k-means

EST0133 - Introdução à Modelagem de Big Data

Marcus Nunes
https://introbigdata.org/
https://marcusnunes.me/

Universidade Federal do Rio Grande do Norte

Introdução

Introdução

- É um método popular de clusterização de dados
- Clusterização significa dividir um conjunto de dados em grupos menores nos quais os pontos dos mesmos grupos são mais similares entre si
- O objetivo é separar n observações em K grupos
- O problema é que não sabemos qual é o valor de K

Introdução

- No k-means, cada observação é designada para o grupo com a média mais próxima
- O método é sensível a dados anômalos e outliers
- Os pontos podem se mover de um grupo para outro, mas a resposta final depende da inicialização dos centros
- Se uma observação estiver igualmente perto de dois ou mais centros, então o grupo deve ser decidido aleatoriamente

Introdução

- Pode ser muito efetivo como um método de previsão *black box*
- Não é útil para entender a natureza da relação entre as características e as classes

k-means

k-means

- Conjunto de treinamento $\{(\boldsymbol{x}_1,g_1),(\boldsymbol{x}_2,g_2),\cdots,(\boldsymbol{x}_n,g_n)\}$, onde os grupos $g_i \in \{1,2,,\cdots,K\}$ e $\boldsymbol{x} \in \mathbb{R}^p$
- Represente o conjunto de treinamento por pontos no espaço das características, também chamados de centroides
- Cada centroide é associado a uma classe e a clusterização de cada $\boldsymbol{x}$ é feito em relação ao centroide mais próximo
- Os métodos diferem de acordo com o número de centroides e suas posições

k-means

- Assuma que há M centroides denotados por $\mathcal{Z} = \{\mathbf{z}_1, \mathbf{z}_2, \cdots, \mathbf{z}_M\}$
- Cada amostra do conjunto de treinamento é designado para um dos centroides
- Denote a função de designação por $A(\cdot)$
- Assim $A(\mathbf{x}_i) = j$ significa que o $i-$ésimo elemento é designado para a $j-$ésima classe

k-means

- O objetivo é minimizar o erro quadrático médio entre as amostras de treinamento e seus centroides de representação
- Isto é equivalente ao traço da matriz de covariância dentro de cada grupo

$$\arg\min_{\mathcal{Z},\mathcal{A}} \sum_{i=1}^{N} \left\| \boldsymbol{x}_i - \boldsymbol{z}_{A(\boldsymbol{x}_i)} \right\|^2$$

- Denote a função objetivo por

$$L(\mathcal{Z}, A) = \sum_{i=1}^{N} \left\| \boldsymbol{x}_i - x_{A(\boldsymbol{x}_i)} \right\|^2$$

k-means

- Intuição: as amostras utilizadas para treinamento estão concentradas em volta dos centroides
- Assim, os centroides servem como uma representação compacta dos dados de treinamento

Condições Necessárias

- Se $\mathcal{Z}$ está fixo, a função de designação $A(\cdot)$ ótima deve seguir a regra do vizinho mais próximo, isto é

$$A(\boldsymbol{x}_i) = \arg \min_{j \in \{1,2,\cdots,M\}} \|\boldsymbol{x}_i - \boldsymbol{z}_j\|$$

- Se $A(\cdot)$ está fixa, o centroide $\boldsymbol{z}_j$ deve ser a média de todas as amostras designadas para o $j-$ésimo centroide

$$\boldsymbol{z}_j = \frac{\sum_{i:A(\boldsymbol{x}_i = j)} \boldsymbol{x}_i}{N_j},$$

onde N_j é o número de amostras designadas ao centroide j

Algoritmo

- Baseado nas condições necessárias, o algoritmo k-means alterna dois passos:
  1. Para um conjunto fixo de centroides, otimize $A(\cdot)$ designando cada amostra ao centroide mais próximo utilizando a distância euclidiana
  2. Atualize os centroides calculando a média de todas as amostras associadas a eles
- O algoritmo converge porque após cada iteração, a função objetivo decresce
- Usualmente, a convergência é rápida
- Se a razão entre o decréscimo e a função objetivo estiver abaixo de um limite, o algoritmo para

