

$X_{train} = \text{Matrix}(\text{Batch_size}, 28 \times 28)$
 $Y_{train} = \text{Matrix}(1 \times \text{batch_size})$

h hidden_layer_size (100)

$W_1 = \text{Matrix}(28 \times 28, h)$
 $b_1 = \text{Matrix}(1, h)$

$W_2 = \text{Matrix}(h, k)$
 $b_2 = \text{Matrix}(1, k)$ } $k=10$ [Anzahl Klassen]

step_size float
 reg float

Learning

for 1 to epochs (100):

forward pass
 $d = \text{mul}(X_{train}, W_1)$ $d_{ik} = \sum_{j=1}^m x_{train,ij} \cdot W_{1,jk}$ Matrix multiplication
 $h_1 = d + b_1$ $h_{1,k} = d_{ik} + b_{1,k}$ Add Matrix & Vector
 $h_{1relu} = \max(0, h_1)$ element wise $\max(0, x)$

$d_2 = \text{mul}(h_{1relu}, W_2)$

$\text{score} = d_2 + b_2$

gradient backpropagation
 $\text{score} \Rightarrow$ alle Werte die ≤ 0 auf 0 setzen
 $\text{scores} \Rightarrow$ den Wert der richtigen Klasse um 1 verringern

$\text{scores} \Rightarrow$ jeden Wert durch batch_size dividieren

$dW_2 = \text{mul}(h_1^T, \text{scores})$

$db_2 = \text{summe scores axis}=0$, wenn batch_size=1 $db_2 = \text{scores}$

$d\text{hidden} = \text{mul}(\text{scores}, W_2^T)$
 $d\text{hidden} \Rightarrow$ alle Werte ≤ 0 auf 0 setzen

$dW_1 = \text{mul}(X_{train}^T, d\text{hidden})$

$db_1 = \text{summe } d\text{hidden axis}=0$, wenn batch_size=1 $db_1 = d\text{hidden}$

$dW_2 += \text{reg} * W_2$
 $dW_1 += \text{reg} * W_1$

$W_1 += -\text{step_size} * dW_1$
 $b_1 += -\text{step_size} * db_1$
 $W_2 += -\text{step_size} * dW_2$
 $b_2 += -\text{step_size} * db_2$