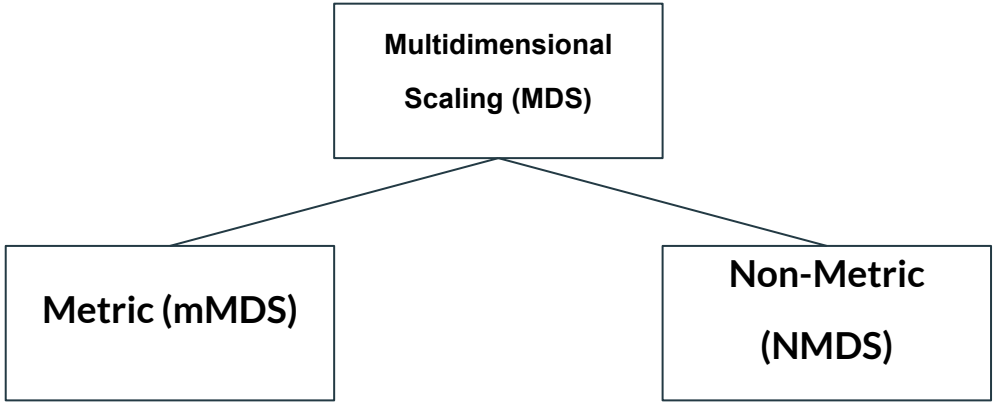
Escalonamento Multidimensional

Introdução

- Técnica de redução de dimensionalidade
- Utilizada para facilitar visualização
- **Transformar distâncias em um plano cartesiano**

Introdução

- Semelhante à análise de componentes principais (PCA), utilizando distâncias em vez de correlação
- Dividido em duas categorias



mMDS - Introdução

- Também conhecido como Análise de Coordenadas Principais (PCoA)
- A partir da matriz de distâncias, **D**, converter uma amostra p-dimensional em k-dimensional, k<p
- Utiliza-se os autovalores e autovetores de **D**
- Se **D** for calculada pela distância euclidiana o resultado será o mesmo da análise de componentes principais (PCA)
- Apenas variáveis numéricas

nMDS - Vantagens

Variedade de formas de se definir distâncias:

- Manhattan
- Mahalanobis
- Chebyshev
- Minkowski

E diversas outras formas específicas para cada problema. Ex: Utilizar distâncias por estrada em vez de geográfica (respeitando as propriedades de distância)

NMDS - Introdução

- Desenvolvido para resolver problemas ecológicos e psicométricos
- Método iterativo
- Dimensão k definida a priori
- Também utiliza matriz de distâncias
- Transforma distâncias em ranks (baseada em estatísticas não paramétrica)
- Cria transformação monotônica não-métrica

NMDS - Introdução

- Maior gasto computacional
- Muito bom para visualização, não para outras análises
- Admite variáveis categóricas
- Recomendado para medidas não matemáticas (notas; percepções; escalas psicométricas)
- Visualizar separação de clusters
- Muito utilizado em dados ambiente-espécie

NMDS - Metodologia

- Minimiza a medida de STRESS (STandard REsiduals Sum of Squares)
- Tem o objetivo de medir da distorção das novas distâncias com as distâncias originais
- A medida de STRESS mais utilizada é a de Kruskal

$$\text{Stress}_{Kruskal} = \sqrt{\frac{\sum_{i<j}(d_{ij} - \hat{d}_{ij})^2}{\sum_{i<j} d_{ij}^2}}$$

- Existem diversas outras medidas de STRESS, como Shepard e Sammon

NMDS - Passos

- Passo 1: Define-se k (quantidade de dimensões)
- Passo 2: Atribuir novas k coordenadas paras as variáveis originais (resultado do PCA ou mMDS, aleatório, posição geográfica)
- Passo 3: Calcular distâncias euclidianas entre as coordenadas
- Passo 4: Calcula o STRESS
- Passo 5: Verificar qualidade do ajuste (próximo slide), se sim o algoritmo é encerrado
- Passo 6: Obter novas estimativas das coordenadas pelo método de otimização steepest descent, a fim de diminuir o STRESS
- Passo 7: Voltar ao passo 3

NMDS - Qualidade do ajuste

Tabela 6.19: Valores de referência - *Stress*

Stress (%)	Qualidade de Ajuste
20	Ruim
10	Razoável
5	Bom
2,5	Excelente
0	Perfeito

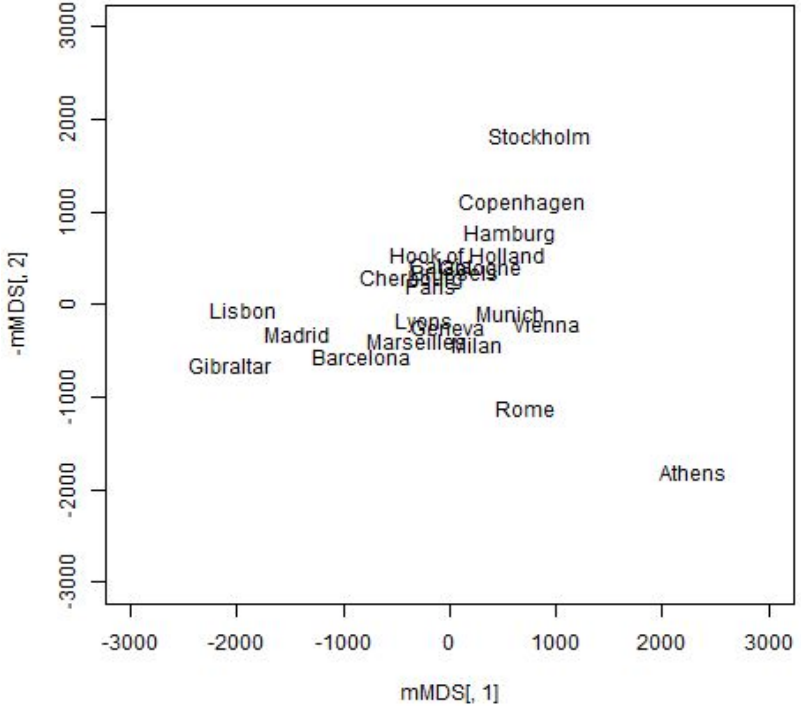
	nMDS	PCoA
Solução	Algoritmo iterativo de aproximação	Autoanálise da matriz de distância
Estabilidade da solução	Pode variar a cada vez que o algoritmo é usado, dependendo do ponto de início (aleatório)	Única. Pode ser usada como "semente" para outras técnicas
Tratamento das dissimilaridades	Distorcidas durante o cálculo, fazendo com que não reflitam as distâncias originais	Não são distorcidas
Construção da solução final ótima	Solução pode depender do critério (tipo de STRESS) usado; impossível saber a priori qual usar	Os primeiros eixos maximizam a variância das observações

