NORMALIZAÇÃO EM LOTE

Zenilton K. G. Patrocínio Jr
zenilton@pucminas.br

Normalização em Lote (*Batch Normalization*)

“Você quer ativações gaussianas unitárias? Apenas faça-as assim.”

[Ioffe & Szegedy, 2015]

Normalização em Lote (*Batch Normalization*)

“Você quer ativações gaussianas unitárias? Apenas faça-as assim.”

[Ioffe & Szegedy, 2015]

Considere um lote de ativações em uma camada qualquer.

Para fazer com que cada dimensão se comporte como gaussiana unitária, aplica-se:

$$\widehat{x}^{(k)} = \frac{x^{(k)} - \mathrm{E}[x^{(k)}]}{\sqrt{\mathrm{Var}[x^{(k)}]}}$$

Normalização em Lote (*Batch Normalization*)

“Você quer ativações gaussianas unitárias? Apenas faça-as assim.”

[Ioffe & Szegedy, 2015]

Considere um lote de ativações em uma camada qualquer.

Para fazer com que cada dimensão se comporte como gaussiana unitária, aplica-se:

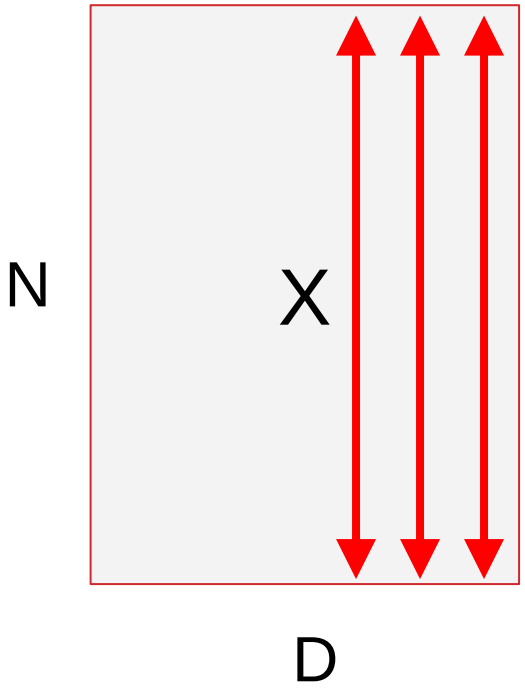
$$\widehat{x}^{(k)} = \frac{x^{(k)} - \mathrm{E}[x^{(k)}]}{\sqrt{\mathrm{Var}[x^{(k)}]}}$$

→ Esta é uma função diferenciável comum ...

Normalização em Lote (*Batch Normalization*)

“Você quer ativações gaussianas unitárias? Apenas faça-as assim.”

[Ioffe & Szegedy, 2015]

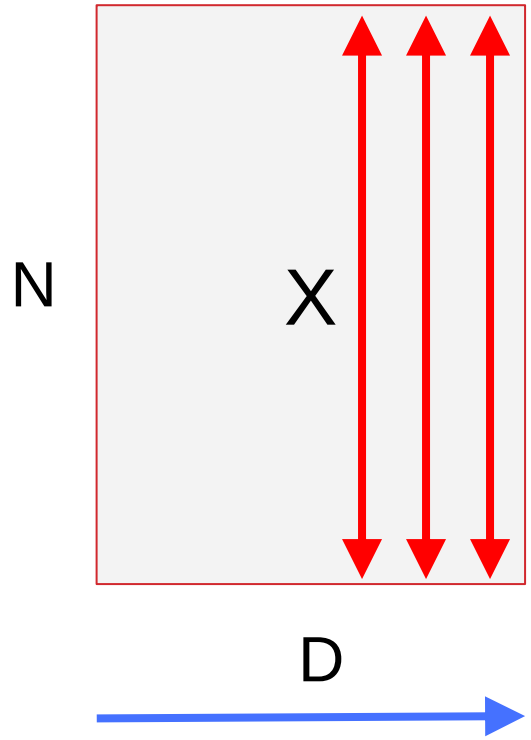


1. Calcular a média e a variância empírica para cada dimensão de forma independente

Normalização em Lote (*Batch Normalization*)

“Você quer ativações gaussianas unitárias? Apenas faça-as assim."

[Ioffe & Szegedy, 2015]

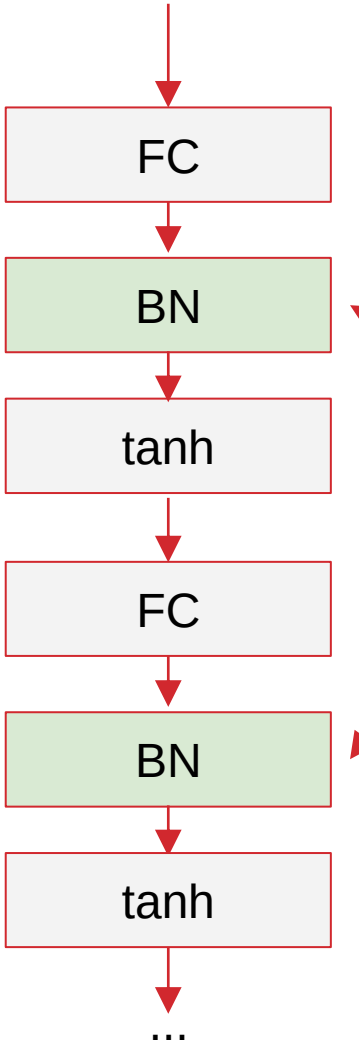


1. Calcular a média e a variância empírica para cada dimensão de forma independente

2. Normalizar

$$\hat{x}^{(k)} = \frac{x^{(k)} - \mathrm{E}[x^{(k)}]}{\sqrt{\mathrm{Var}[x^{(k)}]}}$$

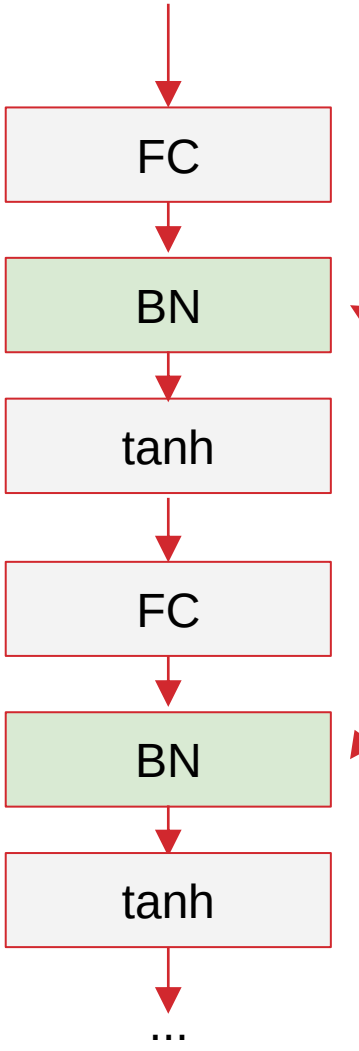
Normalização em Lote (*Batch Normalization*)



Normalmente inserida após as camadas completamente conectadas (ou camadas convolucionais), mas antes da não linearidade.

$$\widehat{x}^{(k)} = \frac{x^{(k)} - \mathrm{E}[x^{(k)}]}{\sqrt{\mathrm{Var}[x^{(k)}]}}$$

Normalização em Lote (*Batch Normalization*)



Normalmente inserida após as camadas completamente conectadas (ou camadas convolucionais), mas antes da não linearidade.

Problema: deseja-se realmente uma entrada gaussiana unitária para a não-linearidade?

$$\hat{x}^{(k)} = \frac{x^{(k)} - \mathrm{E}[x^{(k)}]}{\sqrt{\mathrm{Var}[x^{(k)}]}}$$

Normalização em Lote (*Batch Normalization*)

Solução!

Normalizar:

$$\widehat{x}^{(k)} = \frac{x^{(k)} - \mathrm{E}[x^{(k)}]}{\sqrt{\mathrm{Var}[x^{(k)}]}}$$

Normalização em Lote (*Batch Normalization*)

Solução!

Normalizar:

$$\widehat{x}^{(k)} = \frac{x^{(k)} - \mathrm{E}[x^{(k)}]}{\sqrt{\mathrm{Var}[x^{(k)}]}}$$

E, em seguida, permitir que a rede ajuste a saída para outro intervalo, se quiser:

$$y^{(k)} = \gamma^{(k)}\widehat{x}^{(k)} + \beta^{(k)}$$

em que $\gamma^{(k)}$ e $\beta^{(k)}$ devem ser aprendidos pela rede

Normalização em Lote (*Batch Normalization*)

Solução!

