

# **Métodos Multivariados de Análise de Dados\***

## **4ª Atividade**

Alberson da Silva Miranda

22 de outubro de 2024

\*Código disponível em [https://github.com/albersonmiranda/analise\\_multivariada](https://github.com/albersonmiranda/analise_multivariada).

# Índice

<b>1</b>	<b>Exercício 1</b>	<b>3</b>
1.1	Análise gráfica 3D dos dados e uma análise de estatística descritiva . . . . .	3
1.2	Análise de cluster . . . . .	5
1.3	Anova . . . . .	10
1.4	Método de K-means . . . . .	13

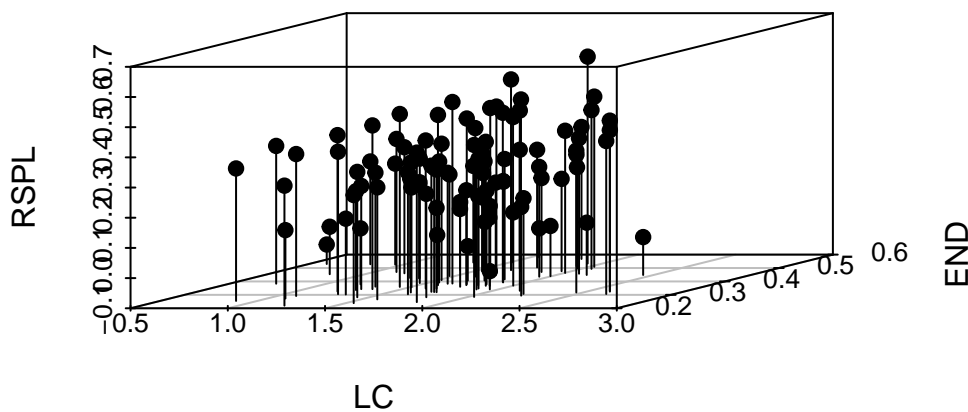
# 1 Exercício 1

## 1.1 Análise gráfica 3D dos dados e uma análise de estatística descritiva

```
1 # Carregando os dados
2 dados <- readxl::read_excel("data-raw/clustering/cluster.xlsx")
3
4 # plot 3D
5 scatterplot3d::scatterplot3d(dados[, 2:4], pch = 19, type = "h", main = "Gráfico 3D dos dados")
```

Warning: Unknown or uninitialised column: `color`.

**Gráfico 3D dos dados**



```
1 # estatísticas descritivas
2 summary(dados)
```

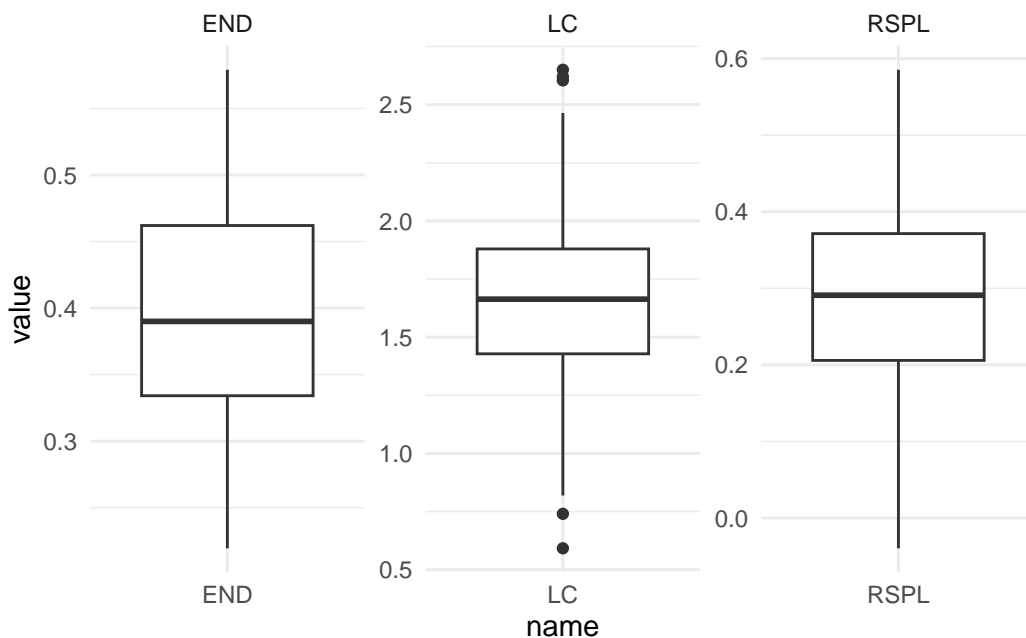
Empresa	LC	END	RSPL	
Min. :	1.00	Min. :0.5919	Min. :0.2195	Min. : -0.03963
1st Qu.:	25.75	1st Qu.:1.4283	1st Qu.:0.3341	1st Qu.: 0.20582

Median :	50.50	Median :	1.6634	Median :	0.3900	Median :	0.29101
Mean :	50.50	Mean :	1.6504	Mean :	0.3989	Mean :	0.27810
3rd Qu.:	75.25	3rd Qu.:	1.8794	3rd Qu.:	0.4621	3rd Qu.:	0.37147
Max. :	100.00	Max. :	2.6500	Max. :	0.5791	Max. :	0.58556

```

1 # boxplot
2 dados_long <- dados >
3   tidyr::pivot_longer(-Empresa)
4
5 dados_long >
6   ggplot(aes(x = name, y = value)) +
7     geom_boxplot() +
8     facet_wrap(~name, scales = "free") +
9     theme_minimal()