Exemplo

- Conjunto de treinamento: $\{1{,}2; 5{,}6; 3{,}7; 0{,}6; 0{,}1; 2{,}6\}$
- Aplicando k-means com 2 centroides $\{\boldsymbol{z}_1, \boldsymbol{z}_2\}$

Exemplo

- Escolha aleatoriamente dois centroides $\boldsymbol{z}_1 = 2$, $\boldsymbol{z}_2 = 5$

Fixo	Atualização
2	$\{1{,}2; 0{,}6; 0{,}1; 2{,}6\}$
5	$\{5{,}6; 3{,}7\}$
$\{1{,}2; 0{,}6; 0{,}1; 2{,}6\}$	1,125
$\{5{,}6; 3{,}7\}$	4,650
1,125	$\{1{,}2; 0{,}6; 0{,}1; 2{,}6\}$
4,650	$\{5{,}6; 3{,}7\}$

- Os dois centroides são $\boldsymbol{z}_1 = 1{,}125$ e $\boldsymbol{z}_2 = 4{,}650$
- A função objetivo é dada por $L(\mathcal{Z}, A) = 5{,}3125$

Algoritmo

- Escolha aleatoriamente dois centroides $\boldsymbol{z}_1 = 0{,}8$, $\boldsymbol{z}_2 = 3{,}8$

Fixo	Atualização
0,8	$\{1{,}2; 0{,}6; 0{,}1\}$
3,8	$\{5{,}6; 3{,}7; 2{,}6\}$
$\{1{,}2; 0{,}6; 0{,}1\}$	0,633
$\{5{,}6; 3{,}7; 2{,}6\}$	3,967
0,633	$\{1{,}2; 0{,}6; 0{,}1\}$
3,967	$\{5{,}6; 3{,}7; 2{,}6\}$

- Os dois centroides são $\boldsymbol{z}_1 = 0{,}633$ e $\boldsymbol{z}_2 = 3{,}967$
- A função objetivo é dada por $L(\mathcal{Z}, A) = 5{,}2133$

k-means

- Note que, iniciando em valores diferentes, o algoritmo k-means converge para mínimos locais diferentes
- Podemos mostrar que $\{\boldsymbol{z}_1 = 0{,}633, \boldsymbol{z}_2 = 3{,}967\}$ é o solução ótima global

Simulação

Simulação

- Duas classes seguem a distribuição normal com matriz de covariância comum, dada por

$$\boldsymbol{\Sigma} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

- A média das duas classes são

$$\boldsymbol{\mu}_1 = \begin{pmatrix} 0{,}0 \\ 0{,}0 \end{pmatrix} \text{ e } \boldsymbol{\mu}_2 = \begin{pmatrix} 1{,}5 \\ 1{,}5 \end{pmatrix}$$

