Do We Need Zero Training Loss After Achieving Zero Training Error?

By

Takashi Ishida et al. ICML 2020

Overfitting

Overfitting refers to the condition when a model memorizes the training data too well and therefore fails in generalize to the underlying function completely.

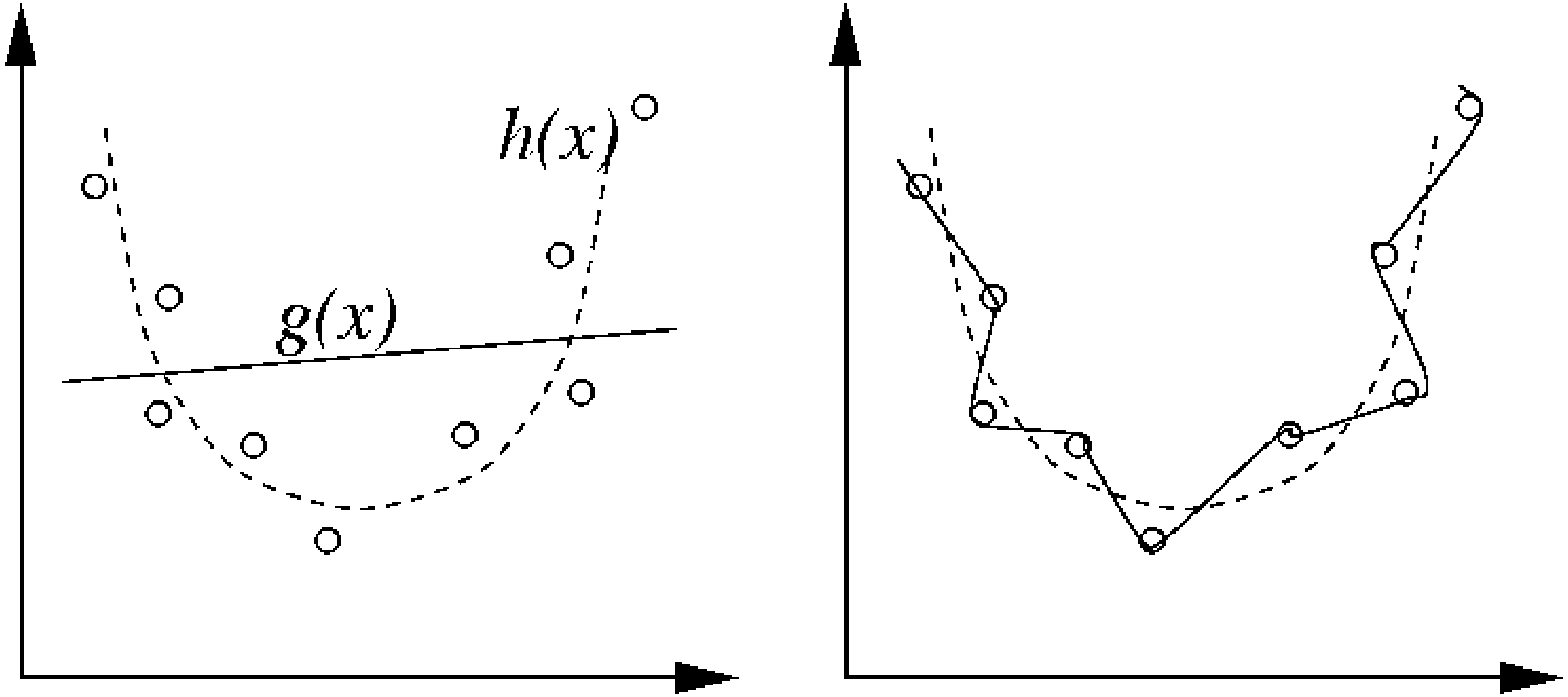


Image source: https://cnl.salk.edu/~schraudo/teach/NNcourse/overfitting.html

Regularization Methods

- Regularizing weights (L1,L2)
- Dropping layer
- Smoothing training labels
- Early stopping

