

Mar 04, 19 20:00

sgd.m

Page 1/1

```

function model = sgd(param, model, net, data)

lr = param.lr;
batch_size = param.bsize;
var_ptr = model.var_ptr;
ch_input = [model.ch_input;model.full_neurons];
num_w = var_ptr(2:end) - var_ptr(1:end-1) - ch_input(2:end);

for k = 1 : param.iter_max
    for j = 1 : ceil(data.l/batch_size)
        batch_idx = randsample(data.l,batch_size);
        [net, loss] = lossgrad_subset(param, model, net, data, batch_idx
, 'fungrad');
        for m = 1 : param.L
            gradW = [model.weight{m}(:);model.bias{m}]/param.C + [ne
t.dlossdvecW{m};net.dlossdb{m}]/batch_size;
            model.weight{m} = model.weight{m} - lr*reshape(gradW(1:n
um_w(m)), ch_input(m+1), []);
            model.bias{m} = model.bias{m} - lr*gradW(num_w(m)+1:end)
;
        end
    end
    fprintf(' %d-epoch loss: %g\n', k, loss/batch_size);
end

model.param = param;

```