- As prioridades *a priori* das duas classes são $p_1 = 0{,}5$ e $p_2 = 0{,}5$

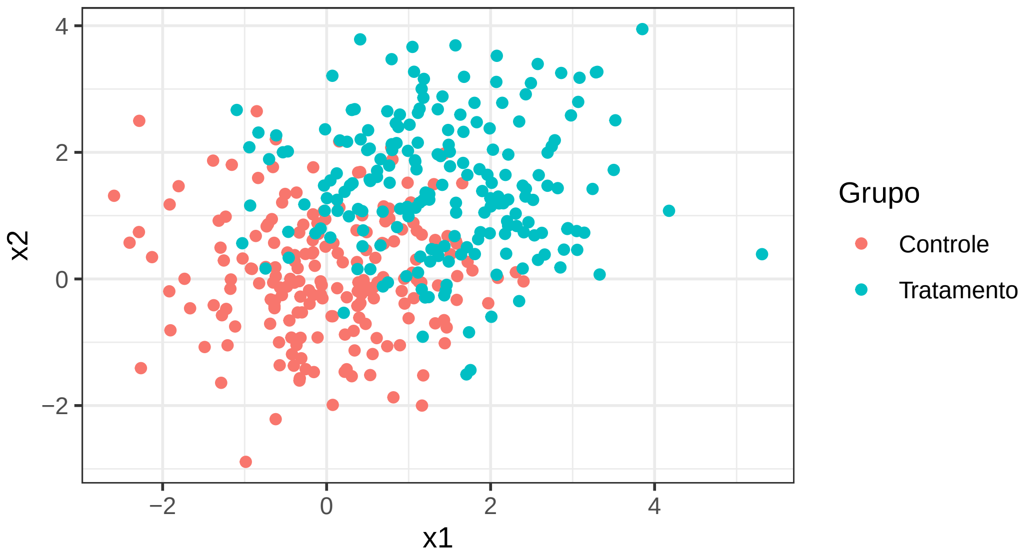
Simulação

```
> library(mvtnorm)
> library(ggplot2)
> set.seed(1)
>
> N     <- 200
> mu1   <- c(0, 0)
> mu2   <- c(1.5, 1.5)
> Sigma <- matrix(c(1, 0, 0, 1), ncol = 2)
>
> ctrl <- rmvnorm(n = N, mean = mu1, sigma = Sigma)
> trt  <- rmvnorm(n = N, mean = mu2, sigma = Sigma)
>
> dados <- data.frame(rbind(ctrl, trt),
+     rep(c("Controle", "Tratamento"), each = N))
> names(dados) <- c("x1", "x2", "Grupo")
```

Simulação

```
> ggplot(dados, aes(x = x1, y = x2)) +
+   geom_point(aes(colour = Grupo))
```

Simulação



Simulação

```
> class <- kmeans(dados[, 1:2], 2)
> names(class)
```

```
## [1] "cluster"      "centers"      "totss"
## [4] "withinss"     "tot.withinss" "betweenss"
## [7] "size"         "iter"         "ifault"
```

Simulação

```
> head(class$cluster)

## [1] 1 1 1 1 1 2

> sum(class$cluster[1:N] == 1)/N

## [1] 0.845

> sum(class$cluster[(N+1):(2*N)] == 2)/N

## [1] 0.765
```

Simulação

```
> class$centers
```

```
##             x1         x2
## 1 -0.09553168 0.05964305
## 2  1.66385702 1.56815154
```

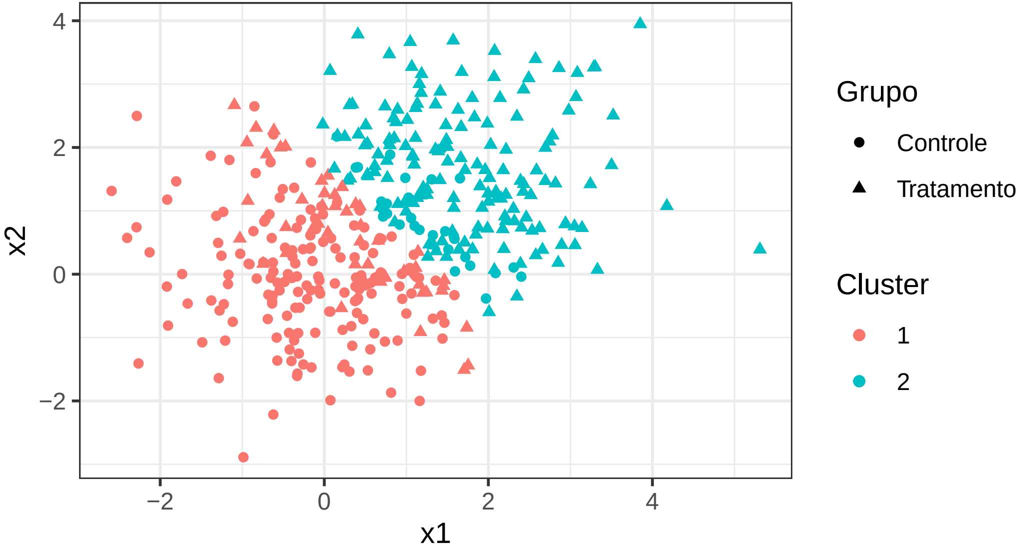
```
> class$size
```

```
## [1] 216 184
```

