Batch-Normalization

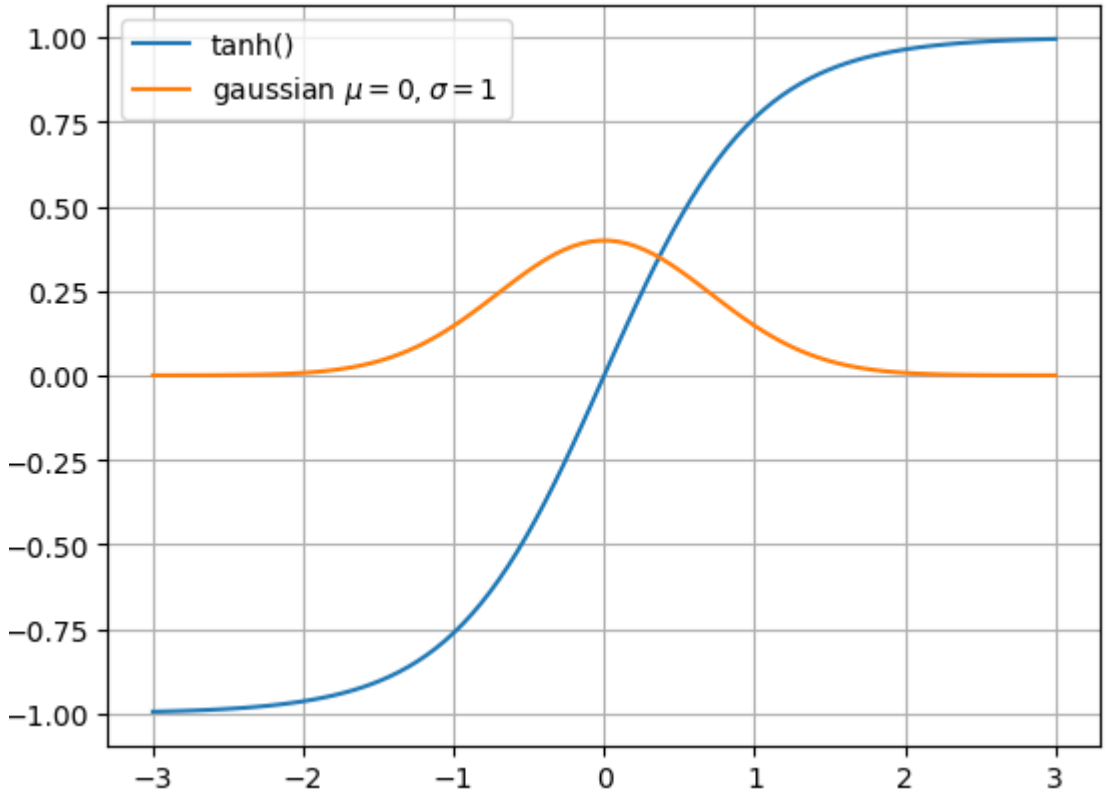
Piya Amara Palamure

03-29-2024

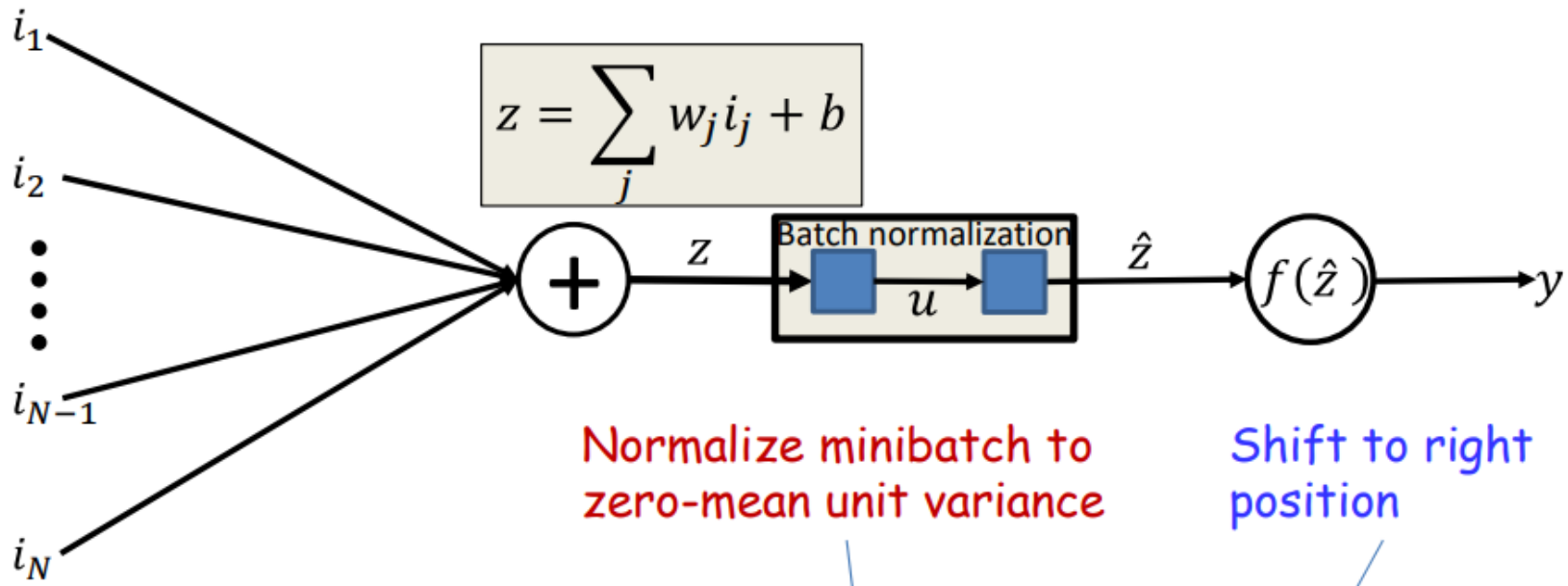
Batch Normalization

- Note: Dividing the weights of the neural network (NN) by the square root of fan-in is a common practice in weight initialization methods.
- Batch normalization is a technique that greatly aids in training very deep neural networks reliably. It operates by taking pre-activation values and normalizing them across mini-batches to have a unit variance.
- This normalization ensures that the pre-activation values neither become too large nor too small, keeping them within the active region. This is crucial for ensuring that the derivatives of the loss with respect to the weights remain non-zero or within reasonable bounds during backpropagation.
- When employing batch normalization, the use of bias weights in the hidden layers becomes unnecessary, as the parameter β assists in shifting the distribution appropriately.
- Batch normalization couples the inputs within the batch through mean and standard deviation, providing an additional regularization effect.

- For example, when using the tanh() activation function, it is beneficial to maintain pre-activation values with a zero mean and unit variance. This helps to prevent issues such as vanishing or exploding gradients during the backpropagation process.



