	20.98±1.475	1.6683±0.0198	1.7602±0.0176
V-TST Z2, M=10	46.23±1.94	23.12±1.725	54.79±0.586	1.5823±0.0206	1.6579±0.0191
V-TST Z256, M=1	92.14±0.05	2.91±0.18	16.04±1.074	0.0731±0.0007	0.305±0.0043
V-TST Z256, M=10	92.56±0.03	1.78±0.161	28.01±9.02	0.0628±0.0003	0.2583±0.0025
V-TST Z32, M=1	91.09±0.06	1.52±0.144	14.77±1.781	0.1125±0.001	0.3732±0.0043
V-TST Z32, M=10	92.52±0.02	1.68±0.125	**13.990±1.515**	0.0885±0.0009	0.2802±0.0011
V-TST Z512, M=1	92.23±0.04	3.15±0.157	16.5±1.149	0.0676±0.0004	0.2934±0.0041
V-TST Z512, M=10	**92.590±0.03**	2.26±0.174	21.6±6.763	0.0601±0.0002	0.2572±0.0021
V-TST Z8, M=1	88.34±0.1	7.09±0.187	19.48±2.124	0.2417±0.0022	0.4746±0.0032
V-TST Z8, M=10	92.21±0.04	12.88±0.213	23.3±0.677	0.2057±0.0022	0.3869±0.003
VAR. E-TO-E Z128, M=1	92.0	6.33	35.97	0.1283	0.6234
VAR. E-TO-E Z128, M=10	92.23	5.66	37.41	0.097	0.4672
VAR. E-TO-E Z32, M=1	91.65	5.85	26.07	0.1482	0.6622
VAR. E-TO-E Z32, M=10	92.16	4.72	22.81	0.1075	0.4779
WRN 28-10	92.53	5.88	35.84	0.1038	0.4944

Table 8. CIFAR10 WRN Shifted Data Test Results

MODEL	SHIFT ECE	SHIFT MCE
E-TO-E Z128	20.81	45.58
E-TO-E Z32	20.67	47.13
TST Z128	11.62±0.75	25.39±1.74
TST Z2	11.2±0.58	22.92±2.07
TST Z256	13.13±0.03	28.88±0.08
TST Z32	12.21±0.48	26.43±1.08
TST Z512	15.69±0.03	35.83±1.25
TST Z8	11.66±0.31	23.82±0.78
TEMP. WRN 28-10	16.45	36.77
V-TST Z128, M=1	13.77±0.22	27.28±0.49
V-TST Z128, M=10	10.41±0.22	22.92±0.57
V-TST Z2, M=1	**1.610±0.18**	13.2±1.87
V-TST Z2, M=10	16.44±1.31	52.7±1.0
V-TST Z256, M=1	13.42±0.43	28.02±0.93
V-TST Z256, M=10	10.76±0.4	23.97±0.94
V-TST Z32, M=1	12.89±0.3	21.01±0.58
V-TST Z32, M=10	7.33±0.24	16.36±0.44
V-TST Z512, M=1	13.58±0.42	29.46±0.9
V-TST Z512, M=10	11.6±0.42	26.03±0.95
V-TST Z8, M=1	2.54±0.25	**3.810±0.25**
V-TST Z8, M=10	6.51±0.29	9.64±0.37
VAR. E-TO-E Z128, M=1	21.73	41.69
VAR. E-TO-E Z128, M=10	20.54	44.35
VAR. E-TO-E Z32, M=1	21.63	34.72
VAR. E-TO-E Z32, M=10	19.83	37.32
WRN 28-10	20.27	45.06

