Franciszek Piszcz, Haemanth Santhi Ponnusamy

Exercise 1 Multilayer Perceptrons

a. Implementation Back-Propagation

i. MultiLayerPerceptron.java, lines 206-243 and 326-391

b. XOR

i. With the given parameters 10000 epochs was not enough to reduce the error below the constraint (< 0.01). But increasing the momentum coefficient (0.5 to 0.95) helped in learning faster.

ii. Also we reduced the number of units in first hidden layer (20 to 2). Since 2 two units are sufficient enough to learn XOR.

```
epoch: 9993 training error: 0.0030821856118021795
epoch: 9994 training error: 0.0030820316494192257
epoch: 9995 training error: 0.0030818723001621307
epoch: 9996 training error: 0.003081714075022224
epoch: 9997 training error: 0.003081557702704213
epoch: 9998 training error: 0.0030814014871231865
epoch: 9999 training error: 0.003081242603812623
epoch: 10000 training error: 0.0030810876520673824
```

c. Geometry:

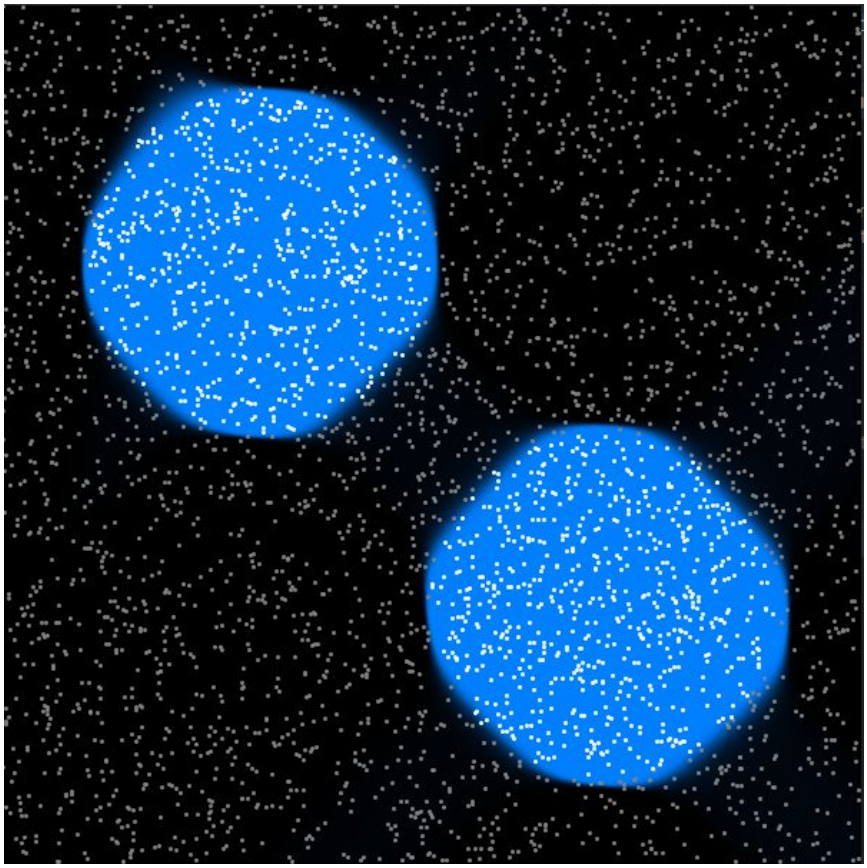
i. No, It is not solvable by one hidden layer. (since the data is not linearly separable)

ii. We need two hidden layers: one to learn set of linear separators and the second layer to learn to combine them to create the decision boundary.

iii. Increasing the number of units in the first hidden layer smoothens the decision boundary with better approximation.

