

Trabalho de IA - Redes Neurais: Multilayer Perceptron e Learning Vector Quantization

Sarajane M. Peres

16 de março de 2015

Material baseado em:

HAN, J. & KAMBER, M. Data Mining: Concepts and Techniques. 2nd. 2006

FAUSETT, L. Fundamentals of Neural Networks: Architectures, Algorithms, and Applications. 1994

Do objetivo e formação de grupos

Objetivo

Resolver a tarefa de classificação multi-classe em um conjunto de dados, usando duas redes neurais artificiais Multilayer Perceptron com Backpropagation e Learning Vector Quantization, e comparar e avaliar os resultados obtidos.

O trabalho deverá ser realizado em grupos de até **quatro** alunos (ou mesmo individualmente, contudo a realização do trabalho de forma individual não é recomendada, visto que parte do trabalho deverá estar apoiada em discussões e análises que são mais bem feitas dentro de uma discussão em grupo).

Das entregas e apresentações

Data de Entrega

A data de entrega do trabalho é: **18 de maio** (ou antes, se o grupo assim desejar) e não serão admitidos atrasos. O grupo recebe nota **zero** no trabalho se não o entregar na até a data limite.

Apresentação

No dia da entrega do trabalho, durante o horário de aula, a data e horário da apresentação de cada um dos grupos serão agendados.

Os artefatos produzidos no trabalho deverão ser entregues via e-Tidia - artefatos de implementação (códigos), conjuntos de dados que sofreram pré-processamento, a melhor rede neural MLP obtida, a melhor rede neural LVQ obtida, e o relatório (em .pdf) sobre o trabalho. Caso os artefatos gerem arquivos grandes, que não podem ser anexados via e-Tidia, o grupo deverá entregar estes artefatos em um CD/DVD, para o professor da disciplina, impreterivelmente até o horário de aula do dia 18 de maio. **Ou seja, deixar para anexar os artefatos do trabalho próximo ao horário final de entrega no e-Tidia não será uma boa estratégia.**

Da Avaliação

Processo de avaliação

Todos os alunos participarão das apresentações que serão realizadas apenas para a professora da disciplina. Todos os alunos devem estar cientes das motivações e justificativas para as tomadas de decisão realizadas pelo grupo, bem como deve ter um claro entendimento do código que implementa as redes neurais, o pré-processamento e os procedimentos de testes e avaliações. A nota final do aluno será uma composição da nota recebida pelo relatório do grupo e da nota recebida na apresentação.

Requisitos obrigatórios X mínimos X desejáveis

O trabalho está especificado sob a forma de requisitos que deverão ser atendidos pelos grupos. Os requisitos podem ser:

- obrigatórios: se não forem atendidos o trabalho não será considerado e o grupo receberá nota zero;
- mínimos: se todos forem atendidos e com a qualidade esperada, o grupo pode receber nota máxima no trabalho;
- desejáveis: permitem que os alunos recebam notas (ainda) mais altas na avaliação do trabalho, desde que já não tenham recebido nota máxima com a avaliação dos requisitos mínimos.

Do plágio

Se o plágio, de qualquer natureza, for constatado, o grupo não terá o seu trabalho avaliado.

Referências

Material de terceiros, disponíveis publicamente da Internet, provenientes de **homepages acadêmicas ou sites especializados no assunto**, poderão ser usados DESDE QUE esteja CLARAMENTE referenciados no relatório do grupo. Se corretamente referenciados, seu uso não se constitui em plágio.

Do conjunto de dados

Optical Recognition of Handwritten Digits Data Set

A tarefa consiste em classificar um dado como pertencente a uma de 10 classes. Cada dado representa um dígito manuscrito. Algumas características do conjunto de dados:

- número de instâncias: 3823 de treinamento e 1797 teste
- número de atributos: 65
- distribuição das classes: de 376 a 389 instâncias por classe no conjunto de treinamento; de 174 a 182 instâncias por classe no conjunto de teste.

Acesse o conjunto em:

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

- `optdigits.tra` ou `optdigits.tes`: dados de treinamento e de teste sugeridos pelo criador do conjunto de dados;
- `optdigits.names`: informações importantes sobre o conjunto de dados;

Obs.: O trabalho exigirá conjuntos de treino, validação e teste. O grupo deve adaptar o conjunto de dados de treino e teste sugerido pelo autor do conjunto para atender às especificações do trabalho.

