

A identificação de grupos de aprendizes no ensino presencial utilizando técnicas de clusterização

Edson P. Pimentel^{1,2}, Vilma F. de França¹, Nizam Omar^{1,3}

¹Instituto Tecnológico da Aeronáutica – (ITA)
Praça Mal. Eduardo Gomes, 50 – 12228-900 – São José dos Campos – SP – Brasil

²Centro Universitário Municipal de São Caetano do Sul (IMES)
Av. Goiás, 3400 – 09550-051 – São Caetano do Sul– SP – Brasil

³Universidade Presbiteriana Mackenzie
Rua da Consolação, 930 – 01302-907 – São Paulo – SP – Brasil

epiment@imes.edu.br, vilmaff@comp.ita.br, omar@mackenzie.com.br

Resumo. A utilização de pedagogias diferenciadas que considerem a heterogeneidade dos alunos de uma mesma sala-de-aula é, com certeza, uma ferramenta capaz de resgatar aprendizes condenados à não-aprendizagem e ao fracasso profissional. No entanto, aulas personalizadas no ensino presencial é algo ineficaz, além de impraticável. O "atendimento personalizado" em grupos homogêneos é uma possibilidade a se considerar. Este artigo descreve uma experiência de categorização de alunos utilizando técnicas de clusterização por aprendizado não-supervisionado com dados obtidos através de questionários e avaliações. Com isto, foi possível identificar grupos similares de aprendizes. Esta é mais uma técnica que poderá ser utilizada para criar e manter o modelo do estudante num Sistema Tutor Inteligente.

Palavras-chave: Ensino-Aprendizagem, Clusterização, Formação de grupos, Modelo do Estudante

Abstract. The use of a different pedagogy that consider the students' heterogeneity in the same classroom is, certainty, a useful tool to rescue learners that was convicted to not learning and to a professional weakness. However, personalized lessons in classrooms are inefficient and impracticable. The "personalized attendance" in homogeneous groups is a possibility to take into consideration. This paper describes an experience of students' categorization using clustering techniques for unsupervised learning with data obtained from questionnaires and assessments. In this way, it was possible to identify similar groups of learners. This is one more technique that could be used to create and maintain the student model in an Intelligent Tutorial System.

Key words: Teaching and Learning, Clustering, Groups Formation, Student Model

1. Introdução

A realidade do ensino presencial nas universidades brasileiras coloca numa mesma sala-de-aula uma grande quantidade de alunos com habilidades e conhecimentos heterogêneos. Desta forma, a mesma aula é dada para quem sabe muito, pouco ou nada sobre determinado tópico. Isto gera um círculo de injustiças que condena muitos alunos à *não-aprendizagem*, gerando sucessivas reprovações e/ou colocando no mercado de trabalho profissionais sem uma formação consistente.

Segundo David Ausubel, o fator mais importante influenciando a aprendizagem é aquilo que o aprendiz já sabe [Moreira 2001]. É necessário, então determinar continuamente o que o aluno conhece e ensiná-lo de acordo, colocando-o numa situação de aprendizagem ótima para ele, em sua zona de desenvolvimento proximal, ou seja, considerando a distância entre o nível real de desenvolvimento e o nível de desenvolvimento potencial, conforme Vygostky em [Baquero 1998], priorizando aqueles que têm mais a aprender.

Perrenoud afirma que é preciso diferenciar, rompendo com a pedagogia frontal que impõe a mesma lição, os mesmos exercícios para todos [Perrenoud 2000]. No entanto, a personalização do ensino ou “educação sob medida” no ambiente presencial, geralmente composto de grandes grupos, é muito difícil. Isto parece mais indicado para a modalidade de ensino não-presencial possivelmente apoiado por Sistemas Tutores Inteligentes (STI) em que isto pode ser resolvido de diferentes maneiras [Stankov 1996].

Quanto ao ensino presencial, ainda predominante em relação ao ensino à distância, o problema parecerá insolúvel enquanto se imaginar que, para criar uma situação de aprendizagem ótima para cada aluno sejam necessárias “aulas personalizadas” ou aulas particulares. Este modelo não seria possível nem mesmo com a diminuição do efetivo das turmas.