Limitation

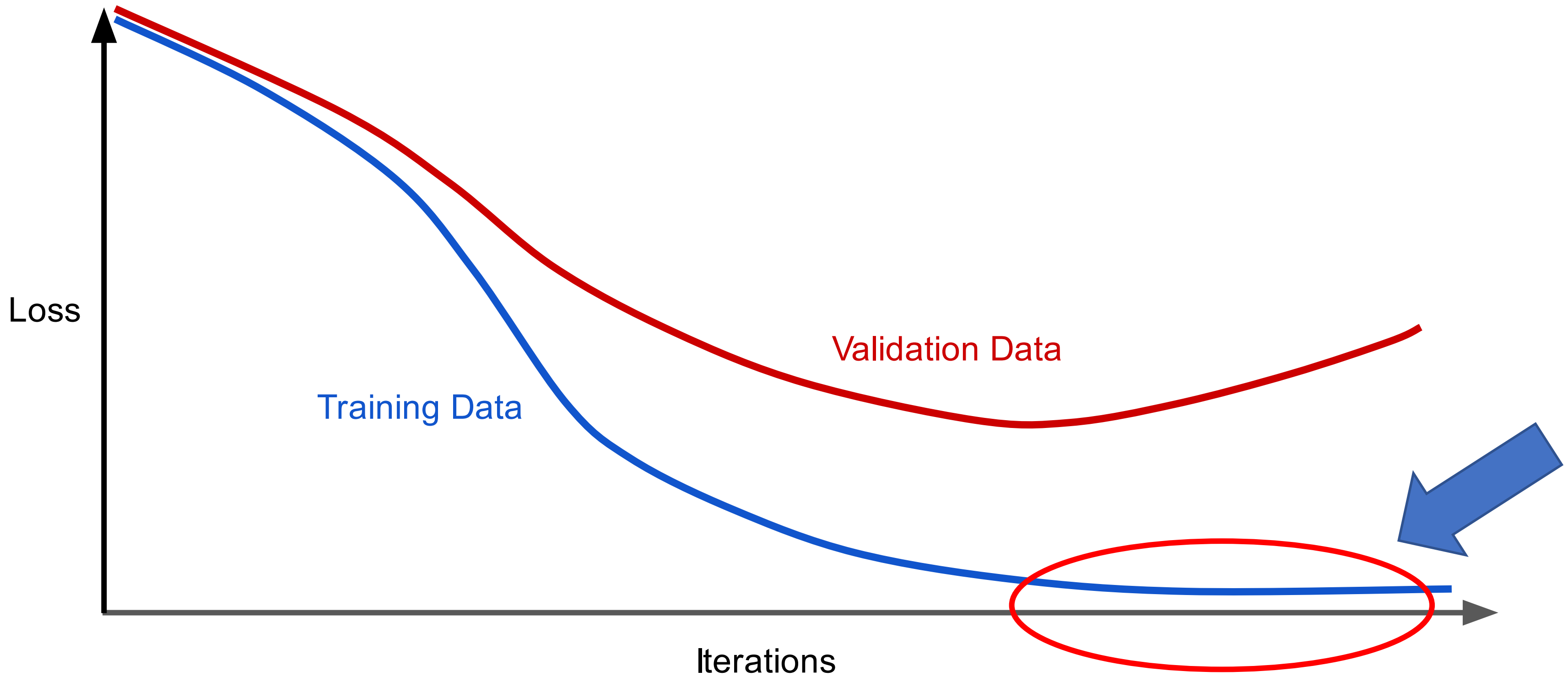


Image source: https://developers.google.com/machine-learning/crash-course/regularization-for-simplicity/l2-regularization

Flooding

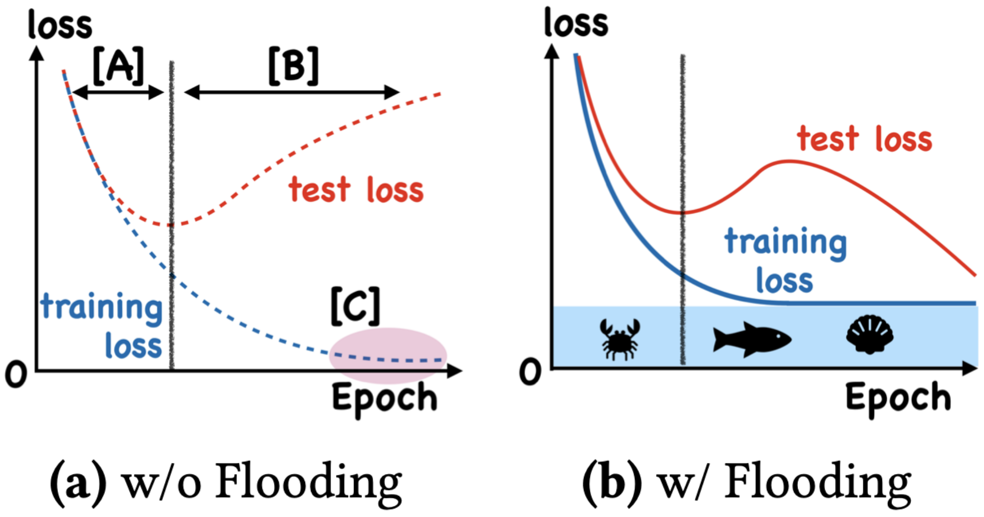
Proposed Objective function

$$\tilde{J}(\theta) = |J(\theta) - b| + b \ ,$$

b = Flooding constant

Algorithm

1. outputs = model(inputs)
2. loss = criterion(outputs, labels)
3. flood = (loss-b).abs()+b # This is it!
4. optimizer.zero_grad()
5. flood.backward()
6. optimizer.step()



