Projeto da Disciplina

Algoritmos e Estruturas de Dados – BCC

Prof. Luciano Demétrio Santos Pacífico
{ldsp@deinfo.ufrpe.br}

Conteúdo

- **Disposições Gerais**
- **Tema**
- **Etapas do Projeto**
- **Avaliação**

Disposições Gerais

UNIVERSIDADE
FEDERAL RURAL
DE PERNAMBUCO

Objetivos

- O projeto da disciplina de Algoritmos e Estruturas de Dados tem por objetivos:
  - Oferecer aos alunos a oportunidade do desenvolvimento de um sistema computacional para a solução de um problema real;
  - Consolidar os conceitos teóricos e práticos abordados durante o decorrer do curso;
  - Incentivar o aluno ao estudo mais aprofundado das estruturas de dados e algoritmos vistos, de forma a adaptá-las ao problema em análise;

Objetivos

- O projeto da disciplina de Algoritmos e Estruturas de Dados tem por objetivos: (Cont.)
  - Incentivar o aluno a buscar conhecimentos adicionais através da atividade de pesquisa e revisão da literatura sobre problema abordado;
  - Apresentar ao aluno diversas áreas da ciência da computação para o auxílio na escolha de seu perfil acadêmico;
  - Fomentar as atividades de escrita e apresentação de trabalhos por parte dos alunos.

Objetivos

- Ao fim do projeto, espera-se que os alunos estejam aptos a:
  - Fazer o mapeamento de problemas reais às estruturas de dados e algoritmos vistos, de forma a solucionar tais problemas;
  - Escreverem relatórios técnicos descrevendo os sistemas computacionais desenvolvidos;
  - Apresentarem os sistemas desenvolvidos ao público de modo geral.

Observações

- O projeto será desenvolvido por equipes de até dois alunos.
- Um membro de cada equipe será responsável pelo envio do nome da dupla ao e-mail ldsp.ufrpe@gmail.com.
- As bases de dados necessárias para o projeto serão alocadas por ordem de envio de e-mail.
- **Data limite para a alocação: 02-12-2014**.

Observações

- Da mesma forma que as listas de exercícios, as esquipes devem desenvolver suas estruturas de dados e algoritmos, sendo proibido o uso de estruturas de dados e algoritmos prontos em pacotes da linguagem escolhida.
- O projeto deve ser desenvolvido em uma das seguintes linguagens de programação: C, C++ ou JAVA.
- Cada equipe deve desenvolver seu código, sendo a **Regra de Ouro** aplicada também aos projetos.

Tema

UNIVERSIDADE
FEDERAL RURAL
DE PERNAMBUCO

Tema

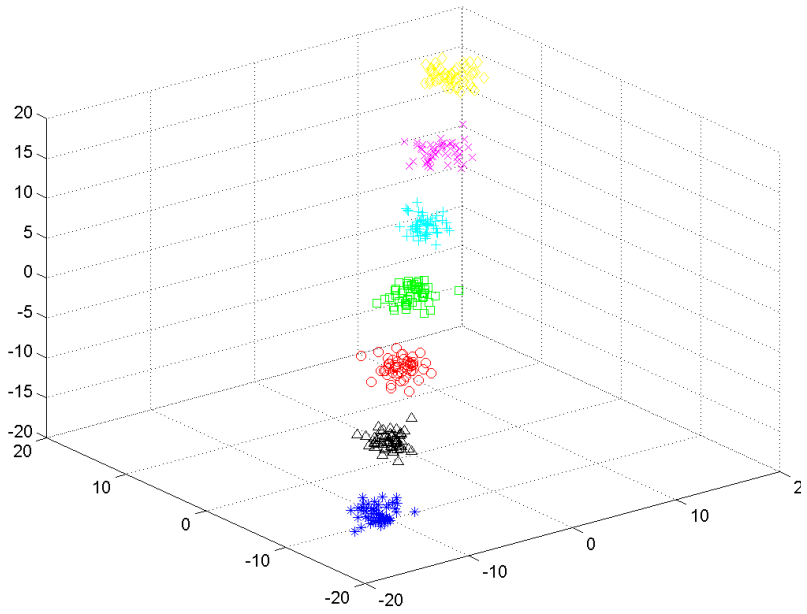
- Neste período, o tema do projeto será único para todas as equipes.
- O tema será Análise de Agrupamentos através de Algoritmos de Particionamento.
- Os algoritmos adotados serão o *Hard K-Means* [1] e o *Fuzzy C-Means* [2].
- Cada equipe deverá usar os algoritmos acima para a realização da tarefa de agrupamento tanto com bases de dados benchmark reais quanto com base de dados sintéticas.