Uma proposta para enfrentar este dilema entre um ensino presencial ineficaz e um ensino individualizado impraticável está no uso de uma pedagogia diferenciada, criando dispositivos múltiplos, não baseando tudo na intervenção do professor. A organização diferenciada do ensino poderia contar, por exemplo, com a criação de novos espaços-tempos de formação, reais ou virtuais, jogando com os reagrupamentos e fazendo uso mais adequado das tecnologias educacionais. É claro que não existem fórmulas mágicas. Se isso fosse possível, os especialistas apresentariam pedagogias diferenciadas e “prontas para uso” [Perrenoud 2000].

Caminhando nesta direção, este trabalho propõe maneiras possíveis para “personalizar” o ensino presencial. Uma idéia é que este “atendimento personalizado” aconteça não de forma individualizada, mas em grupos homogêneos compostos por alunos que possuam, mais ou menos, as mesmas lacunas de aprendizagem. A primeira tentação seria separar fisicamente estes alunos em salas diferentes, formando grupos homogêneos, através de um diagnóstico prévio. É sabido, porém que no decorrer de um espaço de tempo não muito longo, a heterogeneidade se recria. Portanto, propõe-se caminho alternativo. A base da proposta é identificar grupos similares, dentro de uma turma normal de sala-de-aula, não com o intuito de separá-los fisicamente, mas para que o professor possa fazer uso de uma pedagogia diferenciada adequada.

A identificação destes grupos é uma tarefa relativamente árdua para o ser humano, uma vez que pode envolver grande volume de informações, em várias dimensões além destes grupos serem móveis. Modelos matemáticos e estatísticos encontram-se à disposição para ajudar nesta tarefa. Além disso, algoritmos para reconhecimento de padrões, com ênfase em técnicas de clusterização (agrupamento) podem ser utilizados para este fim. Em [Jain et al 1999] são apresentadas diferentes abordagens de clusterização, tais como: algoritmos de hierarquização, de particionamento (K-means), vizinho mais próximo, algoritmos *fuzzy*, algoritmos de redes neurais artificiais (Self-Organizing Maps), algoritmos evolucionários, dentre outros.

Este trabalho apresenta os resultados da aplicação de duas técnicas de clusterização: K-means e Self-Organizing Maps, na tentativa de classificar estudantes em grupos com características similares, a partir de um questionário inicial que levantou o grau de confiança de cada aprendiz nos tópicos de uma disciplina de Programação de Computadores.

O artigo está organizado como segue. A seção 2 apresenta as duas técnicas de clusterização. A seção 3 descreve a metodologia empregada no estudo de caso. Na seção 4 são apresentados os resultados da aplicação da metodologia. Na seção 5 são feitas algumas considerações acerca dos resultados obtidos com este trabalho e são apresentadas propostas de aprofundamentos necessários.

2. Clusterização

Tendo em vista que o objetivo deste trabalho é identificar grupos de alunos com características similares, fez-se necessário pesquisar técnicas de agrupamento (*clustering*) que explorem semelhanças entre padrões e agrupem os padrões parecidos em categorias ou grupos.

Classificar ou agrupar objetos em categorias é atividade bastante comum e vem sendo intensificada devido ao número elevado de informações que estão disponíveis atualmente [Backer, 1995]. Para realizar esta tarefa emprega-se um mecanismo denominado análise de *cluster* ou clusterização.

De acordo com [Jain and Dubes 1988], a clusterização é um método que utiliza o aprendizado não supervisionado ou auto-organizável, ou seja, não há um “professor” ou “crítico” que lhe indique o que cada padrão representa. Basicamente, a previsão será feita através da representação interna montada pelo sistema utilizando um conjunto de dados de entrada.

A seguir serão descritas as duas técnicas de clusterização utilizadas neste trabalho.

2.1. K-means

K-means é uma técnica que usa o algoritmo de agrupamento de dados por K-médias (*K-means clustering*). O objetivo deste algoritmo é encontrar a melhor divisão de P dados em K grupos $C_i, i = 1, \dots, K$, de maneira que a distância total entre os dados de um grupo e o seu respectivo centro, somada por todos os grupos, seja minimizada.

