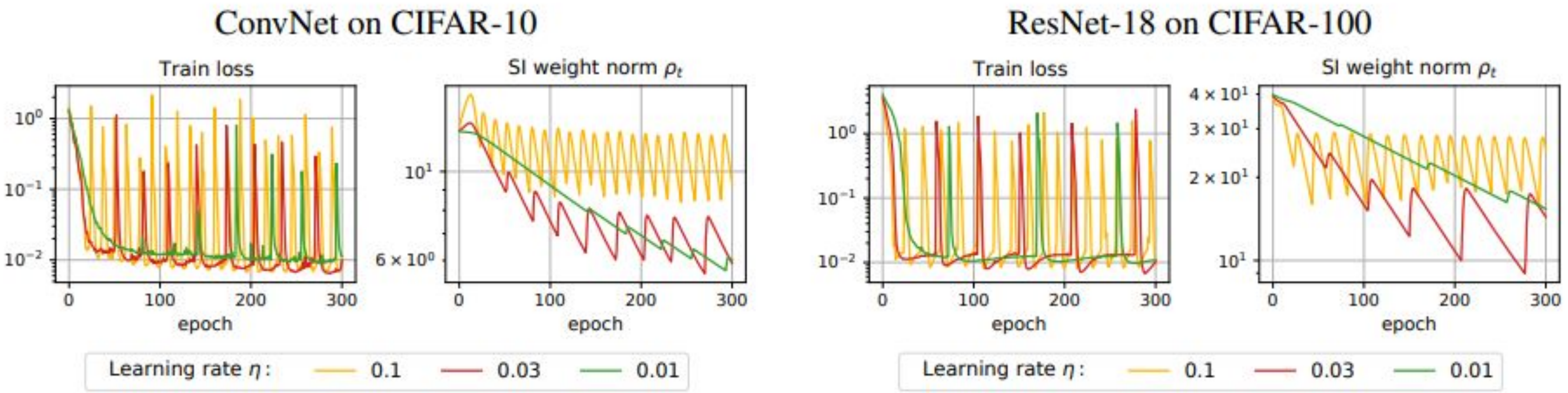
On the Periodic Behavior of Neural Network Training with Batch Normalization and Weight Decay

Ekaterina Lobacheva, Maxim Kodryan,
Nadezhda Chirkova, Andrey Malinin , Dmitry Vetrov

Докладчик: Александра Бакалова
Рецензент: Алексей Цеховой
Хакер: Андрей Боревский

Periodic effect

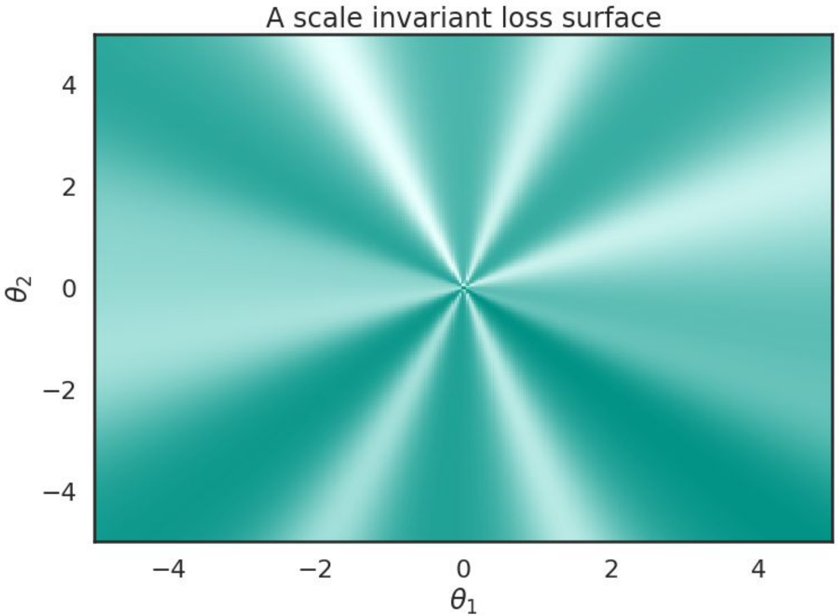
Making an SGD step in the direction of the loss gradient always increases the norm of scale-invariant parameters (those with BN), while WD aims at decreasing the weight norm



Background. Scale invariance of weights with batchnorm.

