Inputnodes =	784											
Hiddennodes =	200											
Outputnodes =	10											
Learningrate =	0.1											
neuralNetwork												
inodes =	Inputnodes =	784							Dot		=	
hnodes =	Hiddennodes =	200										
onodes =	Outputnodes =	10										
lr =	Learningrate =	0.1										
wih	大小:	200 * 784										
who	大小:	10 * 200										
Train												
inputs	input_list轉至	1 * 784	->	784 * 1								
targets	targets_list轉至	1 * 10	->	10 * 1								
hidden_inputs	Matrix_Dot(wih, inputs)	200 * 784	784 * 1	200 * 1								
hidden_outputs	Activation_Function(hidden_inputs);	200 * 1	200 * 1	200 * 1								
final_inputs	Matrix_Dot(who, hidden_outputs);	10 * 200	200 * 1	10 * 1								
final_outputs	Activation_Function(final_inputs);	10 * 1	->	10 * 1								
output_errors	Matrix_Subtract(targets, final_outputs);	10 * 1	10 * 1	10 * 1								
hidden_errors	Matrix_Dot(Matrix_Transpose(who), output_errors);	200 * 10	10 * 1	200 * 1								
who1	(1 - final_outputs)	10 * 1	10 * 1	10 * 1								
who2	output_errors * final_outputs	10 * 1	10 * 1	10 * 1								
who3	(output_errors * final_outputs * (1 - final_outputs)	10 * 1	10 * 1	10 * 1								
who4	numpy.transpose(hidden_outputs)	200 * 1	->	1 * 200								
who5	numpy.dot((who3, who4	10 * 1	1 * 200	10 * 200								
who6	self.lr * numpy.dot	10 * 200	10 * 200	10 * 200								
who	self.who +=	10 * 200	10 * 200	10 * 200								
wih1	(1 - hidden_outputs)	200 * 1	200 * 1	200 * 1								
wih2	hidden_errors * hidden_outputs	200 * 1	200 * 1	200 * 1								
wih3	(hidden_errors * hidden_outputs * (1 - hidden_outputs)	200 * 1	200 * 1	200 * 1								
wih4	numpy.transpose(inputs)	784 * 1	->	1 * 784								
wih5	numpy.dot((wih3, wih4	200 * 1	1 * 784	200 * 784								
wih6	self.lr * numpy.dot	200 * 784	200 * 784	200 * 784								
wih	self.wih +=	200 * 784	200 * 784	200 * 784								
Query												
input		1 * 784										
inputs	Matrix_Transpose(input);	1 * 784	->	784 * 1								
hidden_inputs	Matrix_Dot(wih, inputs);	200 * 784	1 * 784	200 * 1								
hidden_outputs	Activation_Function(hidden_inputs);	200 * 1	->	200 * 1								
final_inputs	Matrix_Dot(who, hidden_outputs);	10 * 200	200 * 1	10 * 1								
final_outputs	Activation_Function(final_inputs);	10 * 1	->	10 * 1								