Este método consiste em usar os valores dos primeiros n casos em um arquivo de dados, como estimativas temporárias das médias dos k *clusters*, onde k é o número de *clusters* especificado pelo usuário. Assim, o centro do *cluster* inicial é formado para cada caso em torno dos dados mais próximos e, então, comparados com os pontos mais distantes e os outros *clusters* formados. A partir daí, dentro de um processo de atualização contínua e de um processo iterativo encontram-se os centros dos *clusters* finais [Monteiro e Silva et al 2001].

Em outras palavras, o algoritmo atribui aleatoriamente os P pontos a K grupos e calcula as médias dos vetores de cada grupo. Em seguida, cada ponto é deslocado para o grupo correspondente ao vetor médio do qual ele está mais próximo. Com este novo rearranjo dos pontos em K grupos, novos vetores médios são calculados. O processo de re-alocação de pontos a novos grupos cujos vetores médios são os mais próximos deles continua até que se chegue a uma situação em que todos os pontos já estejam nos grupos dos seus vetores médios mais próximos.

2.2. Self-organizing Maps (Mapas Auto-Organizáveis)

As redes SOM (*Self-organizing Maps*) são redes neurais artificiais que possuem a capacidade de auto-organização [Kohonen 1990]. Utilizam o paradigma do aprendizado não-supervisionado e são formadas por uma camada de entrada (fonte) e uma camada de saída (representação).

Quando um padrão de entrada P é apresentado, a rede SOM procura por uma unidade mais semelhante a P . Durante o treinamento, a rede aumenta a semelhança do nó escolhido e de seus vizinhos ao padrão P [Braga et al 2000].

Após o treinamento, a rede SOM organiza os padrões de entrada em *clusters*. Algumas aplicações requerem a rotulação dos nós de saída para indicar os *clusters* que representam. Isto é útil para a classificação de padrões desconhecidos [Braga et al 2000]. O algoritmo SOM geralmente leva a uma representação organizada de padrões de ativação retirados do espaço de entrada, desde que seus parâmetros sejam ajustados adequadamente [Haykin, 2001].

3. Estudo de Caso

Como cenário para validação das técnicas de agrupamento escolheu-se algumas **turmas** de graduação em cursos de Informática de duas instituições de ensino superior, sendo duas **destas** de primeiro semestre de uma instituição (INST1) na disciplina Introdução à Programação (IP-INF) e cinco turmas de segundo ano de outra instituição (INST2) na disciplina de Estruturas de Dados (ED-INF). Neste último caso, os alunos que já cursaram a disciplina IP-INF, tendo ou não sido aprovados.

Para a obtenção de dados que serviriam de *input*, aplicou-se um questionário com o objetivo de identificar o grau de confiança dos alunos em cada tópico da disciplina IP-INF. Para a elaboração do questionário, criou-se previamente uma ontologia da disciplina IP-INF utilizando a ferramenta Protégé-2000 [Protégé-2000 2002]. Um fragmento desta ontologia pode ser visto na Figura 1.

