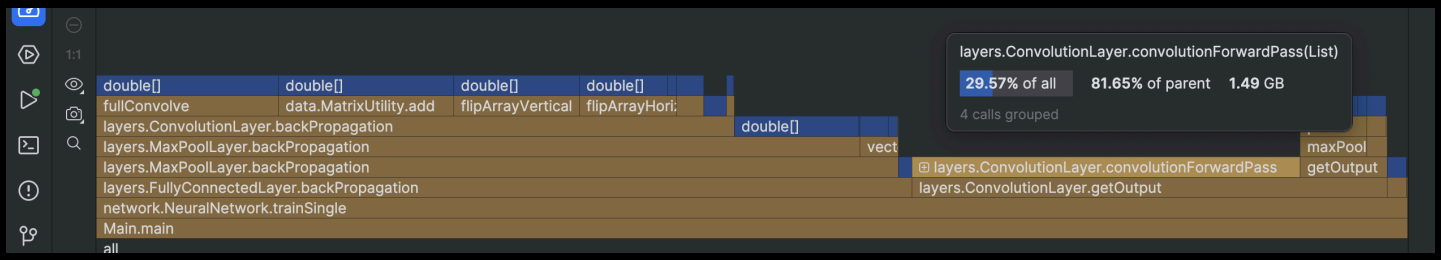
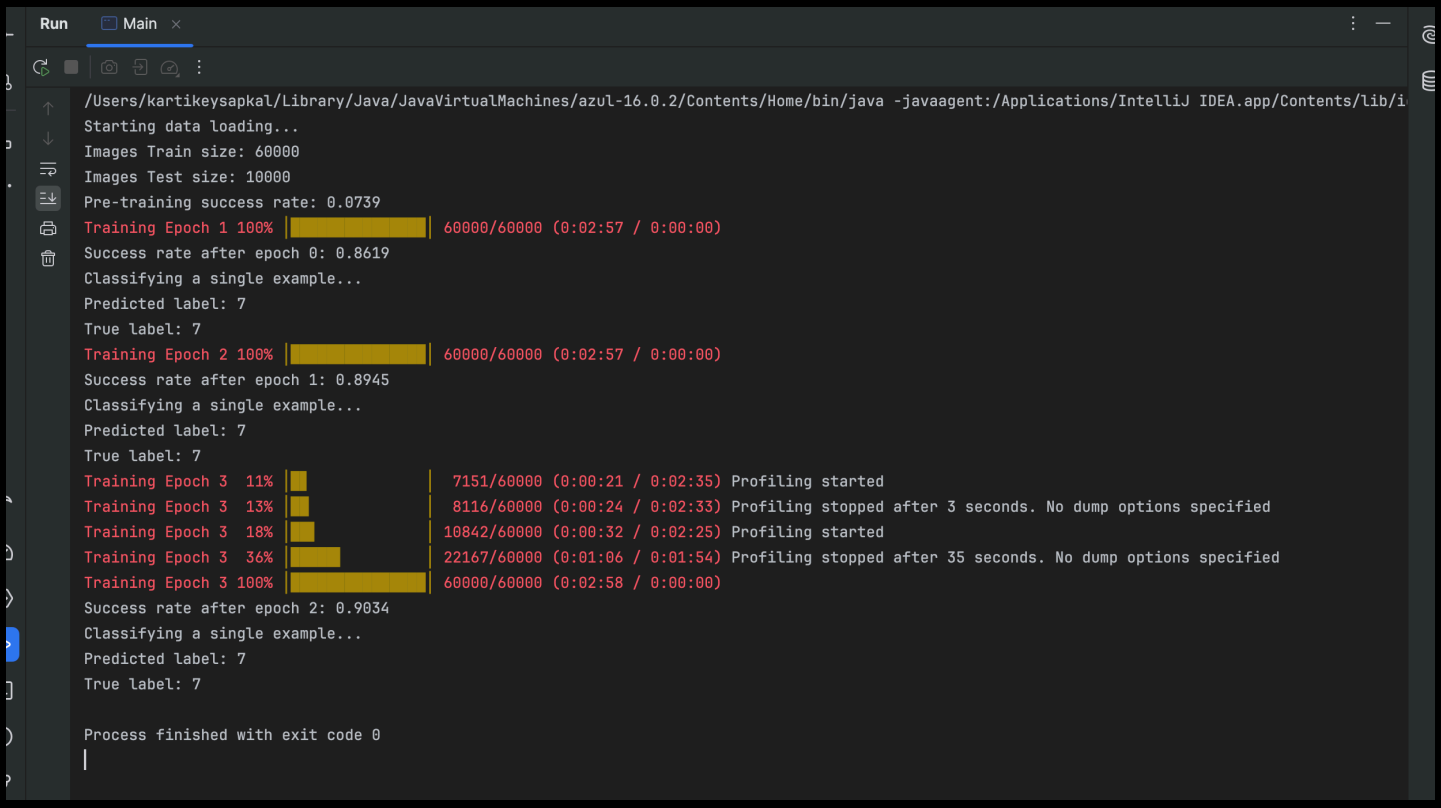
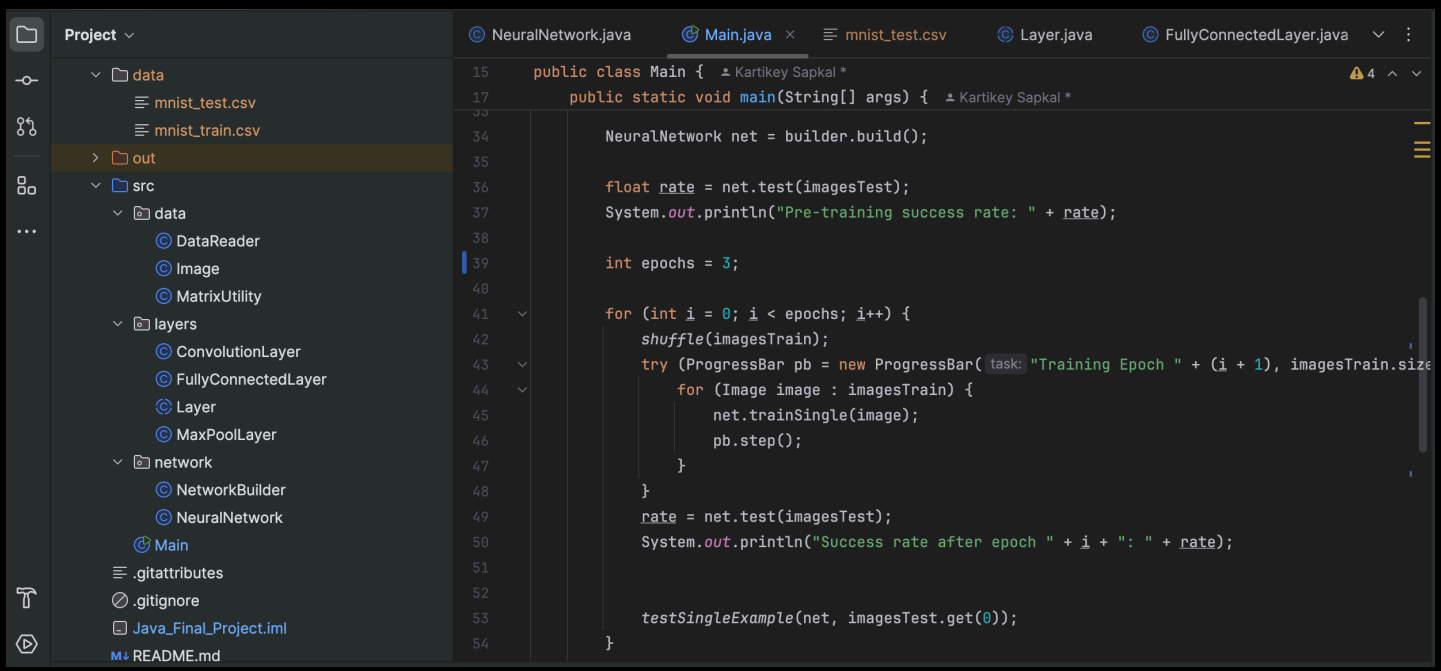
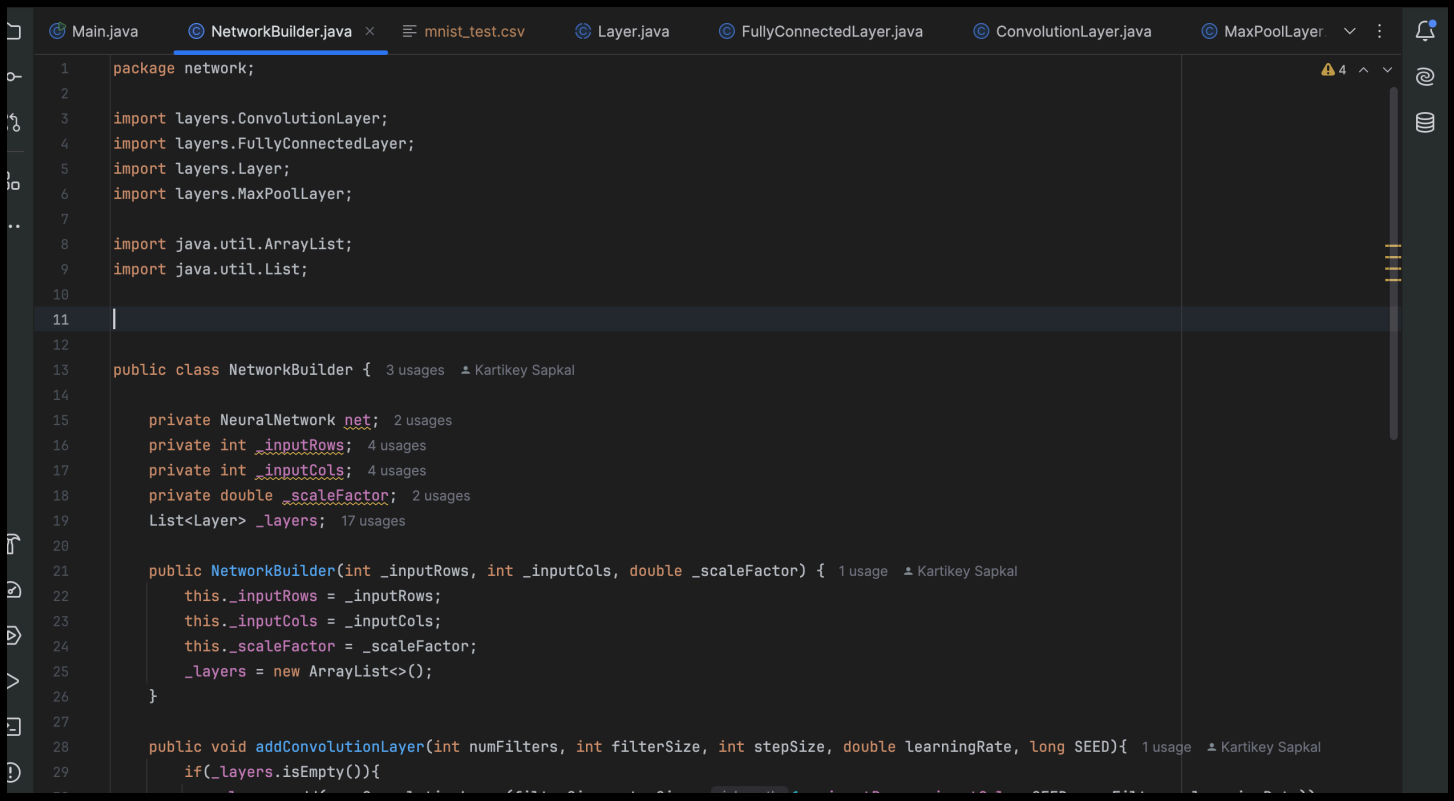
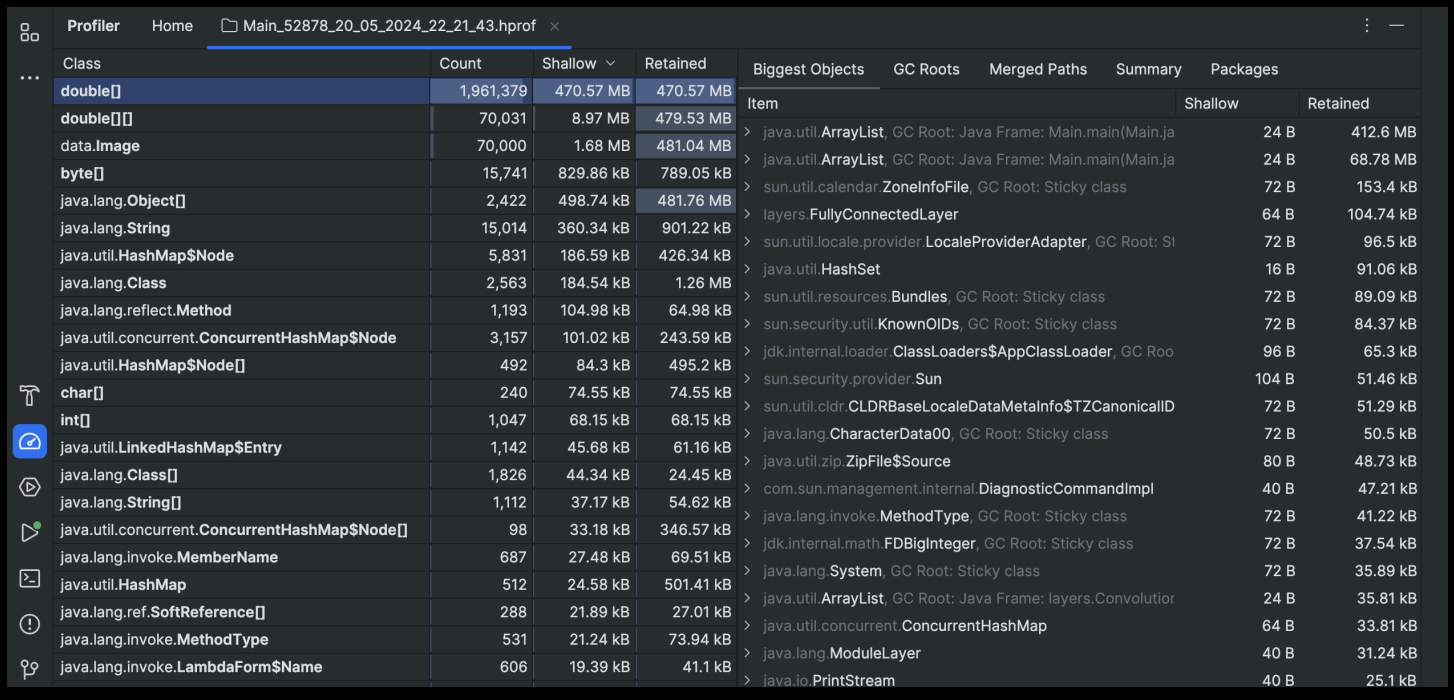

layers.ConvolutionLayer.convolutionForwardPass(List)
29.57% of all
81.65% of parent
1.49 GB
4 calls grouped
double[]
double[]
double[]
double[]
fullConvolve
data.MatrixUtility.add
flipArrayVertical
flipArrayHori
layers.ConvolutionLayer.backPropagation
double[]
layers.MaxPoolLayer.backPropagation
vect
maxPool
layers.MaxPoolLayer.backPropagation
layers.ConvolutionLayer.convolutionForwardPass
getOutput
layers.FullyConnectedLayer.backPropagation
layers.ConvolutionLayer.getOutput
network.NeuralNetwork.trainSingle
Main.main
all


Run
Main
/Users/kartikeysapkal/Library/Java/JavaVirtualMachines/azul-16.0.2/Contents/Home/bin/java -javaagent:/Applications/IntelliJ IDEA.app/Contents/lib/i
Starting data loading...
