EXPLORING THE SIMILARITY BETWEEN MEMO AND SOFT-VOTING

Mengying Lin, Yu Sun
University of California, Berkeley
cassie_lin@berkeley.edu, yusun@berkeley.edu.

ABSTRACT

The article written by Zhang et al. (2021) published in NeurIPS'22 proposed a test time adaptation method called MEMO, which suggests feeding the model with the augmentations of one test point and then minimizing the entropy of marginal output over augmentations during optimization. This report aims to prove this method is actually equivalent to soft-voting both theoretically and empirically.

1 THEORETICAL ANALYSIS OF MEMO

The loss of MEMO is defined as follows

$$\ell(\theta; x) \triangleq H\left(\bar{p}_\theta(\cdot|x)\right) = -\sum_{y \in \mathcal{Y}} \bar{p}_\theta(y|x) \log \bar{p}_\theta(y|x),$$

where x is a single test point, $\bar{p}_\theta(y|x)$ is the model's average predictions of the augmentations. To minimize the loss term, the components of $\bar{p}_\theta(y|x)$ are expected to be driven to either 0 or 1, which indicates the model tends to predict uniformly and confidently if the augmented data are derived from the same data point.

However, when we further examine how the final predictions will look like, it turns out MEMO makes no huge difference compared with soft-voting.

Denote $\bar{p}_\theta(y|x)$ as $\{\bar{p}_{\theta 1}, ..., \bar{p}_{\theta n}\}$, where $\bar{p}_{\theta i}$ is the predicted average probability that x belongs to class i. When implementing gradient decent using the loss function, the gradients over $\bar{p}_{\theta i}$ will be

$$\frac{\partial \ell}{\partial \bar{p}_{\theta i}} = -log(\bar{p}_{\theta i}) - 1.$$

According to the update rules,

$$\bar{p}_{\theta i} \leftarrow \bar{p}_{\theta i} - \eta \frac{\partial \ell}{\partial \bar{p}_{\theta i}}.$$

Those $\bar{p}_{\theta i}$ that are close to 1 will head towards 1 even closer with a even faster pace, which means we are actually averaging the predictions to obtain $\{\bar{p}_{\theta i}\}$ and then trying to pick out the i that maximize $\bar{p}_{\theta i}$, which is exactly what soft-voting will do. Hence, MEMO seems to be equivalent to soft-voting from a theoretical point of view.

2 EMPIRICAL PROOF WITH EXPERIMENTS

To verify the equivalence between soft-voting and MEMO, a series of tests are carried out on Cifar10-C dataset, based on the "episodic" adaptation protocol (i.e. The model is reset after each test batch.) given in the article. To be consistent with the settings of the articles, batch size is set to 32, which means in exp2 and exp3 models will be adapted on 32 augmentations before testing for a single test point.

Exp1 is testing without any adaptations, which serves as a baseline. Exp2 is simply averaging the prediction during test time. Exp3 is implementing MEMO when testing.

The results results for CIFAR-10, CIFAR-10.1, and CIFAR-10-C are presented in the table, with full results for CIFAR-10-C attached in the appendix.

Table 1: Test error (%) on CIFAR-10 series, ResNet-26 with Batch Normalization.

	CIFAR-10	CIFAR-10.1	CIFAR-10-C
ResNet-26	9.17	18.40	22.54
+ Avg	7.32	14.60	20.06
+ MEMO	7.25	14.60	19.55

It is clearly seen that averaging the predictions achieves similar accuracies compared with MEMO.

3 Discussion

We have examined the similarity between MEMO and soft-voting in a both theoretical and empirical manner. They both favor results that are most confident in average when analyzing mathematically, and reduce the testing errors to resemble values when put into practice. However, MEMO seems to always outperform soft-voting slightly when tested on CIFAR-10-C, and the reasons behind are unclear yet.

Despite of the less competitive performance over CIFAR-10-C, it is worth mentioning that soft-voting takes far shorter time than MEMO (9min vs 14min), and manage maintain relatively low testing errors at the same time, which means when there is a demand for faster inference, it is might be more effective to simply implement soft-voting than MEMO with “episodic” adaptation protocol.

At the same time, we should bear in mind soft-voting will not help if the model has a poor performance over augmentations. In the experiments the results see an improvement maybe because the model is pretrained on heavily augmented pictures, when tested on those augmented data it can get better results. It might be the same case for MEMO if the model is performing terribly over augmented data, leading to a decrease in accuracies, which demands further investigations.

References

Marvin Zhang, Sergey Levine, and Chelsea Finn. Memo: Test time robustness via adaptation and augmentation. *arXiv preprint arXiv:2110.09506*, 2021.