(a) w/o Flooding **(b)** w/ Flooding

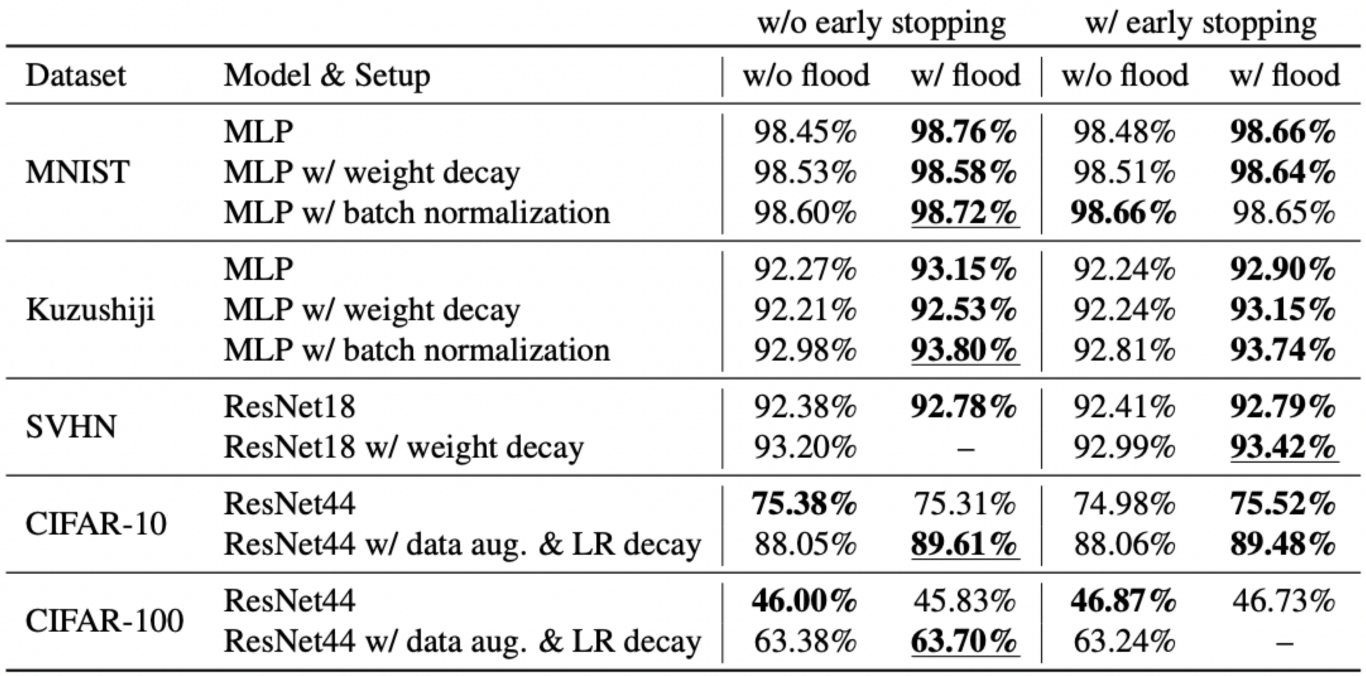
Image source: *Takashi Ishida, Ikko Yamane, Tomoya Sakai, Gang Niu, and Masashi Sugiyama. 2020. Do we need zero training loss after achieving zero training error? In Proceedings of the 37th International Conference on Machine Learning (ICML'20). JMLR.org, Article 428, 4604–4614*

Why this paper?