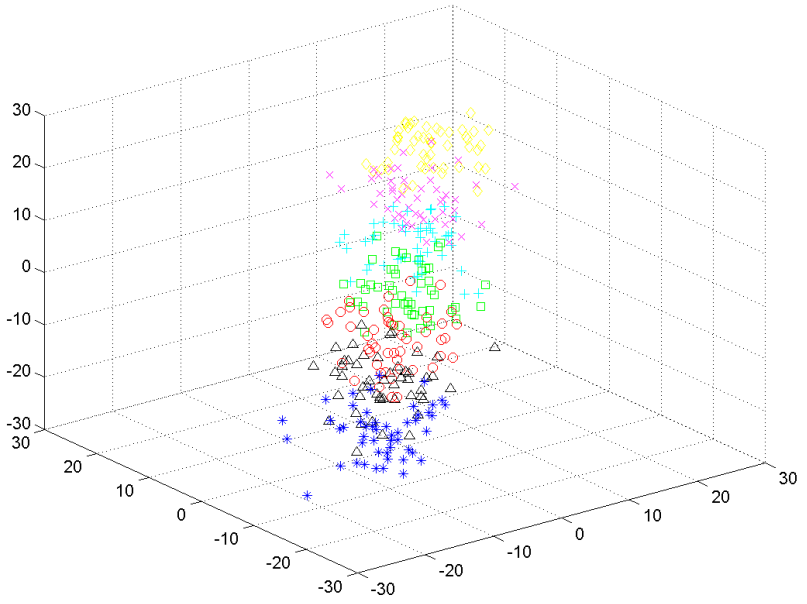
Análise de Agrupamentos

- Problema: Análise de Agrupamentos

- Dada uma base de dados qualquer, o algoritmo de agrupamento deve encontrar uma divisão do conjunto de dados em grupos, de tal forma que cada grupo contenha padrões que sejam mais semelhantes entre si, e grupos diferentes possuam objetos que sejam o mais diferentes entre si.