Do pré-processamento dos dados

Um bom trabalho de classificação de dados exige que os dados sejam pré-processados antes de serem submetidos ao algoritmo de classificação. Assim, o grupo deve se preocupar em estudar algumas estratégias de pré-processamento apresentadas na bibliografia indicada, verificando os efeitos que tais estratégias provocam no resultado de classificação obtido pelos algoritmos.

Exemplos de estratégias

- formatação do conjunto em atributos descritivos atributo de classe: requisito obrigatório
- exclusão de atributos indesejáveis para a tarefa de classificação: requisito mínimo
- normalização: requisito mínimo
- seleção de atributos/dados: requisito desejável
- outros procedimentos a critério do grupo: requisito desejável

Dica ...

Esta base de dados já foi estudada por outras pessoas e existem artigos publicados relatando os experimentos. É uma boa estratégia ler alguns desses artigos para conhecer melhor as possibilidades de pré-processamento da base.

Sobre normalização

Procedimento para colocar os valores de um atributo dentro de um intervalo pequeno e específico, por exemplo $[-1.0;1.0]$ ou $[0.0;1.0]$. É útil para algoritmos de classificação ou agrupamento como redes neurais ou aqueles que usam medidas de distância.

Existem várias formas para normalizar os dados. Por exemplo:

- min-max: executa uma transformação linear; preserva os relacionamentos existentes entre os valores originais dos dados.
- z-score normalization: valores para um atributo são normalizados com base na média e no desvio padrão dos valores assumidos no atributo; é especialmente útil quando o valor mínimo e máximo não são conhecidos ou quando outliers estão presentes;

Para o trabalho, aplique os procedimentos de normalização a cada um dos atributos. (requisito mínimo)

Sobre normalização

Min Max

$$v' = \frac{v - \min_A}{\max_A - \min_A} (\text{newMax}_A - \text{newMin}_A) + \text{newMin}_A$$

Média Zero

$$v' = \frac{v - \bar{A}}{\sigma_A}$$

onde v é o valor assumido pelo atributo, \bar{A} é a média dos valores assumidos pelo atributo e σ_A é o desvio-padrão.

Da codificação

A codificação das rotinas necessárias no trabalho é de responsabilidade de cada grupo. Será permitido o código de terceiros, para qualquer uma das rotinas necessárias, porém o código deve ser comentado pelo grupo de forma a deixar claro que houve o estudo do mesmo e entendimento do mesmo. (requisito obrigatório)

Da proveniência de código de terceiros.

No caso do grupo optar pelo uso de códigos de terceiros, os mesmos deverão ser **NECESSARIAMENTE** obtidos a partir de homepages acadêmicas ou de sites especializados no assunto.

Se código de terceiros foram escolhidos para compor o trabalho, esses deverão se adequar totalmente aos requisitos do trabalho. Estratégias implementadas nesses códigos que fogem do escopo do trabalho não deverão ser usadas em detrimento das estratégias aqui requisitadas. (requisito obrigatório)

O uso de estratégias não requisitadas no trabalho pode ser apresentada, desde que sejam **ADICIONAIS** e estejam completamente entendidas por todos os membros dos grupos. (requisito desejável)

Linguagem de programação

Qualquer linguagem de programação ou ambiente de programação poderá ser usado pelo grupo, **DESDE QUE** a professora da disciplina tenha acesso (em sua máquina) ou que o grupo lhe permita acesso (em suas máquinas particulares). JAVA, Matlab e R são as possibilidades de acesso via máquina da professora. Outras linguagens ou ambientes a serem usados devem ser informados à professora **ANTES** de iniciar a confecção do trabalho.

Da codificação

Dos comentários

- O código deverá ser comentado de forma que seja possível identificar as funções/métodos/linhas de código que implementam as características da rede neural artificial bem como a estratégia de aprendizado utilizada. Cada uma das características de construção da arquitetura da rede (exemplo: estrutura de armazenamento dos neurônios da camada escondida, estrutura de dados de armazenamento de pesos, escolha de unidades de saída e suas classes etc ...) e de implementação do aprendizado (exemplo: fase feedforward, computação da ativação do neurônio, fase backpropagation, cálculo do erro, cálculo da informação de erro, alteração de pesos, alteração de bias, cálculo de distância etc ...), e uso de estratégias diferentes daquelas vistas em aula (por exemplo: uso de momentum, uso de taxa adaptativa, etc ...) devem estar explicitamente indicadas no código.

