Text file to create

We need to set red words

The blue part can be omitted

■ Neural network setting「NET.txt」

CrossEntropy	**Square** or **CrossEntropy**
BATCH_SIZE **50**	Mini batch size
EPOCH **10**	Epoch number
LAMBDA **0.000000**	Load decay (weight decay)
EPS **0.001000**	Learning rate
OPTIMIZER **Adam**	Optimizer
VALIDATION_NUM **100**	Number of Validation data
ERROR_PLOT_STEP **10**	Graph data output interval(gnuplot format)
TEST_SAMPLE **10**	Number of test data samples
	※optimizer solver is listed on the final page

~~* Leave the setting of LAMBDA to 0 (there is a problem)~~

■ Layer setting「LAYER.txt」

LAYER **4**	Number of layers
1 [**28**, **28**]	Input feature map from left, input unit width, input unit height
Each layer setting See layer description	※ Width and height are numbers when input units are regarded as a matrix
END	

■ Describing layers

Fully Connected layer

LAYER_TYPE_FullyConnected**/layerName** **1** [*7, 7*] -> [**1**, **10**] **Softmax**	From the left, input of feature map, output unit width, output unit height, start function ※ The width and height are the numbers when looking at the input device as a matrix ※ Activation function described on final page

Convolutional layer

LAYER_TYPE_Convolutional**/layerName** **20** [*28, 28*]->(**5**, **5**)->[*28, 28*] st **1** **ReLU**	From the left, input feature map, convolution width, convolution height, stride, activation function

maxPooling layer

LAYER_TYPE_maxPooling**/layerName** **20** [*28, 28*]->(**4**, **4**)->[*7, 7*] st **4** **Identity**	From the left, input feature map, convolution width, convolution height, stride, activation function

■ **Activation function can use**

Identity	$h(x) = x$
ReLU	$h(x) = \max(0, x)$
Sigmoid	$h(x) = \frac{1}{1+e^{-x}} = \frac{\tanh(x/2)+1}{2}$
Tanh	$h(x) = \tanh(x)$
Abs	$h(x) = \lvert x \rvert$
Softsign	$h(x) = \frac{x}{1+\lvert x \rvert}$
Softplus	$h(x) = \log(1+e^{x})$
LReLU	$h(x) = \max(0.01x, x)$
ELU	$h(x) = \begin{cases} e^{x}-1 & x < 0 \\ x & x \geq 0 \end{cases}$

■Optimizer solver

Adam

$$
\begin{aligned}
m_{t+1} &= \beta_1 m_t + (1-\beta_1)\nabla E(\mathbf{w}^t) \\
v_{t+1} &= \beta_2 v_t + (1-\beta_2)\nabla E(\mathbf{w}^t)^2 \\
\hat{m} &= \frac{m_{t+1}}{1-\beta_1^t} \\
\hat{v} &= \frac{v_{t+1}}{1-\beta_2^t} \\
\mathbf{w}^{t+1} &= \mathbf{w}^t - \alpha\frac{\hat{m}}{\sqrt{\hat{v}}+\epsilon}
\end{aligned}
$$

α=0.001,β 1 =0.9,β 2 =0.999,ϵ=10E−8

AdaGrad

$$
\begin{aligned}
h_0 &= \epsilon \\
h_t &= h_{t-1} + \nabla E(\mathbf{w}^t)^2 \\
\eta_t &= \frac{\eta_0}{\sqrt{h_t}} \\
\mathbf{w}^{t+1} &= \mathbf{w}^t - \eta_t \nabla E(\mathbf{w}^t)
\end{aligned}
$$

ϵ=10E−8 ,η 0 =0.001

RMSprop

$$
\begin{aligned}
h_t &= \alpha h_{t-1} + (1-\alpha)\nabla E(\mathbf{w}^t)^2 \\
\eta_t &= \frac{\eta_0}{\sqrt{h_t}+\epsilon} \\
\mathbf{w}^{t+1} &= \mathbf{w}^t - \eta_t \nabla E(\mathbf{w}^t)
\end{aligned}
$$

α=0.99,ϵ=10E−8,η0=0.01

AdaDelta

$$
\begin{aligned}
h_t &= \rho h_{t-1} + (1-\rho)\nabla E(\mathbf{w}^t)^2 \\
v_t &= \frac{\sqrt{s_t+\epsilon}}{\sqrt{h_t+\epsilon}}\nabla E(\mathbf{w}^t) \\
s_{t+1} &= \rho s_t + (1-\rho)v_t^2 \\
\mathbf{w}^{t+1} &= \mathbf{w}^t - v_t
\end{aligned}
$$

ρ=0.95,ϵ=10E−6

SGD

$$
\mathbf{w}^{t+1} \leftarrow \mathbf{w}^t - \eta\frac{\partial E(\mathbf{w}^t)}{\partial \mathbf{w}^t}
$$

η=0.01