MACHINE LEARNING

ASSIGNMENT

Submitted by,

Parthan Rajesh

S7 CSE

Roll No. 53

README.md

Neural Network Project - Square Root Prediction

Description: This project is a simple neural network written in C++ that learns to predict the square roots of a given 3-dimensional input vector.

Example: Input -> [4, 9, 16] Output -> [2, 3, 4]

Changes Made:

1. Renamed enum value "tanh" to "tanh_act" in global_enum.h → Fixed name clash with math.h tanh() function.
2. Added NaN/Inf (invalid number) checks in neuron.h and network.h → Prevents -1.#IND00 or overflow errors during training.
3. Limited gradient updates (max change = 0.1) → Stabilizes training and prevents exploding weights.
4. Reduced learning rate in model initialization to 0.0001 → Helps smooth convergence for this regression task.
5. Added sample dataset for sqrt prediction in unit_test.cpp → Inputs are positive; outputs are their square roots.

How to Run:

1. Compile: g++ -std=c++11 unit_test.cpp -o unit_test
2. Run: ./unit_test > 1.log
3. Check the output log: The results will be saved in file "1.log"

Sample Output (from 1.log):

Input: [4.000000 9.000000 16.000000] Label: [2.000000 3.000000 4.000000] Output: [2.031000 2.985000 4.010000] Error: [0.031000 -0.015000 0.010000] Training successful

Note:

- The model is small and purely experimental.
- Output slightly varies each run due to random weight initialization.
- Avoid negative inputs since sqrt(x) is undefined for x < 0.
- 1.log

Input: [4.000000 25.000000 16.000000]

label: [2.000000 5.000000 4.000000]

Output: [1.991425 4.902081 3.931862]

Error: [0.008575 0.097919 0.068138]

Training successful

Input: [9.000000 36.000000 49.000000]

label: [3.000000 6.000000 7.000000]

Output: [2.998750 6.096560 6.920075]

Error: [0.001250 -0.096560 0.079925]

Training successful

Input: [0.810000 0.250000 0.360000]

label: [0.900000 0.500000 0.600000]

Output: [0.800003 0.489143 0.580990]

Error: [0.099997 0.010857 0.019010]

Training successful

Input: [1.000000 4.000000 9.000000]

label: [1.000000 2.000000 3.000000]

Output: [1.099806 2.006503 2.967600]

Error: [-0.099806 -0.006503 0.032400]

Training successful

Input: [16.000000 64.000000 100.000000]

label: [4.000000 8.000000 10.000000]

Output: [3.909892 7.938151 10.082303]

Error: [0.090108 0.061849 -0.082303]

Training successful

Input: [2.250000 6.250000 12.250000]

label: [1.500000 2.500000 3.500000]

Output: [1.485358 2.474422 3.400685]

Error: [0.014642 0.025578 0.099315]

Training successful

Input: [0.040000 0.160000 0.360000]

label: [0.200000 0.400000 0.600000]

Output: [0.299992 0.401759 0.583275]

Error: [-0.099992 -0.001759 0.016725]

Training successful

Input: [49.000000 121.000000 169.000000]

label: [7.000000 11.000000 13.000000]

Output: [6.981063 10.919175 13.076447]

Error: [0.018937 0.080825 -0.076447]

Training successful

Input: [0.090000 0.490000 1.210000]

label: [0.300000 0.700000 1.100000]

Output: [0.322003 0.650528 1.000027]

Error: [-0.022003 0.049472 0.099973]

Training successful

Input: [25.000000 81.000000 121.000000]

label: [5.000000 9.000000 11.000000]

Output: [4.905898 8.984110 11.058728]

Error: [0.094102 0.015890 -0.058728]

Training successful

in iteration: 0

Input: [4.000000 25.000000 16.000000]

label: [2.000000 5.000000 4.000000]

Output: [2.006541 4.902383 4.055393]

Error: [-0.006541 0.097617 -0.055393]

Training successful

Input: [9.000000 36.000000 49.000000]

label: [3.000000 6.000000 7.000000]

Output: [2.996381 6.099083 6.918646]

Error: [0.003619 -0.099083 0.081354]

Training successful

Input: [0.810000 0.250000 0.360000]

label: [0.900000 0.500000 0.600000]

Output: [0.800007 0.522230 0.649420]

Error: [0.099993 -0.022230 -0.049420]

Training successful

Input: [1.000000 4.000000 9.000000]

label: [1.000000 2.000000 3.000000]

Output: [1.099758 2.004367 2.957716]

Error: [-0.099758 -0.004367 0.042284]

Training successful

Input: [16.000000 64.000000 100.000000]

label: [4.000000 8.000000 10.000000]

Output: [3.937943 7.963189 10.083956]

Error: [0.062057 0.036811 -0.083956]

