Relatório de trabalho prático
Métodos Estatísticos
Geração de dendrograma

João Paulo Pizani Flor
Mauricio Oliveira Haensch

6 de junho de 2011

1 Introdução

Para o módulo de métodos estatísticos da disciplina, o primeiro trabalho prático tem como objetivo implementar a geração de um dendrogama com base num conjunto de dados, além de utilizar o dendrograma gerado para separar os padrões de entrada num número arbitrário de classes. A descrição do trabalho pode ser vista a seguir:

Implementar o Método de Unificação ou Agrupamento em Árvore (dendrograma). O sistema deve ser capaz de ler um conjunto qualquer de dados em formato texto, por exemplo separado por *tabs*. Deve possuir interface gráfica que apresente o dendrograma gerado. Bole um "analisador de dendrograma" que encontre o local ótimo de corte, dados dois limites: *maxClass* e *minClass*. Para testes tome um conjunto de quatro *sets* de dados:

- (2) Os dados da flor do Gênero Iris e dos carros disponíveis na página;
- (1) Os dados de câncer cerebral (gliomas): http://www.inf.ufsc.br/ patrec/glioma-daumas-duport.xls
- (1) Outro conjunto qualquer (por exemplo, veja os "Links Úteis")

2 Ferramentas utilizadas

A linguagem utilizada para implementação do programa era de livre escolha dos alunos, assim como outras ferramentas que pudessem ser utilizadas para representação gráfica dos padrões, por exemplo. As ferramentas escolhidas para a implementação do trabalho foram:

Linguagem de programação - Python: Escolhida por ser uma linguagem já bem conhecida pela equipe, que está sendo utilizada para todos os trabalhos práticos até então. Conta com boas bibliotecas e documentação, além de agilizar o desenvolvimento.

Interface gráfica - PyQt: *Binding* do framework gráfico Qt para a linguagem Python, escolhida por já ser conhecida pelos integrantes e também é o framework gráfico sendo utilizado nos demais trabalhos da disciplina.

Pydot: *Binding* para a linguagem Python do software de visualização de grafos GraphViz. A representação gráfica do dendrograma gerado é feita através dessa ferramenta.

O trabalho foi desenvolvido no ambiente Linux/Ubuntu e para executá-lo são necessários alguns pacotes[1] para a interface gráfica:

- pyqt4-dev-tools
- libqtgui4
- libqtcore4
- python-pydot

[1] os nomes dos pacotes citados são específicos para a distribuição Ubuntu, para outras distribuições devem ser procurados os pacotes correspondentes.

3 Alguns resultados

A interface para este trabalho ficou simples, dado o número pequeno de funcionalidade exigidas. É possível carregar um conjunto de dados separados por vírgulas (.csv) para que seja analisado, gerar o dendrograma, que é exibido em uma janela separada, e encontrar classes para os dados analisados com base no dendrograma gerado e no número de classes desejada (passado por parâmetros como um intervalo). O resultado da classificação das entradas é apresentado em forma de texto.

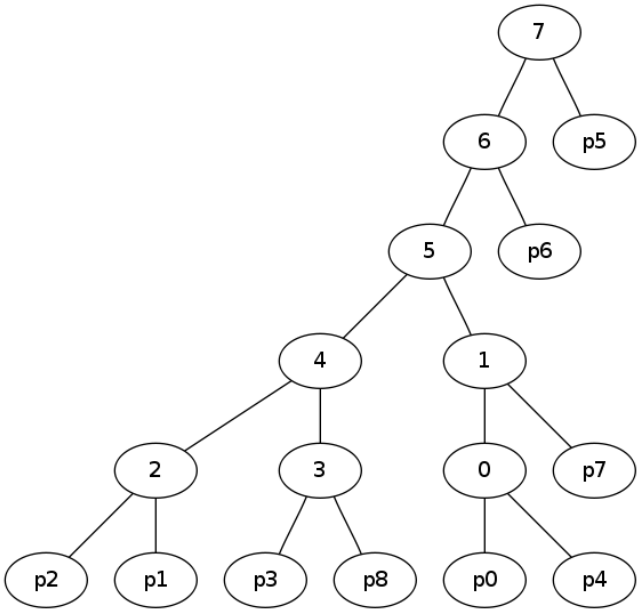
3.1 Leitura de dados de entrada

Os dados de entrada a serem carregados devem seguir o padrão a seguir, com um caso por linha, com os valores dos atributos podendo ser inteiros ou ponto flutuante:

```
atributo1,atributo2,atributo3,atributo4
atributo1,atributo2,atributo3,atributo4
atributo1,atributo2,atributo3,atributo4
atributo1,atributo2,atributo3,atributo4
```

3.2 Geração de dendrograma

Para demonstrar o resultado de nossa implementação utilizaremos um caso de uso simples. Na imagem abaixo é exibido o dendrograma gerado com um subconjunto dos casos da flor de gênero Iris. A árvore foi gerada com o algoritmo de distância simples (menor distância) e a representação gráfica foi feita com a biblioteca *pydot*.



Um dendrograma gerado para um subconjunto pequeno de casos da flor de gênero Iris.

Os nodos folha são nomeados seguindo o padrão *"p"+ número* (como *p0*, *p1*, *p2*), e os nodos que unem as folhas da árvore foram nomeados apenas com números em ordem crescente, indicando a ordem em que foram feitas as junções. Por exemplo, conforme a figura demonstrada, a primeira junção feita (nodo "0") uniu as folhas p0 e p4, indicando que os casos de entrada representados por esses nodos eram os mais próximos segundo a métrica utilizada, distância euclidiana. Um dendrograma mais complexo está anexo ao relatório, baseado no arquivo inteiro das flores de gênero Iris (150 casos).

3.3 Agrupamento em classes

A partir do dendrograma gerado é possível que o usuário escolha o número mínimo e máximo de classes a serem encontradas pelo algoritmo (que podem eventualmente ter o mesmo valor para que seja encontrado um número específico de classes). A saída gerada para a aplicação do algoritmo de análise do dendrograma gerado pelo caso anterior, com o subconjunto dos casos da flor Iris, pode ser visto a seguir. Para este caso, foram utilizados como número mínimo de classes a serem encontradas 4, e o máximo 7.

```
1.788854382,1.90065778087,2.39791576166,2.5,0
-0.894427191,0.22360679775,-0.417028828114,1.0,1
-0.894427191,-0.782623792125,0.521286035143,-0.5,2
0.111803398875,-1.11803398875,-0.417028828114,-0.5,2
-0.559016994375,-0.4472135955,-1.35534369137,-0.5,2
-1.56524758425,-1.45344418537,-0.417028828114,-0.5,2
0.782623792125,0.559016994375,-0.417028828114,-0.5,3
0.4472135955,0.894427191,-0.417028828114,-0.5,3
0.782623792125,0.22360679775,0.521286035143,-0.5.3
```

Conforme o exemplo acima, na saída gerada após o agrupamento dos vetores os atributos iniciais correspondem aos valores estandardizados dos vetores de entrada, enquanto o último valor separado por vírgula corresponde à classe designada àquela entrada específica.

4 Referências

- Python - http://www.python.org/
- Qt - http://doc.qt.nokia.com/
- pyQt - http://www.riverbankcomputing.co.uk/software/pyqt/intro
- Pydot - http://code.google.com/p/pydot/