Table 9. CIFAR10 WRN OOD Data Test Results

MODEL	OOD AUROC	OOD FPR95
E-TO-E Z128	0.915	0.561
E-TO-E Z32	**0.922**	**0.494**
TST Z128	0.874±0.002	0.699±0.009
TST Z2	0.739±0.033	0.856±0.036
TST Z256	0.877±0.002	0.696±0.009
TST Z32	0.879±0.003	0.68±0.012
TST Z512	0.879±0.001	0.683±0.006
TST Z8	0.875±0.006	0.68±0.026
TEMP. WRN 28-10	0.872	0.746
V-TST Z128, M=1	0.775±0.004	0.783±0.003
V-TST Z128, M=10	0.821±0.003	0.751±0.002
V-TST Z2, M=1	0.593±0.008	0.924±0.003
V-TST Z2, M=10	0.681±0.022	0.868±0.017
V-TST Z256, M=1	0.802±0.007	0.776±0.005
V-TST Z256, M=10	0.835±0.005	0.75±0.002
V-TST Z32, M=1	0.722±0.005	0.824±0.006
V-TST Z32, M=10	0.79±0.002	0.758±0.003
V-TST Z512, M=1	0.823±0.005	0.751±0.005
V-TST Z512, M=10	0.846±0.003	0.743±0.002
V-TST Z8, M=1	0.729±0.01	0.843±0.011
V-TST Z8, M=10	0.818±0.003	0.763±0.009
VAR. E-TO-E Z128, M=1	0.772	0.732
VAR. E-TO-E Z128, M=10	0.752	0.715
VAR. E-TO-E Z32, M=1	0.713	0.809
VAR. E-TO-E Z32, M=10	0.694	0.756
WRN 28-10	0.891	0.653

E.2. CIFAR100 WRN Two-Stage Results

Table 10. CIFAR100 WRN In Dist. Results.

MODEL	ACCURACY	ECE	MCE	TRAIN NLL	TEST NLL
TST Z128	70.19±0.08	9.41±0.356	21.14±0.994	0.0976±0.0015	1.2556±0.0066
TST Z2	6.96±0.17	3.5±0.197	47.52±1.227	3.3507±0.0232	4.0506±0.0171
TST Z256	70.9±0.06	8.88±0.054	19.14±0.655	0.0875±0.0002	1.1923±0.002
TST Z32	65.57±0.72	11.97±1.359	23.86±2.658	0.1684±0.0043	1.5957±0.0398
TST Z512	71.26±0.06	7.07±0.084	16.74±1.942	0.0887±0.0003	**1.133±0.002**
TST Z8	49.52±0.35	9.34±0.857	16.83±2.705	0.882±0.0235	2.6482±0.0225
TEMP. WRN 28-10	71.55	15.24	33.21	**0.074**	1.3708
V-TST Z128, M=1	69.18±0.09	7.34±0.291	16.85±0.694	0.1094±0.0009	1.4315±0.0104
V-TST Z128, M=10	71.64±0.07	3.84±0.082	**7.240±0.305**	0.0945±0.0013	1.2129±0.0021
V-TST Z2, M=1	2.83±0.07	**0.180±0.027**	22.81±6.562	4.2878±0.0131	4.431±0.009
V-TST Z2, M=10	4.55±0.15	2.52±0.141	33.87±13.028	4.2054±0.0134	4.3332±0.0094
V-TST Z256, M=1	69.89±0.08	8.48±0.367	19.76±0.974	0.0942±0.001	1.3975±0.0127
V-TST Z256, M=10	71.9±0.05	4.87±0.118	11.68±0.585	0.0827±0.0012	1.2072±0.0034
V-TST Z32, M=1	65.79±0.09	3.35±0.124	8.61±0.466	0.205±0.0021	1.5907±0.0027
V-TST Z32, M=10	69.82±0.06	4.9±0.13	12.56±0.454	0.1761±0.0019	1.3437±0.0026
V-TST Z512, M=1	70.59±0.06	8.81±0.333	20.54±0.828	0.0858±0.0007	1.3416±0.0135
V-TST Z512, M=10	**71.930±0.04**	5.83±0.143	14.03±0.59	0.078±0.001	1.202±0.0052
V-TST Z8, M=1	45.06±0.26	19.75±0.15	43.42±0.383	1.5655±0.0137	2.6529±0.0126
V-TST Z8, M=10	53.44±0.25	30.45±0.135	55.65±0.239	1.4756±0.0135	2.4792±0.0119
WRN 28-10	71.56	21.74	82.08	0.1161	2.2588

Table 11. CIFAR100 WRN Shift Results.