Images Train size: 60000
Images Test size: 10000
Pre-training success rate: 0.0739
Training Epoch 1 100% 60000/60000 (0:02:57 / 0:00:00)
Success rate after epoch 0: 0.8619
Classifying a single example...
Predicted label: 7
True label: 7
Training Epoch 2 100% 60000/60000 (0:02:57 / 0:00:00)
Success rate after epoch 1: 0.8945
Classifying a single example...
Predicted label: 7
True label: 7
Training Epoch 3 11% 7151/60000 (0:00:21 / 0:02:35) Profiling started
Training Epoch 3 13% 8116/60000 (0:00:24 / 0:02:33) Profiling stopped after 3 seconds. No dump options specified
Training Epoch 3 18% 10842/60000 (0:00:32 / 0:02:25) Profiling started
Training Epoch 3 36% 22167/60000 (0:01:06 / 0:01:54) Profiling stopped after 35 seconds. No dump options specified
Training Epoch 3 100% 60000/60000 (0:02:58 / 0:00:00)
Success rate after epoch 2: 0.9034
Classifying a single example...
Predicted label: 7
True label: 7
Process finished with exit code 0


Project
data
mnist_test.csv
mnist_train.csv
out
src
data
DataReader
Image
MatrixUtility
layers
ConvolutionLayer
FullyConnectedLayer
Layer
MaxPoolLayer
network
NetworkBuilder
NeuralNetwork
Main
.gitattributes
.gitignore
Java_Final_Project.iml
README.md
NeuralNetwork.java
Main.java
mnist_test.csv
Layer.java
FullyConnectedLayer.java
public class Main { Kartikey Sapkal *
public static void main(String[] args) { Kartikey Sapkal *
NeuralNetwork net = builder.build();
float rate = net.test(imagesTest);
System.out.println("Pre-training success rate: " + rate);
int epochs = 3;
for (int i = 0; i < epochs; i++) {
shuffle(imagesTrain);
try (ProgressBar pb = new ProgressBar(task: "Training Epoch " + (i + 1), imagesTrain.size
for (Image image : imagesTrain) {
net.trainSingle(image);
pb.step();
}
}
rate = net.test(imagesTest);
System.out.println("Success rate after epoch " + i + ": " + rate);
testSingleExample(net, imagesTest.get(0));
}


Main.java
NetworkBuilder.java
mnist_test.csv
Layer.java
FullyConnectedLayer.java
ConvolutionLayer.java
MaxPoolLayer
package network;
import layers.ConvolutionLayer;
import layers.FullyConnectedLayer;
import layers.Layer;
import layers.MaxPoolLayer;
import java.util.ArrayList;
import java.util.List;
public class NetworkBuilder { 3 usages Kartikey Sapkal
private NeuralNetwork net; 2 usages
private int _inputRows; 4 usages
private int _inputCols; 4 usages
private double _scaleFactor; 2 usages
List<Layer> _layers; 17 usages
public NetworkBuilder(int _inputRows, int _inputCols, double _scaleFactor) { 1 usage Kartikey Sapkal
this._inputRows = _inputRows;
this._inputCols = _inputCols;