Simulação

```
> dados$Cluster <- as.factor(class$cluster)
> ggplot(dados, aes(x = x1, y = x2)) +
+   geom_point(aes(shape = Grupo,
+     colour = Cluster))
```

Simulação



Simulação

- Vamos mudar os parâmetros da simulação
- Teremos a seguinte matriz de covariâncias, dada por

$$\boldsymbol{\Sigma} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

- As médias das duas classes vão ser diferentes do caso anterior:

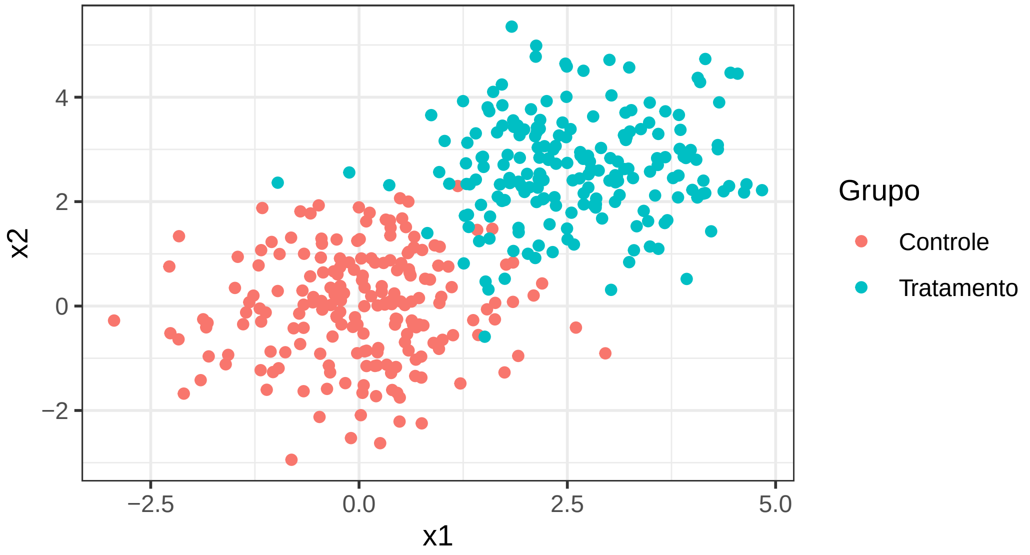
$$\boldsymbol{\mu}_1 = \begin{pmatrix} 0{,}0 \\ 0{,}0 \end{pmatrix} \text{ e } \boldsymbol{\mu}_2 = \begin{pmatrix} 2{,}5 \\ 2{,}5 \end{pmatrix}$$

- As prioridades *a priori* das duas classes são $p_1 = 0{,}5$ e $p_2 = 0{,}5$

Simulação

```
> N     <- 200
> mu1   <- c(0, 0)
> mu2   <- c(2.5, 2.5)
> Sigma <- matrix(c(1, 0, 0, 1), ncol = 2)
>
> ctrl <- rmvnorm(n = N, mean = mu1, sigma = Sigma)
> trt  <- rmvnorm(n = N, mean = mu2, sigma = Sigma)
>
> dados <- data.frame(rbind(ctrl, trt),
+     rep(c("Controle", "Tratamento"), each = N))
> names(dados) <- c("x1", "x2", "Grupo")
>
> ggplot(dados, aes(x = x1, y = x2)) +
+   geom_point(aes(colour = Grupo))
```

Simulação



Simulação

```
> class <- kmeans(dados[, 1:2], 2)
> names(class)

## [1] "cluster"      "centers"      "totss"
## [4] "withinss"     "tot.withinss" "betweenss"
## [7] "size"         "iter"         "ifault"
```

