_____ / _____ / _____

Algoritmo dos $k-$vizinhos

Suponha que você é o programador do NetFlix e quer implantar um supersistema de sugestão de filmes. Para isso, você vai criar e implantar o algoritmo conhecido como os $k-$vizinhos. A idéia é simples. Cada um dos filmes do catálogo vai receber uma nota entre o 0 (ruim) e 5 (excelente). Você não pode obrigar os usuários a dar notas. Então, você vai dar um mimo (um brinde ou um desconto) toda vez que o usuário atribuir essas notas a um filme que acabou de ver.

Mensalmente, você vai coletar todas as notas atribuídas aos filmes e vai calcular uma média da nota de cada filme. É essa nota que o algoritmo vai usar.

A nota vai colocar o filme em questão em um eixo ou dimensão e o nosso espaço de filmes vai ser hexa-dimensional. (6 dimensões). Note aqui a beleza da matemática: enquanto na física (no mundo, no universo) estamos limitados a 3 ou 4 dimensões e sabemos via Teorema de Pitágoras, calcular distâncias, nada nos impede de usar o mesmo teorema num espaço hexa-dimensional. Ninguém tem idéia de como seria no universo esse espaço, mas na matemática teórica ele está perfeitamente definido. *Santa Matemática!*

As dimensões atribuídas aos filmes serão:

riso Quão comédia é este filme: se você riu muito, atribua 5, se você não riu nada, ou se mesmo tendo a pretensão de ser uma comédia o filme não te fez rir (ou fez chorar) atribua 0. Atribua notas proporcionais ao quesito.

filme cult Quão famoso e cultuado o filme é. Ou você acha que é. Afinal é a sua opinião que está sendo pedida. Atribua notas proporcionais ao quesito.

atuação do ator principal Que nota você dá à atuação do personagem principal do filme. Atribua notas proporcionais ao quesito.

efeitos especiais adequados Quão bem inseridos no filme estão os efeitos especiais, se ele existem.Atribua notas proporcionais ao quesito.

drama Como é construído o drama do filme. Atribua notas proporcionais ao quesito.

romance Qual a nota para romance que o filme merece. Atribua notas proporcionais ao quesito.

No arquivo anexo, existem 1000 filmes, identificados por um código numérico, e seguido pelas 6 notas médias atribuídas no último mês pelos usuários do serviço.

Do ponto de vista matemático, cada filme destes define um ponto no espaço de 6 dimensões.

Sugestão

Para que o sistema possa fazer uma recomendação adequada, o interessado deve dar o código do último filme de que ele gostou. Por hipótese é um dos códigos do acervo no arquivo. Agora, o seu algoritmo vai usar este filme de que o usuário gostou como padrão e vai calcular a distância hexadimensional de todos os 999 filmes restantes até o padrão. Feito isso, o algoritmo deve ordenar o acervo em ordem crescente por distância e devolver o segundo, terceiro e quarto filmes do acervo. Note que o primeiro terá distância 0 já que ele é o padrão. Por isso, ele ficou de fora da nossa relação.

Supondo que o filme k é o padrão. A distância de qualquer filme i para o filme k é dada pelo Teorema de Pitágoras expandido para 6 dimensões:

$$d_{ik} = \sqrt{\sum_{m=1}^{6} (x_{im} - x_{km})^2}$$

Note que o filme indicado pelo cliente é indicado pelo seu código, então antes de descobrir qual é o padrão, você precisa varrer o acervo para localizar o índice k do padrão.

Depois disso, pode aplicar o algoritmo acima.

Exemplo

Para testar o seu algoritmo, localize no AVA o arquivo EXEMPLO668. Ele é uma instância completa deste exercício. As primeiras 5 linhas do arquivo são

```
 3 1.8    1.405  1.4125 4.02   3.195  2.3125
 4 1.87   2.3625 4.55   0.655  4.35   1.9375
 6 1.655  2.5975 0.1675 1.52   4.3375 1.545
 7 0.7975 1.215  0.645  3.7    4.7    2.44
14 2.2425 4.295  4.475  4.8525 4.8525 2.92
```

Se o padrão escolhido for o filme da metade do arquivo (índice=500) o cálculo das distâncias vai gerar (apenas as primeiras 5 linhas)

```
 3 1.8  1.405 1.412 4.02 3.195 2.312 4.4559
 4 1.87 2.362 4.55  0.65 4.35  1.937 5.4483
 6 1.65 2.597 0.167 1.52 4.337 1.545 4.6661
 7 0.79 1.215 0.645 3.7  4.7   2.44  5.3738
14 2.24 4.295 4.475 4.85 4.852 2.92  4.4343
```