Da codificação

O grupo deverá preparar o seu código de forma que os seguintes parâmetros possam ser passados na linha de código de execução do programa:

- nome do arquivo do conjunto de dados de treino
- nome do arquivo do conjunto de dados de validação
- nome do arquivo do conjunto de dados de teste
- taxa de aprendizado inicial
- número de neurônios na cada escondida (para a rede MLP)
- número de neurônios para cada classe (para a rede LVQ)
- inicialização de pesos (zero/aleatória)
- ... (o que mais o grupo implementar em relação os requisitos mínimos e desejáveis)

Acompanhado de um arquivo readme.txt sobre quais são os parâmetros a mais.

Como resultado, a implementação deverá:

- mostrar na tela a progressão do treinamento (pode ser um contador de épocas ou um indicador do erro a cada época);
- gravar em arquivo texto as respostas da rede para o conjunto de teste, as medidas de avaliação e época de parada do treinamento;

Do uso de funções de ativação

O grupo deve atentar-se para o fato que no problema de classificação multiclasse, a saída da rede neural pode ser projetada de, pelo menos, duas formas:

- usando neurônios na camada de saída que compoem a resposta final da rede, usando por exemplo, respostas desejadas binárias ou bipolares
- usando neurônios na camada de saída com possibilidade de respostas em um intervalo onde divisões representam a resposta desejada.

O uso de mais de uma estratégia é **requisito desejável**.

Dica: para modelar a camada de saída, diferentes estratégias podem ser implementadas: usando uma função linear como função de ativação ou escalonando a função logística para extendê-la a intervalos de respostas maiores do que $[0,1]$ ou $[-1,1]$, se for o caso. Veja uma discussão sobre isso no livro da FAUSETT Seção 6.2.2.

Da métrica de distância

A métrica de distância mais comumente usada é a Distância Euclidiana. Seu uso será requisito mínimo.

O teste de outras métricas de distância será requisito desejável.

Da estratégia de teste do classificador e critério de parada do treinamento

Para testar um classificador, é necessário executar uma estratégia de teste. Várias são sugeridas na literatura: ressubstituição, holdout, cross-validation, leave-one-out, bootstrapping.

A estratégia HOLDOUT deverá ser usada neste trabalho como um **requisito mínimo**. Nesta estratégia, a divisão dos conjuntos deverão ser feitas usando 60% 20% e 20% dos dados, respectivamente para treino, validação e teste.

Neste trabalho, o número de épocas de treinamento deverá ser controlado pelo comportamento do erro da rede neural **requisito mínimo**, durante as épocas, usando o conjunto de validação e o erro quadrado total da rede (o erro quadrado para um dado é a soma dos erros – ao quadrado – obtidos em cada neurônio de saída da rede; o erro total da rede é a soma dos erros obtidos em uma época). O erro quadrado médio também pode ser usado. Veja as definições no Glossário do livro da FAUSETT.

Das medidas de avaliação do classificador

Matriz de confusão: tabulação cruzada entre as classes preditas pelo modelo e a classe real de cada exemplo. **Requisito mínimo** no trabalho.

		Classe Predita	
		C+	C-
Classe Real	C+	Verdadeiro Positivo (True Positive – TP)	Falso Negativo (False Negative – FN)
	C-	Falso Positivo (False Positive – FP)	Verdadeiro Negativo (True Negative – TN)

A matriz de confusão apresentada aqui está preparada para um classificador binário. Ela precisa ser estendida para que um problema multiclasse seja avaliado.