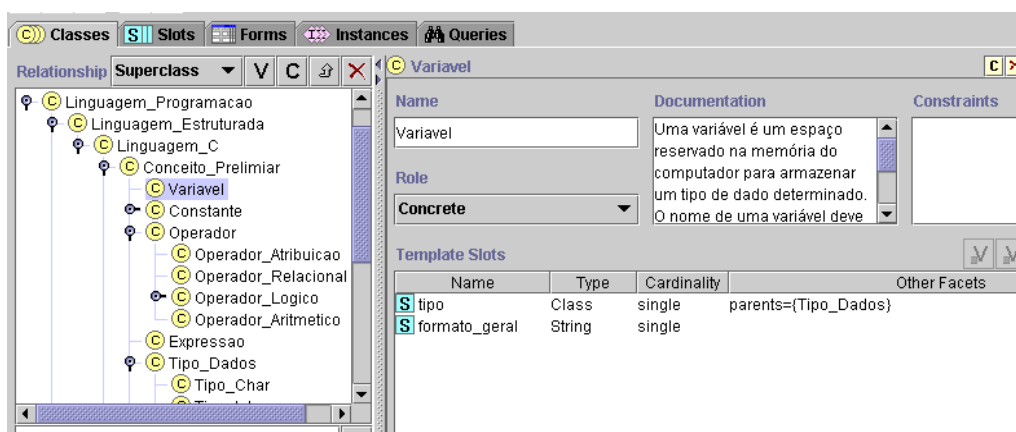


Figura 1. Fragmento da ontologia de linguagem de programação

Os alunos responderam ao questionário, indicando o seu grau de confiança em relação a cada tópico da ontologia (68 itens), registrando valores entre zero e cinco. Quanto mais próximo de cinco, maior a confiança e quanto mais próximo de zero, menor a confiança do aluno sobre seus conhecimentos naquele tópico. A tabela 1 apresenta a quantidade de alunos por turma e por instituição que responderam o questionário.

Tabela 1. Quantidade de alunos por turma x instituição

INST1 (Introdução à Programação)		INST2 (Estruturas de Dados)				
T1	T2	C2A	C2B	C2C	C2D	S2A
26	17	71	68	54	27	39

A análise das respostas de cada aluno, por grupos de conceitos, permitiu identificar os itens do programa em que os alunos, no conjunto, apresentam maior dificuldade, ou seja, menor grau de confiança. O uso de algoritmos para reconhecimento de padrões, baseados em técnicas de clusterização ajudam na identificação de grupos similares. A próxima seção mostra como esses algoritmos foram utilizados e quais resultados foram alcançados.

4. Experimentos e Resultados

Para aplicação dos algoritmos de agrupamentos, foram utilizados dois tipos de conjuntos de dados de entrada: um deles envolvendo cada um dos 68 itens do questionário e outro com as médias das respostas por grupos de conceitos de acordo com a ontologia. Apesar de parecer mais correto o uso do primeiro conjunto de dados, a análise de resultados tornou-se mais viável em cima do segundo conjunto, por envolver uma menor quantidade de variáveis (colunas) a serem comparadas. Contudo, comparando-se os resultados de uma única turma, com os dois tipos de conjuntos de dados, os agrupamentos formados pelo algoritmo K-means foram semelhantes. Um fragmento do arquivo de entrada (com as médias por grupo de conceitos) pode ser observado na Figura 2. A primeira coluna (matric) é a identificação do

aluno e as outras colunas (a, b, c, d, f) são os grupos de conceitos presentes no questionário. Dependendo do algoritmo utilizado, pequenas modificações no formato dos dados foram necessárias.

matric	a	b	c	d	e	f
73166	2.2	1.3	2	1.1	0.5	1.5
77138	2.2	2.6	1.2	2.5	1.4	0.5
77176	1.2	1.5	1.8	1	1.3	0
77104	1.2	1.7	2.3	0.9	0.6	1
77168	1.2	1.7	2.3	1.3	0.6	2
77181	1.3	1.9	2	0.9	1.1	1.5
77105	2.4	3.1	2.3	1.4	0.8	2.5

Figura 2. Exemplo do arquivo de entrada de dados

4.1. Análise de Resultados utilizando K-means

Para obtenção dos resultados foi utilizada a versão 3.0 do software Cluster desenvolvido inicialmente por Michael Eisen na Universidade de Stanford e modificado por Michiel de Hoon do Centro de Genoma Humano da Universidade de Tokyo [De Hoon 2002a].

Este software foi criado como um ambiente computacional para analisar dados de experimentos de DNA ou outros conjuntos de genomas. Apesar disto a ferramenta serviu ao nosso propósito, uma vez que o algoritmo K-means descrito no manual é padrão. Dentre vários outros softwares pesquisados este foi escolhido pela boa documentação o que possibilitou formatar os dados de entrada de acordo com o requerido e escolher a quantidade de *clusters* (grupos) para cada turma.