Simulação

```
> head(class$cluster)

## [1] 1 1 1 1 1 1

> sum(class$cluster[1:N] == 1)/N

## [1] 0.98

> sum(class$cluster[(N+1):(2*N)] == 2)/N

## [1] 0.96
```

Simulação

```
> class$centers
```

```
##           x1          x2
## 1 0.02883039 -0.01144434
## 2 2.63350228  2.66714325
```

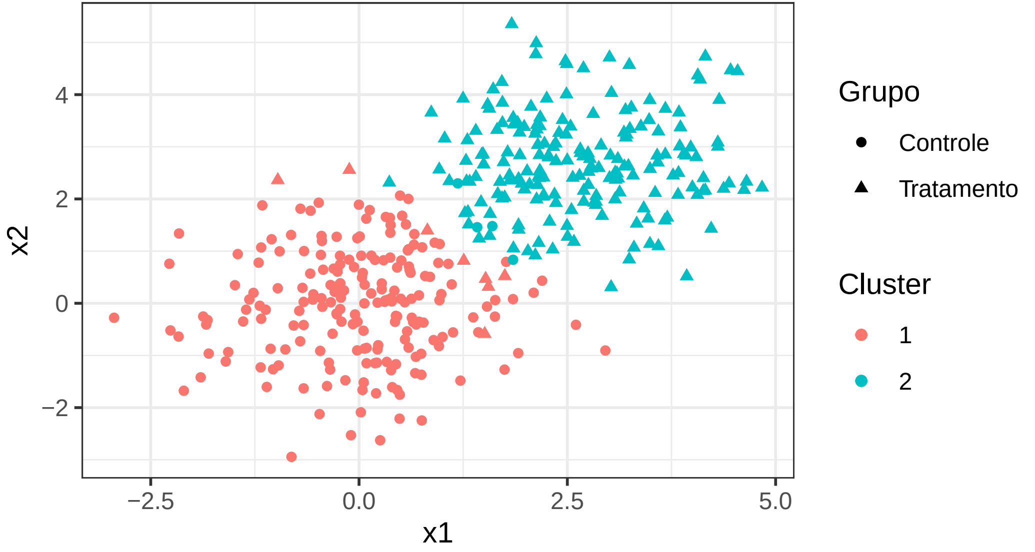
```
> class$size
```

```
## [1] 204 196
```

Simulação

```
> dados$Cluster <- as.factor(class$cluster)
> ggplot(dados, aes(x = x1, y = x2)) +
+   geom_point(aes(shape = Grupo,
+     colour = Cluster))
```

Simulação



Escolha do Número de Clusters

Escolha do Número de Clusters

- Já vimos como clusterizar um conjunto de dados
- Mas lembre-se que esta é uma atividade de aprendizagem não-supervisionada: precisamos determinar o número correto de clusters
- Essa é uma tarefa importante, porém muito difícil
- Existem alguns métodos bastante conhecidos para fazer isto e veremos dois deles

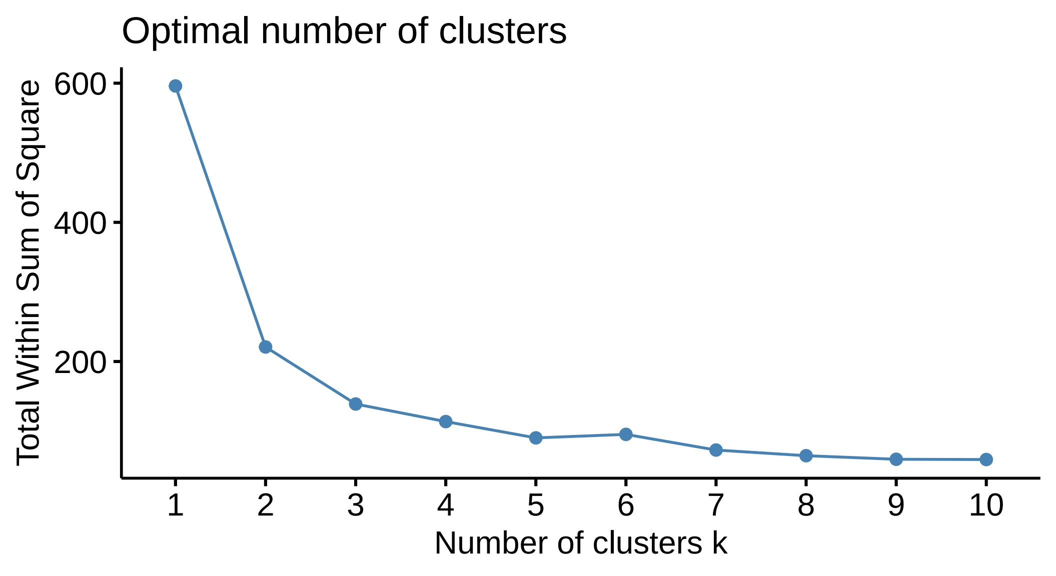
Escolha do Número de Clusters - Cotovelo