MODEL	SHIFT ECE	SHIFT MCE
TST Z128	21.13±0.52	33.52±0.87
TST Z2	3.39±0.16	40.69±1.85
TST Z256	20.43±0.06	32.76±0.1
TST Z32	21.38±1.29	33.53±1.36
TST Z512	17.44±0.08	28.51±0.14
TST Z8	21.95±1.0	30.06±1.54
TEMP. WRN 28-10	29.58	47.67
V-TST Z128, M=1	16.8±0.4	30.72±0.63
V-TST Z128, M=10	9.61±0.24	19.16±0.4
V-TST Z2, M=1	**0.500±0.04**	35.29±7.44
V-TST Z2, M=10	1.35±0.08	51.41±13.12
V-TST Z256, M=1	18.16±0.46	33.31±0.73
V-TST Z256, M=10	12.09±0.25	23.5±0.41
V-TST Z32, M=1	12.35±0.18	20.97±0.34
V-TST Z32, M=10	3.07±0.2	**7.380±0.3**
V-TST Z512, M=1	18.7±0.42	34.25±0.76
V-TST Z512, M=10	14.3±0.23	26.96±0.46
V-TST Z8, M=1	7.52±0.1	34.16±0.36
V-TST Z8, M=10	15.52±0.11	47.98±0.36
WRN 28-10	40.47	60.38

Table 12. CIFAR100 WRN OOD Results.

MODEL	OOD AUROC	OOD FPR95
TST Z128	0.806±0.006	0.8±0.016
TST Z2	0.612±0.021	0.925±0.008
TST Z256	0.812±0.004	0.791±0.008
TST Z32	0.778±0.011	0.829±0.019
TST Z512	**0.823±0.002**	**0.764±0.006**
TST Z8	0.722±0.018	0.862±0.025
TEMP. WRN 28-10	0.778	0.816
V-TST Z128, M=1	0.783±0.004	0.822±0.006
V-TST Z128, M=10	0.792±0.002	0.81±0.006
V-TST Z2, M=1	0.552±0.005	0.94±0.002
V-TST Z2, M=10	0.637±0.013	0.913±0.006
V-TST Z256, M=1	0.788±0.002	0.815±0.005
V-TST Z256, M=10	0.793±0.002	0.808±0.005
V-TST Z32, M=1	0.769±0.006	0.84±0.01
V-TST Z32, M=10	0.791±0.005	0.803±0.01
V-TST Z512, M=1	0.791±0.002	0.815±0.005
V-TST Z512, M=10	0.791±0.002	0.809±0.004
V-TST Z8, M=1	0.711±0.011	0.868±0.009
V-TST Z8, M=10	0.738±0.012	0.853±0.01
WRN 28-10	0.785	0.892

E.3. SVHN WRN Two-Stage Results

Table 13. SVHN WRN In Dist. Results.

MODEL	ACCURACY	ECE	MCE	TRAIN NLL	TEST NLL
TST Z128	**95.100±0.01**	**0.430±0.016**	25.04±9.617	0.0439±0.0001	**0.184±0.0004**
TST Z2	87.16±0.64	9.38±0.424	25.54±1.541	0.3045±0.0063	0.6263±0.0109
TST Z256	**95.100±0.01**	1.43±0.01	**9.180±0.429**	**0.042±0.0**	0.1891±0.0003
TST Z32	95.02±0.01	0.82±0.046	26.6±6.255	0.0504±0.0003	0.1922±0.0011
TST Z512	**95.100±0.01**	2.26±0.009	15.33±0.392	0.0433±0.0001	0.2028±0.0004
TST Z8	94.65±0.04	0.86±0.085	27.18±7.062	0.0592±0.0009	0.2234±0.003
TEMP. WRN 28-10	95.07	1.88	19.51	**0.042**	0.1926
V-TST Z128, M=1	94.27±0.02	0.82±0.111	17.66±7.441	0.0781±0.0008	0.2574±0.002
V-TST Z128, M=10	94.19±0.02	0.81±0.118	21.33±6.973	0.0781±0.0008	0.2588±0.002
V-TST Z2, M=1	43.87±0.42	7.04±0.36	16.88±0.875	1.5066±0.0079	1.5777±0.0081
V-TST Z2, M=10	43.79±0.41	6.97±0.328	16.4±0.914	1.506±0.0075	1.5777±0.0082
V-TST Z256, M=1	94.5±0.02	0.55±0.09	14.43±1.996	0.0667±0.0004	0.2345±0.0015
V-TST Z256, M=10	94.51±0.03	0.52±0.076	10.85±2.212	0.0669±0.0004	0.235±0.0013
V-TST Z32, M=1	92.8±0.04	1.58±0.11	23.54±6.687	0.1202±0.0014	0.3214±0.0021
V-TST Z32, M=10	92.78±0.03	1.54±0.132	24.63±6.551	0.1196±0.0015	0.3218±0.0017
V-TST Z512, M=1	94.73±0.02	0.48±0.067	22.63±6.889	0.059±0.0003	0.2193±0.0011
V-TST Z512, M=10	94.76±0.01	0.47±0.066	12.67±1.958	0.0587±0.0003	0.2195±0.0012
V-TST Z8, M=1	89.16±0.08	7.48±0.092	18.18±0.402	0.2531±0.0018	0.449±0.002
V-TST Z8, M=10	89.11±0.09	7.44±0.093	17.38±0.327	0.2538±0.0016	0.4493±0.0024
WRN 28-10	95.07	3.48	26.61	0.0605	0.2869

Table 14. SVHN WRN Shift Results.