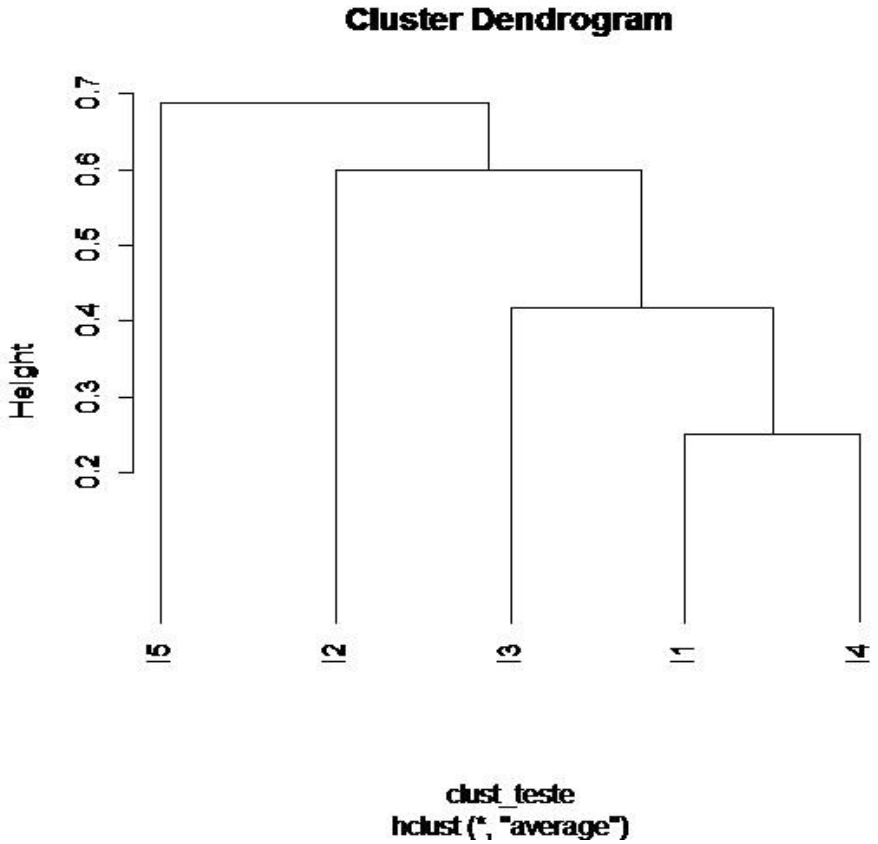
Análise de Agrupamentos





Análise de Agrupamentos

- Exemplo:



Análise de Agrupamentos

- Detalhes do Sistema:

- O sistema deve ler um arquivo contendo uma base de dados e ser capaz de formar grupos homogêneos entre si, a partir de um algoritmo de agrupamento de particionamento.

Hard K-Means

- O Hard K-Means é um dos mais populares algoritmos de agrupamento de particionamento.
- Sua popularidade é dada por seu fácil entendimento e implementação.
- Nesse algoritmo, um conjunto de dados $\boldsymbol{P}$ contendo n padrões m-dimensionais $(x_i \in \mathbb{R}^m, i = 1, 2, \ldots, n)$ é dividido em C grupos de acordo com suas relações de similaridade.

Hard K-Means

- No K-Means, a cada iteração t, um padrão x_i é associado a um único grupo k, sendo tal grupo o mais próximo a x_i.
- A medida de dissimilaridade adotada o quadrado da Distância Euclidiana.

$$d(x_i, g_k)^2 = \sum_{j=1}^{m} (x_{ij} - g_{kj})^2$$

- Assim, o padrão x_i pertencerá ao grupo k tal que:

$$k = \min_{1 \leq k \leq C} d(x_i, g_k)^2$$

Hard K-Means

- Para cada grupo k $(k \in C)$, um representante $\mathbf{g}_k$ $(g_k \in \mathbb{R}^m)$ é calculado como o ponto médio entre todos os padrões atualmente associados ao mesmo.

$$g_k = \frac{1}{n_k} \sum_{i \in k} x_i$$

onde n_k representa o número de padrões associados ao grupo k.

Hard K-Means

- Algoritmos de particionamento realizam a tarefa de agrupamento visando a minimização de uma função critério, que serve como medida para a indicação do quão boa a solução apresentada pelo algoritmo é.
- A função objetivo do K-Means é dada abaixo:

$$J(P, C) = \sum_{k=1}^{C} \sum_{i \in k} d(x_i, g_k)^2$$

Hard K-Means

- A etapa de inicialização do algoritmo K-Means pode ser executada de duas formas:
  - Centro Aleatório: *C* indivíduos distintos da base de dados são escolhidos como os primeiros centros de grupo. Os demais padrões são associados aos grupos mais próximos aos mesmos;
  - Afetação Aleatória: cada padrão é associado a um grupo aleatoriamente.

Hard K-Means

- O K-Means será executado durante um número máximo *maxIt* de iterações.
- A cada iteração do algoritmo, duas etapas são realizadas:
  - Afetação: todos os padrões da base de dados são associados ao grupo mais próximo ao mesmo;
  - Determinação dos novos centros de grupo: os novos centros de grupo são calculados.