- Faça a clusterização dos dados para vários valores de k (digamos de 1 a 10)
- Para cada k, calcule a soma de quadrados dentro (wss)
- Faça o gráfico de wss em função do número de clusters k
- O valor ótimo de k é aquele em que a curva se estabiliza

Escolha do Número de Clusters - Cotovelo

```
> library(factoextra)
> library(NbClust)
>
> x <- scale(iris[, 1:4])
>
> fviz_nbclust(x, kmeans, method = "wss")
```

Escolha do Número de Clusters - Cotovelo



Escolha do Número de Clusters - Silhueta

- Assuma que os dados foram clusterizados em k clusters
- Para cada dado i, seja $a(i)$ a distância média entre i e todos os outros dados no mesmo cluster
- Seja $b(i)$ a menor distância média de i a todos os pontos em qualquer outro cluster que i não pertença
- A silhueta $s(i)$ do ponto i é definida como

$$s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}}$$

Escolha do Número de Clusters - Silhueta

- Esta expressão pode ser escrita como

$$s(i) = \left\{ \begin{array}{ll} 1 - a(i)/b(i), & \text{se } a(i) < b(i) \\ 0, & \text{se } a(i) = b(i) \\ b(i)/a(i) - 1, & \text{se } a(i) > b(i) \end{array} \right\}$$

- Ou seja, $-1 \leq s(i) \leq 1$
- Se $s(i)$ estiver próxima de 1, então i pertence ao cluster
- Se $s(i)$ estiver próxima de -1, então i não pertence ao cluster

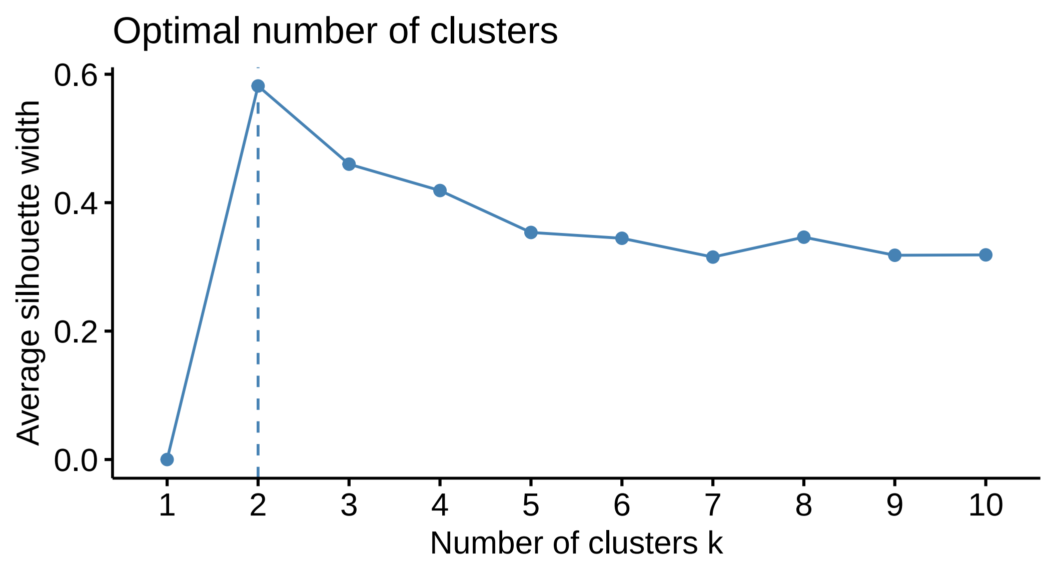
Escolha do Número de Clusters - Silhueta