MODEL	SHIFT ECE	SHIFT MCE
TST Z128	30.97±0.07	48.39±0.09
TST Z2	27.77±1.03	37.64±1.13
TST Z256	35.13±0.06	53.44±0.09
TST Z32	28.21±0.15	44.2±0.24
TST Z512	39.0±0.05	57.24±0.06
TST Z8	29.18±0.26	45.22±0.36
TEMP. WRN 28-10	37.53	55.09
V-TST Z128, M=1	37.33±0.6	51.5±0.65
V-TST Z128, M=10	37.28±0.62	51.73±0.76
V-TST Z2, M=1	**7.940±0.4**	15.36±1.35
V-TST Z2, M=10	**7.940±0.41**	**14.430±0.39**
V-TST Z256, M=1	35.95±0.55	51.39±0.6
V-TST Z256, M=10	35.93±0.54	51.23±0.54
V-TST Z32, M=1	38.18±0.39	50.43±0.5
V-TST Z32, M=10	38.22±0.37	50.39±0.45
V-TST Z512, M=1	35.67±0.59	52.33±0.59
V-TST Z512, M=10	35.72±0.58	52.36±0.59
V-TST Z8, M=1	28.91±0.13	37.36±0.2
V-TST Z8, M=10	28.84±0.13	37.39±0.19
WRN 28-10	47.06	80.65

Table 15. SVHN WRN OOD Results.

MODEL	OOD AUROC	OOD FPR95
TST Z128	0.935±0.001	0.415±0.006
TST Z2	0.8±0.024	0.757±0.031
TST Z256	0.936±0.0	0.412±0.003
TST Z32	0.933±0.002	0.432±0.011
TST Z512	0.937±0.0	0.404±0.002
TST Z8	0.918±0.003	0.495±0.018
TEMP. WRN 28-10	0.934	0.461
V-TST Z128, M=1	0.816±0.004	0.645±0.004
V-TST Z128, M=10	0.831±0.003	0.614±0.004
V-TST Z2, M=1	0.591±0.004	0.926±0.001
V-TST Z2, M=10	0.577±0.005	0.928±0.001
V-TST Z256, M=1	0.844±0.004	0.593±0.005
V-TST Z256, M=10	0.856±0.003	0.571±0.004
V-TST Z32, M=1	0.776±0.003	0.741±0.004
V-TST Z32, M=10	0.787±0.002	0.728±0.002
V-TST Z512, M=1	0.859±0.004	0.549±0.006
V-TST Z512, M=10	0.872±0.004	0.537±0.003
V-TST Z8, M=1	0.749±0.002	0.8±0.002
V-TST Z8, M=10	0.756±0.001	0.798±0.002
WRN 28-10	**0.945**	**0.344**

F. CIFAR10 CNN Two-Stage Results

Table 16. CIFAR10 CNN In Dist. Results.