this._scaleFactor = _scaleFactor;
_layers = new ArrayList<>();
}
public void addConvolutionLayer(int numFilters, int filterSize, int stepSize, double learningRate, long SEED){ 1 usage Kartikey Sapkal
if(_layers.isEmpty()){


Profiler
Home
Main_52878_20_05_2024_22_21_43.hprof
Class
Count
Shallow
Retained
double[] 1,961,379 470.57 MB 470.57 MB
double[][] 70,031 8.97 MB 479.53 MB
data.Image 70,000 1.68 MB 481.04 MB
byte[] 15,741 829.86 kB 789.05 kB
java.lang.Object[] 2,422 498.74 kB 481.76 MB
java.lang.String 15,014 360.34 kB 901.22 kB
java.util.HashMap$Node 5,831 186.59 kB 426.34 kB
java.lang.Class 2,563 184.54 kB 1.26 MB
java.lang.reflect.Method 1,193 104.98 kB 64.98 kB
java.util.concurrent.ConcurrentHashMap$Node 3,157 101.02 kB 243.59 kB
java.util.HashMap$Node[] 492 84.3 kB 495.2 kB
char[] 240 74.55 kB 74.55 kB
int[] 1,047 68.15 kB 68.15 kB
java.util.LinkedHashMap$Entry 1,142 45.68 kB 61.16 kB
java.lang.Class[] 1,826 44.34 kB 24.45 kB
java.lang.String[] 1,112 37.17 kB 54.62 kB
java.util.concurrent.ConcurrentHashMap$Node[] 98 33.18 kB 346.57 kB
java.lang.invoke.MemberName 687 27.48 kB 69.51 kB
java.util.HashMap 512 24.58 kB 501.41 kB
java.lang.ref.SoftReference[] 288 21.89 kB 27.01 kB
java.lang.invoke.MethodType 531 21.24 kB 73.94 kB
java.lang.invoke.LambdaForm$Name 606 19.39 kB 41.1 kB
Biggest Objects
GC Roots
Merged Paths
Summary
Packages
Item
Shallow
Retained
java.util.ArrayList, GC Root: Java Frame: Main.main(Main.ja 24 B 412.6 MB
java.util.ArrayList, GC Root: Java Frame: Main.main(Main.ja 24 B 68.78 MB
sun.util.calendar.ZoneInfoFile, GC Root: Sticky class 72 B 153.4 kB
layers.FullyConnectedLayer 64 B 104.74 kB
sun.util.locale.provider.LocaleProviderAdapter, GC Root: St 72 B 96.5 kB
java.util.HashSet 16 B 91.06 kB
sun.util.resources.Bundles, GC Root: Sticky class 72 B 89.09 kB
sun.security.util.KnownOIDs, GC Root: Sticky class 72 B 84.37 kB
jdk.internal.loader.ClassLoaders$AppClassLoader, GC Roo 96 B 65.3 kB
sun.security.provider.Sun 104 B 51.46 kB
sun.util.cldr.CLDRBaseLocaleDataMetaInfo$TZCanonicalID 72 B 51.29 kB
java.lang.CharacterData00, GC Root: Sticky class 72 B 50.5 kB
java.util.zip.ZipFile$Source 80 B 48.73 kB
com.sun.management.internal.DiagnosticCommandImpl 40 B 47.21 kB
java.lang.invoke.MethodType, GC Root: Sticky class 72 B 41.22 kB
jdk.internal.math.FDBigInteger, GC Root: Sticky class 72 B 37.54 kB
java.lang.System, GC Root: Sticky class 72 B 35.89 kB
java.util.ArrayList, GC Root: Java Frame: layers.Convolutior 24 B 35.81 kB
java.util.concurrent.ConcurrentHashMap 64 B 33.81 kB
java.lang.ModuleLayer 40 B 31.24 kB
java.io.PrintStream 40 B 25.1 kB