Normalizar:

$$\widehat{x}^{(k)} = \frac{x^{(k)} - \mathrm{E}[x^{(k)}]}{\sqrt{\mathrm{Var}[x^{(k)}]}}$$

E, em seguida, permitir que a rede ajuste a saída para outro intervalo, se quiser:

$$y^{(k)} = \gamma^{(k)}\widehat{x}^{(k)} + \beta^{(k)}$$

em que $\gamma^{(k)}$ e $\beta^{(k)}$ devem ser aprendidos pela rede

Observe que a rede poderia aprender, de modo que:

$$\gamma^{(k)} = \sqrt{\mathrm{Var}[x^{(k)}]}$$

$$\beta^{(k)} = \mathrm{E}[x^{(k)}]$$

e, assim, recuperar o mapeamento de identidade

Normalização em Lote (*Batch Normalization*)

$$\frac{\partial \ell}{\partial \widehat{x}_i} = \frac{\partial \ell}{\partial y_i} \cdot \gamma$$

$$\frac{\partial \ell}{\partial \sigma_{\mathcal{B}}^2} = \sum_{i=1}^{m} \frac{\partial \ell}{\partial \widehat{x}_i} \cdot (x_i - \mu_{\mathcal{B}}) \cdot \frac{-1}{2} (\sigma_{\mathcal{B}}^2 + \epsilon)^{-3/2}$$

$$\frac{\partial \ell}{\partial \mu_{\mathcal{B}}} = \left(\sum_{i=1}^{m} \frac{\partial \ell}{\partial \widehat{x}_i} \cdot \frac{-1}{\sqrt{\sigma_{\mathcal{B}}^2 + \epsilon}} \right) + \frac{\partial \ell}{\partial \sigma_{\mathcal{B}}^2} \cdot \frac{\sum_{i=1}^{m} -2(x_i - \mu_{\mathcal{B}})}{m}$$

$$\frac{\partial \ell}{\partial x_i} = \frac{\partial \ell}{\partial \widehat{x}_i} \cdot \frac{1}{\sqrt{\sigma_{\mathcal{B}}^2 + \epsilon}} + \frac{\partial \ell}{\partial \sigma_{\mathcal{B}}^2} \cdot \frac{2(x_i - \mu_{\mathcal{B}})}{m} + \frac{\partial \ell}{\partial \mu_{\mathcal{B}}} \cdot \frac{1}{m}$$

$$\frac{\partial \ell}{\partial \gamma} = \sum_{i=1}^{m} \frac{\partial \ell}{\partial y_i} \cdot \widehat{x}_i$$

$$\frac{\partial \ell}{\partial \beta} = \sum_{i=1}^{m} \frac{\partial \ell}{\partial y_i}$$

Não necessários diretamente, são subexpressões para os outros gradientes.

Considerar como “backprop” para $\hat{x}$, $\sigma_{\mathcal{B}}^2$, $\mu_{\mathcal{B}}$, que são internos a atualização do minibatch.

Normalização em Lote (*Batch Normalization*)

$$\frac{\partial \ell}{\partial \widehat{x}_i} = \frac{\partial \ell}{\partial y_i} \cdot \gamma$$

$$\frac{\partial \ell}{\partial \sigma_{\mathcal{B}}^2} = \sum_{i=1}^{m} \frac{\partial \ell}{\partial \widehat{x}_i} \cdot (x_i - \mu_{\mathcal{B}}) \cdot \frac{-1}{2}(\sigma_{\mathcal{B}}^2 + \epsilon)^{-3/2}$$

$$\frac{\partial \ell}{\partial \mu_{\mathcal{B}}} = \left(\sum_{i=1}^{m} \frac{\partial \ell}{\partial \widehat{x}_i} \cdot \frac{-1}{\sqrt{\sigma_{\mathcal{B}}^2 + \epsilon}} \right) + \frac{\partial \ell}{\partial \sigma_{\mathcal{B}}^2} \cdot \frac{\sum_{i=1}^{m} -2(x_i - \mu_{\mathcal{B}})}{m}$$

$$\frac{\partial \ell}{\partial x_i} = \frac{\partial \ell}{\partial \widehat{x}_i} \cdot \frac{1}{\sqrt{\sigma_{\mathcal{B}}^2 + \epsilon}} + \frac{\partial \ell}{\partial \sigma_{\mathcal{B}}^2} \cdot \frac{2(x_i - \mu_{\mathcal{B}})}{m} + \frac{\partial \ell}{\partial \mu_{\mathcal{B}}} \cdot \frac{1}{m}$$

Gradientes para propagar para a camada de entrada

$$\frac{\partial \ell}{\partial \gamma} = \sum_{i=1}^{m} \frac{\partial \ell}{\partial y_i} \cdot \widehat{x}_i$$

$$\frac{\partial \ell}{\partial \beta} = \sum_{i=1}^{m} \frac{\partial \ell}{\partial y_i}$$

Normalização em Lote (*Batch Normalization*)

$$\frac{\partial \ell}{\partial \widehat{x}_i} = \frac{\partial \ell}{\partial y_i} \cdot \gamma$$

$$\frac{\partial \ell}{\partial \sigma_{\mathcal{B}}^2} = \sum_{i=1}^{m} \frac{\partial \ell}{\partial \widehat{x}_i} \cdot (x_i - \mu_{\mathcal{B}}) \cdot \frac{-1}{2}(\sigma_{\mathcal{B}}^2 + \epsilon)^{-3/2}$$

$$\frac{\partial \ell}{\partial \mu_{\mathcal{B}}} = \left(\sum_{i=1}^{m} \frac{\partial \ell}{\partial \widehat{x}_i} \cdot \frac{-1}{\sqrt{\sigma_{\mathcal{B}}^2 + \epsilon}} \right) + \frac{\partial \ell}{\partial \sigma_{\mathcal{B}}^2} \cdot \frac{\sum_{i=1}^{m} -2(x_i - \mu_{\mathcal{B}})}{m}$$

$$\frac{\partial \ell}{\partial x_i} = \frac{\partial \ell}{\partial \widehat{x}_i} \cdot \frac{1}{\sqrt{\sigma_{\mathcal{B}}^2 + \epsilon}} + \frac{\partial \ell}{\partial \sigma_{\mathcal{B}}^2} \cdot \frac{2(x_i - \mu_{\mathcal{B}})}{m} + \frac{\partial \ell}{\partial \mu_{\mathcal{B}}} \cdot \frac{1}{m}$$

$$\frac{\partial \ell}{\partial \gamma} = \sum_{i=1}^{m} \frac{\partial \ell}{\partial y_i} \cdot \widehat{x}_i$$

$$\frac{\partial \ell}{\partial \beta} = \sum_{i=1}^{m} \frac{\partial \ell}{\partial y_i}$$

Gradientes para os parâmetros γ e β.

Normalização em Lote (*Batch Normalization*)

Input: Values of x over a mini-batch: $\mathcal{B} = \{x_{1...m}\}$;
Parameters to be learned: γ, β

Output: $\{y_i = \mathrm{BN}_{\gamma,\beta}(x_i)\}$

$$\mu_{\mathcal{B}} \leftarrow \frac{1}{m}\sum_{i=1}^{m} x_i \qquad \text{// mini-batch mean}$$

$$\sigma^2_{\mathcal{B}} \leftarrow \frac{1}{m}\sum_{i=1}^{m}(x_i - \mu_{\mathcal{B}})^2 \qquad \text{// mini-batch variance}$$

$$\widehat{x}_i \leftarrow \frac{x_i - \mu_{\mathcal{B}}}{\sqrt{\sigma^2_{\mathcal{B}} + \epsilon}} \qquad \text{// normalize}$$

$$y_i \leftarrow \gamma\widehat{x}_i + \beta \equiv \mathrm{BN}_{\gamma,\beta}(x_i) \qquad \text{// scale and shift}$$

- Melhora o fluxo gradiente através da rede

Normalização em Lote (*Batch Normalization*)

Input: Values of x over a mini-batch: $\mathcal{B} = \{x_{1...m}\}$;
Parameters to be learned: γ, β

Output: $\{y_i = \mathrm{BN}_{\gamma,\beta}(x_i)\}$

$$\mu_{\mathcal{B}} \leftarrow \frac{1}{m}\sum_{i=1}^{m} x_i \qquad \text{// mini-batch mean}$$

$$\sigma^2_{\mathcal{B}} \leftarrow \frac{1}{m}\sum_{i=1}^{m}(x_i - \mu_{\mathcal{B}})^2 \qquad \text{// mini-batch variance}$$

$$\widehat{x}_i \leftarrow \frac{x_i - \mu_{\mathcal{B}}}{\sqrt{\sigma^2_{\mathcal{B}} + \epsilon}} \qquad \text{// normalize}$$

$$y_i \leftarrow \gamma\widehat{x}_i + \beta \equiv \mathrm{BN}_{\gamma,\beta}(x_i) \qquad \text{// scale and shift}$$

- Melhora o fluxo gradiente através da rede
- Permite taxas de aprendizagem mais altas

Normalização em Lote (*Batch Normalization*)

Input: Values of x over a mini-batch: $\mathcal{B} = \{x_{1...m}\}$;
Parameters to be learned: γ, β