Batch normalization: Training



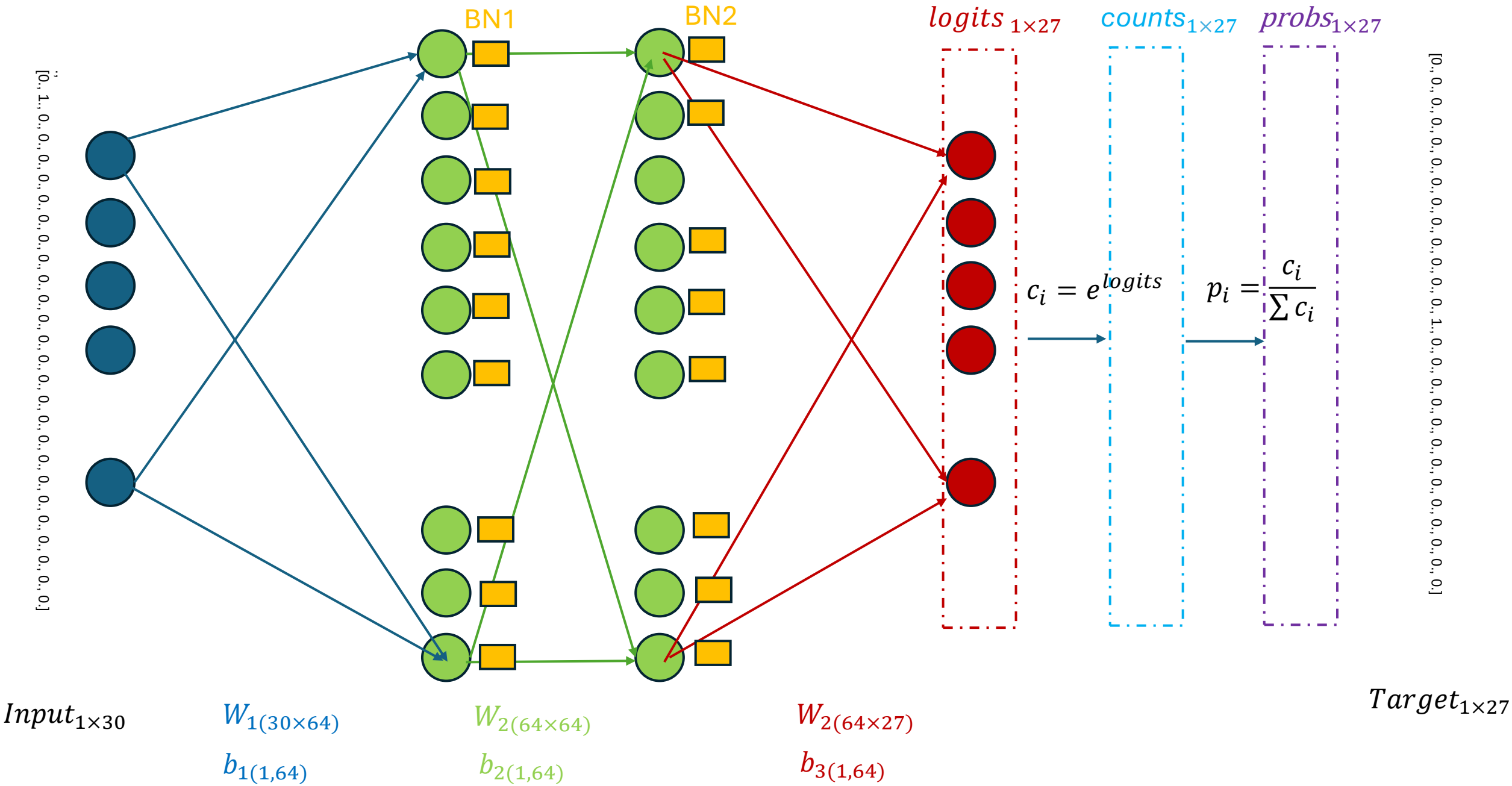
$$\mu_B = \frac{1}{B}\sum_{i=1}^{B} z_i$$

$$\sigma_B^2 = \frac{1}{B}\sum_{i=1}^{B} (z_i - \mu_B)^2$$

$$u_i = \frac{z_i - \mu_B}{\sqrt{\sigma_B^2 + \epsilon}}$$

$$\hat{z}_i = \gamma u_i + \beta$$

- BN aggregates the statistics over a minibatch and normalizes the batch by them
- Normalized instances are "shifted" to a *unit-specific* location


BN1
BN2
$logits_{1\times27}$
$counts_{1\times27}$
$probs_{1\times27}$
$c_i = e^{logits}$
$p_i = \frac{c_i}{\sum c_i}$
[0., 1., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.]
[0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 1., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.]
$Input_{1\times30}$
$W_{1(30\times64)}$
$b_{1(1,64)}$
$W_{2(64\times64)}$
$b_{2(1,64)}$
$W_{2(64\times27)}$
$b_{3(1,64)}$
$Target_{1\times27}$