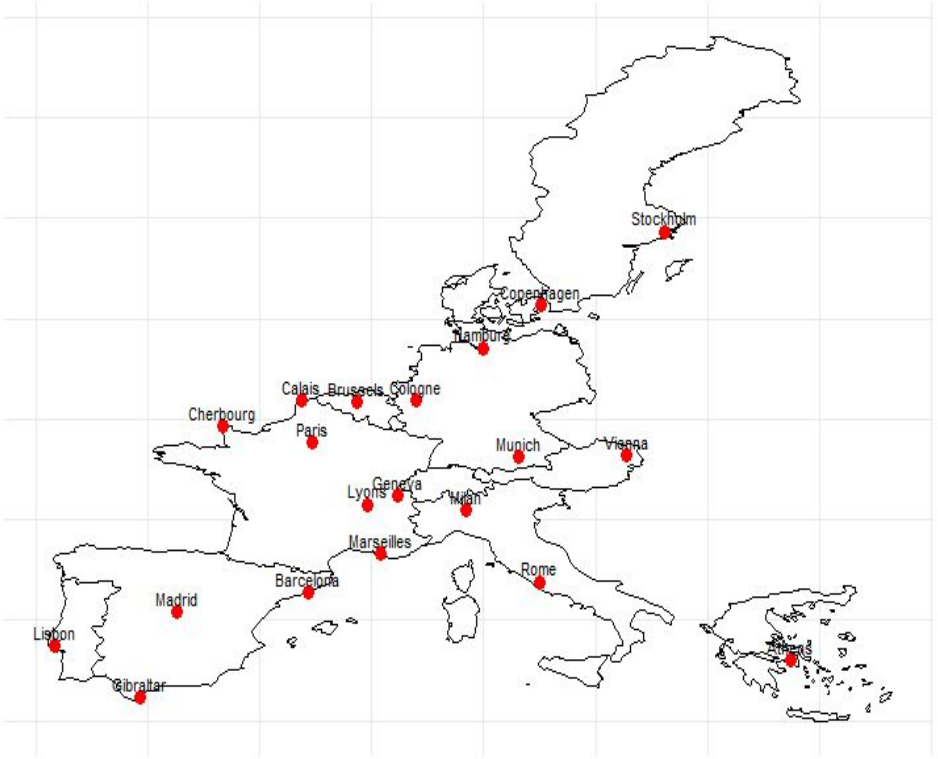
Aplicações

- O conjunto de dados ***eurodist*** em R fornece as distâncias rodoviárias entre 21 cidades europeias
- Para realizar o mMDS utilizaremos a função ***cmdscale*** encontrada no software R
- Para realizar o NMDS utilizaremos ***isoMDS***, do pacote MASS no software R

