Comparison between Adam variants

Here we compare different variations and implementations of Adam:

1. Adam - The default Adam implementation included in Tensorflow 2.x.
2. my Adam - My custom implementation of the same Adam algorithm using the Tensorflow API.
3. cAdam - My proposed variant "centered Adam" using m_t as the estimate of $\mathbb{E}[g_t]$.
4. cAdam hat - My proposed variant "centered Adam" using $\hat{m}_t$ as the estimate of $\mathbb{E}[g_t]$.

best and worst values regarding training time

parameter name	**best values**				
training time	53.155	53.89	56.399	56.989	57.293
neurons per layer	(50, 10)	(50, 10)	(50, 10)	(50, 10)	(50, 10)
activation functions	ReLU	ReLU	ReLU	ReLU	ReLU
last activation function	sigmoid	softmax	softmax	sigmoid	sigmoid
loss function	cat-cross	cat-cross	cat-cross	cat-cross	cat-cross
training data percentage	1.0	1.0	1.0	1.0	1.0
number of epochs	50	50	50	50	50
batch size	100	100	100	100	100
optimizer	Adam	Adam	my Adam	cAdam	my Adam
learning rate	0.001	0.001	0.001	0.001	0.001
ε	10^{-7}	10^{-7}	10^{-7}	10^{-7}	10^{-7}

Table 0.1: best settings regarding *training time* for the MNIST dataset

parameter name	**worst values**				
training time	56.989	57.293	58.796	59.013	59.761
neurons per layer	(50, 10)	(50, 10)	(50, 10)	(50, 10)	(50, 10)
activation functions	ReLU	ReLU	ReLU	ReLU	ReLU
last activation function	sigmoid	sigmoid	softmax	softmax	sigmoid
loss function	cat-cross	cat-cross	cat-cross	cat-cross	cat-cross
training data percentage	1.0	1.0	1.0	1.0	1.0
number of epochs	50	50	50	50	50
batch size	100	100	100	100	100
optimizer	cAdam	my Adam	cAdam	c adam hat	c adam hat
learning rate	0.001	0.001	0.001	0.001	0.001
ε	10^{-7}	10^{-7}	10^{-7}	10^{-7}	10^{-7}

Table 0.2: worst settings regarding *training time* for the MNIST dataset

parameter influence on training time

parameter name	**parameter values**				**win ratios in %**				**avg. differences in s**				**best value**
last activation	softmax		sigmoid		50.0		50.0		0.635		0.41		sigmoid
optimizer	Adam	my Adam	cAdam	cAdam hat	100.0	0	0	0	0	3.324	4.37	5.864	Adam

Table 0.3: parameter influence regarding *training time* for the MNIST dataset

best and worst values regarding accuracy

parameter name	best values			
test accuracy	0.96826	0.96692	0.96828	0.9686
final validation accuracy	0.96788	0.96418	0.96795	0.96622
training time	57.293	56.399	53.155	53.89
neurons per layer	(50, 10)	(50, 10)	(50, 10)	(50, 10)
activation functions	ReLU	ReLU	ReLU	ReLU
last activation function	sigmoid	softmax	sigmoid	softmax
loss function	cat-cross	cat-cross	cat-cross	cat-cross
training data percentage	1.0	1.0	1.0	1.0
number of epochs	50	50	50	50
batch size	100	100	100	100
optimizer	my Adam	my Adam	Adam	Adam
learning rate	0.001	0.001	0.001	0.001
ε	10^{-7}	10^{-7}	10^{-7}	10^{-7}

Table 0.4: best settings regarding *test accuracy* for the MNIST dataset

parameter name	worst values				
test accuracy	0.96826	0.96788	0.96692	0.4184	0.18496
final validation accuracy	0.96788	0.96762	0.96418	0.41733	0.18395
training time	57.293	58.796	56.399	59.013	59.761
neurons per layer	(50, 10)	(50, 10)	(50, 10)	(50, 10)	(50, 10)
activation functions	ReLU	ReLU	ReLU	ReLU	ReLU
last activation function	sigmoid	softmax	softmax	softmax	sigmoid
loss function	cat-cross	cat-cross	cat-cross	cat-cross	cat-cross
training data percentage	1.0	1.0	1.0	1.0	1.0
number of epochs	50	50	50	50	50
batch size	100	100	100	100	100
optimizer	my Adam	cAdam	my Adam	c adam hat	c adam hat
learning rate	0.001	0.001	0.001	0.001	0.001
ε	10^{-7}	10^{-7}	10^{-7}	10^{-7}	10^{-7}

Table 0.5: worst settings regarding *test accuracy* for the MNIST dataset

parameter influence on accuracy

parameter name	parameter values				win ratios in %				avg. differences				best value
last activation	softmax		sigmoid		50.0		50.0		0.0		0.058		softmax
optimizer	Adam	my Adam	cAdam	cAdam hat	50.0	0	50.0	0	0.0	0.001	0.0	0.667	Adam

Table 0.6: parameter influence regarding *test accuracy* for the MNIST dataset