Screenshot of a successful training run
Parameters (learning rate: 0.01, momentum: 0.95, 20 units in hidden layer)

```
epoch: 1995 training error: 0.022602483988094214
epoch: 1996 training error: 0.02061975422544343
epoch: 1997 training error: 0.023962082012653898
epoch: 1998 training error: 0.02646158333376371
epoch: 1999 training error: 0.02514416648485564
epoch: 2000 training error: 0.02323801198981493
```



Exercise 2 Recurrent Neural Networks

a. Implementation Back-Propagation Through Time

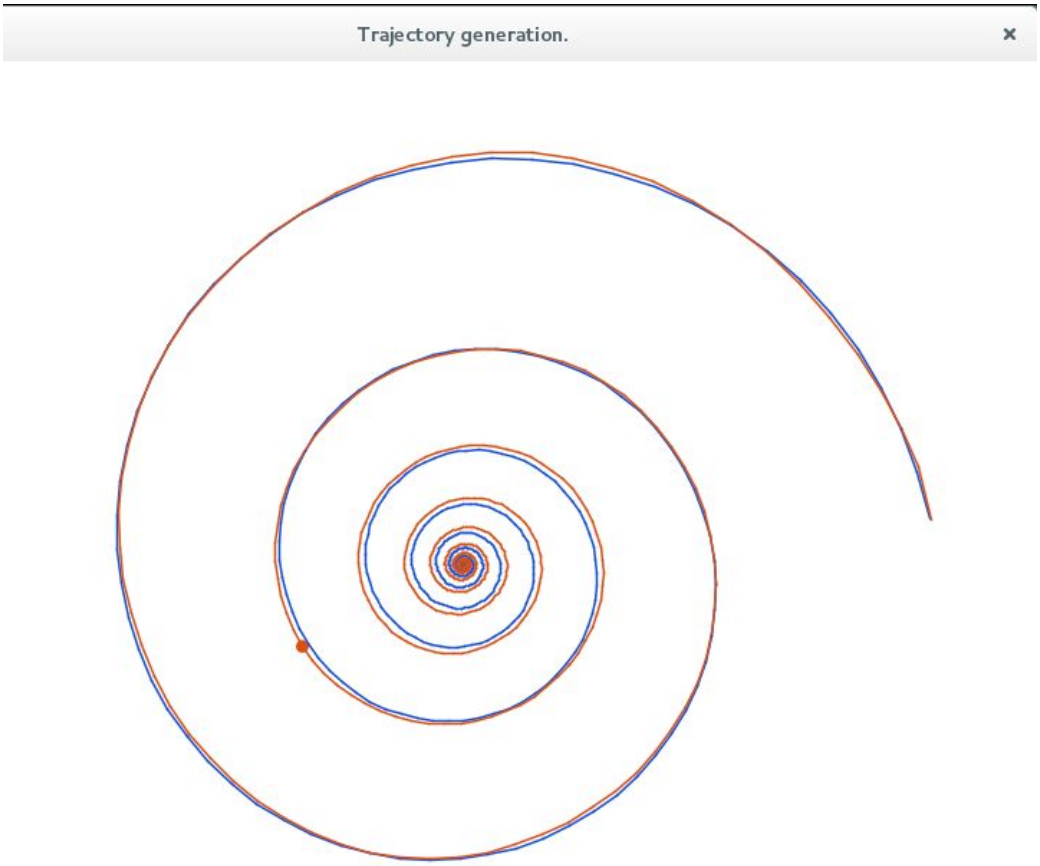
i. RecurrentNeuralNetwork.java, lines 325-423 and 514-582