EURODIST - mMDS

```
mMDS = cmdscale(eurodist)
```





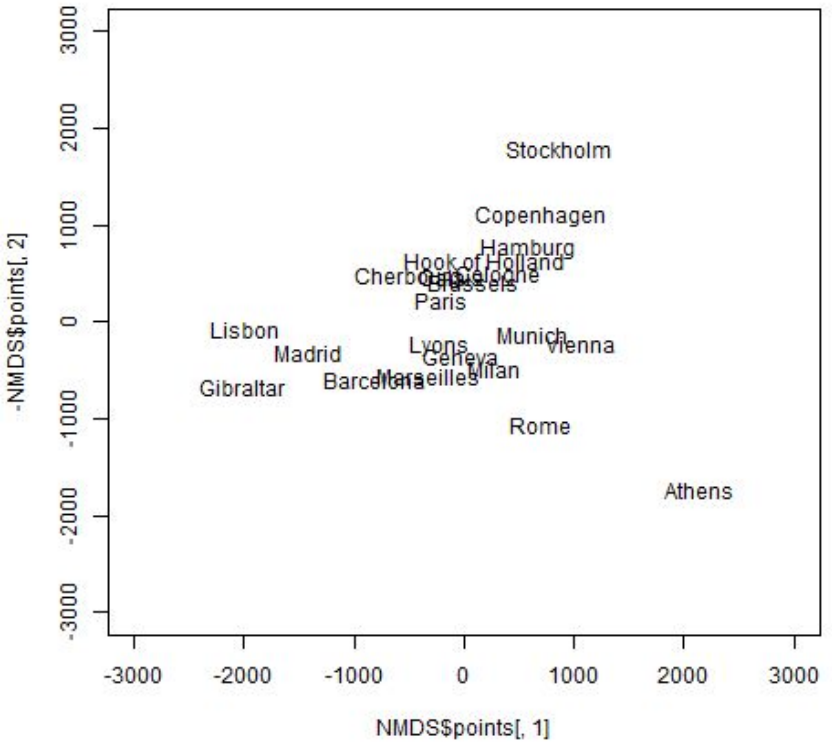
EURODIST - NMDS

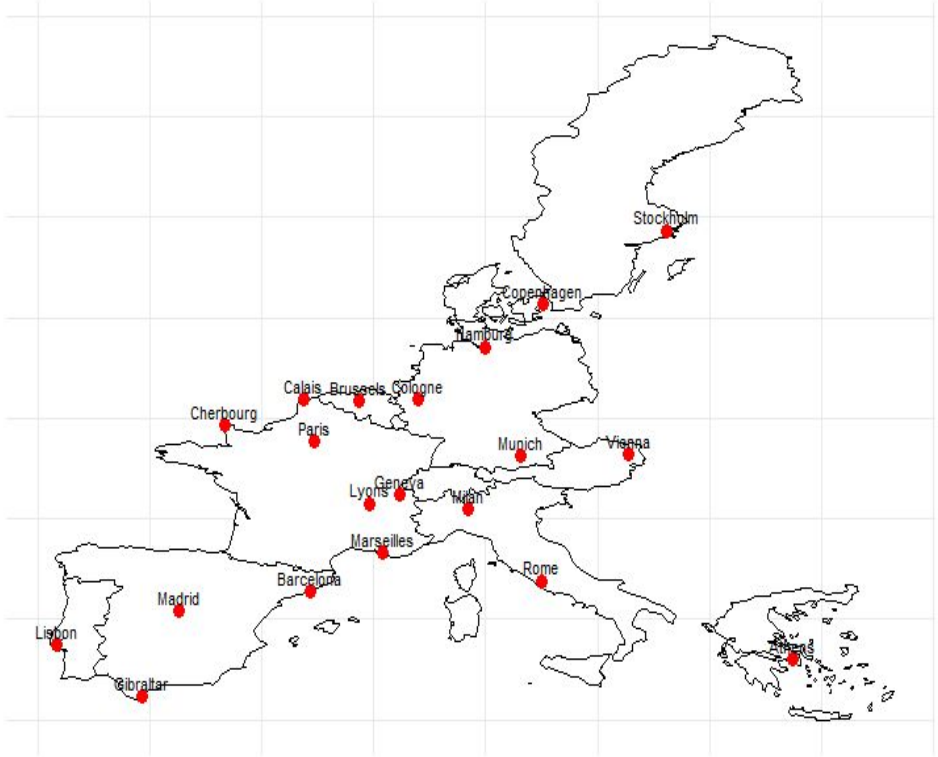
```
NMDS = MASS::isoMDS(eurodist, tol = 1E-6)
NMDS$stress

png("plot_mMDS.png")
plot(
  NMDS$points[,1], -NMDS$points[,2],
  col = "white",
  xlim=c(-3000,3000),
  ylim=c(-3000,3000)
)
text(
  NMDS$points[,1], -NMDS$points[,2],
  rownames(mMDS)
)
dev.off()
```

```
> NMDS = MASS::isoMDS(eurodist, tol = 1E-6)
initial  value 7.505733
iter   5 value 6.217447
iter  10 value 6.083762
final  value 6.013445
converged
> NMDS$stress
[1] 6.013445
```

EURODIST - NMDS





DUNA - NMDS

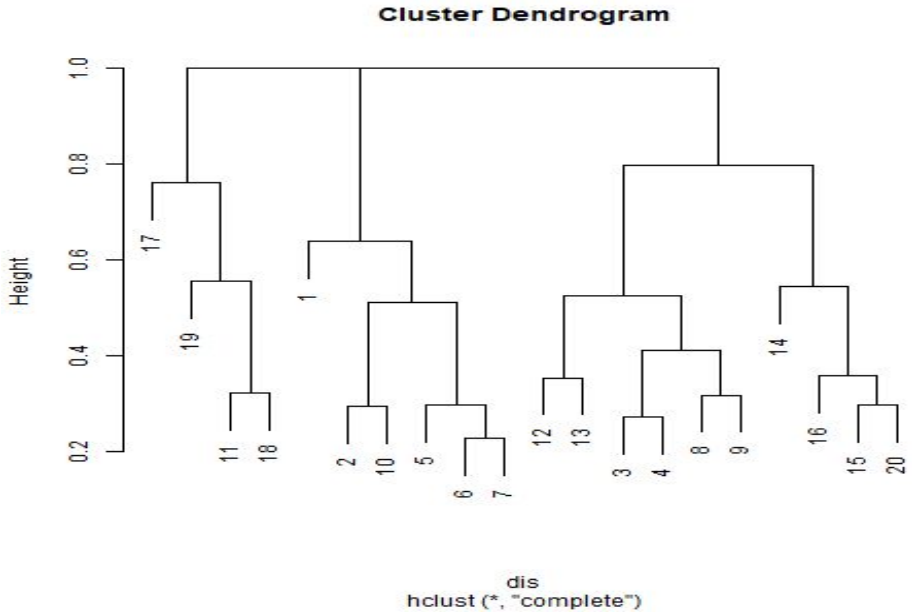
- Os dados de vegetação de prados dunares, duna, têm valores de classe de cobertura de 30 espécies em 20 locais
- O interesse é comparar a semelhança entre os locais com base nas espécies encontradas

DUNA - NMDS

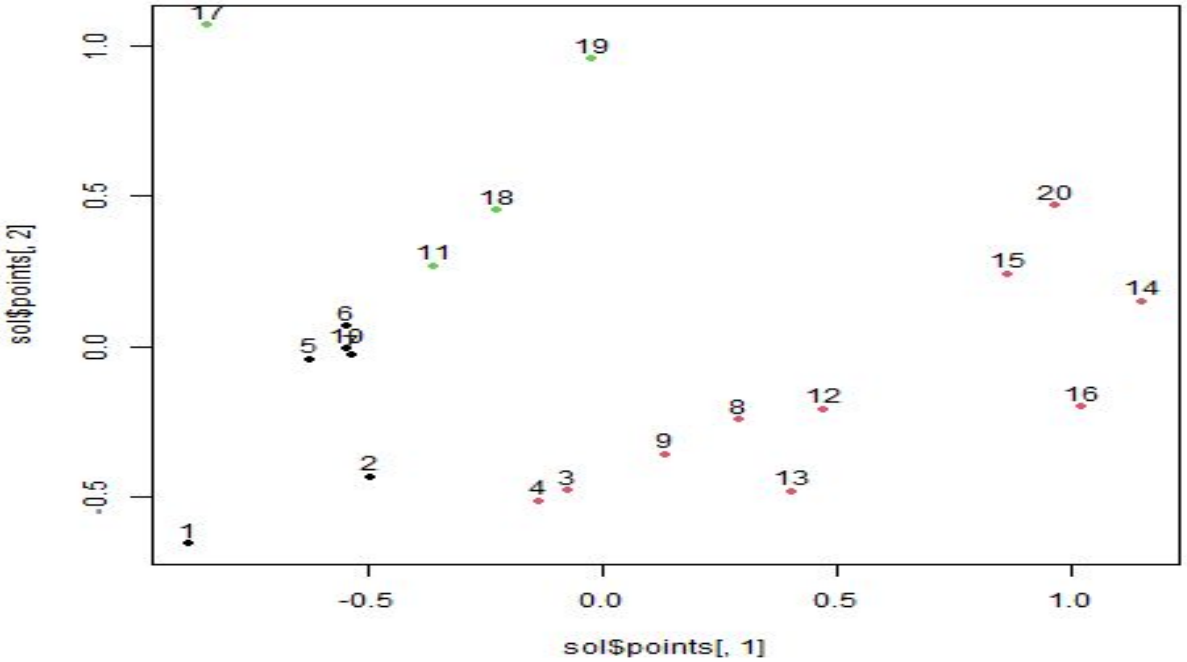
```r
library(vegan)
library(vegclust)

data(dune)
sol <- metaMDS(dune)
dis <- vegdist(dune, method = 'bray')
cluster <- hclust (d = dis, method = 'complete')
plot(cluster)

plot(
  sol$points[,1],
  sol$points[,2],
  pch = 20,
  col = cluster |>
    cutree(3)
  )
text(
  sol$points[,1],
  sol$points[,2]+0.05,
  labels = 1:20
)
```