A APPENDIX

A.1 RESULTS ON CIFAR-10-C, LEVEL 1-5

In terms of testing time for each sub-dataset, baseline takes approximately **30˜40s**, soft-voting takes **8min 50s ˜ 9min**, and MEMO takes **14 min 30s ˜ 15min.**

Table 2: Test error (%) on CIFAR-10-C level 1 corruptions, ResNet-26 with Batch Normalization.

	gauss	shot	impul	defoc	glass	motn	zoom	snow	frost	fog	brit	contr	elast	pixel	jpeg
ResNet-26	20.8	16.5	15.7	9.2	38.9	11.8	12.8	13.9	13.4	9.7	9.4	9.6	13.1	12.0	16.4
+SV	16.3	12.9	11.7	7.4	34.9	11.1	12.8	10.9	10.9	7.5	7.4	7.7	11.6	9.2	12.6
+MEMO	16.0	12.8	11.8	7.4	34.9	10.3	12.1	10.9	10.8	7.5	7.3	7.5	11.0	9.2	12.4
+MEMO,SV	0.2	0.1	0.1	0.1	0.4	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1

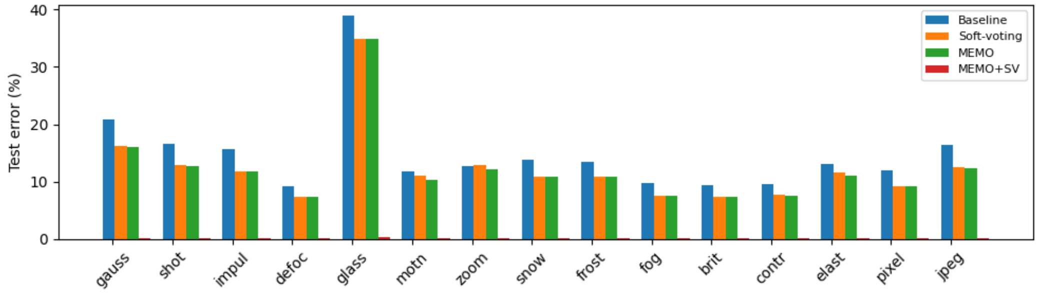


Table 3: Test error (%) on CIFAR-10-C level 2 corruptions, ResNet-26 with Batch Normalization.

	gauss	shot	impul	defoc	glass	motn	zoom	snow	frost	fog	brit	contr	elast	pixel	jpeg
ResNet-26	30.1	21.8	21.2	9.7	38.3	15.3	13.8	21.2	17.6	10.5	9.7	11.6	12.9	15.4	21.3
+SV	25.4	17.2	15.7	8.7	34.0	15.8	13.9	18.1	14.8	8.7	7.9	10.0	11.2	11.9	16.4
+MEMO	25.1	16.9	15.9	8.4	33.4	14.8	13.1	17.7	14.6	8.5	7.7	9.7	10.8	12.0	16.1
+MEMO,SV	0.2	0.2	0.2	0.1	0.3	0.1	0.1	0.2	0.1	0.1	0.1	0.1	0.1	0.1	0.2

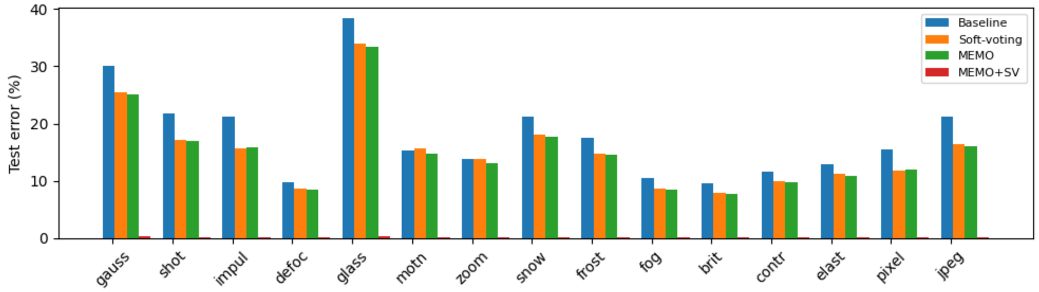


Table 4: Test error (%) on CIFAR-10-C level 3 corruptions, ResNet-26 with Batch Normalization.

	gauss	shot	impul	defoc	glass	motn	zoom	snow	frost	fog	brit	contr	elast	pixel	jpeg
ResNet-26	40.0	33.8	26.3	11.5	37.3	20.0	16.6	20.0	24.7	12.2	10.5	13.6	15.0	18.4	22.7
+SV	34.6	28.3	19.8	11.5	32.8	21.1	17.3	16.2	21.5	10.2	8.7	12.7	14.0	14.8	17.4
+MEMO	34.4	28.2	20.1	10.9	32.6	19.7	16.2	16.0	20.8	9.9	8.5	11.8	13.1	14.2	17.2
+MEMO,SV	0.3	0.3	0.2	0.1	0.3	0.2	0.2	0.2	0.2	0.1	0.1	0.1	0.1	0.1	0.2