Hard K-Means

procedimento *k_means*(*X*, *maxIt*, *C*)

$t \leftarrow 0$;

Inicialização: Centro Aleatório ou Afetação Aleatória;

enquanto $t \leq maxIt$ **faça**

$t \leftarrow t + 1$;

Determine os novos centros de grupo;

Associe os padrões aos novos grupos;

se nenhum padrão mudou de grupo **então**

pare;

fim_se

fim_enquanto

fim_k_means

Fuzzy C-Means

- A principal diferença entre o Fuzzy C-Means e o Hard K-Means está no fato de que no K-Means, em uma determinada iteração, um padrão da base de dados é associado a um único grupo k, enquanto no Fuzzy C-Means cada padrão é associado a todos os grupos de acordo com um grau de pertinência μ, tal que:
- $0 \leq \mu_k \leq 1, k = 1, 2, \ldots, C$
- $\sum_{k=1}^{C} \mu_k = 1$

Fuzzy C-Means

- A pertinência de um padrão x_i ao grupo k é dada por:

$$u_{ik} = \begin{cases} 1, & se\ d(x_i, g_k)^2 = 0 \\ 0, & se\ d(x_i, g_k)^2 \neq 0\ e\ \exists j \neq k, d(x_i, g_j)^2 = 0 \\ \dfrac{1}{\sum_{j=1}^{C} \left(\dfrac{d(x_i, g_k)^2}{d(x_i, g_j)^2}\right)^{\frac{2}{l-1}}}, & caso\ contrário \end{cases}$$

onde $l > 1$ é uma constante real (geralmente, $l = 2$) chamada fuzzifier.

Fuzzy C-Means

- Os centros de grupo são determinados de acordo com a equação abaixo:

$$g_k = \frac{\sum_{i=1}^{n} \mu_{ik}^{l} x_i}{\sum_{i=1}^{n} \mu_{ik}^{l}}$$

Fuzzy C-Means

- O critério de parada para o Fuzzy C-Means é dado abaixo:

$$\max_{ik}\left\{\left|\mu_{ik}^{t}-\mu_{ik}^{t-1}\right|\right\}<\varepsilon$$

onde ε é uma constante real positiva com valor baixo (exemplo, $\varepsilon = 10^{-5}$).

- O algoritmo Fuzzy C-Means minimizará a seguinte função critério:

$$W(P,C)=\sum_{k=1}^{C}\sum_{i=1}^{n}\mu_{ik}^{l}d(x_i,g_k)^2$$

Fuzzy C-Means

procedimento *fuzzy_c_means*(X, *maxIt*, *C*, *l*, ε)

$t \leftarrow 0$;

Inicialização: Centro Aleatório ou Afetação Aleatória para a matriz μ_{ik}^{0};

enquanto $t \leq maxIt$ **faça**

$t \leftarrow t + 1$;

Determine os novos centros de grupo;

Atualize a matriz μ_{ik}^{t};

se $\max_{ik}\left\{\left|\mu_{ik}^{t} - \mu_{ik}^{t-1}\right|\right\} < \varepsilon$ **então**

pare;

fim_se

fim_enquanto

fim_fuzzy_c_means

Etapas do Projeto

UNIVERSIDADE
FEDERAL RURAL
DE PERNAMBUCO

Etapas do Projeto

- Inicialmente, deve-se enviar o nome dos membros da equipe para o professor.

- Após o envio, as bases de dados que serão usadas por cada equipe serão enviadas.

- Munidos das bases de dados, a equipe poderá passar para a etapa de experimentação.

Experimentos

- Cada base de dados (tanto as reais quanto as sintéticas) serão bases de problemas de classificação.
- Haverá em cada base um atributo indicando qual a classe real a qual cada padrão da base pertence.
- A comparação será realizada em relação às classes reais de cada padrão e o grupo ao qual cada padrão foi associado pelo algoritmo de agrupamento utilizado.

Experimentos

- Para cada base de dados, experimentos deverão ser feito com as quatro variações possíveis dos métodos adotados:
  - Hard K-Means com Afetação Aleatória;
  - Hard K-Means com Centro Aleatório;
  - Fuzzy C-Means com Afetação Aleatória;
  - Fuzzy C-Means com Centro Aleatório.