- O valor médio de $s(i)$ sobre todos os dados em um cluster é uma medida de quão próximos os dados deste cluster estão
- Desta forma, a média dos $s(i)$ sobre todos os dados do conjunto de dados se torna uma medida de quão bem os dados foram clusterizados
- Portanto, quanto maior o valor de $s(i)$, melhor a clusterização

Escolha do Número de Clusters - Silhueta

```
> fviz_nbclust(x, kmeans, method = "silhouette")
```

Escolha do Número de Clusters - Silhueta



Aplicação

Aplicação

- Vamos aplicar o k-means no conjunto de dados iris
- Queremos ver como será o desempenho do algoritmo na identificação dos clusters
- Note que, normalmente, não utilizamos algoritmos de clusterização em conjuntos de dados para o qual conhecemos as classes
- Além disso, como a inicialização do algoritmo é aleatória, os seus resultados podem variar em relação aos apresentados aqui

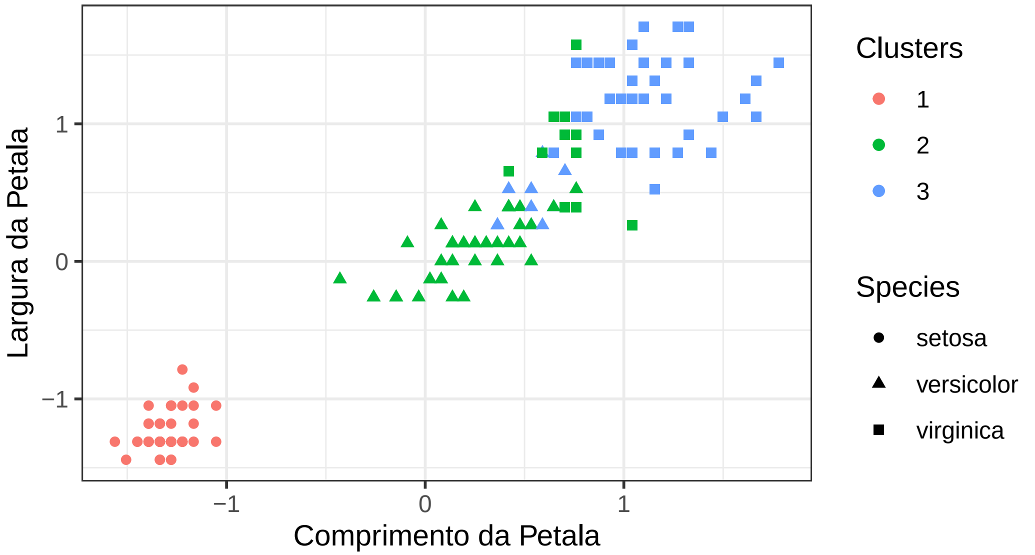
Aplicação

```
> x <- scale(iris[, 1:4])
>
> iris.kmeans <- kmeans(x, centers = 3)
>
> iris.pca <- prcomp(x, center = TRUE, scale. = TRUE)
>
> iris.plot <- data.frame(x,
+                         iris.pca$x,
+                         Species = iris$Species,
+                         Clusters = as.character(iris.kmeans$cluster))
```

Aplicação

```
> ggplot(iris.plot, aes(x = Petal.Length, y = Petal.Width)) +
+   geom_point(aes(shape = Species, colour = Clusters)) +
+   labs(x = "Comprimento da Petala", y = "Largura da Petala")
```