Inicialmente o Cluster 3.0 associa os elementos (alunos) aleatoriamente a um cluster. Em virtude desta aleatorização, diferentes execuções do K-means podem gerar agrupamentos diferentes, o que realmente foi observado nos resultados dos experimentos. Para lidar com isto, o Cluster 3.0 executa várias vezes o algoritmo K-means, e cada vez com uma clusterização inicial diferente. A soma das distâncias dentro dos “*clusters*” é usada para comparar diferentes soluções de clusterização e a que tiver menor distância é escolhida como resultado. O número de tentativas a ser feita depende da dificuldade de encontrar solução ótima, que por sua vez depende do número de elementos envolvidos e da quantidade de clusters a se formar. A cada resultado, o software apresenta a quantidade de vezes que a solução ótima foi encontrada. Se esta solução for encontrada muitas vezes, isto indica que provavelmente possui a menor soma de distância possível dentro dos *clusters* [De Hoon 2002b].

Um fragmento do arquivo de resultado gerado pelo Cluster 3.0 pode ser visualizado na figura 3.

matric	GROUP
77155	0
77167	0
77134	1
77140	1
77114	1
77115	1
77146	1
73166	2
77138	2
77176	2
77104	2
77168	2

Figura 3. Exemplo do arquivo de saída de dados

O aplicativo foi executado para que formasse cinco “*clusters*” (grupos), para cada uma das turmas em 200 tentativas. Escolheram-se, então, soluções ótimas encontradas em pelo menos 10 tentativas. Para uma das turmas (C2D), efetuou-se várias execuções do K-means com o objetivo de fazer comparações entre os diferentes agrupamentos formados. A tabela 2 apresenta um detalhamento de 10 execuções e os agrupamentos formados com os 27 alunos. Cada linha da tabela representa um aluno e o(s) grupo(s) em que este foi alocado em cada uma das 10 execuções. A última linha (Nº de vezes)

corresponde a quantidade de vezes em que o mesmo agrupamento foi obtido, em 200 tentativas, indicando ser uma "solução ótima".

Tabela 2. Agrupamentos da turma C2D em 10 execuções do K-means

IDENTIFICAÇÃO DO ALUNO	EXECUÇÕES / GRUPOS									
	[01]	[02]	[03]	[04]	[05]	[06]	[07]	[08]	[09]	[10]
1	2	2	2	2	2	2	2	2	2	2
2	1	1	1	1	1	1	1	1	1	1
3	1	1	1	1	1	1	1	1	1	1
4	2	2	2	2	2	2	2	2	2	2
5	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0
7	4	4	4	4	4	4	4	4	4	4
8	0	0	0	0	0	0	0	0	0	0
9	1	1	1	1	1	1	1	1	1	1
10	0	0	0	0	0	0	0	0	0	0
11	4	4	4	4	4	4	4	4	4	4
12	0	0	0	0	0	0	0	0	0	0
13	1	1	1	1	1	1	1	1	1	1
14	1	1	1	1	1	1	1	1	1	1
15	0	0	0	0	0	0	0	0	0	0
16	4	4	4	4	4	4	4	4	4	4
17	4	4	1	4	4	4	4	4	4	4
18	3	3	3	3	3	3	3	3	3	3
19	4	4	1	4	4	4	4	4	4	4
20	3	3	3	3	3	3	3	3	3	3
21	4	4	4	4	4	4	4	4	4	4
22	0	0	0	0	0	0	0	0	0	0
23	4	4	1	1	4	1	1	4	4	1
24	4	4	4	4	4	4	4	4	4	4
25	3	3	3	3	3	3	3	3	3	3
26	4	1	1	4	1	4	1	4	4	1
27	1	3	1	1	3	1	1	1	1	1
Nº de Vezes	21	25	18	23	17	18	14	19	16	21

Dentre os 27 alunos da turma C2D, apenas 4 alunos não foram alocados sempre no mesmo grupo nas 10 execuções: o aluno 19 foi inserido 9 vezes no grupo 4 e apenas uma vez no grupo 1; os

alunos 23 e 26 foram colocados metade das execuções no grupo 4 e outra metade no grupo 1; e o aluno 27 apareceu apenas 2 vezes no grupo 3 e 8 vezes no grupo 1.

