

```

1 package teste;
2
3 import java.io.FileNotFoundException;
4 import java.io.PrintWriter;
5 import java.io.UnsupportedEncodingException;
6
7 import classes.MultiLayerPerceptron;
8
9 /**
10  * created by lgcaobianco on 2018-04-29
11  */
12
13 public class TestarImplementacao {
14     public static void main(String[] args) throws
15     FileNotFoundException, UnsupportedEncodingException {
16         MultiLayerPerceptron mlp = new
17         MultiLayerPerceptron();
18         double epsilon = Math.pow(10, -6);
19         int contadorEpocas = 0;
20         double erroAnterior = 10.0;
21         double erroAtual = 0;
22         PrintWriter writer = new PrintWriter("/home/
23         lgcaobianco/repositorios/epc-rna/epc4/src/base/eqm.csv", "
24         UTF-8");
25
26         // fase de treino
27         while (Math.abs((erroAtual - erroAnterior)) >
28         epsilon) {
29             for (int i = 0; i < mlp.getMatrizInputs().size
30             (); i++) {
31                 mlp.forwardPropagation(i);
32                 mlp.backwardPropagation(i);
33                 mlp.forwardPropagation(i);
34             }
35
36             erroAnterior = erroAtual;
37             erroAtual = mlp.calcularErroTotal();
38             writer.println(contadorEpocas+", "+erroAtual);
39             contadorEpocas++;
40         }
41         writer.close();
42         System.out.println(contadorEpocas);
43         // trocar a matriz para o conjunto de operacao
44         mlp.setMatrizInputs(mlp.getMatrizOperacao());
45
46         double erroRelativoMedio = 0.0,
47         variancaErroRelativo = 0.0, erroRelativo;
48         for (int i = 0; i < mlp.getMatrizOperacao().size()
49         ; i++) {
50             mlp.forwardPropagation(i);

```

```
43         erroRelativoMedio += (mlp.calcularErroRelativo
44         (i) / mlp.getMatrizInputs().size());
45         mlp.imprimirY2();
46     }
47     for (int i = 0; i < mlp.getMatrizOperacao().size()
48     ; i++) {
49         mlp.forwardPropagation(i);
50         erroRelativo = mlp.calcularErroRelativo(i);
51         variancaErroRelativo += Math.pow(Math.abs(
52         erroRelativo/100 - erroRelativoMedio/100), 2)/mlp.
53         getMatrizInputs().size());
54     }
55     System.out.println("Erro relativo medio: " +
56     erroRelativoMedio);
57     System.out
58         .println("Variancia do erro relativo médio
59     : " + variancaErroRelativo*100);
60 }
```