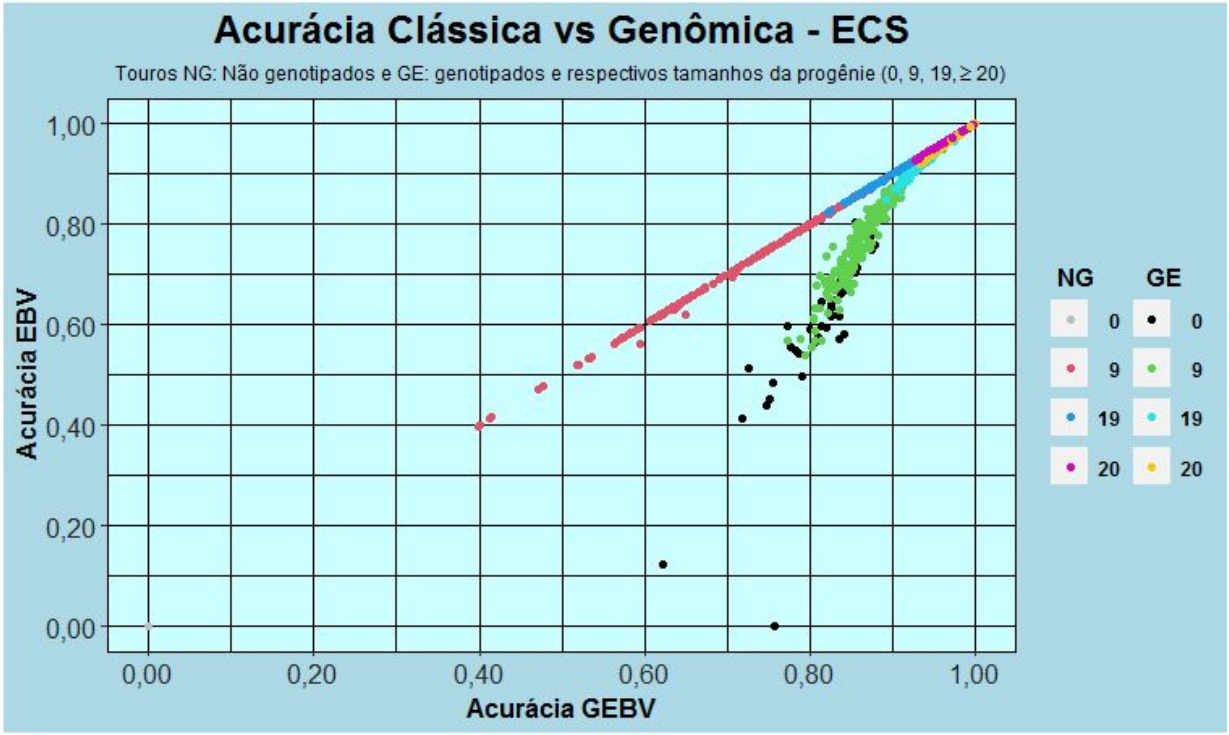


DUNA - NMDS

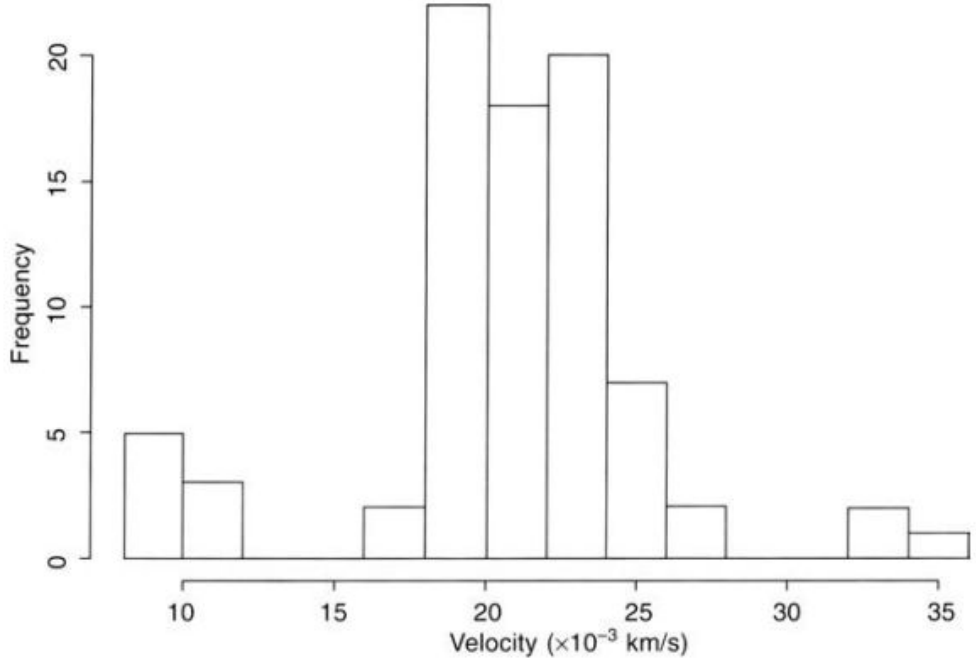

sol$points[, 2]
sol$points[, 1]
1.0
0.5
0.0
-0.5
-0.5
0.0
0.5
1.0
17
19
18
20
11
15
14
6
5
19
16
12
8
9
2
4
3
13
1