Uma análise sobre os dados de entrada da turma C2D, mostra que a migração de determinado aluno para um grupo ou outro, deve-se ao fato dos valores das respostas do aluno presentes no questionário, estarem numa "distância" mais próxima de um grupo ou de outro, dependendo do grupo inicial, que por sua vez é aleatório em cada nova tentativa. Quando a "distância" é bem definida, o aluno é alocado sempre no mesmo grupo.

Outro dado a se observar nas execuções do K-means é a quantidade de elementos por grupo. Conforme mostra a tabela 3, não existe um valor padrão para o tamanho dos grupos. A alocação dos alunos em cada grupo depende apenas da "proximidade" ou "similaridade" das suas respostas em cada conceito.

A tabela 3 registra ainda que para as duas turmas da INST1, o algoritmo K-means foi executado para formar apenas 3 grupos ao invés de 5. Convém lembrar que essas duas turmas correspondem a alunos de uma disciplina introdutória em que muitas respostas se concentram com valores iguais a zero para a maioria dos conceitos perguntados, indicando ausência de conhecimento. Ao tentar executar o K-means para a formação de 5 grupos, os agrupamentos formados eram conseguidos apenas 1 vez em 200 tentativas, o que pela documentação do software não corresponderia a uma "solução ótima" [De Hoon 2002b].

Tabela 3. Quantidade de elementos por grupo obtidos pelo K-means

	TURMAS POR INSTITUIÇÃO						
	INST1		INST2				
Nº de vezes	26	19	21	32	20	25	20
GRUPO	T1	T2	C2A	C2B	C2C	C2D	S2A
0	16	3	20	5	12	7	7
1	4	2	12	19	16	6	6
2	6	12	23	14	5	2	10
3	-	-	13	12	7	4	5
4	-	-	3	18	14	8	11
TOTAL	26	17	71	68	54	27	39

Com o objetivo de verificar se o K-means forneceu uma divisão de dados similar à classificação que uma pessoa faria, solicitou-se ao professor das turmas C2D e S2A, escolhidas por terem uma menor quantidade de alunos, que fizesse uma análise comparativa dos agrupamentos formados pelo algoritmo K-means, em relação ao desempenho real destes alunos na sala-de-aula e nas avaliações. Segundo o professor, se ele tivesse que agrupar os alunos usando como critério as avaliações (provas e trabalhos) e as observações de sala-de-aula faria praticamente os mesmos agrupamentos. Este observou ainda que algumas inclusões de elementos em grupos "errados" foram decorrentes de respostas equivocadas destes alunos em alguns conceitos. Isto pode significar que de fato, grande parte dos alunos foi coerente em suas respostas e que o algoritmo K-means serve ao propósito de agrupamento de alunos.

Quanto à atribuição de rótulos aos grupos formados, observou-se que alguns grupos são bem definidos (vide, por exemplo, o grupo 1 na Tabela 4). Utilizando os resultados obtidos com a turma S2A, e com o auxílio do referido professor que analisou os resultados e o desempenho dos alunos em sala-de-aula, os grupos foram classificados em níveis de 'A' a 'E', indicando valores crescentes de domínio e conhecimento expressos pelo grau de confiança, ou seja:

- Nível A – subgrupo 1, com valores concentrados entre 3.2 e 5.0
- Nível B – subgrupo 4, com valores concentrados entre 2.8 e 4.5
- Nível C – subgrupo 3, com valores concentrados entre 2.1 e 3.0

- Nível D – subgrupo 2, com valores concentrados entre 1.0 e 2.2
- Nível E – subgrupo 0, com muitos valores próximos de 0

A Tabela 4 apresenta a visão parcial de um agrupamento formado pelo algoritmo K-means na turma S2A. Quando se compara as mesmas colunas dentro do mesmo grupo, é possível observar a proximidade dos valores demonstrando assim a coerência dos agrupamentos formados (vide a área escura na tabela 4, referente ao grupo 1).