Output: $\{y_i = \mathrm{BN}_{\gamma,\beta}(x_i)\}$

$$\mu_{\mathcal{B}} \leftarrow \frac{1}{m}\sum_{i=1}^{m} x_i \qquad \text{// mini-batch mean}$$

$$\sigma^2_{\mathcal{B}} \leftarrow \frac{1}{m}\sum_{i=1}^{m}(x_i - \mu_{\mathcal{B}})^2 \qquad \text{// mini-batch variance}$$

$$\widehat{x}_i \leftarrow \frac{x_i - \mu_{\mathcal{B}}}{\sqrt{\sigma^2_{\mathcal{B}} + \epsilon}} \qquad \text{// normalize}$$

$$y_i \leftarrow \gamma\widehat{x}_i + \beta \equiv \mathrm{BN}_{\gamma,\beta}(x_i) \qquad \text{// scale and shift}$$

- Melhora o fluxo gradiente através da rede
- Permite taxas de aprendizagem mais altas
- Reduz a forte dependência da inicialização

Normalização em Lote (*Batch Normalization*)

Input: Values of x over a mini-batch: $\mathcal{B} = \{x_{1...m}\}$;
Parameters to be learned: γ, β

Output: $\{y_i = \mathrm{BN}_{\gamma,\beta}(x_i)\}$

$$\mu_{\mathcal{B}} \leftarrow \frac{1}{m}\sum_{i=1}^{m} x_i \qquad \text{// mini-batch mean}$$

$$\sigma^2_{\mathcal{B}} \leftarrow \frac{1}{m}\sum_{i=1}^{m}(x_i - \mu_{\mathcal{B}})^2 \qquad \text{// mini-batch variance}$$

$$\widehat{x}_i \leftarrow \frac{x_i - \mu_{\mathcal{B}}}{\sqrt{\sigma^2_{\mathcal{B}} + \epsilon}} \qquad \text{// normalize}$$

$$y_i \leftarrow \gamma\widehat{x}_i + \beta \equiv \mathrm{BN}_{\gamma,\beta}(x_i) \qquad \text{// scale and shift}$$

- Melhora o fluxo gradiente através da rede
- Permite taxas de aprendizagem mais altas
- Reduz a forte dependência da inicialização
- Atua como uma forma de regularização diferente, reduzindo talvez a necessidade de *dropout*

Normalização em Lote (*Batch Normalization*)

Input: Values of x over a mini-batch: $\mathcal{B} = \{x_{1...m}\}$;
Parameters to be learned: γ, β

Output: $\{y_i = \mathrm{BN}_{\gamma,\beta}(x_i)\}$

$$\mu_{\mathcal{B}} \leftarrow \frac{1}{m}\sum_{i=1}^{m} x_i \qquad \text{// mini-batch mean}$$

$$\sigma^2_{\mathcal{B}} \leftarrow \frac{1}{m}\sum_{i=1}^{m}(x_i - \mu_{\mathcal{B}})^2 \qquad \text{// mini-batch variance}$$

$$\widehat{x}_i \leftarrow \frac{x_i - \mu_{\mathcal{B}}}{\sqrt{\sigma^2_{\mathcal{B}} + \epsilon}} \qquad \text{// normalize}$$

$$y_i \leftarrow \gamma\widehat{x}_i + \beta \equiv \mathrm{BN}_{\gamma,\beta}(x_i) \qquad \text{// scale and shift}$$

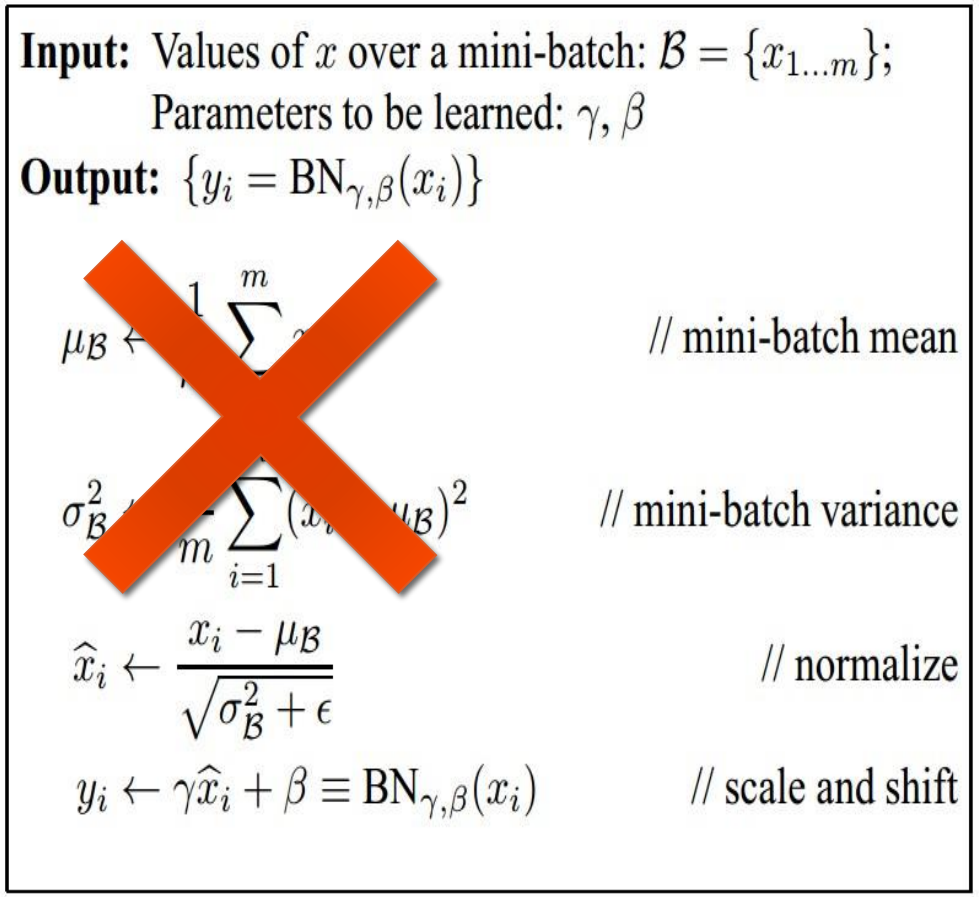
- Melhora o fluxo gradiente através da rede
- Permite taxas de aprendizagem mais altas
- Reduz a forte dependência da inicialização
- Atua como uma forma de regularização diferente, reduzindo talvez a necessidade de *dropout*

Desnormalização!!

Aprende-se γ e β (mesmas dims que μ e σ^2)

Permite (se necessário) aprender o mapeamento identidade!

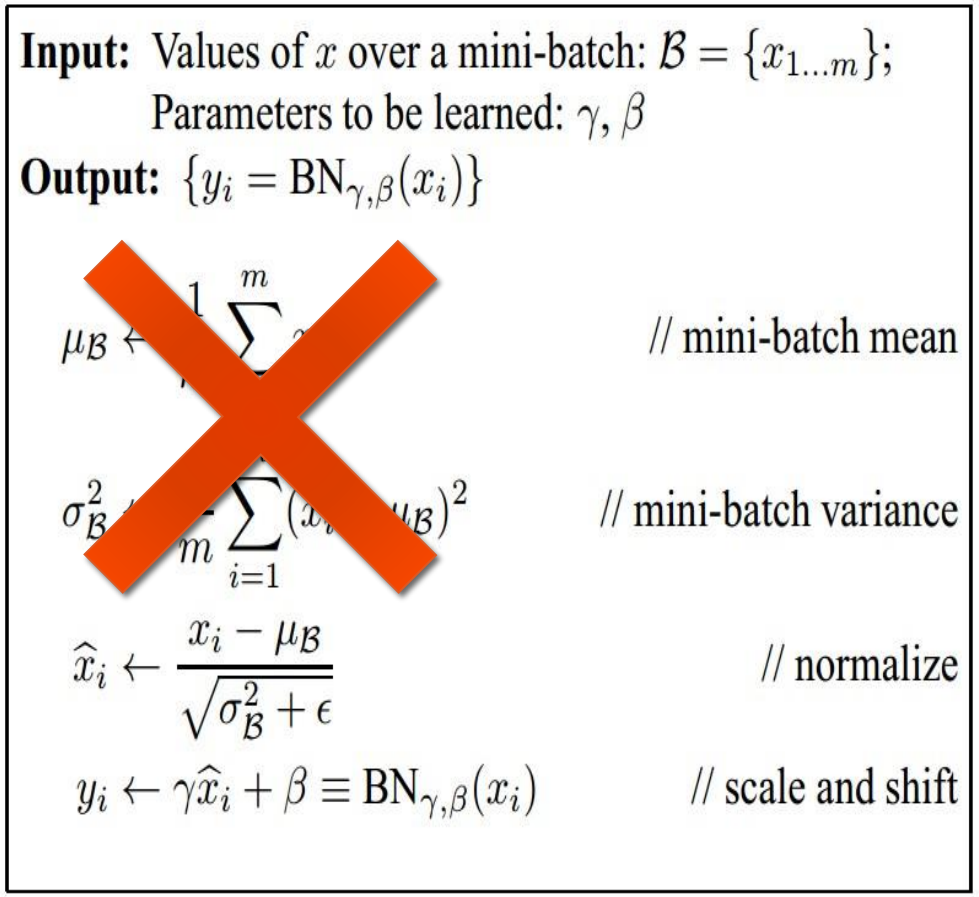
Normalização em Lote (*Batch Normalization*)