b. Trajectory Generation

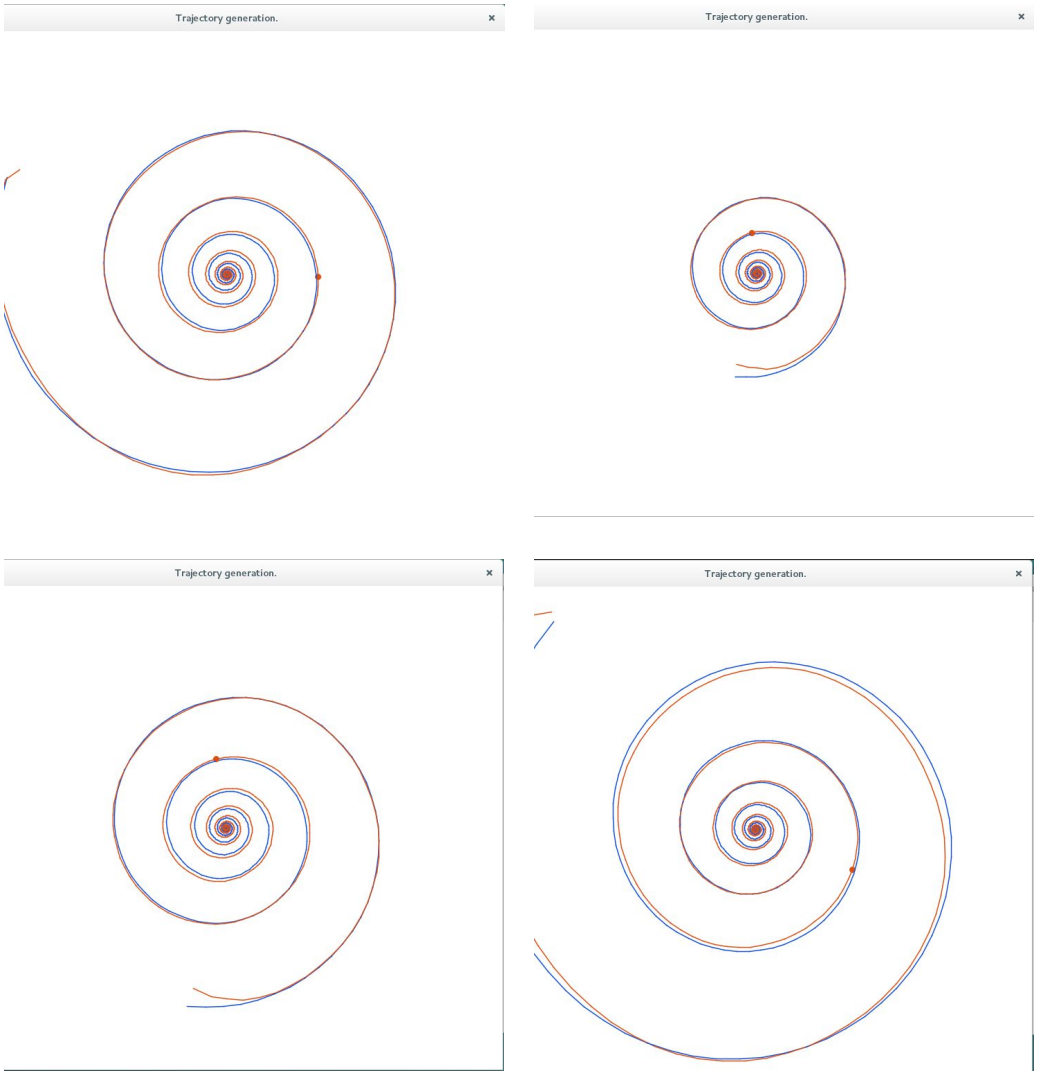
i. We found that the initial settings (learning rate = 0.00002, momentum rate = 0.95, one hidden layer of 16 neurons) work pretty well:

```
franek@debian:java$ javac de/cogmod/anns/exercisesheet01/RNNTrajectory.java && java
epoch: 1 training error: 0.4672477719933927
epoch: 1001 training error: 0.46478949589974766
epoch: 2001 training error: 0.42308284249417766
epoch: 3001 training error: 0.46097955407332786
epoch: 4001 training error: 0.4123991598705403
epoch: 5001 training error: 0.4078908795002563
epoch: 6001 training error: 0.45531733332275587
epoch: 7001 training error: 0.0727340287844561
epoch: 8001 training error: 0.04995516531206807
epoch: 9001 training error: 0.040903661495657606
epoch: 10001 training error: 0.03546548223652344
epoch: 11001 training error: 0.031686386862729024
epoch: 12001 training error: 0.028870290149261155
epoch: 13001 training error: 0.026680946749573848
epoch: 14001 training error: 0.024925860300448372
epoch: 15001 training error: 0.023484321797008992
epoch: 16001 training error: 0.022276696987753468
epoch: 17001 training error: 0.021248494144430242
```

ii. Below is a spiral corresponding to the above training run:



iii. We implemented the “tricky” part, that is forcing trajectory to start from a chosen point. You can run the RNNTrajectory.java file and after clicking it will start the spiral from the point you clicked. We implemented it in the function forceTrajectoryByInitialization() in RecurrentNeuralNetwork.java, lines 630-661.