Simplicity

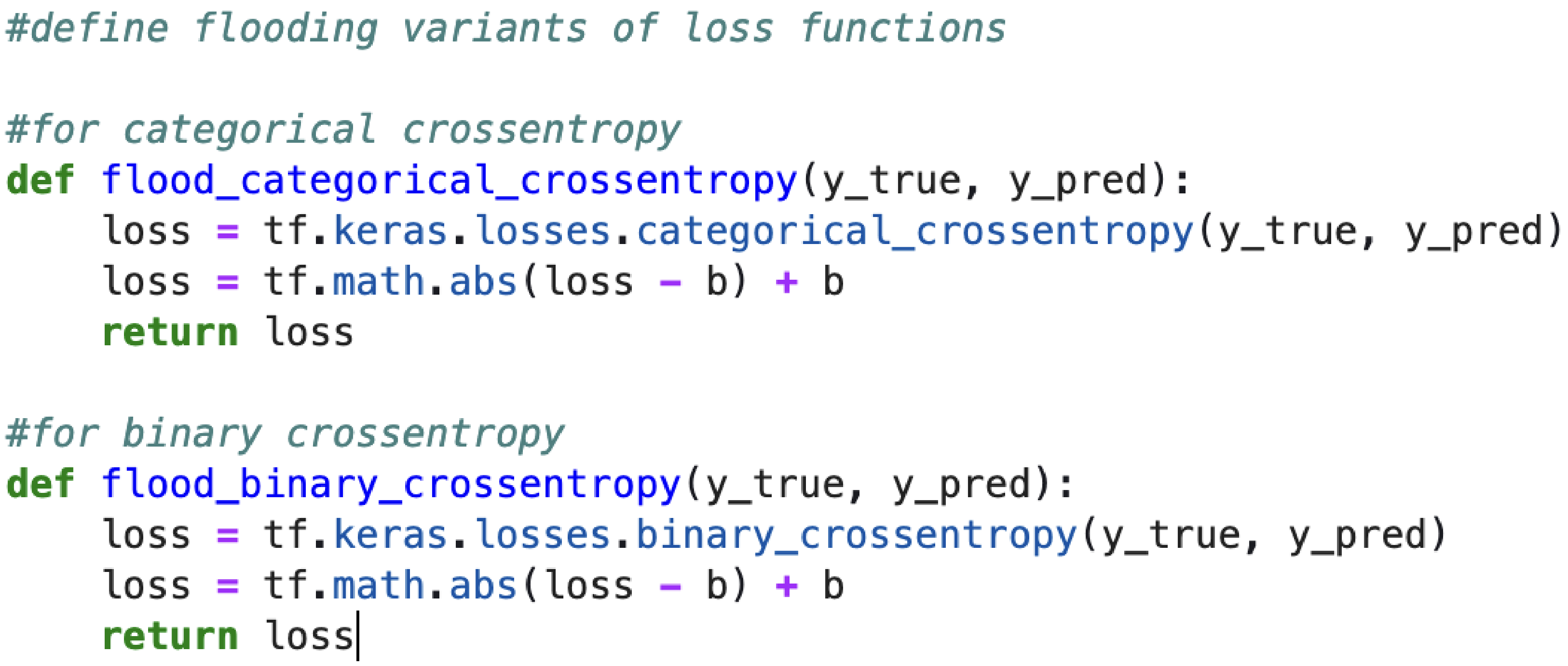
- No extra computational cost (Adding Dropping layers etc.)
- Applicable to lots of machine learning and deep learning models
- Avoids Zero Training Loss

Results



		w/o early stopping		w/ early stopping	
Dataset	Model & Setup	w/o flood	w/ flood	w/o flood	w/ flood
MNIST	MLP	98.45%	**98.76%**	98.48%	**98.66%**
	MLP w/ weight decay	98.53%	**98.58%**	98.51%	**98.64%**
	MLP w/ batch normalization	98.60%	**98.72%**	**98.66%**	98.65%
Kuzushiji	MLP	92.27%	**93.15%**	92.24%	**92.90%**
	MLP w/ weight decay	92.21%	**92.53%**	92.24%	**93.15%**
	MLP w/ batch normalization	92.98%	**93.80%**	92.81%	**93.74%**
SVHN	ResNet18	92.38%	**92.78%**	92.41%	**92.79%**
	ResNet18 w/ weight decay	93.20%	–	92.99%	**93.42%**
CIFAR-10	ResNet44	**75.38%**	75.31%	74.98%	**75.52%**
	ResNet44 w/ data aug. & LR decay	88.05%	**89.61%**	88.06%	**89.48%**
CIFAR-100	ResNet44	**46.00%**	45.83%	**46.87%**	46.73%
	ResNet44 w/ data aug. & LR decay	63.38%	**63.70%**	63.24%	–