Nota: no momento do **teste**, a camada de normalização em lote funciona diferente**:**

- A **média e a variância não são calculadas** com base no lote

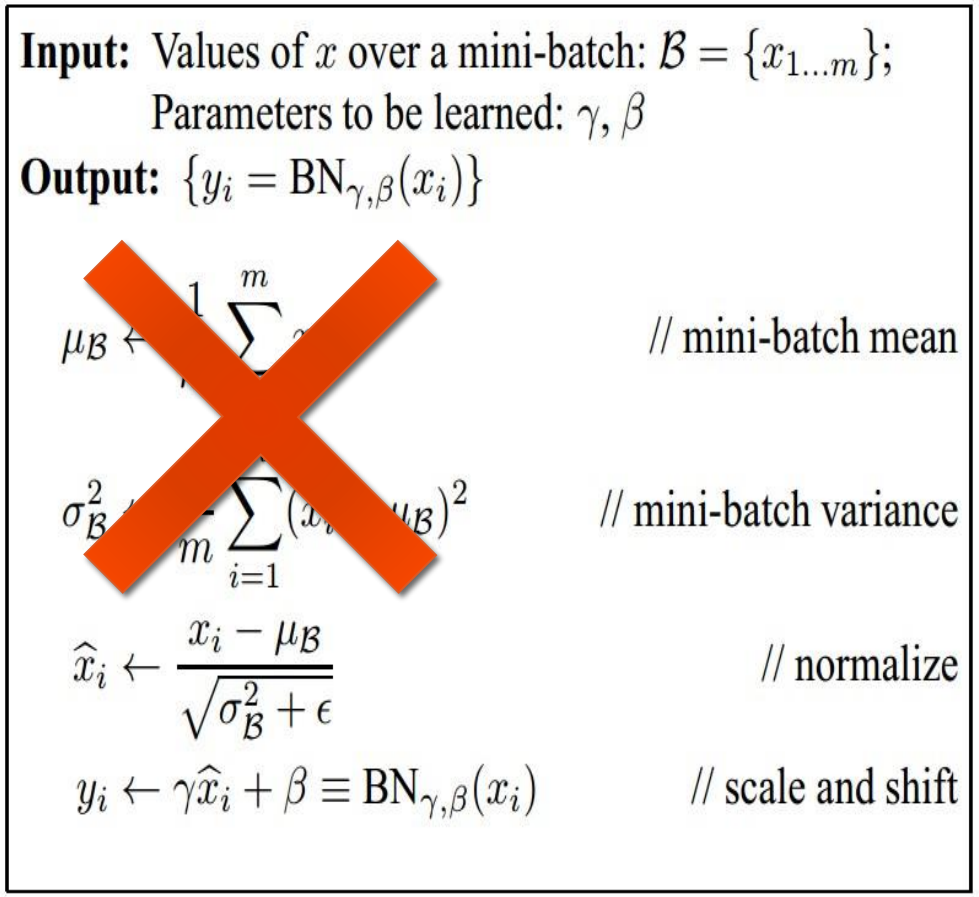
Normalização em Lote (*Batch Normalization*)



Nota: no momento do **teste**, a camada de normalização em lote funciona diferente**:**

- A **média e a variância não são calculadas** com base no lote
- Em vez disso, **utiliza-se um único par fixo de média e variância** empírica de ativações obtido durante o treinamento

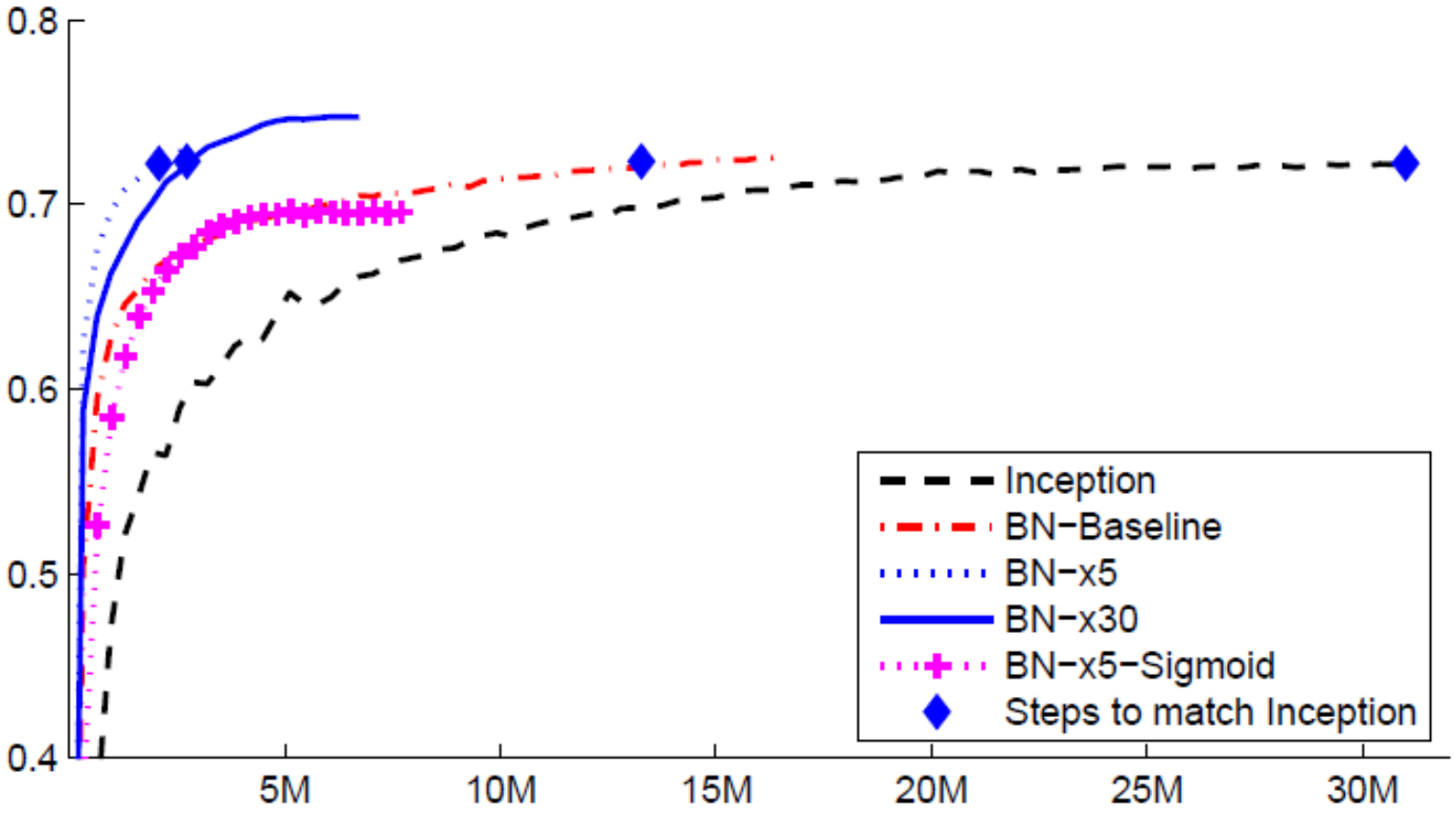
Normalização em Lote (*Batch Normalization*)



Nota: no momento do **teste**, a camada de normalização em lote funciona diferente**:**