Das medidas de avaliação do classificador

Problema de classificação binária

Para avaliar um classificador binário, algumas medidas devem ser calculadas sobre os dados da matriz de confusão:

- sensibilidade ou taxa de verdadeiros positivos ou revocação: porcentagem de verdadeiros positivos dentre todos os exemplos cuja classe real é positiva $TPR = TP / (TP + FN)$
- taxa de falsos positivos: porcentagem de exemplo cuja classe real é negativa que são classificados como positivos $FPR = FP / (TN + FP)$
- especificidade ou taxa de verdadeiros negativos: proporção de rejeições corretas entre os exemplos cuja classe real é negativa $SPC = TN / (FP + TN)$
- precisão (precision ou preditividade positiva): proporção de acertos dentre todos os exemplos preditos como positivos: $PPV = TP / (TP + FP)$
- preditividade negativa: proporção de rejeições corretas dentre os exemplos preditos como negativos: $NPV = TN / (TN + FN)$
- taxa de falsas descobertas: denota o número de falsos positivos dentre os exemplos classificados como positivos: $FDR = FP / (TP + FP)$
- acurácia: quantidade (ou taxa) de exemplos classificados corretamente;
- erro: quantidade (ou taxa) de exemplos classificados incorretamente.

Para uso dessas medidas no trabalho, com exceção da acurácia e erro, estratégias como **oneXone** e **oneXall** deverão ser implementadas. (requisito desejável)

Das medidas de avaliação do classificador

F-Score

Considera medidas de precisão e revocação, ou seja, considera tanto a capacidade do classificador em reconhecer exemplos positivos dentre todos os exemplos positivos disponíveis quanto a capacidade de não considerar exemplos negativos como positivos.

$$F_score = \frac{(TPR * PPV)}{(TPR + PPV) / 2}$$

A discussão do F-Score é um **requisito desejável** no trabalho, desde que o tratamento do problema multiclasse como problemas de classificação binária deveria ser implementada.

Das medidas de avaliação do classificador

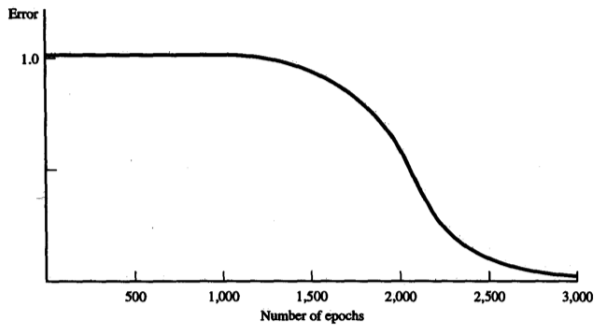
Problema de classificação multiclasse - Requisito mínimo

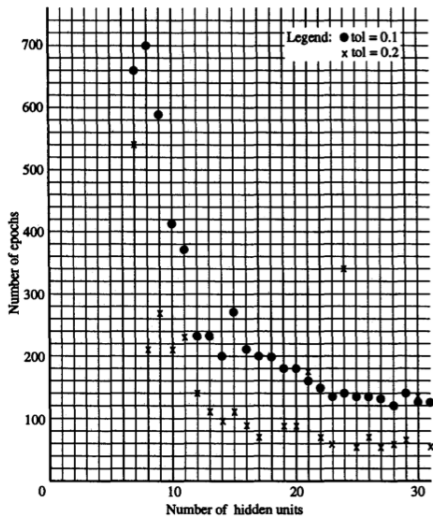
Para avaliar um classificador multiclasse, a matriz de confusão deve ser usada para entender quais são as dificuldades do classificador. Além disso, neste trabalho, a média e desvio padrão do número de instâncias classificadas corretamente, em várias execuções (pelo menos 10 por arquitetura e conjunto de parâmetros) do holdout, deverão ser apresentadas.

Ajudando a entender o treinamento e os resultados

Alguns gráficos podem ser gerados para ajudar no entendimento do comportamento dos parâmetros que estão sendo calibrados nos experimentos, no entendimento do comportamento do treinamento e no entendimento dos resultados obtidos. Esses gráficos são:

- comportamento do erro no tempo (durante as épocas), nos conjuntos de treinamento e de validação **requisito mínimo**;
- comportamento do erro com aumento de número de neurônios e aumento de épocas **requisito desejável**;
- ... **requisito desejável**.





Conteúdo do relatório

A organização do relatório deve seguir o especificado aqui. Informações a mais são muito bem vindas, porém devem estar graficamente destacadas no texto, com uso de uma fonte de letra menor e margens reduzidas. As informações a mais devem dizer respeito exclusivamente ao resultado do trabalho.