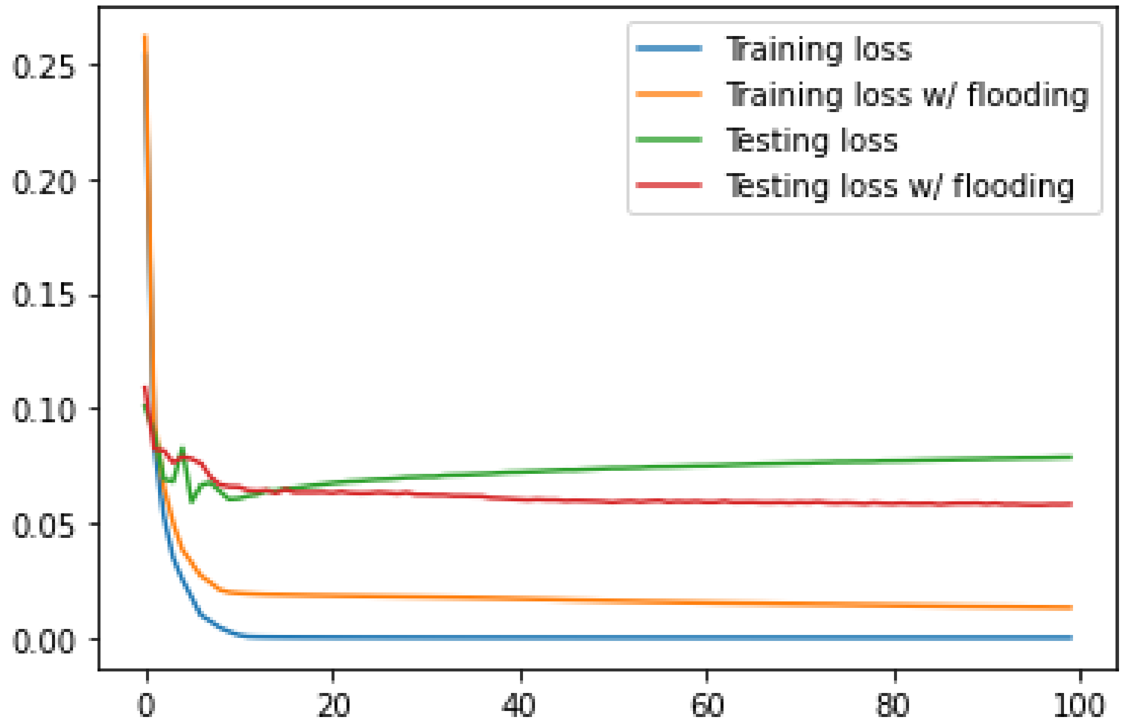
Implementation

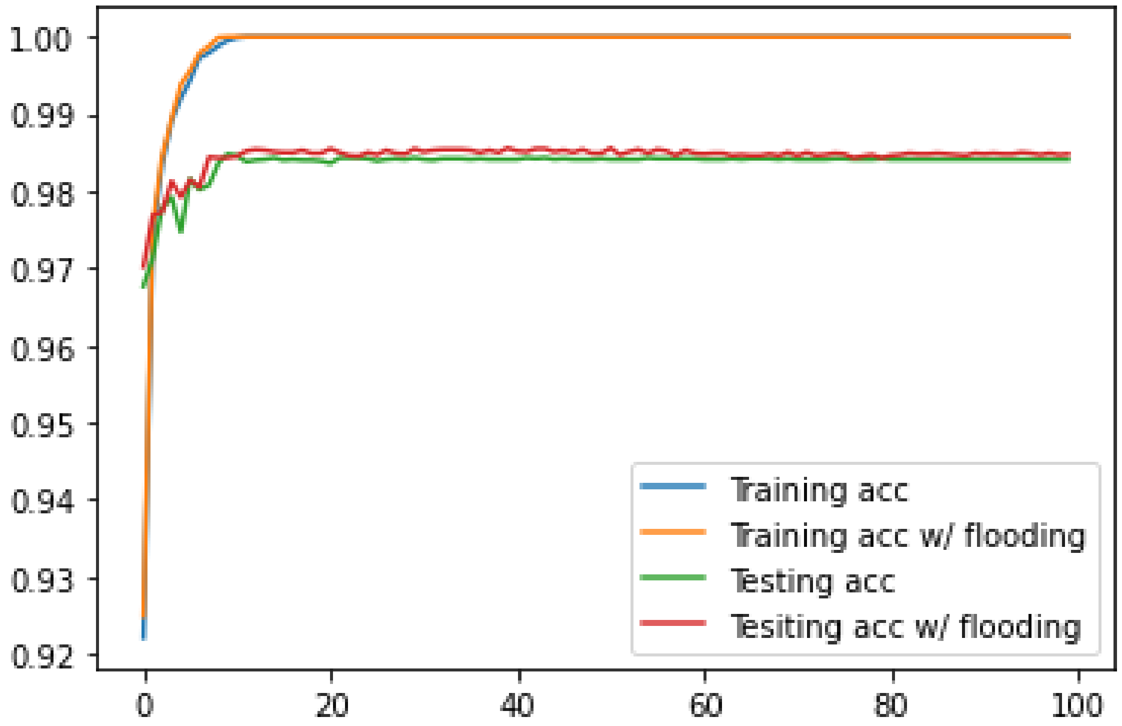


```
#define flooding variants of loss functions

#for categorical crossentropy
def flood_categorical_crossentropy(y_true, y_pred):
    loss = tf.keras.losses.categorical_crossentropy(y_true, y_pred)
    loss = tf.math.abs(loss - b) + b
    return loss

#for binary crossentropy
def flood_binary_crossentropy(y_true, y_pred):
    loss = tf.keras.losses.binary_crossentropy(y_true, y_pred)
    loss = tf.math.abs(loss - b) + b
    return loss
```