Visualização Gráfica

Gráfico de dispersão



Histograma



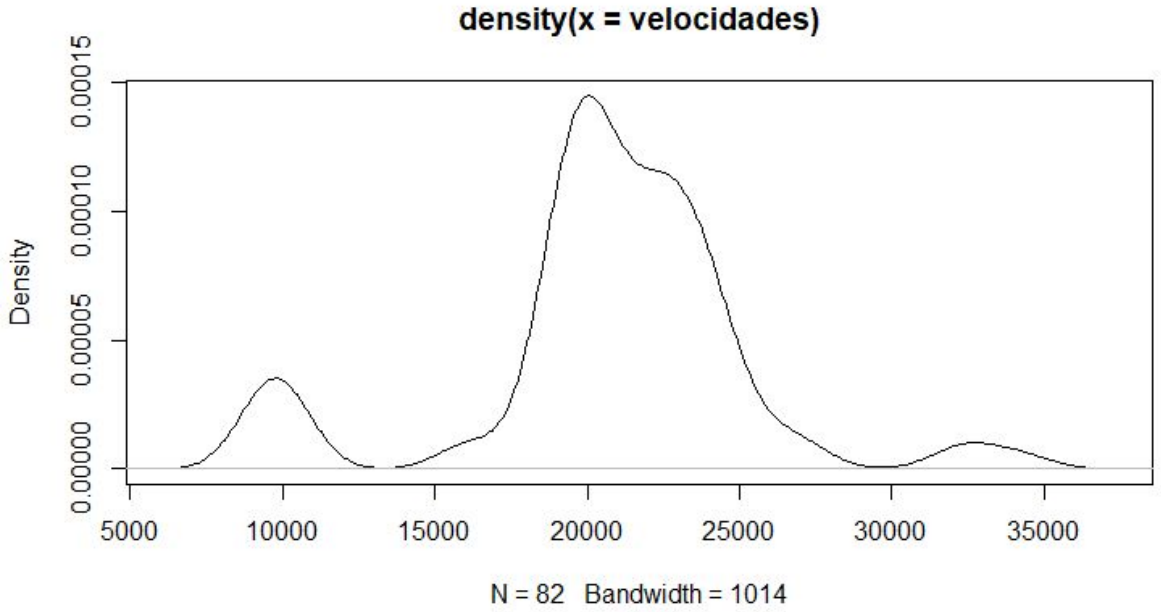
Densidade univariada

$$\hat{f}_h(x) = \frac{1}{n}\sum_{i=1}^{n} K_h(x - x_i) = \frac{1}{nh}\sum_{i=1}^{n} K(\frac{x - x_i}{h})$$

$$\frac{0.9 \times min(s, \frac{IQR}{1.34})}{n^{1/5}}$$

s = Desvio padrão amostral
IQR = distância interquartílica (quantil 75% - quantil 25%)
n = tamanho da amostra

```
plot(density(velocidades))
```



Densidade bivariada

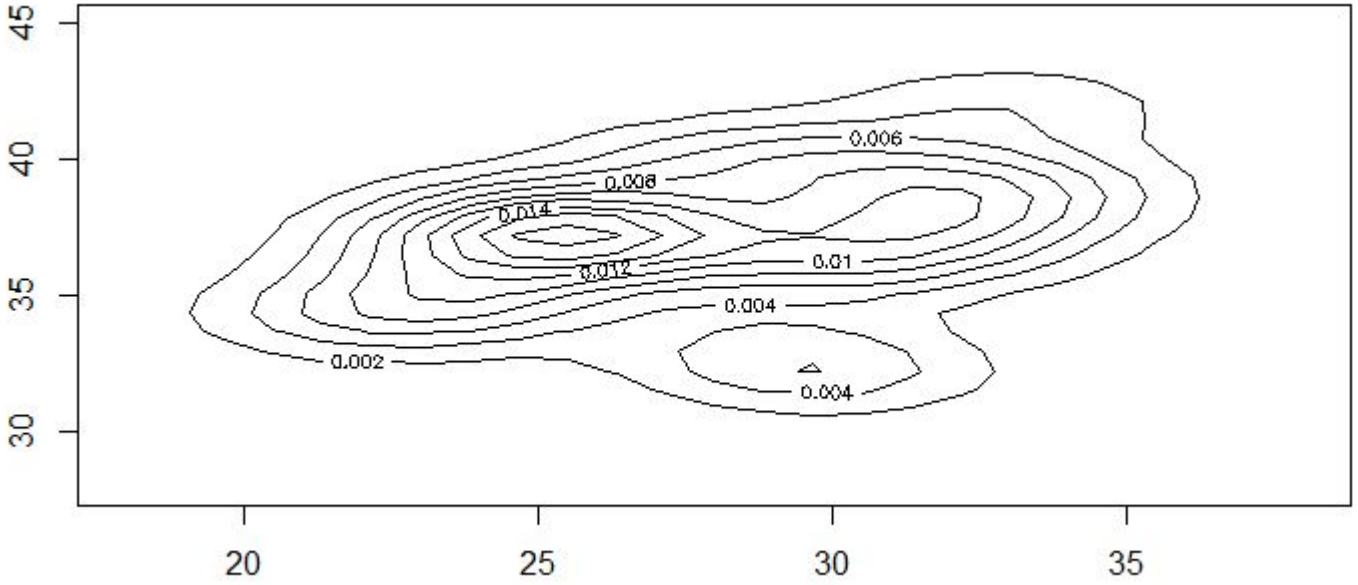
$$\frac{1}{nh_xh_y}\sum_{i=1}^{n}K(\frac{x-x_i}{h_x},\frac{y-y_i}{h_y})$$

$$\frac{4\times 1.06\times min(s,\frac{IQR}{1.34})}{n^{1/5}}$$

Table 2.2 Body measurements data (inches).

Subject	Chest	Waist	Hips
1	34	30	32
2	37	32	37
3	38	30	36
4	36	33	39
5	38	29	33
6	43	32	38
7	40	33	42
8	38	30	40
9	40	30	37
10	41	32	39
11	36	24	35
12	36	25	37
13	34	24	37
14	33	22	34
15	36	26	38
16	37	26	37
17	34	25	38
18	36	26	37
19	38	28	40
20	35	23	35

```
library(MASS)
contour(kde2d(ex$waist, ex$Hips, lims = c(18,38, 28, 45)))
```