Formate seu relatório espaçamento $1\frac{1}{2}$ entre as linhas, margens justificadas e fonte 12 - Times ou Times New Roman.

Coloque no início de seu relatório os nomes completos dos alunos que formam o grupo de trabalho e seus respectivos números USP.

OBS.: Não é esperado que os alunos apresentem seções como: introdução, objetivos, teoria de redes neurais ou informações descritiva sobre os conjuntos de dados.

Conteúdo do relatório

SEÇÃO 1: Sobre a implementação

- (a) das redes neurais
- (b) dos procedimentos de pré-processamento
- (c) dos procedimentos de avaliação do classificador

Indique se as implementações para os quesitos (a), (b) e (c) usados em seu trabalho foram feitas pelo grupo, se são baseada em algum código de terceiros, se usam código de terceiros modificado ou se usam exatamente o código obtido de terceiros.

Se o código é de terceiros, indique a autoria do código e a fonte onde o código foi obtido.

Conteúdo do relatório

SEÇÃO 2: Sobre o pré-processamento

Descreva:

- tomadas de decisões sobre exclusão de atributos indesejáveis para a tarefa de classificação (se não houve, justifique);
- normalizações aplicadas: inclua análise sobre a eficiência do(s) método(s) aplicados;
- seleção de atributos/dados (se houver): explique quais procedimentos foram aplicados e por que;
- outros procedimentos: explique quais procedimentos foram aplicados e por que.

Se algum procedimento de pré-processamento foi aplicado segundo observado em algum artigo publicado, indique aqui os artigos utilizados.

Conteúdo do relatório

SEÇÃO 3: Divisão dos conjuntos

Explique qual foi a estratégia que você usou para dividir os conjuntos de dados em conjuntos de treino, validação e teste.

SEÇÃO 4: Descrição da sintonização de parâmetros

Descreva:

- descreva as diferentes arquiteturas de MLP (com as diferentes combinações de parâmetros) e a estratégia de sintonização adotada;
- descreva as diferentes arquiteturas de LVQ (com as diferentes combinações de parâmetros) e a estratégia de sintonização adotada;
- escolha arquiteturas onde você observou comportamentos interessantes e ilustrativos do comportamento da rede neural e comente-os através de texto e gráficos.
- escolha a melhor arquitetura e outras duas para comparação, indique-as e justifique sua escolha: essas arquiteturas deverão ser usadas na estratégia holdout e deverão fomentar as avaliações de classificadores via medidas de avaliação.

Conteúdo do relatório

OBS.1: Para mostrar as variações de parâmetros que definem as diferentes arquiteturas de redes neurais e também a estratégia de teste destes parâmetros, crie tabelas onde nas colunas são definidos os parâmetros que estão variando e nas linhas são definidas cada arquitetura de rede neural (os valores para os parâmetros).

O que sintonizar

- número de neurônios na camada escondida (no caso da MLP);
- número de neurônios por classe (no caso da LVQ);
- inicialização de pesos;
- inicialização da taxa de aprendizado e variação da mesma durante o treinamento;
- formatação da camada de saída;
- funções de ativação;
- número de épocas;
- ...

OBS.2: Lembre-se que a sintonização dos parâmetros deverá ser feita com conjuntos de treino e validação. Conjuntos de testes deverão ser usados para aferir a qualidade do modelo em termos de medidas de avaliação.

OBS.3: Cuidado com as medidas escolhidas para avaliar os classificadores. Eventualmente alguma medida analisada de forma isolada pode lhe passar uma impressão errada sobre o classificador. Estude o significado de cada medida.

Conteúdo do relatório

Seção 5: Avaliação dos classificadores

Para cada uma das arquiteturas escolhidas na fase de sintonização, execute o teste e:

- mostre as medidas de avaliação organizadas em tabelas;
- mostre os gráficos de convergência do erro;
- analise os resultados obtidos e teça seus comentários sobre a qualidade dos classificadores obtidos

Trabalho de IA - Redes Neurais: Multilayer Perceptron e Learning Vector Quantization

● Sarajane M. Peres - sarajane@usp.br

Disciplina de Inteligência Artificial
Graduação em Sistemas de Informação - SI
Escola de Artes, Ciências e Humanidades - EACH
Universidade de São Paulo - USP