

Segundo Trabalho de Implementação Computacional

Rede "Backpropagation" e problema MNIST

1. Introdução

O(a) estudante deverá demonstrar conhecer a Rede Neural Artificial conhecida como "Backpropagation" (ou "Perceptron Multicamada"), aplicando-a na solução de um problema de reconhecimento de padrões.

Usaremos a base de dados "MNIST", disponível em <http://yann.lecun.com/exdb/mnist/> . Trata-se de um conjunto de imagens manuscritas dos dígitos de 0 a 9, em níveis de cinza. Cada imagem é um conjunto de 28x28 pixels em níveis de cinza de 0 (branco) a 255 (preto). As imagens foram centralizadas pelo método do centro de gravidade. Há 60000 imagens para treinamento e 10000 para teste. Outros detalhes do pré-processamento e dos formatos dos arquivos, bem como os arquivos propriamente ditos, podem ser buscados no sítio indicado acima.

2. Requisitos mínimos

a) Demonstrar um código computacional capaz de:

a1) Ler os arquivos de entrada (ou algum outro que tenha sido preparado a partir deste para facilitar a montagem da rede). Os dados propriamente ditos (quantidade e conteúdo das imagens) não deve ser modificado, para efeito de comparação, mas pré-processamento adicional dos dados é permitido.

a2) Treinar uma rede *Backpropagation* com uma camada escondida, usando algoritmo padrão.

a3) Testar a rede contra a saída desejada em um arquivo de teste, informando a taxa de erro (porcentagem de exemplos erradamente classificados).

b) Fazer uma análise dos resultados obtidos, verificando a evolução da taxa de erro (e o erro quadrático) no arquivo de treinamento, a taxa de erro no arquivo de teste, as dificuldades encontradas, as soluções propostas, os valores usados para os parâmetros de treinamento. O arquivo de teste não deve ser usado para nenhuma otimização da rede. Apenas para testar uma rede já treinada.

3. Requisitos adicionais (pontuação extra)

Pontos adicionais serão atribuídos para os trabalhos que envolvam um ou mais destas técnicas:

a) Uso de múltiplas camadas escondidas.

b) Uso de Entropia Cruzada como função de custo.

c) Uso de Regularização (L1, L2, Dropout).

d) Uso de saída em camada softmax (Regressão Logística) e custo log-likelihood.

e) Uso de sequências de camadas RBM com treinamento não-supervisionado (Contrast Divergence)

f) Teste com padrões adicionais (obtidos pelos autores).

3. Regras gerais e observações

1) Aceitam-se trabalhos individuais ou em dupla.

- 2) Não será pré-definida uma linguagem de programação, nem a estrutura da rede. Deve-se notar, porém, que uma rede demasiado pequena terá desempenho insuficiente.
- 3) Observe, entre as informações contidas no sítio indicado, o desempenho de MLPs com uma camada escondida situa-se na faixa de 4% de erro. Não se espera necessariamente alcançar, em um primeiro trabalho de implementação, desempenho igual ou superior a este, mas pode-se tomá-lo como um parâmetro de comparação.
- 4) O algoritmo deve ser implementado pelo estudante. Não se deve simplesmente utilizar um simulador ou outro software já disponível (incluindo o Toolbox do MATLAB). Podem ser utilizadas bibliotecas matemáticas para as operações necessárias (operações com matrizes, leituras de arquivos, etc.).
- 5) Apresentar o código, os resultados obtidos e a análise.
- 6) Limite de entrega: 15 de maio de 2019 23h55min, no ambiente Moodle. Haverá penalidade para entrega em atraso.