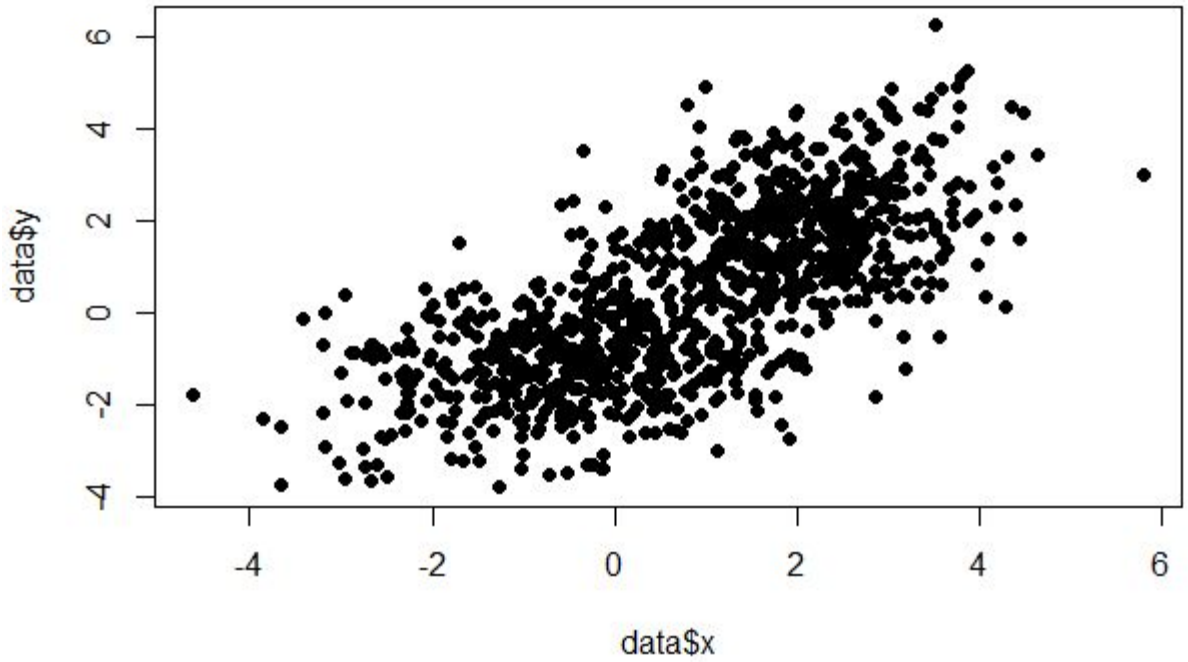


A densidade conjunta é muito importante, pois nem sempre as densidades univariadas (ou o gráfico de dispersão) nos permitem ver os grupos que a bivariada permite.

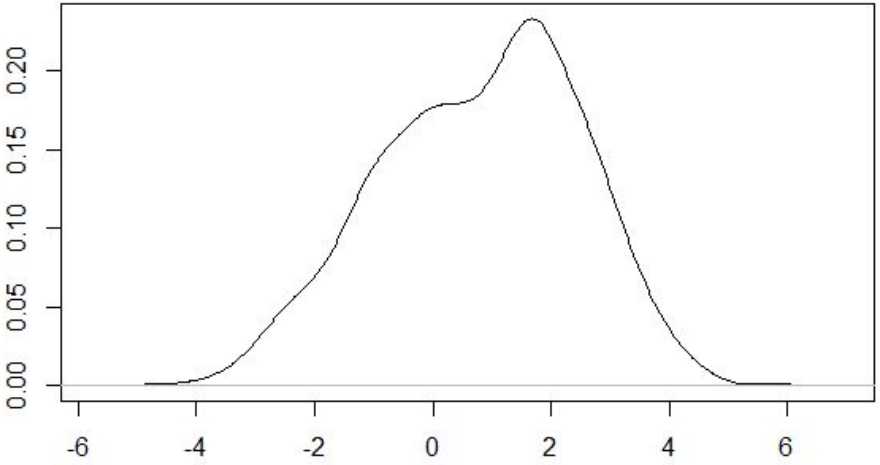
Para exemplo faremos uma simulação

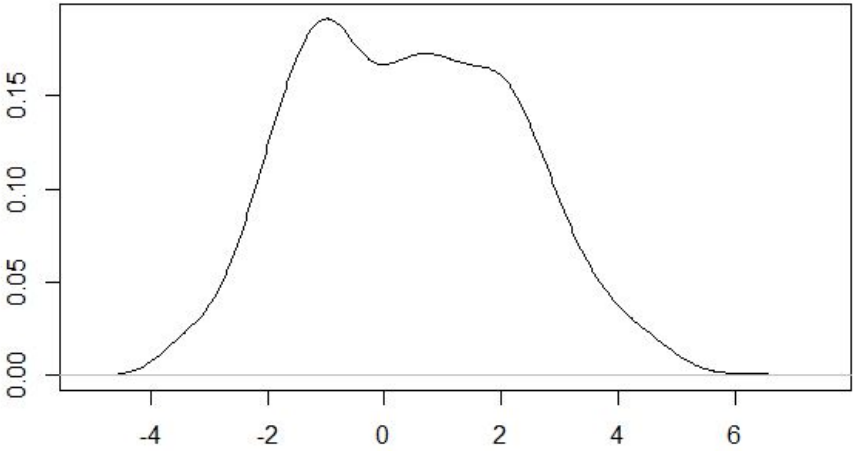
```
x1 <- rnorm(n, mean = 2)
x2 <- rnorm(n, mean = -2) + 3/4*x1
y1 <- 0.5 * x2 + rnorm(n, mean = 2)
y2 <- 0.5 * x1 + rnorm(n, mean = -2)

data <- data.frame(x = c(x1, x2), y = c(y1, y2))
```