Consider an arbitrary scale-invariant function f(x), i.e., f(ax) = f(x), ∀ x and ∀ a > 0. Then:

$$\begin{cases} \langle \nabla f(x), x \rangle = 0, \ \forall x \\ \nabla f(\alpha x) = \dfrac{1}{\alpha} \nabla f(x), \ \forall x, \ \alpha > 0. \end{cases}$$



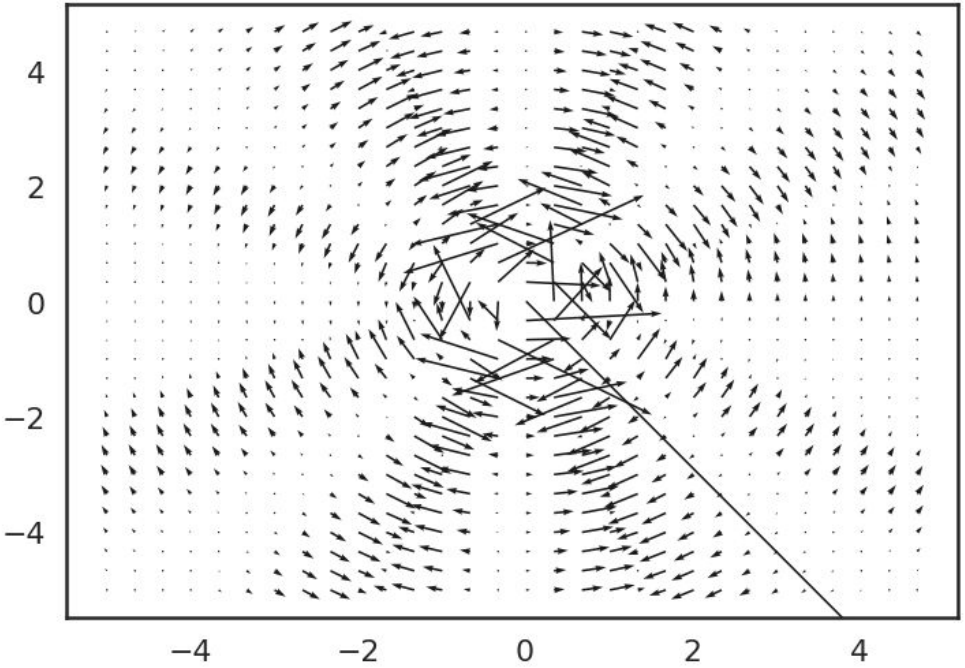
*https://www.inference.vc/exponentially-growing-learning-rate-implications-of-scale-invariance-induced-by-batchnorm/

Background. Scale invariance of weights with batchnorm.

Consider an arbitrary scale-invariant function f(x), i.e., f(ax) = f(x), ∀ x and ∀ a > 0.

Then:

$$\begin{cases} \langle \nabla f(x), x \rangle = 0, \ \forall x \\ \nabla f(\alpha x) = \dfrac{1}{\alpha} \nabla f(x), \ \forall x, \ \alpha > 0. \end{cases}$$



Background. Weight decay.

(S)GD optimization step with weight decay:

$$x_{t+1} = (1 - \eta\lambda)x_t - \eta\nabla f(x_t)$$

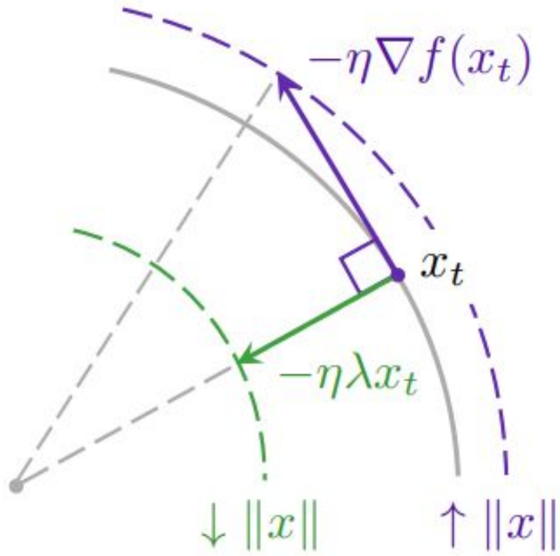
Corollaries. “Centripetal” and “centrifugal” forces.

(S)GD optimization step:

$$x_{t+1} = (1 - \eta\lambda)x_t - \eta\nabla f(x_t)$$

Properties of scale-invariant weights:

$$\begin{cases} \langle \nabla f(x), x \rangle = 0, \ \forall x \\ \nabla f(\alpha x) = \dfrac{1}{\alpha}\nabla f(x), \ \forall x, \ \alpha > 0. \end{cases}$$



Corollaries. Optimization steps.