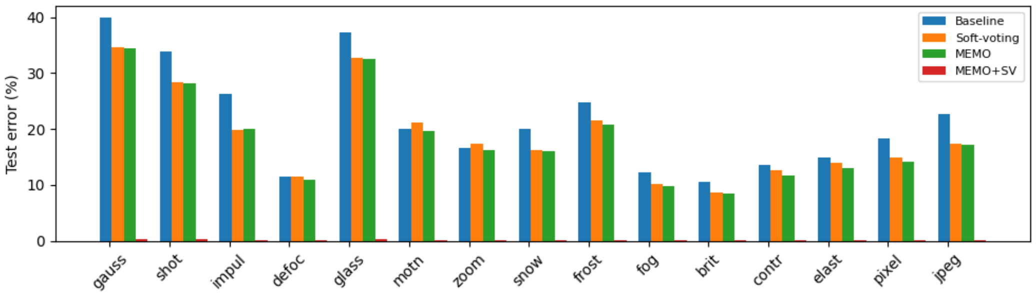


Table 5: Test error (%) on CIFAR-10-C level 4 corruptions, ResNet-26 with Batch Normalization.

	gauss	shot	impul	defoc	glass	motn	zoom	snow	frost	fog	brit	contr	elast	pixel	jpeg
ResNet-26	43.8	37.2	39.3	14.8	48.0	19.9	18.7	22.0	24.9	15.1	11.4	16.8	19.1	27.9	24.9
+SV	39.5	32.5	31.9	15.6	45.0	21.4	20.7	18.6	21.9	13.1	9.2	16.9	17.8	25.8	19.4
+MEMO	39.4	32.2	32.2	14.9	44.9	19.8	19.1	18.7	21.3	12.4	9.5	15.4	17.0	25.4	19.2
+MEMO,SV	0.3	0.3	0.2	0.1	0.3	0.2	0.2	0.2	0.2	0.1	0.1	0.1	0.1	0.1	0.2

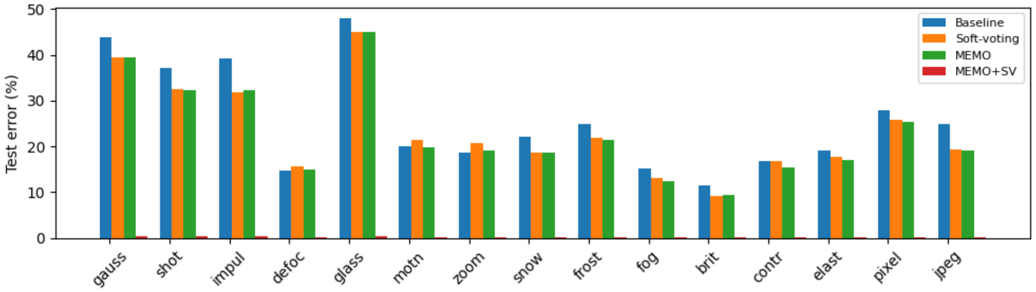
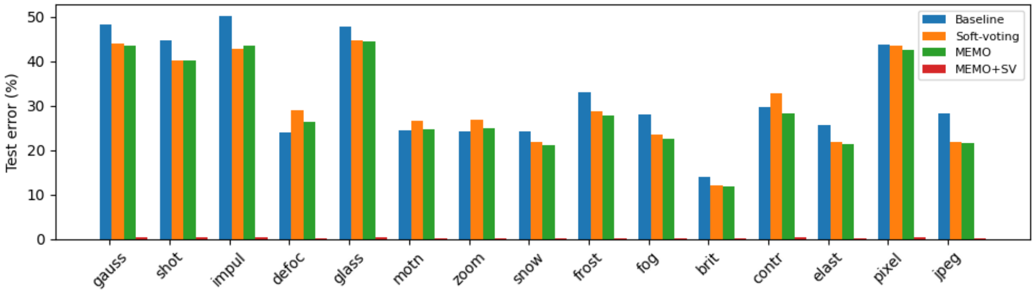


Table 6: Test error (%) on CIFAR-10-C level 5 corruptions, ResNet-26 with Batch Normalization.

	gauss	shot	impul	defoc	glass	motn	zoom	snow	frost	fog	brit	contr	elast	pixel	jpeg
ResNet-26	48.4	44.8	50.3	24.1	47.7	24.5	24.1	24.1	33.1	28.0	14.1	29.7	25.5	43.7	28.3
+SV	44.0	40.1	42.9	28.9	44.7	26.6	26.9	21.9	28.7	23.6	12.0	32.8	21.8	43.5	21.8
+MEMO	43.6	40.1	43.5	26.4	44.5	24.8	24.9	21.2	27.8	22.6	11.8	28.3	21.3	42.5	21.7
+MEMO,SV	0.4	0.3	0.3	0.2	0.5	0.2	0.2	0.2	0.2	0.1	0.1	0.2	0.2	0.3	0.2


Test error (%)
Baseline
Soft-voting
MEMO
MEMO+SV
gauss
shot
impul
defoc
glass
motn
zoom
snow
frost
fog
brit
contr
elast
pixel
jpeg