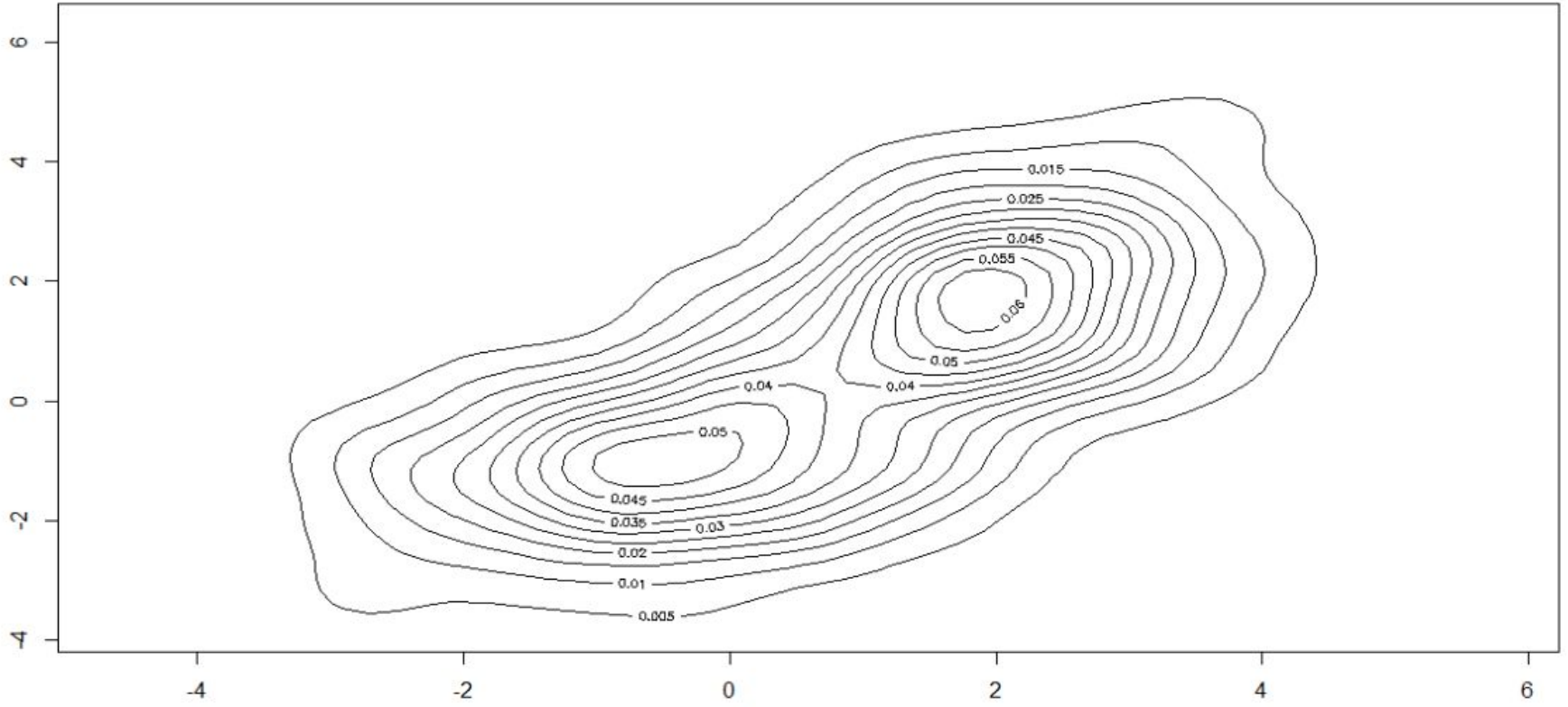

data$y
6
4
2
0
-2
-4
-4
-2
0
2
4
6
data$x

Densidades univariadas de x e y





Densidade bivariada de x e y



SNE - Stochastic Neighbor Embedding

$$p_{j|i} = \frac{\exp\left(-\|x_i - x_j\|^2 / 2\sigma_i^2\right)}{\sum_{k \neq i} \exp\left(-\|x_i - x_k\|^2 / 2\sigma_i^2\right)},$$

Transforma as distâncias euclidianas de alta dimensão em probabilidade condicionais

$$q_{j|i} = \frac{\exp\left(-\|y_i - y_j\|^2\right)}{\sum_{k \neq i} \exp\left(-\|y_i - y_k\|^2\right)}.$$

Transforma as distâncias euclidianas de baixa dimensão em probabilidade condicionais

$$C = \sum_i KL(P_i||Q_i) = \sum_i \sum_j p_{j|i} \log \frac{p_{j|i}}{q_{j|i}},$$

Uma medida de qualidade do ajuste é a divergência de KullbackLeib

$$\frac{\delta C}{\delta y_i} = 2 \sum_j (p_{j|i} - q_{j|i} + p_{i|j} - q_{i|j})(y_i - y_j).$$

$$Perp(P_i) = 2^{H(P_i)},$$

$$H(P_i) = -\sum_j p_{j|i} \log_2 p_{j|i}.$$

A perplexidade ajuda a definir um sigma bom, e é interpretado como uma forma de medir quantos vizinhos se tem efetivamente. Grandes valores de p buscam manter mais a estrutura global, enquanto pequenos valores focam na estrutura local

t-SNE (t-Distributed Stochastic Neighbor Embedding)

$$q_{ij} = \frac{(1 + \|y_i - y_j\|^2)^{-1}}{\sum_{k \neq l} (1 + \|y_k - y_l\|^2)^{-1}}.$$

Agora consideramos distribuições t-student a baixa dimensão com 1 grau de liberdade

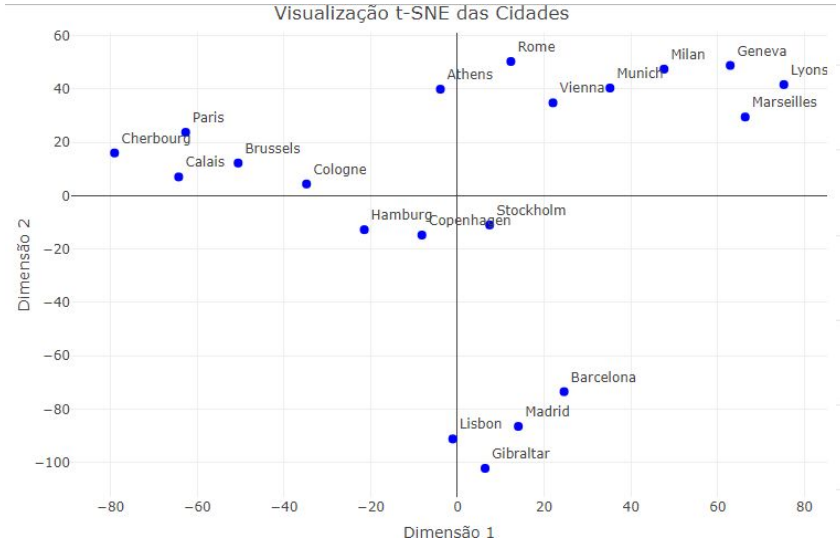
$(1 + \|y_i - y_j\|^2)^{-1}$ é uma aproximação de $\|y_i - y_j\|$