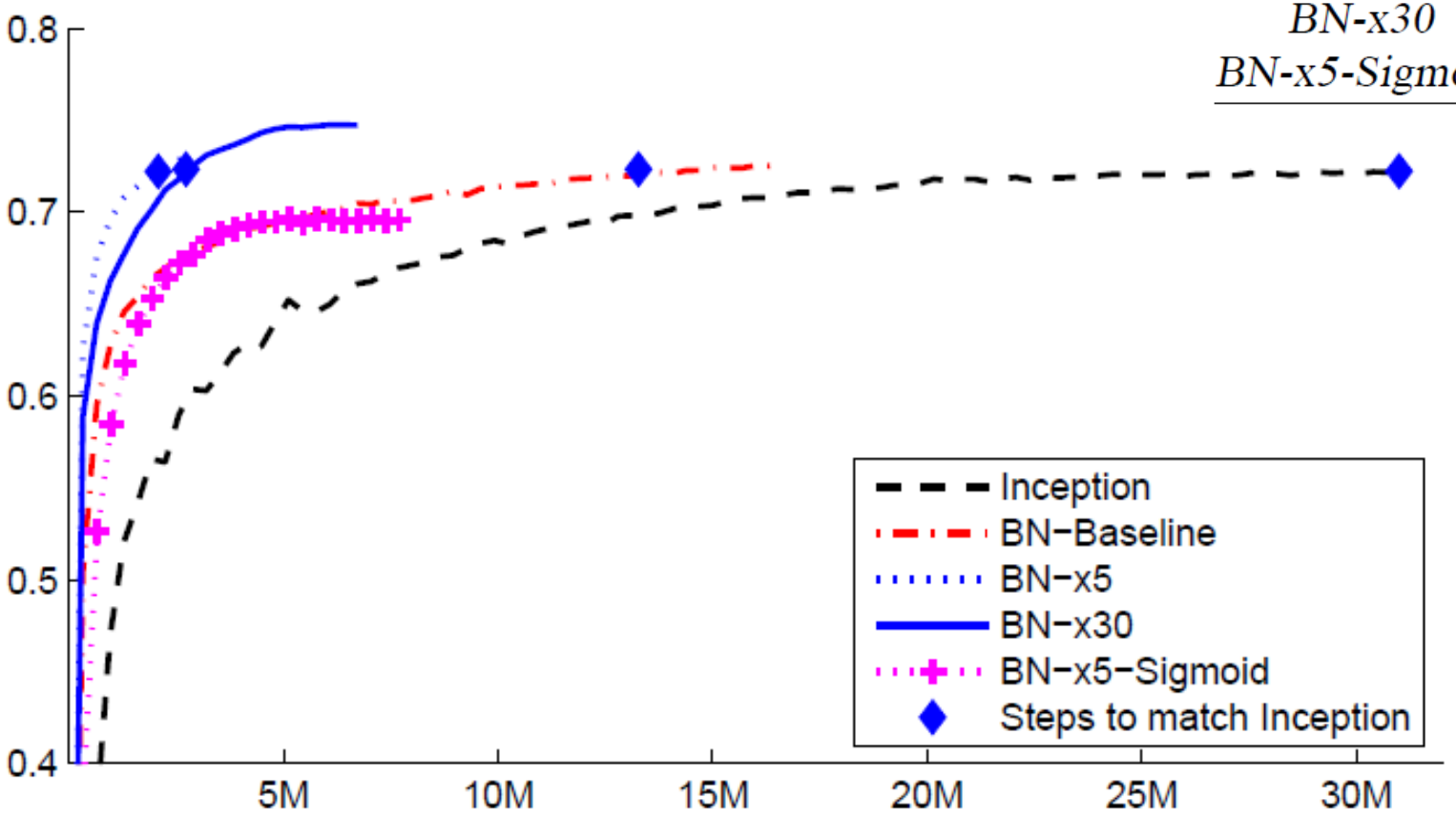
- A **média e a variância não são calculadas** com base no lote
- Em vez disso, **utiliza-se um único par fixo de média e variância** empírica de ativações obtido durante o treinamento
- Por exemplo, pode-se estimar o par durante o treinamento por meio de médias móveis

Normalização em Lote (*Batch Normalization*)

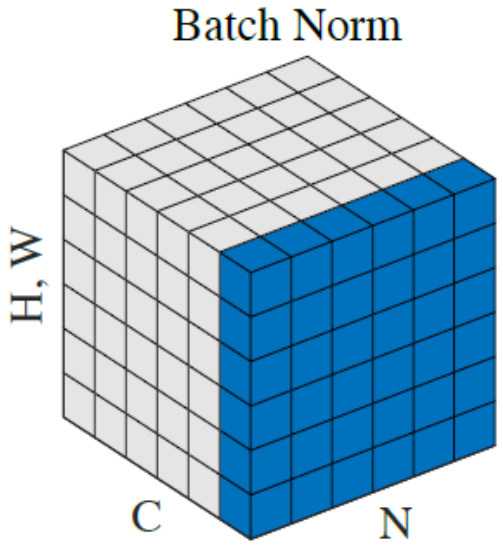


Normalização em Lote (*Batch Normalization*)

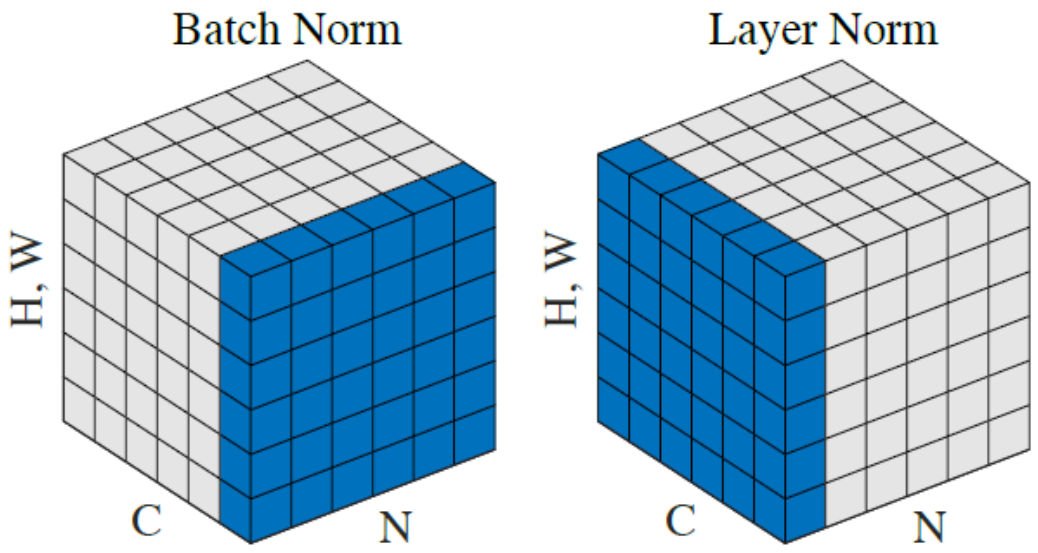
Model	Steps to 72.2%	Max accuracy
Inception	$31.0 \cdot 10^6$	72.2%
BN-Baseline	$13.3 \cdot 10^6$	72.7%
BN-x5	$2.1 \cdot 10^6$	73.0%
BN-x30	$2.7 \cdot 10^6$	74.8%
BN-x5-Sigmoid		69.8%