c. Gradient Measurement

i. We implemented gradient measurement in GradientMeasurement.java, closely following the instructions. The result indeed shows that vanishing gradient is a problem:

```
franek@debian:java$ javac de/cogmod/anns/exercisesheet01/GradientMeasurement.java && java -ea de.cogmod.anns.exercisesheet01.Gradien
tMeasurement
100
[[-1.0782431235968582E-117, -1.6007426315158488E-116, -2.377730299871274E-115, -3.531995457343568E-114, -5.246610105502545E-113, -7.
7935893950667738E-112, -1.1577005941302315E-110, -1.7197091080316978E-109, -2.5545459952229O9E-108, -3.794656440007762E-107, -5.63678
1457301306E-106, -8.373170456852046E-105, -1.2437946021250606E-103, -1.8475976575986886E-102, -2.744518346141632E-101, -4.0768513216
767057E-100, -6.055968517180166E-99, -8.99585287451541E-98, -1.3362911103376798E-96, -1.9849968162843038E-95, -2.94861825404431E-94,
 -4.380032016554155E-93, -6.506329003330914E-92, -9.664841932568473E-91, -1.435666249488968E-89, -2.1326138537001027E-88, -3.1678963
342716313E-87, -4.705759163704131E-86, -6.990181170772696E-85, -1.0383581288457396E-83, -1.542431558495418E-82, -2.2912086365493716E
-81, -3.4034813326299204E-80, -5.055709461276174E-79, -7.510015674769892E-78, -1.1155770691983706E-76, -1.6571366175735155E-75, -2.4
6159754007518 36E-74, -3.656585935670678E-73, -5.431684297399903E-72, -8.06850839161512E-71, -1.1985385029966996E-69, -1.780371877233
7028E-68, -2.6446576503962238E-67, -3.9285130130603445E-66, -5.835619022928074E-65, -8.668534192847513E-64, -1.287669478719O3E-62, -
1.912771697656388E-61, -2.8413312793551345E-60, -4.22066232417255E-59, -6.269592913760017E-58, -9.313181743809844E-57, -1.3834288028
952075E-55, -2.0550176140953708E-54, -3.0526308151198174E-53, -4.5345377234253806E-52, -6.735839874027074E-51, -1.0005769402720837E-
49, -1.4863094018381065E-48, -2.207841845117 51E-47, -3.2796439335064128E-46, -4.8717549014535217E-45, -7.2367599352352O9E-44, -1.074
986230210398E-42, -1.5968408590085468E-41, -2.3720310617374915E-40, -3.523539196849532E-39, -5.2340497019634E-38, -7.774931610557315
E-37, -1.1549290700500509E-35, -1.7155921410748755E-34, -2.5484304368495322E-33, -3.7855720692402333E-32, -5.6232870570827764E-31, -
8.353125167869567E-30, -1.2408169698932887E-28, -1.843174526699734E-27, -2.7379479958006686E-26, -4.0670913790954896E-25, -6.0414705
87199972E-24, -8.974316889855507E-23, -1.3330920423610431E-21, -1.9802447531301187E-20, -2.9415592905004 64E-19, -4.3695462623256O1E-
18, -6.490752914708458E-17, -9.641704394582515E-16, -1.4322292783916167E-14, -2.1275083967879634E-13, -3.1603124211273525E-12, -4.69
4493621839854E-11, -6.973446744747324E-10, -1.0358723095414178E-8, -1.5387389922822895E-7, -2.2857235052630258E-6, -3.39533343128113
E-5, -5.043606141788656E-4, -0.007492036769976366, -0.11129063884986831]]
```

ii. And below you can find a log-scale plot of the above sequence (reversed in time):