(S)GD optimization step:

$$x_{t+1} = (1 - \eta\lambda)x_t - \eta\nabla f(x_t)$$

Properties of scale-invariant weights:

$$\begin{cases} \langle \nabla f(x), x \rangle = 0, \ \forall x \\ \nabla f(\alpha x) = \dfrac{1}{\alpha}\nabla f(x), \ \forall x, \ \alpha > 0. \end{cases}$$

The larger weight norm, the smaller optimization steps.

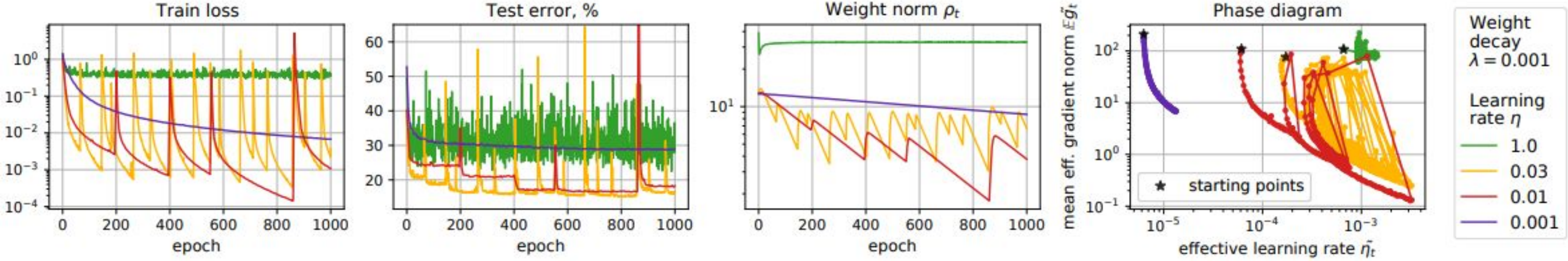
Experimental setup

- All learnable weights of a neural network are scale-invariant
    - Insert additional BN layers and fix the non-scale-invariant weights to be constant
- SGD with constant learning rate, without momentum or data augmentation
- Varied learning rate and fixed weight decay of 0.001

Models: ResNet-18 and ConvNet (simple 3-layer batch-normalized convolutional neural network)