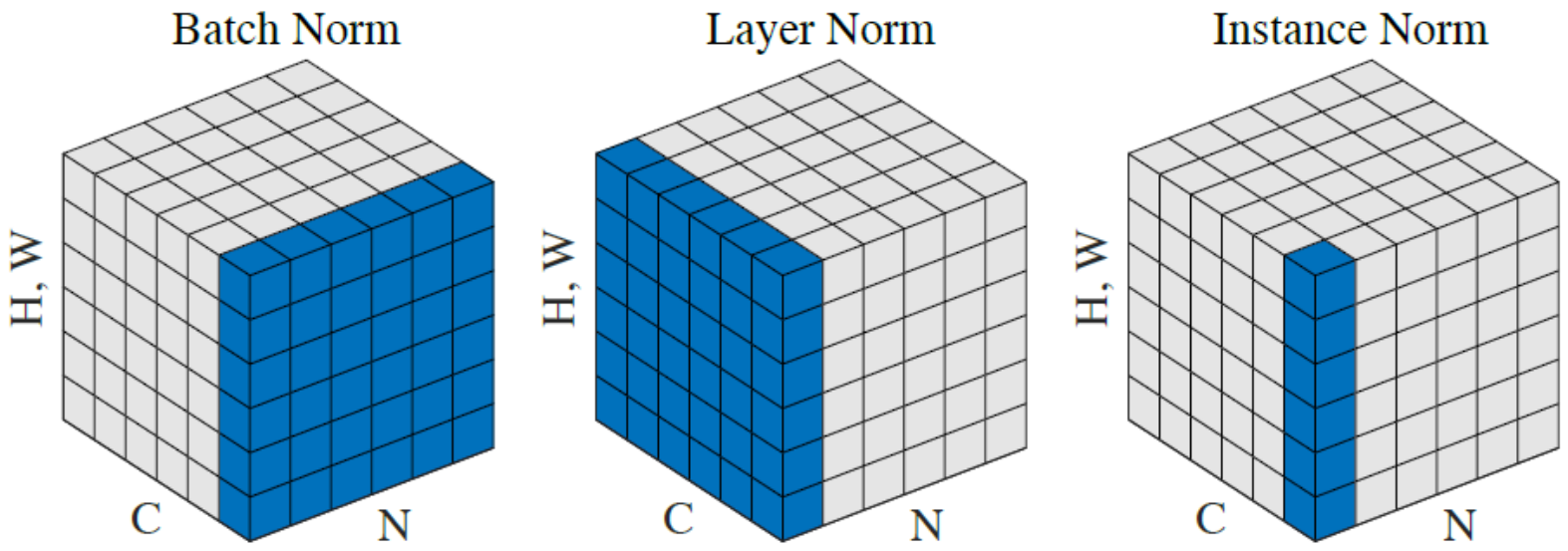
Normalização – Variações



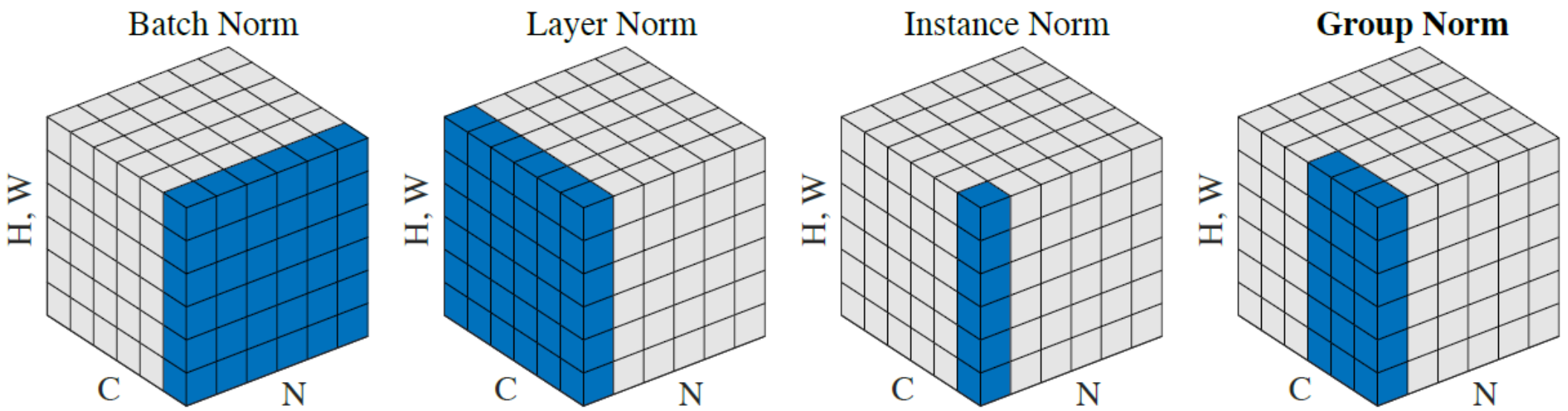
Normalização – Variações



Normalização – Variações

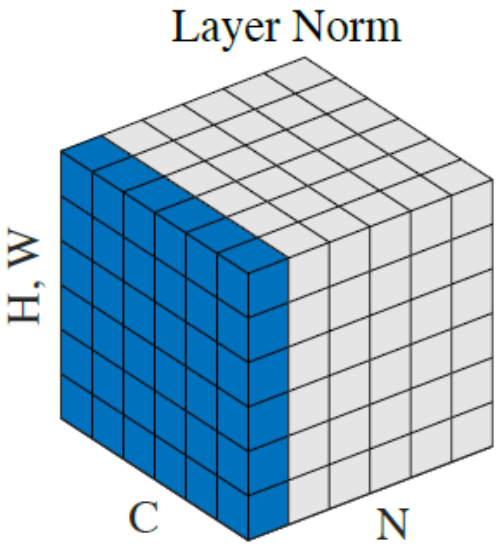


Normalização – Variações



Layer Normalization

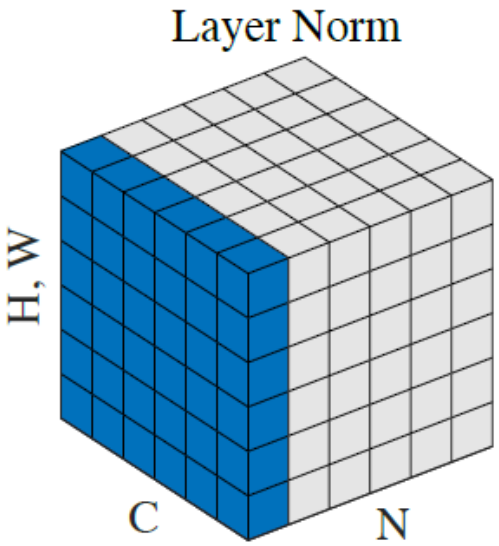
[Ba et al., 2016]



- Insensível ao tamanho do lote

Layer Normalization

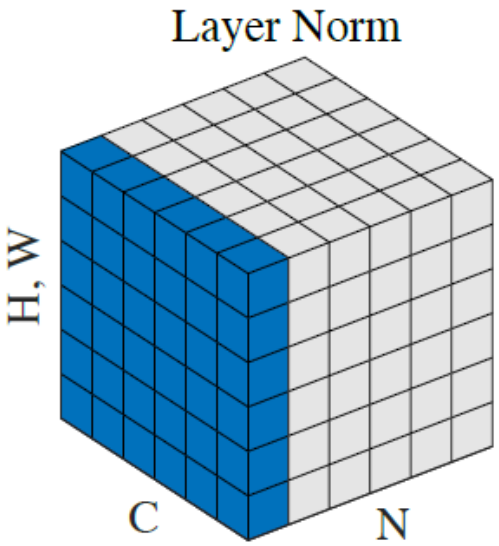
[Ba et al., 2016]



- Insensível ao tamanho do lote
- Ideal para aplicar no processamento de sequências

Layer Normalization

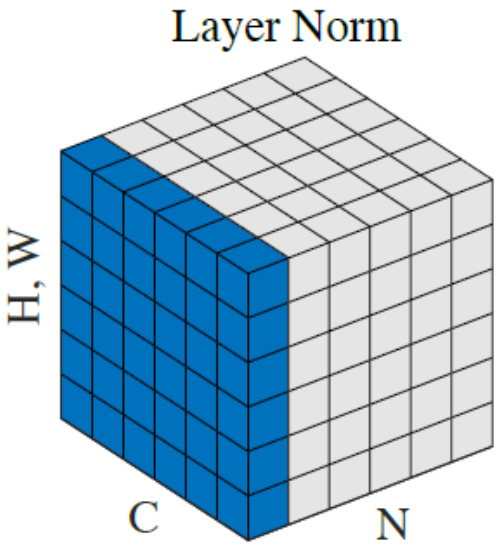
[Ba et al., 2016]



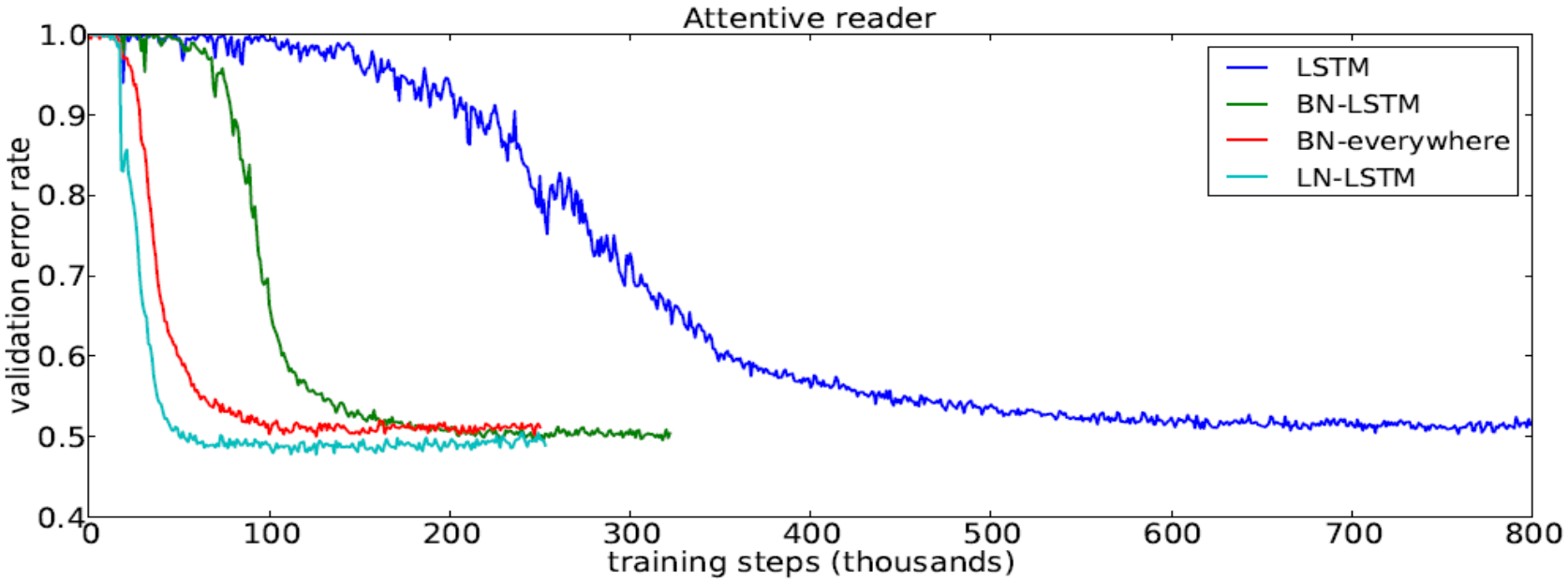
- Insensível ao tamanho do lote
- Ideal para aplicar no processamento de sequências
- Redes recorrentes e *transformers*