MNIST

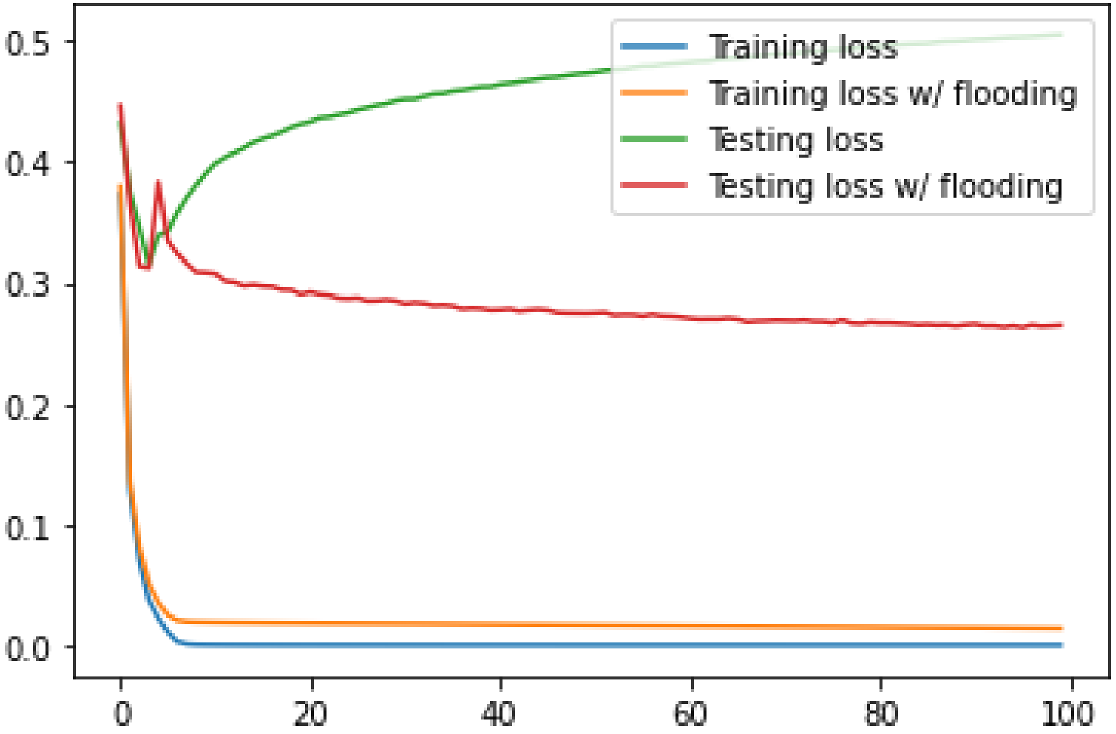
- Testing acc: 98.41%
- Testing acc w/ Flooding: 98.48%