Datasets: CIFAR-10 and CIFAR-100

ConvNet on CIFAR-10

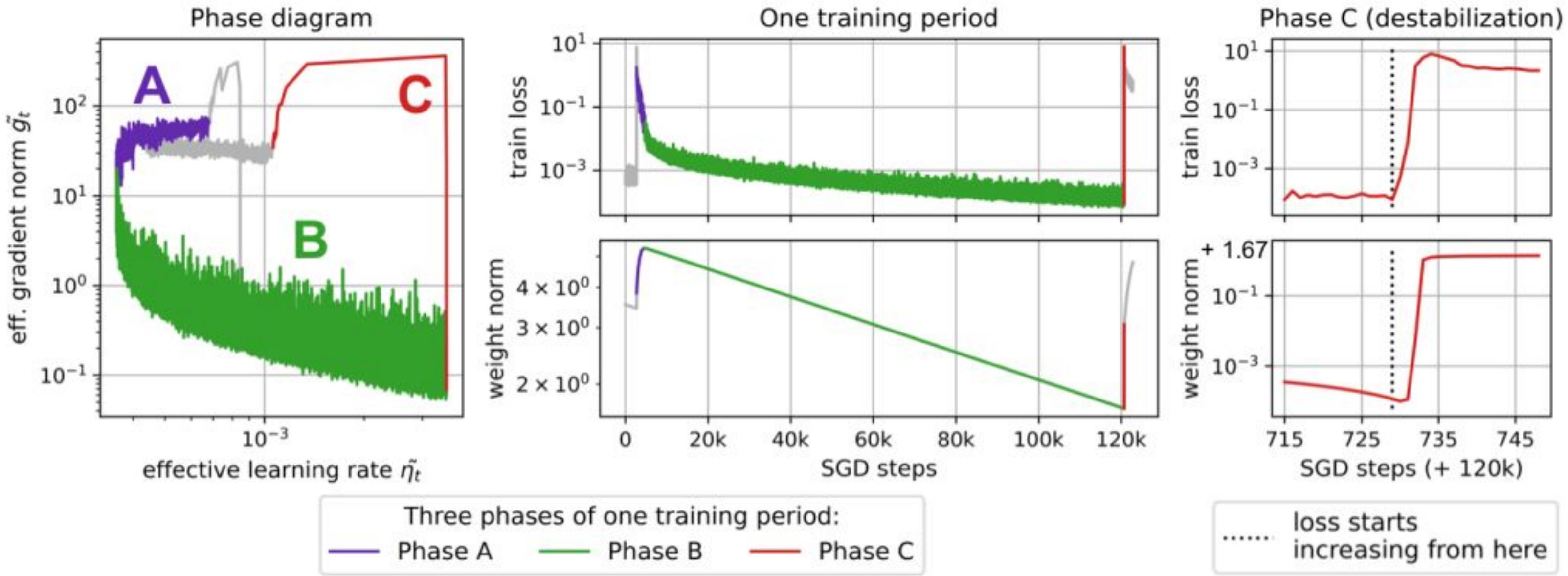


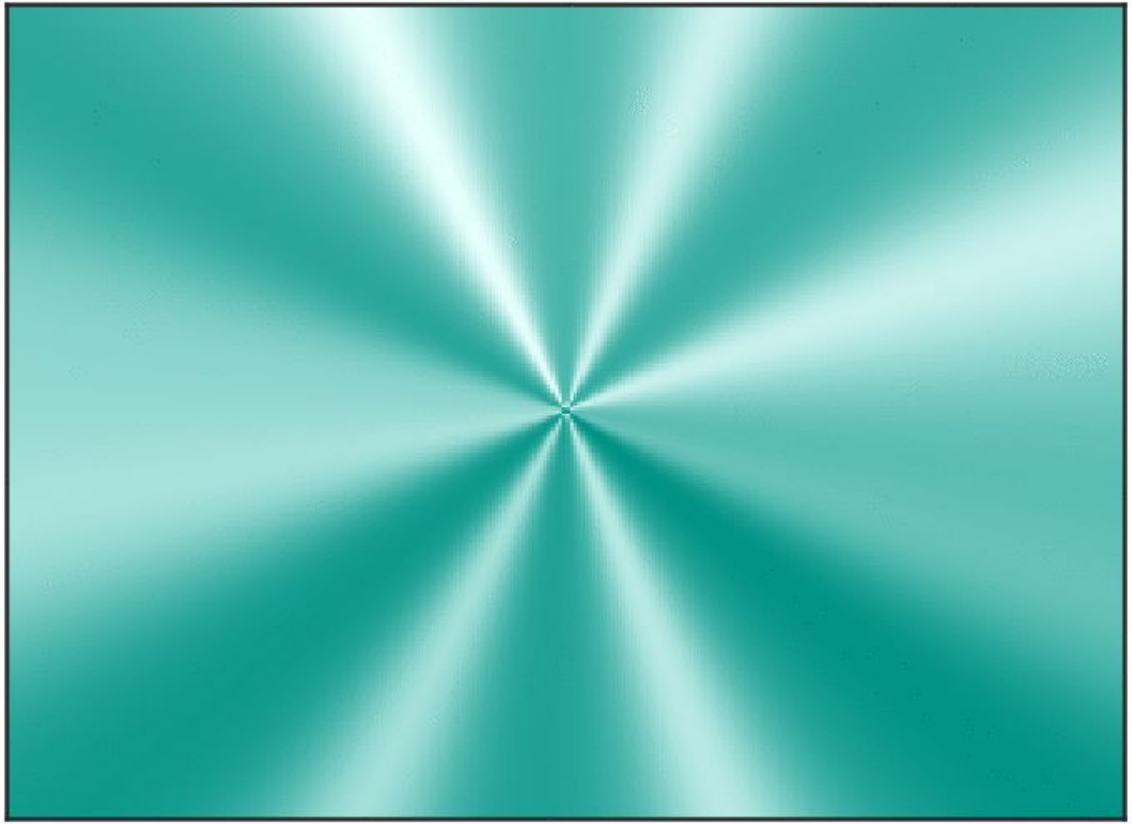
$$x_{t+1} = (1-\eta\lambda)x_t - \eta\nabla f(x_t)$$

$$\begin{cases} \langle \nabla f(x), x\rangle = 0, \ \forall x \\ \nabla f(\alpha x) = \dfrac{1}{\alpha}\nabla f(x), \ \forall x, \ \alpha > 0. \end{cases}$$

Single period of ConvNet on CIFAR-10

weight decay 0.001, learning rate 0.01.





Optimization dynamics of a network with scale-invariant parameters trained with weight decay.

Practical setting. ConvNet on CIFAR-10

- Training non-scale-invariant weights retains the periodic behavior and affects the frequency of periods
- Using momentum does not break the periodic behavior and increases the frequency of periods
- If the number of parameters in the neural network is insufficient to achieve low train loss gradients, phase A never ends (at least in 1000 epochs), resulting in the absence of the periodic behavior