Tabela 4. Visão dos grupos formados pelo K-means

Identificação do aluno	Conceitos (Questionário)						GRUPO
	A	B	C	D	E	F	
77167	3,4	3,9	0,0	0,0	0,0	0,0	0
77155	2,4	4,0	2,5	1,1	0,0	0,0	0
77140	3,4	4,2	4,0	4,1	3,3	4,5	1
77146	3,3	4,6	4,0	3,5	4,5	4,0	1
77114	3,2	4,4	5,0	4,2	4,6	4,5	1
77134	4,5	4,2	5,0	2,9	4,8	4,5	1
77115	4,1	4,6	5,0	4,8	5,0	5,0	1
77143	1,9	3,0	3,0	0,8	0,0	3,5	2
73166	2,2	1,3	2,0	1,1	0,5	1,5	2
77104	1,2	1,7	2,3	0,9	0,6	1,0	2
77168	1,2	1,7	2,3	1,3	0,6	2,0	2
77105	2,4	3,1	2,3	1,4	0,8	2,5	2

4.2. Análise de Resultados utilizando SOM

Para obtenção dos resultados foi utilizado o software SOM_PAK desenvolvido pela equipe “SOM Programming” do Laboratório de Computação e Ciências da Informação da Universidade de Tecnologia de Helsinki [Kohonen et al 1996].

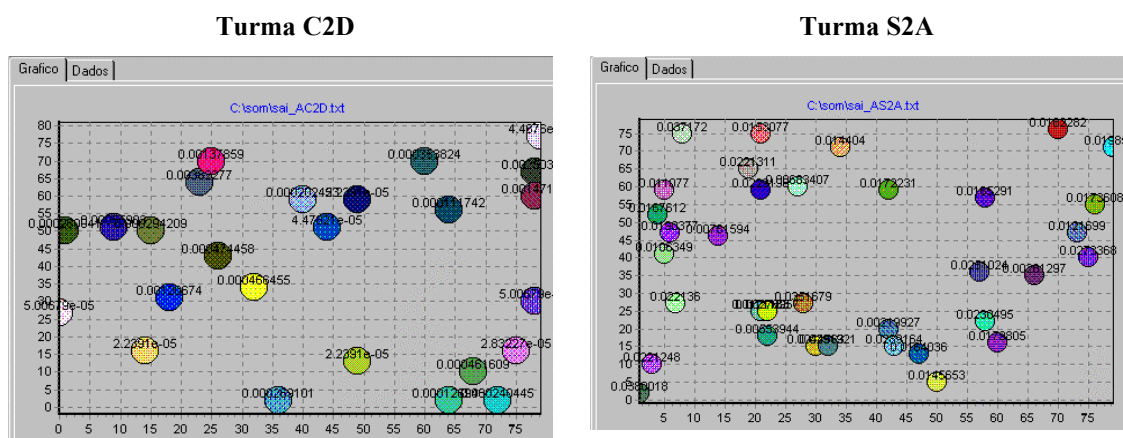


Figura 4. Classificação fornecida pelo Algoritmo SOM

Como pode ser observado na Figura 4, o agrupamento de aprendizes não foi obtido, indicando possivelmente que o questionário e as respostas dos alunos não foram suficientes para tal. Em várias

tentativas, observou-se que os dados ficam bem espalhados no plano, não sendo possível identificar qualquer formação de grupos. Como ressalta [Braga et al 2000] os parâmetros de aprendizagem e da função de vizinhança devem ser cuidadosamente escolhidos.

Embora os dados não tenham convergido para resultados satisfatórios, a técnica continuará a ser pesquisada, uma vez que tem sido utilizada com sucesso e com bom desempenho em sistemas de mineração de dados além de ser indicada em contextos em que não se conhece a *priori* a quantidade de grupos que se deseja formar.

5. Conclusões e aprofundamentos necessários

Os resultados apresentados confirmam que algoritmos que utilizam técnicas de clusterização são bastante úteis para a formação de grupos homogêneos de aprendizes. A identificação de grupos similares em turmas do ensino presencial é uma ferramenta importante que o professor pode utilizar na aplicação de uma pedagogia diferenciada.