MODEL	ACCURACY	ECE	MCE	TRAIN NLL	TEST NLL
CNN	72.48±0.0	1.04±0.0	19.89±0.0	0.7132±0.0	0.81±0.0
TST Z128	73.97±0.2	2.92±0.332	18.66±0.39	0.5951±0.0028	0.7593±0.0027
TST Z256	74.86±0.11	3.9±0.085	15.29±3.719	0.5326±0.0063	0.7478±0.0007
TST Z32	70.77±0.18	**0.760±0.172**	7.79±4.67	0.7706±0.0025	0.834±0.002
TST Z512	74.84±0.16	2.58±0.413	22.87±4.241	0.5596±0.0138	0.7397±0.0018
TST Z8	65.51±0.15	2.39±0.194	14.45±5.075	0.9707±0.0044	0.9889±0.0055
V-TST Z128, M=1	73.22±0.12	1.61±0.299	9.08±5.473	0.6416±0.002	0.7847±0.0022
V-TST Z128, M=10	74.44±0.13	1.26±0.28	14.54±2.986	0.6111±0.0017	0.7474±0.0004
V-TST Z256, M=1	74.23±0.14	3.65±0.278	22.5±4.379	0.5368±0.0029	0.7563±0.0042
V-TST Z256, M=10	75.27±0.11	1.72±0.241	9.96±4.611	0.5127±0.0043	0.7245±0.0024
V-TST Z32, M=1	66.74±0.28	1.09±0.209	7.81±2.272	0.8981±0.0069	0.953±0.0024
V-TST Z32, M=10	69.9±0.13	6.34±0.193	10.59±0.682	0.8332±0.0052	0.8779±0.0027
V-TST Z512, M=1	74.87±0.15	5.77±0.248	21.0±5.511	0.4442±0.0031	0.7735±0.0036
V-TST Z512, M=10	**75.950±0.15**	3.27±0.052	10.65±4.346	**0.415±0.0026**	**0.722±0.0012**
V-TST Z8, M=1	53.54±0.53	3.13±0.393	**6.620±0.857**	1.2905±0.0087	1.3107±0.0058
V-TST Z8, M=10	61.54±0.42	15.53±0.429	22.04±0.728	1.1659±0.0071	1.1765±0.0075

Table 17. CIFAR10 CNN Shift Results.

MODEL	SHIFT ECE	SHIFT MCE
CNN	9.13±0.0	11.8±0.0
TST Z128	9.85±0.18	13.68±0.3
TST Z256	11.03±0.26	15.46±0.22
TST Z32	7.02±0.08	9.39±0.35
TST Z512	9.54±0.61	13.37±0.83
TST Z8	5.53±0.24	8.47±0.07
V-TST Z128, M=1	8.18±0.13	11.15±0.08
V-TST Z128, M=10	5.37±0.23	6.91±0.26
V-TST Z256, M=1	10.23±0.34	14.68±0.53
V-TST Z256, M=10	7.87±0.3	11.06±0.5
V-TST Z32, M=1	6.33±0.23	9.0±0.31
V-TST Z32, M=10	**0.860±0.05**	**1.810±0.04**
V-TST Z512, M=1	13.31±0.14	19.75±0.25
V-TST Z512, M=10	10.06±0.16	14.56±0.27
V-TST Z8, M=1	4.23±0.04	5.9±0.2
V-TST Z8, M=10	7.66±0.34	12.02±0.83

Table 18. CIFAR10 CNN OOD Results.

MODEL	OOD AUROC	OOD FPR95
CNN	0.704±0.034	0.917±0.032
TST Z128	0.727±0.004	0.901±0.001
TST Z256	0.721±0.005	**0.898±0.007**
TST Z32	0.73±0.003	0.911±0.006
TST Z512	0.721±0.002	0.905±0.001
TST Z8	0.699±0.009	0.934±0.008
V-TST Z128, M=1	0.689±0.012	0.936±0.01
V-TST Z128, M=10	**0.731±0.007**	0.902±0.008
V-TST Z256, M=1	0.685±0.011	0.936±0.008
V-TST Z256, M=10	0.717±0.003	0.903±0.002
V-TST Z32, M=1	0.645±0.005	0.946±0.005
V-TST Z32, M=10	0.714±0.001	0.936±0.001
V-TST Z512, M=1	0.688±0.003	0.93±0.002
V-TST Z512, M=10	0.713±0.002	0.9±0.002
V-TST Z8, M=1	0.609±0.015	0.944±0.006
V-TST Z8, M=10	0.664±0.004	0.953±0.002

G. CIFAR10 ViT Two-Stage Results

Table 19. CIFAR10 ViT In Dist. Results.

