

Instituto de Ciências Matemáticas e de Computação
SCC0570 – Introdução a Redes Neurais
(1º semestre de 2012)

Professor: Dr. Zhao Liang (zhao at icmc)
PAE: Fabiano Berardo de Sousa (fabber at icmc)

Descrição do Trabalho 2

1. Objetivos

Cada grupo terá a opção de implementar um dos seguintes modelos de Rede Neural Artificial:

- Rede Multi-Layer Perceptron (MLP);
- Mapa de Kohonen (rede auto-organizável – SOM);
- Rede de Hopfield.

2. Descrição Geral

- O trabalho deve ser realizado pelo mesmo grupo formado para o trabalho 1.
- Cada grupo deve escolher um dos modelos citados na Seção 1 para resolver problemas de classificação / reconhecimento de padrões.
- O programa deve operar em duas fases: (i) etapa de treinamento, na qual os pesos da rede devem ser ajustados e (ii) etapa de teste, na qual exemplos que não pertencem ao conjunto de treinamento devem ser submetidos à rede e classificados/reconhecidos de acordo com a matriz de pesos obtida.
- O programa deve gerar como saída, pelo menos, informações sobre taxa de acerto e tempo decorrido em cada etapa. Obviamente que, dependendo do modelo de RNA escolhido, a saída do programa deverá incorporar informações extras (ver Seção 4).
- Pode-se utilizar qualquer linguagem de programação que o grupo achar adequada.
- Deve-se gerar um pequeno relatório contendo: (i) o nome dos integrantes do grupo, (ii) instruções sobre os procedimentos de compilação e execução do programa e (iii) descrição e resultado de alguns casos de testes. Além disto, informações adicionais (que dependem do modelo escolhido) garantirão os pontos complementares (ver Seção 4).
- O relatório e o código fonte devem ser enviados para o e-mail fabber.usp@gmail.com até o dia 24/05/2012.
- A apresentação (obrigatória) deve durar de 5 a 10 minutos e, nela, os alunos devem: (i) exibir o código fonte e comentar detalhes que o grupo achar relevante sobre o mesmo, (ii) executar alguns casos de teste e (iii) responder adequadamente a perguntas, quando necessário.

3. Avaliação

O grupo que cumprir os itens a seguir garantirá 7 pontos:

- entregar o relatório na data requerida;
- implementar corretamente a rede (sem erros de compilação ou execução);

- apresentar o trabalho na data requerida.

Relatórios contendo as informações adicionais descritas na Seção 4, garantirão os 3 pontos adicionais.

4. Descrição específica para cada modelo

- **Redes Multi-Layer Perceptron:** basicamente, é uma extensão do trabalho 1, sobre o qual o grupo deve implementar a capacidade da rede operar com mais de uma camada. Escolher uma base para treinamento e teste (pode ser a mesma base utilizada no trabalho 1). Para este modelo, o relatório deve conter, além das informações gerais mencionadas na Seção 2, uma tabela com valores de parâmetros utilizados (número de camadas, número de neurônios nas camadas, taxa de aprendizado) pela taxa de acerto obtida. Exemplo:

# Camadas Escondidas (# neurônios em cada camada)	Taxa de Aprendizado	Taxa de Acerto para o treinamento	Taxa de Acerto para os testes
1 (2)	0,25	95,00%	89,00%
2 (5, 2)	0,25	93,74%	81,43%
3 (5, 5, 5)	0,25	64,00%	58,00%
1 (2)	0,5	80,00%	72,11%
2 (5, 2)	0,5	66,00%	61,00%
3 (5, 5, 5)	0,5	58,00%	60,20%

As informações exigidas até aqui garantem 2 pontos. O grupo que mostrar que a rede MLP resolve o problema do XOR garante o ponto adicional.

- **Mapa de Kohonen:** rede auto-organizável que opera com aprendizado competitivo (*winner takes all*). A rede é estruturada em uma única camada, na qual os neurônios são (na forma mais simples) organizados em uma malha bidimensional (neurônios localmente conectados). O modelo de Kohonen pode ser utilizado para tarefas de classificação, tal como a rede MLP. A base de dados utilizada para o trabalho pode ser a mesma utilizada como exemplo visto em sala de aula (exemplo dos 4 vetores). Treinar a rede com os 4 vetores do exemplo e testar com os vetores complementares.

O relatório deve conter a visualização da malha, indicando a classificação dos exemplos em cada uma das etapas (treinamento e teste). Isto garante ao grupo 2 pontos. O ponto adicional vai para o grupo que classificar a base *Íris*. Um exemplo de como utilizar Kohonen com a base *Íris* pode ser encontrado em (texto em espanhol):

<http://halweb.uc3m.es/esp/Personal/personas/jmmarin/esp/DM/tema5dm.pdf>

- **Rede de Hopfield:** o modelo de Hopfield é um modelo de rede recorrente (dinâmico), utilizado, principalmente, em tarefas de reconhecimento de padrão, funcionando como uma memória associativa (aprendizado não-supervisionado). A rede é composta por uma única camada de neurônios globalmente conectados de maneira recorrente. O número de neurônios deve ser igual ao número de elementos de cada padrão a ser armazenado. Neste cenário, o treinamento (aprendizado Hebbiano) é realizado em um único ciclo, no qual os padrões escolhidos são armazenados como pontos fixos de um sistema dinâmico. A etapa de

teste (reconhecimento de padrão) consiste em fornecer à rede um padrão distorcido e verificar se esta recupera corretamente um dos padrões armazenados. O taxa de acerto é calculada como sendo o número de elementos iguais entre o padrão recuperado e o padrão esperado (*distância de Hamming* entre os padrões). Um exemplo de utilização do modelo é para reconhecimento de dígitos. A base a seguir pode ser utilizada no trabalho (matriz de 10x10 composta pelos caracteres '0' e ' '):

Padrões a serem armazenados:
(etapa de treinamento)

000	0000000000	00	00	0000000000
0000	0000000000	00	00	0000000000
00 00	00	00	00	00
00 00	00	00	00	00
00	0000000000	0000000000		00
00	0000000000	0000000000		00
00	00		00	00
00	00		00	00
00000000	0000000000	00		00
00000000	0000000000	00		00

Padrões a serem propagados:
(etapa de teste)

000	00000000		00	0000000000
0000	00000000		0000	0000000000
00		00	00 00	00
00		00	00 00	00
00	000000	0000000000		00
00	000000	0000000000		00
00	00		00	00
00	00		00	00
00	00000000	00		00
00	00000000	00		00

Valendo 2 pontos, o relatório deve apresentar os padrões recuperados (para os padrões propagados) e a taxa de erro/acerto para cada um deles. O grupo que treinar a rede Hopfield para reconhecimento da base de dígitos manuscritos da UCI (link abaixo), garante o ponto adicional.

<http://archive.ics.uci.edu/ml/datasets/Optical+Recognition+of+Handwritten+Digits>

5. Dicas

Informações adicionais e algoritmos para os três modelos aqui propostos (e demais outros) podem ser encontrados no artigo “*An Introduction To Computing With Neural Nets*” de Richard P. Lippmann:

<http://hawk.cs.csuci.edu/William.Wolfe/UCD/engineering/cse/Graduate/courses/CSC5542/Lippmann.pdf>