Redução de dimensionalidade

Jacques Wainer

IC – Unicamp

Novembro 2015

Agrupamento ou clusterização

- caracterizar os dados como membros de grupos.
- os grupos devem "fazer sentido"
- em alguns casos "representantes" dos grupos passam a ser "arquetipos" dos dados.
- em KDD: entender os "representantes dos grupos"
- em processos automáticos: pertencer a um grupo é uma forma de classificar o dado e essa classe é usada nos processamentos seguintes
- nas próximas aulas falaremos apenas de dados vetoriais (pontos no espaço). A última aula dessa disciplina será sobre agrupamento em dados relacionais (grafos).

Tipos de agrupamentos

- **completo** vs parcial: todos os dados pertencerão a um (ou mais grupos) ou apenas parte dos dados serão agrupados. Desta forma dados anômalos não influenciam na definição dos grupos. ▸ link
- **particionamento** vs intersecção (*overlapping*): os dados são membros de apenas um grupo ou podem ser membros simultaneamente de mais de um grupo ▸ link
- **um nivel** vs multi-nivel: os grupos estão todos no mesmo nível ou podem conter outros grupos.

Familias de algoritmos

- baseado em centroides: descobrir o "centro" de cada grupo. Baseado em distancia - cada ponto pertence ao grupo (grupos) cujo centro é mais proximo(s).
- baseado em conectividade (ou agrupamento hierárquico): pontos e sub-grupos perto um dos outros devem pertencer ao mesmo super-grupo
- baseado em densidade: grupos sao regiões de alta densidade de pontos separados por regiões de baixa densidade
- baseado em distribuições probabilísticas: modela-se os dados como sendo gerado por diferentes distribuições probabilísticas (cada uma um grupo).
- outras

k-médias (k-means)

Algoritmo iterativo: dividir os dados em k grupos

1. criar os centros dos grupos aleatoriamente (inicialização)
2. atribuir cada dado ao grupo cujo centro esta mais próximo
3. se não houve mudança de atribuição desde a volta passada, terminar
4. recomputar o centro do grupo como sendo a média dos dados que pertencem ao grupo
5. voltar ao item 2

▸ link e também ▸ link

k-médias inicialização

- pode-se mostrar que o algoritmo k-means a cada passo encontra uma melhor solução para o problema de encontrar centros que minimizam o quadrado da distancia entre o centro e os dados daquele grupo
- como um algoritmo iterativo ele pode ficar preso em soluções "ruins" mas que nao podem ser "localmente" melhoradas (minimo local)
- minimos locais dependem da inicialização
- várias inicializações: centros aleatórios, partição aleatória, centros como dados, kmeans++ (primeiro centro aleatório, segundo dados como centro com probabilidade proporcional a distancia, etc).

k-médias

- k-means é um algoritmo completo (todos os pontos são classificados), particionamento (pontos só pertencem a um grupo) e um nível (não ha subgrupos dentro de super-grupos
- os grupos em k-means são convexos (sem "entradas"), talvez estendendo-se ao infinito em alguma direção ▸ link

k-means em R

`kmeans` ▸ link

- passa-se ou os centros iniciais dos clusters, ou um k, e os centros iniciais serão escolhidos dos dados
- permite multiplas inicializações (`nstart=1`) (falaremos disso abaixo)
- permite controlar o numero máximo de interações (`iter.max = 10`)
- `$cluster` retorna a indicação do clusters de cada dado
- `$centers` são os k centros dos clusters

Há várias outras implementações e funções auxiliares (mais abaixo)

kmeans em R

```
> cl=kmeans(iris[,1:4],3)
> cl
K-means clustering with 3 clusters of sizes 50, 38, 62

Cluster means:
  Sepal.Length Sepal.Width Petal.Length Petal.Width
1     5.006000    3.428000     1.462000    0.246000
2     6.850000    3.073684     5.742105    2.071053
3     5.901613    2.748387     4.393548    1.433871

Clustering vector:
  [1] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 [61] 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 2 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
[121] 2 3 2 3 2 2 3 3 2 2 2 2 2 3 2 2 2 2 3 2 2 2 3 2 2 2 3 2 2 3

Within cluster sum of squares by cluster:
[1] 15.15100 23.87947 39.82097
 (between_SS / total_SS =  88.4 %)
```

kmeans em R

```
x <- rbind(matrix(rnorm(100, sd = 0.3), ncol = 2),
           matrix(rnorm(100, mean = 1, sd = 0.3), ncol = 2))
colnames(x) <- c("x", "y")
plot(x)

cc=kmeans(x,2)
cc

plot(x,col=cc$cluster)
points(cc$centers,col=1:2,pch="+",cex=3)
```

Quando o kmeans não da certo

- kmeans acha um minimo local (que não pode ser melhorado trocando uma atribuição) e portanto soluções ruins são devido a inicialização "errada" ▸ link
- várias heurísticas para escolher a inicialização certa (kmeans++).
- ou roda com varias inicializações e "escolhe a melhor"
- qual a "melhor?". Usar a métrica de minimização do proprio kmeans - soma dos quadrados das distancias ate o centro do grupo.
- `kemans` no R permite várias inicializações `nstart=` e escolhe a melhor solução

Como escolher o k

assunto da próxima aula

Pre-processar os dados?

- deve-se fazer uma padronização dos atributos numéricos que não tem a mesma medida ou o "mesmo sentido"
- deve-se converter os atributos categóricos em múltiplos atributos 0/1 seguindo o "one-hot enconding" (aula 1).
- há algum debate se deve-se padronizar os atributos 0/1 derivados da conversão dos atributos categóricos originais

Limites do kmeans

- kmeans acha grupos "convexos" ▸ link justificativa para clusterização por densidade/proximidade
- kmeans não se da bem com grupos de tamanho e densidade muito diferente ▸ link
- kmeans acaba agrupando dados que não tem estrutura de grupos
- para muitos grupos, há uma maior probabilidade de minimos locais ▸ link
- os grupos do kmeans tem "o mesmo raio" ▸ link

Tarefa

- Usando as 4 primeiras dimensões do dataset `iris` (que ja vem dentro do R), plote as 2 primeiras dimensões dos dados,
- rode o kmeans com k=3, nstart=20,
- plote o resultado da clusterização nas 2 primeiras dimensões
- os grupos parecem suficientemente separados nas 2 primeiras dimensões?
- plote os grupos nos 2 principais componentes dos dados - a separação dos grupos parece melhor?
- repita a clusterização nos dados padronizados (usando o `scale`
- plote os novos grupos nos 2 principais componentes dos dados - ha alguma diferença?
- entregue um pdf com os 4 gráficos (2d dos dados, 2d da clusterização sem padronização, 2PC da clusterização sem padronização e 2PC com padronização.) e com uma resposta curta (1 paragrafo no máximo) para as questões levantadas.