Layer Normalization

[Ba et al., 2016]

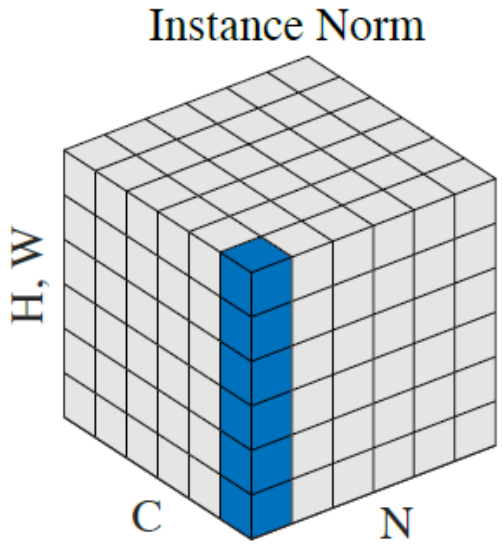


- Insensível ao tamanho do lote
- Ideal para aplicar no processamento de sequências
- Redes recorrentes e *transformers*



Instance Normalization

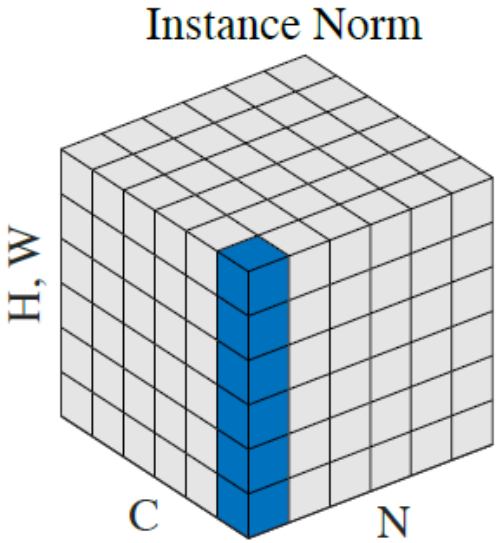
[Ulyanov et al., 2017]



Transferência de Estilo

Instance Normalization

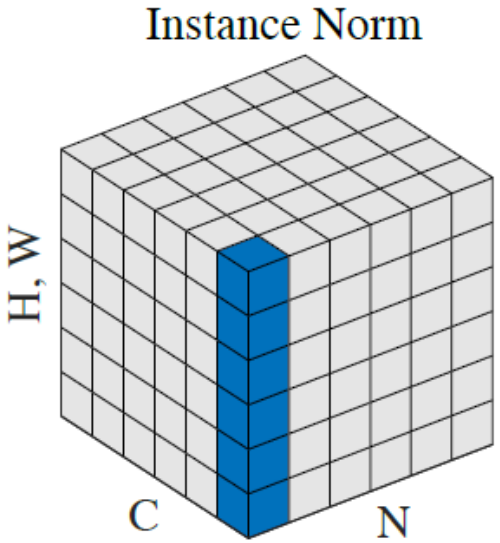
[Ulyanov et al., 2017]



Transferência de Estilo

Instance Normalization

[Ulyanov et al., 2017]



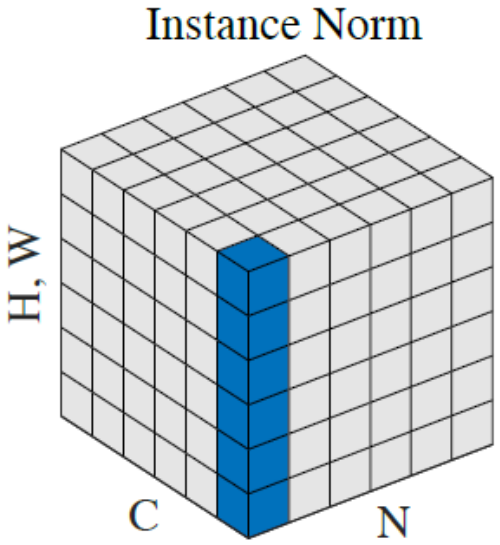
Transferência de Estilo

Imagem

Estilo

Instance Normalization

[Ulyanov et al., 2017]

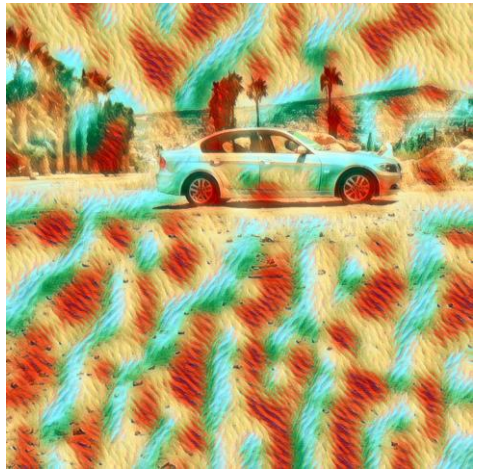


Transferência de Estilo

Imagem

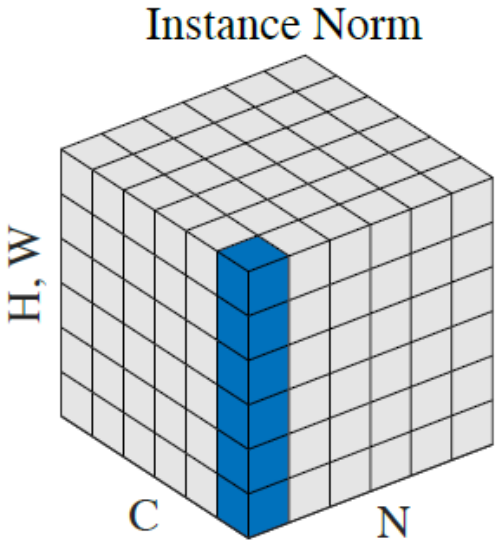
Estilo

Resultado com BN



Instance Normalization

[Ulyanov et al., 2017]