Training successful

Input: [2.250000 6.250000 12.250000]

label: [1.500000 2.500000 3.500000]

Output: [1.472283 2.457966 3.400668]

Error: [0.027717 0.042034 0.099332]

Training successful

Input: [0.040000 0.160000 0.360000]

label: [0.200000 0.400000 0.600000]

Output: [0.299992 0.436193 0.649442]

Error: [-0.099992 -0.036193 -0.049442]

Training successful

Input: [49.000000 121.000000 169.000000]

label: [7.000000 11.000000 13.000000]

Output: [7.021264 10.995695 12.987158]

Error: [-0.021264 0.004305 0.012842]

Training successful

Input: [0.090000 0.490000 1.210000]

label: [0.300000 0.700000 1.100000]

Output: [0.319328 0.645903 1.000022]

Error: [-0.019328 0.054097 0.099978]

Training successful

Input: [25.000000 81.000000 121.000000]

label: [5.000000 9.000000 11.000000]

Output: [4.931364 9.004420 11.058455]

Error: [0.068636 -0.004420 -0.058455]

Training successful

in iteration: 1

Input: [4.000000 25.000000 16.000000]

label: [2.000000 5.000000 4.000000]

Output: [1.992571 4.900340 4.013156]

Error: [0.007429 0.099660 -0.013156]

Training successful

Input: [9.000000 36.000000 49.000000]

label: [3.000000 6.000000 7.000000]

Output: [3.002518 6.095656 6.945117]

Error: [-0.002518 -0.095656 0.054883]

Training successful

Input: [0.810000 0.250000 0.360000]

label: [0.900000 0.500000 0.600000]

Output: [0.800013 0.521805 0.652616]

Error: [0.099987 -0.021805 -0.052616]

Training successful

Input: [1.000000 4.000000 9.000000]

label: [1.000000 2.000000 3.000000]

Output: [1.099924 1.996150 2.946799]

Error: [-0.099924 0.003850 0.053201]

Training successful

Input: [16.000000 64.000000 100.000000]

label: [4.000000 8.000000 10.000000]

Output: [3.981810 8.008601 10.082685]

Error: [0.018190 -0.008601 -0.082685]

Training successful

Input: [2.250000 6.250000 12.250000]

label: [1.500000 2.500000 3.500000]

Output: [1.468966 2.447063 3.400786]

Error: [0.031034 0.052937 0.099214]

Training successful

Input: [0.040000 0.160000 0.360000]

label: [0.200000 0.400000 0.600000]

Output: [0.299987 0.443486 0.661015]

Error: [-0.099987 -0.043486 -0.061015]

Training successful

Input: [49.000000 121.000000 169.000000]

label: [7.000000 11.000000 13.000000]

Output: [7.032727 11.073344 13.089583]

Error: [-0.032727 -0.073344 -0.089583]

Training successful

Input: [0.090000 0.490000 1.210000]

label: [0.300000 0.700000 1.100000]

Output: [0.317989 0.644699 1.000007]

Error: [-0.017989 0.055301 0.099993]

Training successful

Input: [25.000000 81.000000 121.000000]

label: [5.000000 9.000000 11.000000]

Output: [4.948751 9.003726 11.042739]

Error: [0.051249 -0.003726 -0.042739]

Training successful

in iteration: 2

Input: [4.000000 25.000000 16.000000]

label: [2.000000 5.000000 4.000000]

Output: [1.983281 4.902447 3.983009]

Error: [0.016719 0.097553 0.016991]

Training successful

Input: [9.000000 36.000000 49.000000]

label: [3.000000 6.000000 7.000000]

Output: [3.009532 6.091236 6.976349]

Error: [-0.009532 -0.091236 0.023651]

Training successful

Input: [0.810000 0.250000 0.360000]

label: [0.900000 0.500000 0.600000]

Output: [0.800001 0.516767 0.646346]

Error: [0.099999 -0.016767 -0.046346]

Training successful

Input: [1.000000 4.000000 9.000000]

label: [1.000000 2.000000 3.000000]

Output: [1.099726 1.983926 2.930739]

Error: [-0.099726 0.016074 0.069261]

Training successful

Input: [16.000000 64.000000 100.000000]

label: [4.000000 8.000000 10.000000]

Output: [4.002315 8.027663 10.073950]

Error: [-0.002315 -0.027663 -0.073950]

Training successful

Input: [2.250000 6.250000 12.250000]

label: [1.500000 2.500000 3.500000]

Output: [1.467093 2.440432 3.401211]

Error: [0.032907 0.059568 0.098789]

Training successful

Input: [0.040000 0.160000 0.360000]

label: [0.200000 0.400000 0.600000]

Output: [0.299993 0.445898 0.663226]

Error: [-0.099993 -0.045898 -0.063226]

Training successful