O trabalho mostra que os agrupamentos formados pelo K-means são similares a que um humano faria utilizando análise de desempenho do aluno em sala de aula. No entanto, a “qualidade” dos grupos formados depende das “distâncias” dos valores presentes nos dados de entrada, ou seja, nem sempre é possível obter, com garantia de solução ótima, a quantidade de grupos que se deseja. Por outro lado, o experimento utilizando o algoritmo SOM mostrou que os dados obtidos com o questionário e os parâmetros utilizados não foram suficientes para a formação de grupos distintos.

Como aprofundamentos necessários, pretende-se fazer uma análise mais abrangente dos algoritmos utilizados para clusterização. Por exemplo, no algoritmo SOM, a idéia é fazer ajustes em seus parâmetros com o objetivo de obter agrupamentos, no mínimo, semelhantes aos obtidos com o algoritmo K-means.

Outra direção importante a ser explorada é utilizar as informações obtidas com este trabalho para a modelagem do aprendiz em um STI. Com isto será possível utilizar, para os agrupamentos, além dos dados de entrada fornecidos pelo estudante através de questionários, informações geradas pelo registro de uso do STI.

No que diz respeito aos agrupamentos obtidos com este trabalho nas turmas pesquisadas, pretende-se desenvolver experimentos com grupos homogêneos e heterogêneos, através do uso de pedagogias diferenciadas. Os grupos heterogêneos poderão ser obtidos utilizando técnicas de embaralhamento (scrambling) dos membros dos grupos homogêneos.

Referências

- Backer, Eric. Computer Assisted Reasoning in Cluster Analysis. Prentice Hall, New York, 1995.
- Baquero, Ricard. (1998) “Vygotsky e a Aprendizagem Escolar”. Porto Alegre: Editora Artes Médicas.
- Braga, A. P., Ludermir, T. B. e Carvalho, A. C. P. L. Redes Neurais Artificiais: Teoria e Aplicações, LTC, 2000.
- De Hoon, M.; Imoto, S.; Miyano, S. (2002a) “A Comparison of Clustering Techniques for Gene Expression Data” (abstract). Poster apresentado na 10th International Conference on Intelligent Systems for Molecular Biology em Agosto de 2002, Edmonton, Canadá.
- De Hoon, M e Eisen, Michael (2002b). Cluster 3.0 Manual. Universidade de Tokyo. Disponível em <http://bonsai.ims.u-tokyo.ac.jp/~mdehoon/software/cluster/software.htm#ctv>, Junho.
- Kohonen, T. (1990) “The Self-Organizing Map”. Proceedings of the IEEE, vol. 78, n° 9, p. 1464-1480.
- Kohonen T., Hynninen J., Kangas J. e Laaksonen J. (1996) “SOM PAK: The Self-Organizing Map program package”, Report A31, Helsinki University of Technology, Laboratory of Computer and Information Science, <http://citeseer.nj.nec.com/kohonen96som.html>, Junho.
- Haykin, S. Redes Neurais: Princípios e Prática. 2 ed. Porto Alegre: Bookman, 2001.
- Jain, A. K. and Dubes, R. C. Algorithms for Clustering Data. Prentice-Hall, 1988.

- Jain, A.K., Murty M.N., and Flynn P.J. (1999): "Data Clustering: A Review", ACM Computing Surveys, Vol 31, No. 3, 264-323.
- Monteiro E Silva, A. B., Portugal, M. S., Cechin, A. L. (2001). "Redes Neurais Artificiais e Análise de Sensibilidade: Uma Aplicação à Demanda de Importações Brasileira". Revecap vol. 5 n. 4.
- Moreira, Marco A., Masini, Elcie F.S. Aprendizagem Significativa: A Teoria de David Ausubel. São Paulo: Centauro, 2001.
- Perrenoud, Philippe. (2000) "Dez Novas Competências para Ensinar". Trad. Patricia Chittoni Ramos. Porto Alegre: Artes Médicas Sul.
- Protégé-2000 (2002). Protégé-2000 home page: <http://protege.stanford.edu/index.html>.
- Stankov, Slavomir. (1996). Student Model Developing for Intelligent Tutoring Systems. International Journal for Engineering Modeling. Vol 9, N°. 1.