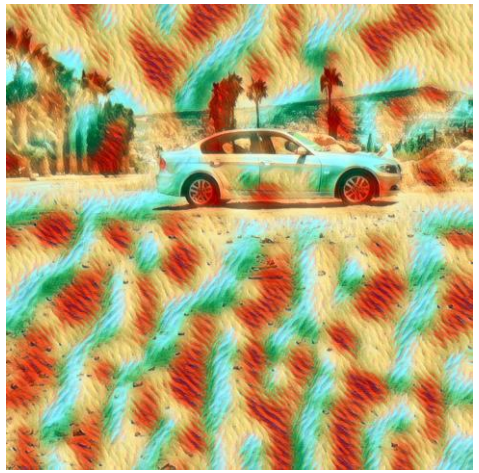


Transferência de Estilo

Imagem

Estilo

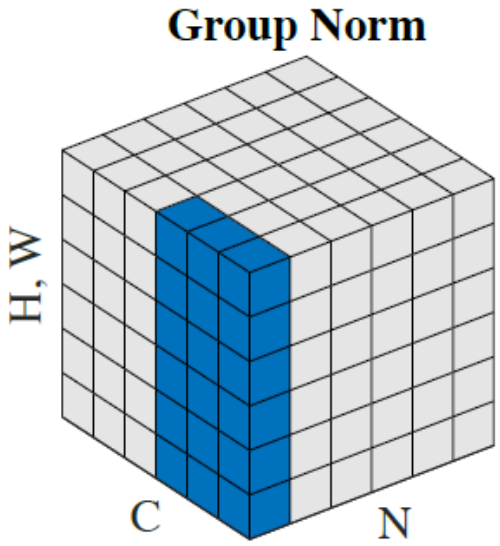
Resultado com BN



Resultado com IN

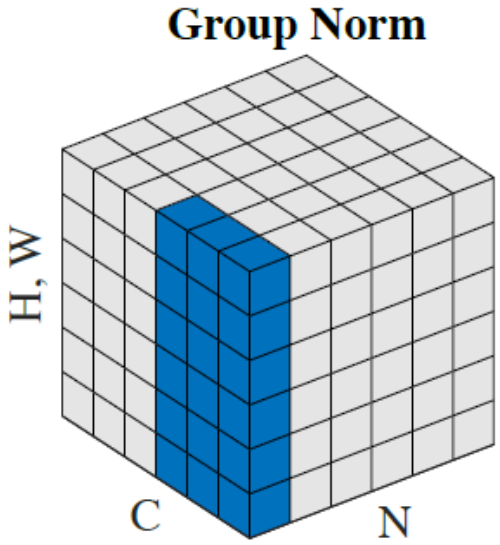
Group Normalization

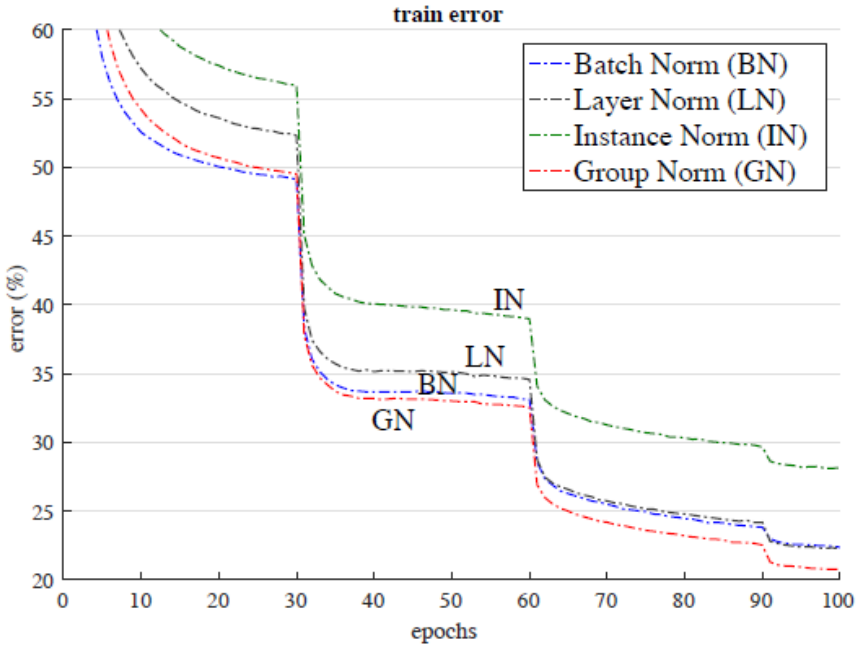
[Wu & He, 2018]

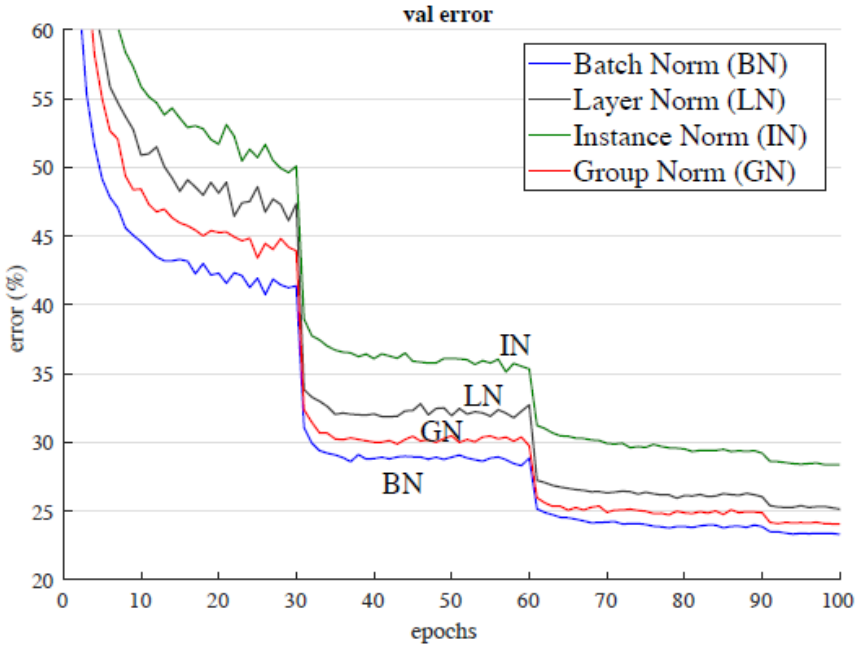


Group Normalization

[Wu & He, 2018]



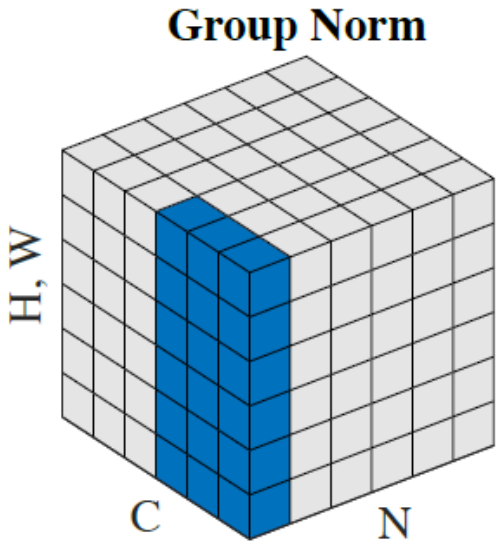


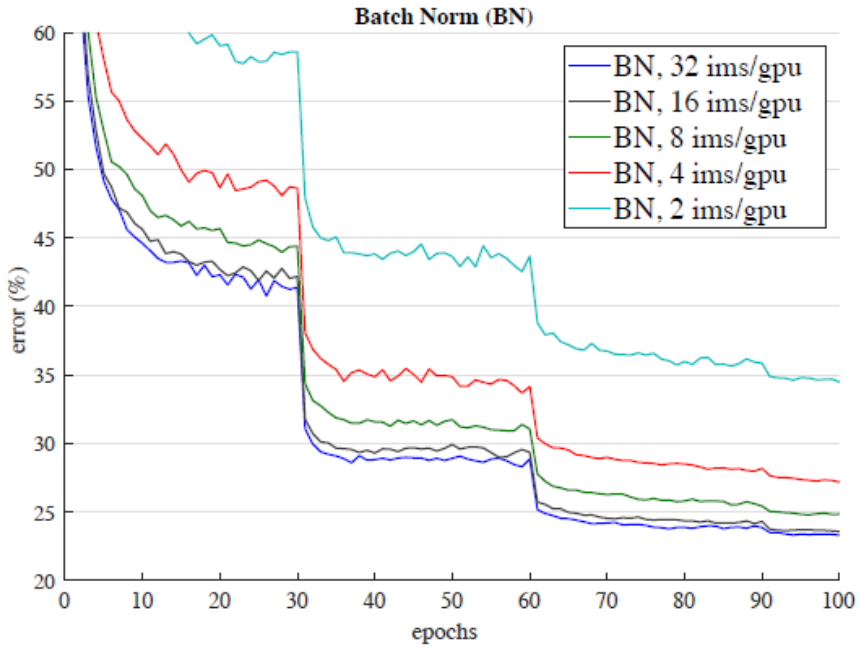


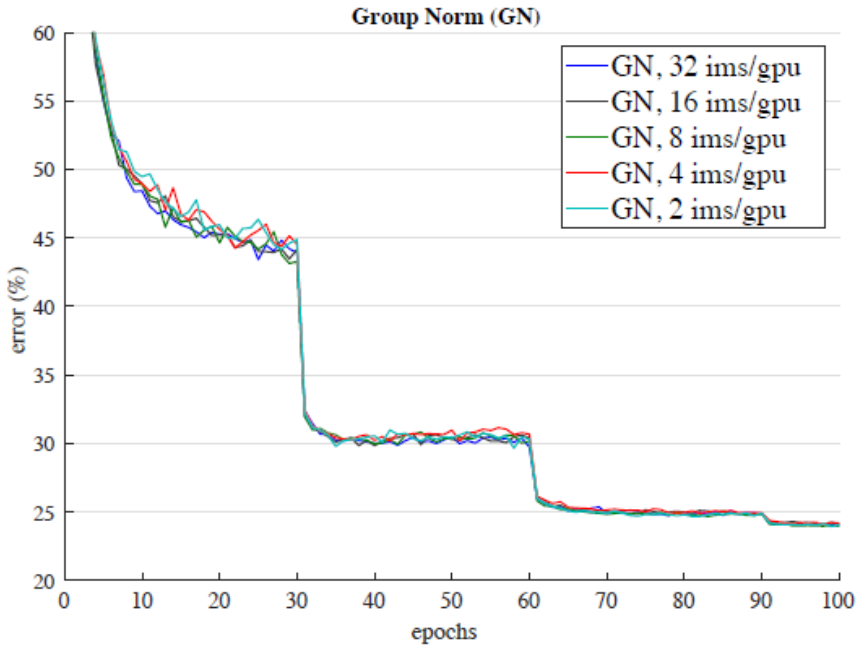
	BN	LN	IN	GN
val error	**23.6**	25.3	28.4	24.1
△ (*vs.* BN)	-	*1.7*	*4.8*	***0.5***

Group Normalization

[Wu & He, 2018]







batch size	32	16	8	4	2
BN	**23.6**	**23.7**	24.8	27.3	34.7
GN	24.1	24.2	**24.0**	**24.2**	**24.1**
△	*0.5*	*0.5*	*-0.8*	*-3.1*	*-10.6*