Finalmente, ordenando este segundo arquivo pela distância vai ficar

```
1543 4.19  4.55  1.607 2.927 3.24  4.05  0
1044 4.065 4.022 1.985 2.172 4.34  3.75  1.51
2495 4.707 4.06  1.697 2.305 2.345 3.187 1.56
 418 3.742 3.225 1.602 3.71  3.137 4.097 1.60
 181 3.7   4.182 1.437 4.362 3.555 4.24  1.61
```

O programa Python

```
def dist(x,y):
  su=(x[1]-y[1])**2
  su=su+(x[2]-y[2])**2
  su=su+(x[3]-y[3])**2
  su=su+(x[4]-y[4])**2
  su=su+(x[5]-y[5])**2
  su=su+(x[6]-y[6])**2
  su=su**0.5
  return su
def f668(arq,qual):
  import numpy
  bd = numpy.zeros((1000,8))
  f=open(arq,'r')
  for i in range(1000):
    ll = f.readline()
    ll = ll+" 0.0"
   #acrescenta a distancia (por enquanto é 0.0)
    bd[i] = [float(x) for x in ll.split()]
  for i in range(1000):
    bd[i][7]=dist(bd[qual],bd[i])
  bd=bd[bd[:,7].argsort()]
      # ordena a matriz pela coluna 7 - sort
  print(int(bd[1][0]),int(bd[2][0]),
       int(bd[3][0]))
       #imprime os filmes. o 0 é o próprio
f668("c:/p/n/668/EXEMPLO668.myd",499)
   #499 é o filme 500
```

Aplicações

Este algoritmo pode ser adaptado – sempre com o objetivo de fazer *machine learning* – a diversas situações possíveis

1. Na área médica visando estabelecer compatibilidade entre doador e receptor de algum órgão a transplantar. Os valores numéricos podem ser fatores laboratoriais ou clínicos importantes para a compatibilidade/ rejeição. Aplicar o algoritmo neste universo de doadores potenciais vai levantar quais os mais próximos ao doador em todo o universo.

2. Na área comportamental para uma possível criação de um aplicativo de encontros amorosos. A expertise aqui deve se concentrar nas perguntas (critérios) a valorar e também na atribuição de valores para garantir *match* de interesses. A regra da distância eventualmente terá que ser reescrita.
   (a) idade ≡ regra de comparação criativa
   (b) sexo ≡ regra de comparação criativa
   (c) situação econômica ≡ regra de comparação criativa
   (d) peso/altura/quantidade de cabelo/características
   (e) cor de pele/diversão preferida/práticas apimentadas
   (f) gosto/disponibilidade para viagens/necessidade de sigilo
   (g) precisão/urgência, pretensão futura: tipos de vínculos

3. Na taxonomia animal, um zoólogo diante de uma espécie nova pode valorar características físicas (número de pernas, peso médio, número de olhos, articulações, apêndices...) e consultar fatias crescentes do reino animal estudando os resultados de cada consulta.

4. Estudando o histórico de compras de uma pessoa em um portal on line um aplicativo poderia sugerir novas compras. Por exemplo suponha valorar itens em um sapato feminino: altura, número, cores, fechado/aberto, de prender ou não,...

   A cada dois meses, o aplicativo pode varrer o arquivo, recuperar itens encalhados, aplicar um desconto a eles, e indicá-los a uma compradora, que não vai resistir...

5. Carros, peças de espetáculos (disk-ingressos), livros, passagens aéreas, compras de supermercado (preço, portabilidade, tipo de item, nacional/importado, cara/barato), pratos em disk-entrega...

☞ Para você fazer

O acervo que você vai consultar está no AVA com o nome de

`F668D001.myd`

1. Processe o algoritmo considerando que o padrão escolhido foi o filme de código

`2846`

e informe aqui o código do filme das 3 sugestões do seu algoritmo achou

1	2	3

2. Processe o algoritmo considerando que o padrão escolhido foi o filme de código

`2008`

e informe aqui o código do filme das 3 sugestões do seu algoritmo achou

4	5	6

3. Processe o algoritmo considerando que o padrão escolhido foi o filme de código

`459`

e informe aqui o código do filme das 3 sugestões do seu algoritmo achou

7	8	9