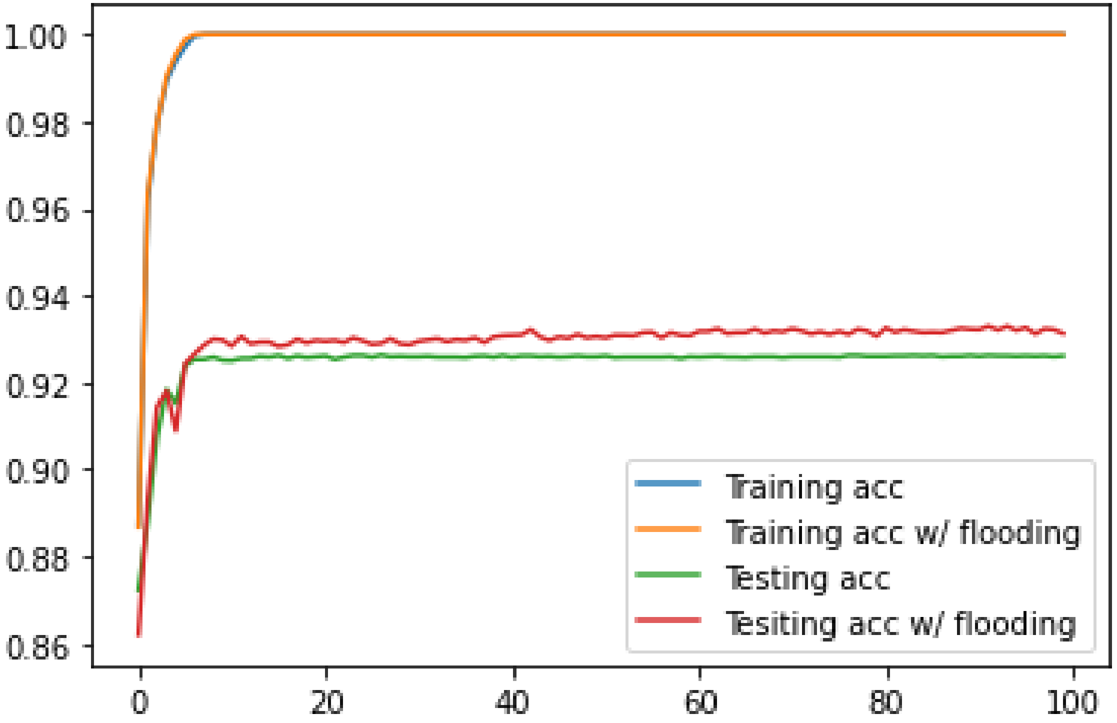




KMNIST

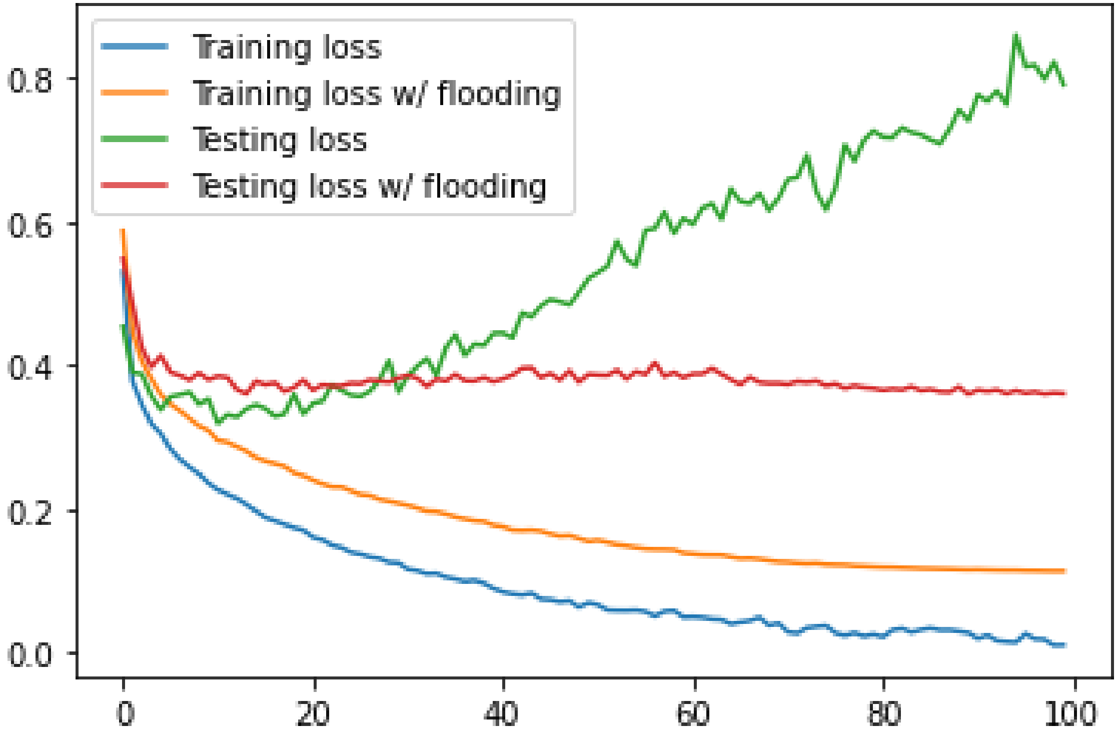
- Testing acc: 92.61%
- Testing acc w/ Flooding: 93.13%