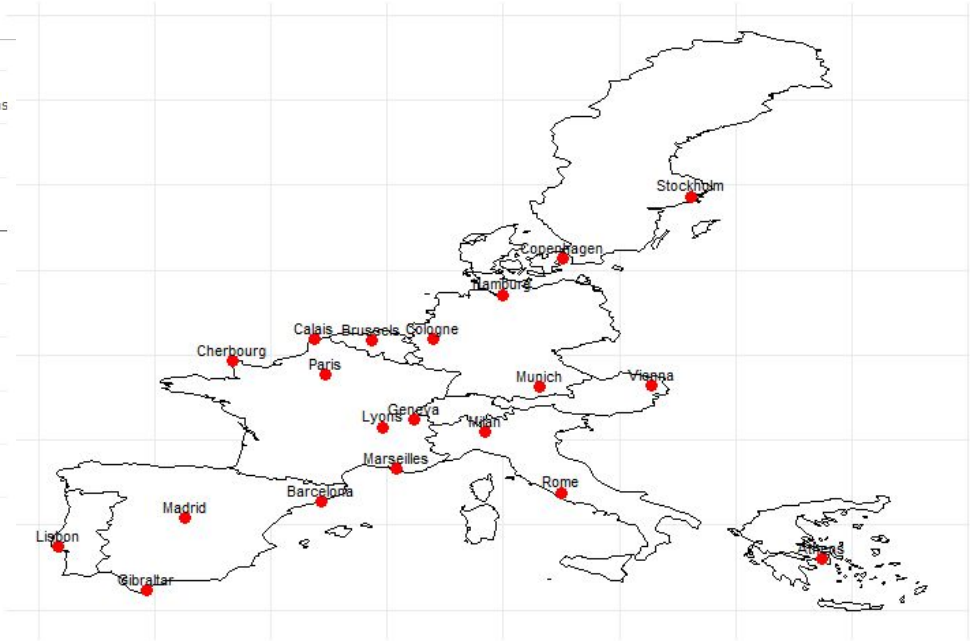
$$C = KL(P||Q) = \sum_i \sum_j p_{ij} \log \frac{p_{ij}}{q_{ij}}.$$

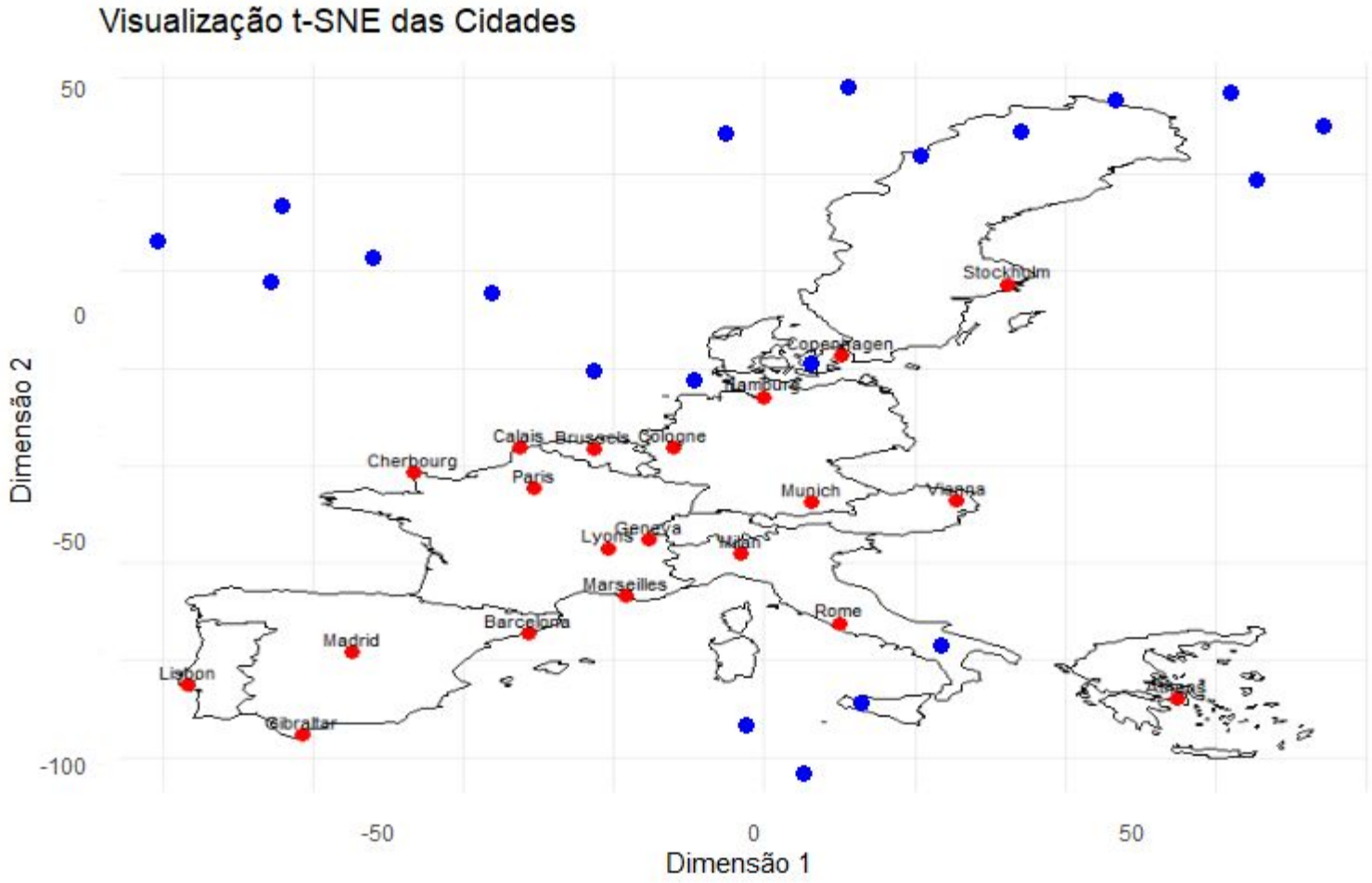
Ao invés de probabilidade condicional agora temos probabilidade conjunta

$$\frac{\delta C}{\delta y_i} = 4 \sum_j (p_{ij} - q_{ij})(y_i - y_j) (1 + \|y_i - y_j\|^2)^{-1}.$$

```
library(M3C)
tsne_result <- tsne(dist_matrix, perplex = 5)
```






Visualização t-SNE das Cidades
Dimensão 2
50
0
-50
-100
-50
0
50
Dimensão 1
Stockholm
Copenhagen
Hamburg
Calais
Brussels
Cologne
Cherbourg
Paris
Munich
Vienna
Lyons
Geneva
Milan
Marseilles
Rome
Barcelona
Madrid
Lisbon
Gibraltar

Referências

https://www.jmlr.org/papers/volume9/vandermaaten08a/vandermaaten08a.pdf

https://uw.pressbooks.pub/appliedmultivariatestatistics/chapter/nmds/

https://dvdscripter.wordpress.com/2016/02/13/nmds-como-funciona/

Cluster Analysis - Brian S. Everitt

Análise de dados através de métodos de estatística multivariada: uma abordagem aplicada - Sueli Aparecida Mingoti