MODEL	ACCURACY	ECE	MCE	TRAIN NLL	TEST NLL
TST Z128	82.59±0.05	3.87±0.049	28.2±8.792	0.1296±0.0003	**0.535±0.001**
TST Z2	60.98±1.19	5.48±0.601	19.11±2.255	0.7726±0.0264	1.3832±0.0314
TST Z256	82.84±0.04	7.03±0.037	25.24±6.103	**0.124±0.0002**	0.5772±0.0009
TST Z32	82.3±0.06	1.78±0.369	17.71±2.201	0.1497±0.0023	0.5452±0.0032
TST Z512	**82.960±0.03**	9.04±0.026	24.89±0.838	0.133±0.0003	0.6463±0.0012
TST Z8	80.49±0.16	4.91±0.462	22.6±6.482	0.1756±0.0058	0.6622±0.008
V-TST Z128, M=1	81.14±0.06	4.2±0.304	17.67±3.616	0.1616±0.0007	0.6538±0.0053
V-TST Z128, M=10	82.84±0.04	**1.230±0.181**	14.93±2.542	0.1407±0.0012	0.5637±0.0012
V-TST Z2, M=1	27.05±0.4	2.61±0.171	21.94±1.933	1.8358±0.014	1.9698±0.0107
V-TST Z2, M=10	38.64±0.91	18.95±0.946	55.29±0.73	1.7446±0.0152	1.8557±0.0132
V-TST Z256, M=1	81.76±0.04	5.11±0.272	18.33±1.854	0.1486±0.0004	0.636±0.0065
V-TST Z256, M=10	82.94±0.04	2.54±0.222	19.51±4.521	0.1319±0.0007	0.56±0.0026
V-TST Z32, M=1	79.42±0.08	2.44±0.267	13.14±1.92	0.2116±0.0013	0.7357±0.0033
V-TST Z32, M=10	82.72±0.03	4.06±0.274	10.55±0.858	0.1759±0.0019	0.5958±0.0011
V-TST Z512, M=1	82.1±0.07	5.57±0.245	27.88±6.765	0.1393±0.0004	0.6186±0.0053
V-TST Z512, M=10	82.92±0.04	3.54±0.149	19.28±1.854	0.1277±0.0006	0.561±0.002
V-TST Z8, M=1	75.18±0.12	4.41±0.226	**9.430±0.367**	0.3676±0.0022	0.8786±0.0036
V-TST Z8, M=10	82.33±0.06	15.57±0.21	23.12±0.318	0.3108±0.0025	0.7181±0.002
VIT	82.56	13.71	34.27	0.2363	1.1753

Table 20. CIFAR10 ViT Shift Results.

MODEL	SHIFT ECE	SHIFT MCE
TST Z128	11.06±0.07	19.03±0.12
TST Z2	8.72±0.8	13.68±1.06
TST Z256	15.34±0.06	27.38±0.1
TST Z32	8.16±0.48	14.14±0.82
TST Z512	18.26±0.03	33.18±0.07
TST Z8	12.98±0.54	21.08±1.02
V-TST Z128, M=1	10.72±0.44	18.28±0.75
V-TST Z128, M=10	5.44±0.35	10.05±0.57
V-TST Z2, M=1	1.69±0.22	12.86±2.16
V-TST Z2, M=10	13.73±0.66	52.9±0.85
V-TST Z256, M=1	12.16±0.4	21.13±0.77
V-TST Z256, M=10	7.87±0.31	13.98±0.53
V-TST Z32, M=1	9.2±0.41	14.32±0.6
V-TST Z32, M=10	**1.290±0.14**	**3.600±0.12**
V-TST Z512, M=1	12.78±0.26	22.14±0.53
V-TST Z512, M=10	9.57±0.2	16.68±0.43
V-TST Z8, M=1	2.84±0.21	4.29±0.13
V-TST Z8, M=10	11.66±0.2	15.65±0.29
VIT	25.46	45.2

Table 21. CIFAR10 ViT OOD Results.

MODEL	OOD AUROC	OOD FPR95
TST Z128	0.821±0.002	0.757±0.006
TST Z2	0.71±0.022	0.87±0.018
TST Z256	0.816±0.001	0.772±0.002
TST Z32	0.823±0.002	0.751±0.007
TST Z512	0.812±0.001	0.791±0.003
TST Z8	0.797±0.004	0.806±0.009
V-TST Z128, M=1	0.81±0.001	0.769±0.002
V-TST Z128, M=10	0.832±0.001	0.727±0.003
V-TST Z2, M=1	0.573±0.006	0.933±0.003
V-TST Z2, M=10	0.667±0.02	0.897±0.012
V-TST Z256, M=1	0.813±0.001	0.765±0.002
V-TST Z256, M=10	0.83±0.001	0.733±0.004
V-TST Z32, M=1	0.789±0.001	0.795±0.002
V-TST Z32, M=10	0.832±0.001	**0.721±0.002**
V-TST Z512, M=1	0.816±0.002	0.765±0.005
V-TST Z512, M=10	0.827±0.002	0.744±0.006
V-TST Z8, M=1	0.764±0.003	0.825±0.004
V-TST Z8, M=10	**0.835±0.003**	**0.721±0.011**
VIT	0.797	0.823