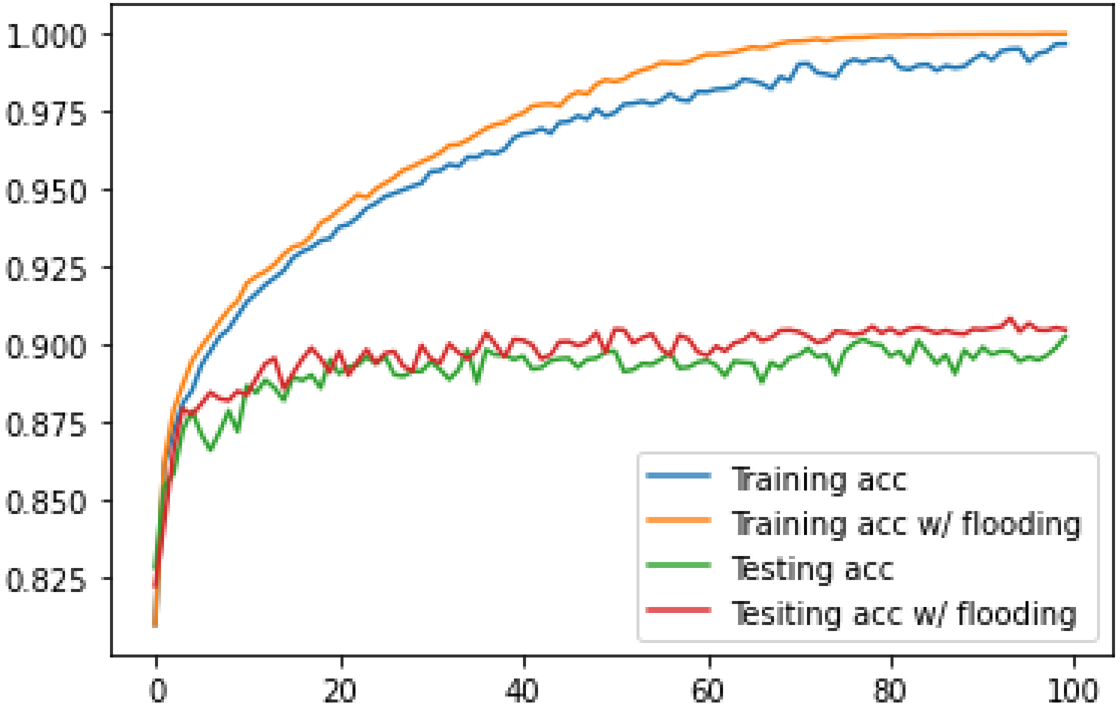




Fashion-MNIST

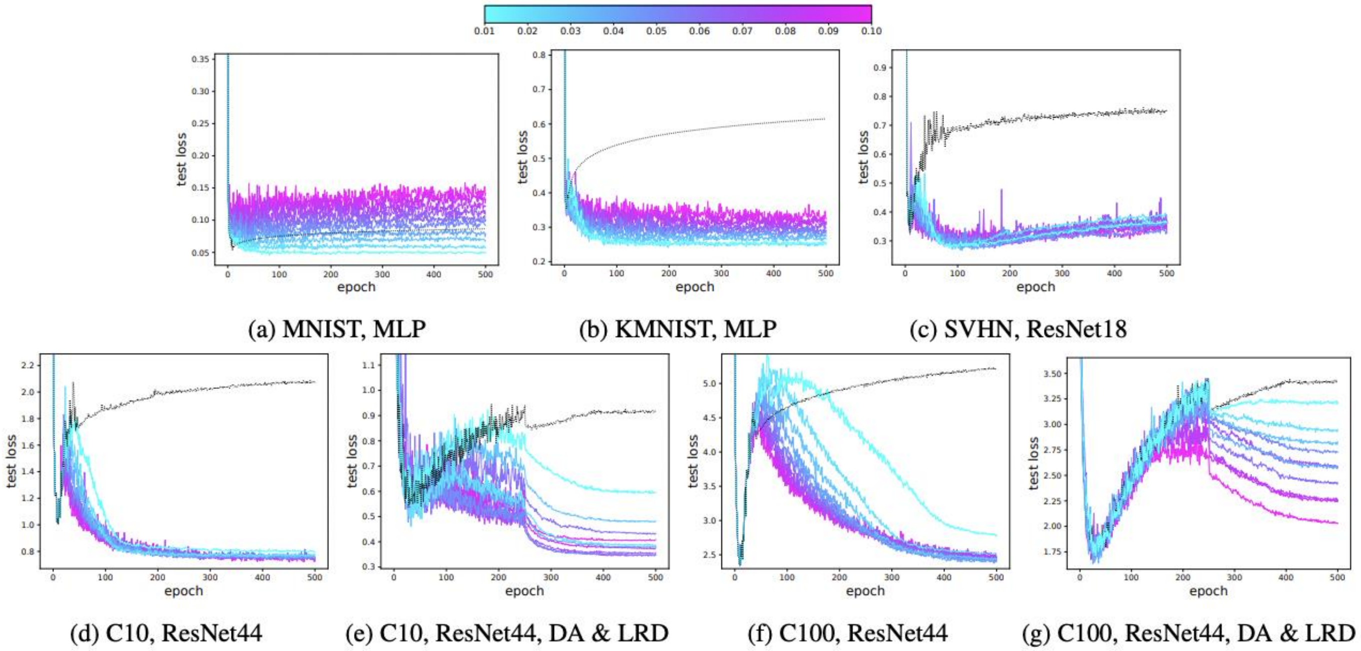
- Testing acc: 90.23%
- Testing acc w/ flooding: 90.45%





Analysis

- Image source: *Takashi Ishida, Ikko Yamane, Tomoya Sakai, Gang Niu, and Masashi Sugiyama. 2020. Do we need zero training loss after achieving zero training error? In Proceedings of the 37th International Conference on Machine Learning (ICML'20). JMLR.org, Article 428, 4604–4614*



(a) MNIST, MLP

(b) KMNIST, MLP

(c) SVHN, ResNet18

(d) C10, ResNet44

(e) C10, ResNet44, DA & LRD

(f) C100, ResNet44

(g) C100, ResNet44, DA & LRD

Figure 2. Learning curves of test loss. The black dotted line shows the baseline without flooding. The colored lines show the results for different flooding levels specified by the color bar. DA and LRD stand for data augmentation and learning rate decay, respectively. We can observe that adding flooding will lead to lower test loss with a double descent curve in most cases. See Fig. 6 in Appendix for other datasets.

Possible PRs

- Test for Deep CNNs
- Test Adaptive Flooding
  - Decay b (Flooding Constant)
  - Example:
    - If $\eta - \eta\%100 = \eta$ then, $b = b - \lambda$

      where, η = number of epochs, λ = constant
- Test effects of learning rate decay along with Adaptive Flooding
- Test on other datasets
- Any other suggestions..!!

References

- *Takashi Ishida, Ikko Yamane, Tomoya Sakai, Gang Niu, and Masashi Sugiyama. 2020. Do we need zero training loss after achieving zero training error? In Proceedings of the 37th International Conference on Machine Learning (ICML'20). JMLR.org, Article 428, 4604–4614*
- *Tarin Clanuwat, Mikel Bober-Irizar, Asanobu Kitamoto, Alex Lamb, Kazuaki Yamamoto, David Ha, "Deep Learning for Classical Japanese Literature", arXiv:1812.01718*.

Thank you very much.

Have a nice day !!!