Aplicação

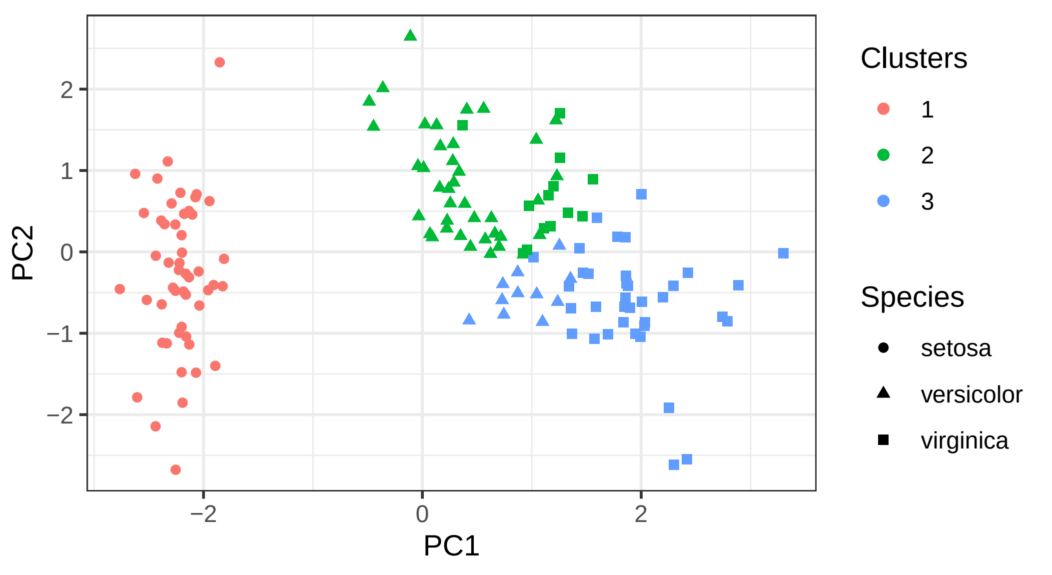


Largura da Petala
1
0
−1
−1
0
1
Comprimento da Petala
Clusters
1
2
3
Species
setosa
versicolor
virginica

Aplicação

```
> ggplot(iris.plot, aes(x = PC1, y = PC2)) +
+   geom_point(aes(shape = Species, colour = Clusters)) +
+   labs(x = "PC1", y = "PC2")
```

Aplicação



PC2
2
1
0
−1
−2
−2
0
2
PC1
Clusters
1
2
3
Species
setosa
versicolor
virginica

Exercícios

Exercícios

O conjunto de dados `vendas.csv` possui os dados de 30 clientes de uma loja especializada em móveis. Em particular, os dados fornecidos dizem respeito a vendas de mesas de jantar. As colunas disponíveis são:

- Idade: idade do cliente (em anos)
- TamanhoMesa: área da mesa comprada (em polegadas quadradas)
- ComprasPorAno: número de compras que o cliente faz na loja por ano
- DolaresPorCompra: quantidade de dólares que o cliente gasta em cada compra na loja

Exercícios

1. Faça a análise exploratória dos dados. É possível perceber algum padrão?
2. Determine o número ótimo de clusters utilizando o método do cotovelo
3. Determine o número ótimo de clusters utilizando o método da silhueta
4. Quantos perfis de clientes esta loja possui? Justifique.

Exercícios

5. Utilize o conjunto de dados `heptatlo.csv` para determinar se haviam grupos diferentes de atletas na disputa do heptatlo nas Olimpíadas de 1988 e quantos eram.
6. Como estes grupos poderiam ser classificados? Ou melhor, que nomes poderiam ser dados a estes grupos de modo que um leigo pudesse ser capaz de entender esta classificação?

k-means

EST0133 - Introdução à Modelagem de Big Data

Marcus Nunes
https://introbigdata.org/
https://marcusnunes.me/

Universidade Federal do Rio Grande do Norte