Experimentos

- Cada algoritmo deve executar por até 100 iterações.
- Para cada algoritmo, 50 experimentos independentes deverão ser executados.
- A análise dos resultados será baseada na média e no desvio padrão da Taxa de Erro Global de Classificação (TEGC) que cada algoritmo obteve durante as 50 execuções.

Experimentos

- A matriz de confusão deverá ser impressa no arquivo de saída dos experimentos.
- Ex.: Base de dados Iris [3].

```
Confusion Matrix:
   Cluster:        1        2        3
Class:  1:        50        0        0
Class:  2:         0       47        3
Class:  3:         0       14       36
```

- Para o Fuzzy C-Means, associe cada padrão ao grupo pelo qual o mesmo apresentou maior grau de pertinência.

Experimentos

- Para o cálculo da Taxa de Erro Global de Classificação, deve-se considerar que cada grupo será associado à classe com mais representantes no grupo (voto majoritário), sendo os padrões das outras classes que pertencem ao grupo considerados erros de classificação.
- A Taxa de Erro Global de Classificação será dada então pela seguinte equação:

$$TEGC = \frac{n_e}{n}$$

onde n_e é o número total de padrões rotulados como erro.

Experimentos

- Deve-se comparar, dentre as quatro variações dos algoritmos a serem testados, quais os que obtiveram os melhores desempenhos para cada uma das bases de dados.
- Métricas:

$$\overline{TEGC} = \frac{\sum_{j=1}^{rep} TEGC_j}{n_{rep}}$$

$$Std = \sqrt{\left(\sum_{j=1}^{n_{rep}} \left(TEGC_j - \overline{TEGC}\right)^2\right) / (n_{rep} - 1)}$$

Avaliação

Avaliação

- A avaliação levará os seguintes pontos em consideração:

  - Corretude da solução proposta;
  - Eficiência da solução proposta;
  - Domínio do conteúdo apresentado por cada membro da equipe;
  - Apresentação do sistema.

Avaliação

- A nota será atribuída aos seguintes fatores:
  - Código desenvolvido;
  - Relatório técnico (artigo) descrevendo o sistema apresentado ao professor;
  - Apresentação em sala de aula do projeto.
  - Participação nas atividades de acompanhamento e apresentações dos outros grupos.
- A nota será atribuída a equipe como um todo, devendo cada membro estar apto a representar o grupo em cada uma das etapas da avaliação.

Artigo

- O relatório do projeto deverá ser escrito em formato de artigo, de acordo com o template da SBC [4] ou do IEEE [5].
- O artigo deve conter ao menos os seguintes tópicos:
  - Introdução;
  - Estado da Arte;
  - Metodologia;
  - Experimentos;
  - Conclusões;
  - Referências.

Avaliação

- No dia da apresentação, cada equipe deverá trazer seu computador para evitar problemas.
- O projeto compõe 60% da nota da segunda verificação de aprendizagem (2ª VA).

- **Data limite da entrega do artigo, código e apresentação do projeto: 04-01-2015**

Dúvidas?

Referências

[1]J. MacQueen et al., “Some methods for classification and analysis of multivariate observations,” in Proceedings of the fifth Berkeley symposium on mathematical statistics and probability, vol. 1, no. 281-297. California, USA, 1967, p. 14.

[2]J. C. Bezdek, *Pattern Recognition with Fuzzy Objective Function Algorithms*. New York: Plenum Press, 1981.

[3]A. Frank and A. Asuncion, “UCI Machine Learning Repository”, Univ. California, Sch. Inform. Comput. Sci., Irvine, CA, 2014 [Online]. Available: http://archive.ics.uci.edu/ml.

[4]Template artigos SBC. Disponível em: http://www.sbc.org.br/index.php?option=com_jdownloads&Itemid=0&task=view.download&catid=32&cid=38. Acesso: 24-11-2014.

[5]Template artigos IEEE. Disponível em: http://www.ieee.org/conferences_events/conferences/publishing/templates.html. Acesso em 24-11-2014.