```



```

1 # histograma com linha da normal teórica
2 dados_long >
3   ggplot(aes(x = value)) +
4     geom_histogram(aes(y = after_stat(density))) +
5     facet_wrap(~name, scales = "free") +
6     geom_line(
7       aes(
8         y = dnorm(
9           value,

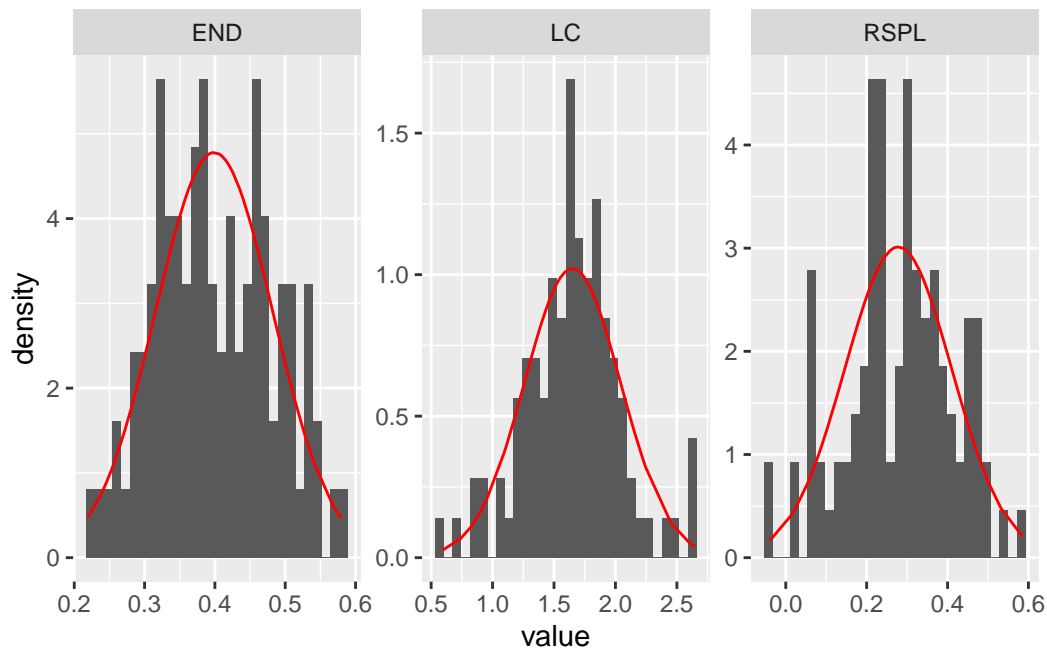
```

```

10     mean = tapply(value, name, mean, na.rm = TRUE)[PANEL],
11     sd = tapply(value, name, sd, na.rm = TRUE)[PANEL]
12   )
13 },
14   color = "red"
15 )

```

``stat_bin()`` using ``bins = 30``. Pick better value with ``binwidth``.



## 1.2 Análise de cluster

Com *single linkage*, o dendograma não parece muito promissor, com o primeiro *split* já não trazendo muito ganho de informação:

```

1 # Carregando os pacotes
2 library(mlr3verse)

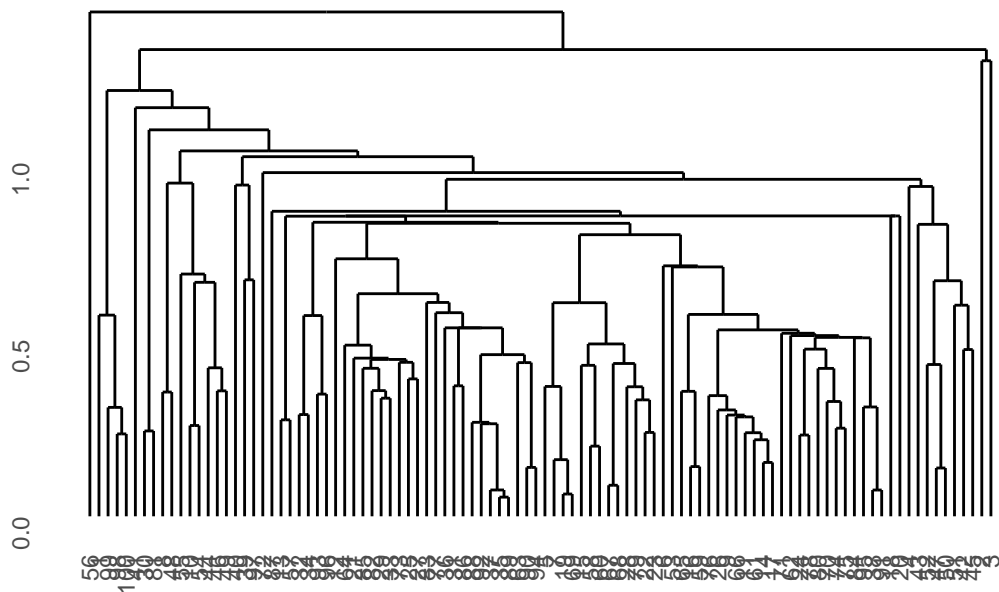
```

Loading required package: mlr3

```

1 library(mlr3cluster)
2
3 # normalizando os dados
4 dados_scale <- as.data.frame(scale(dados[, 2:4]))
5
6 # Criando tarefa
7 task <- TaskClust$new(id = "cluster", backend = dados_scale)
8
9 # criando learner
10 learner <- lrn(
11   "clust.hclust",
12   distmethod = "euclidean",
13   method = "single"
14 )
15
16 # treinando o modelo
17 model <- learner$train(task)
18
19 # plotando dendograma
20 autoplot(model)

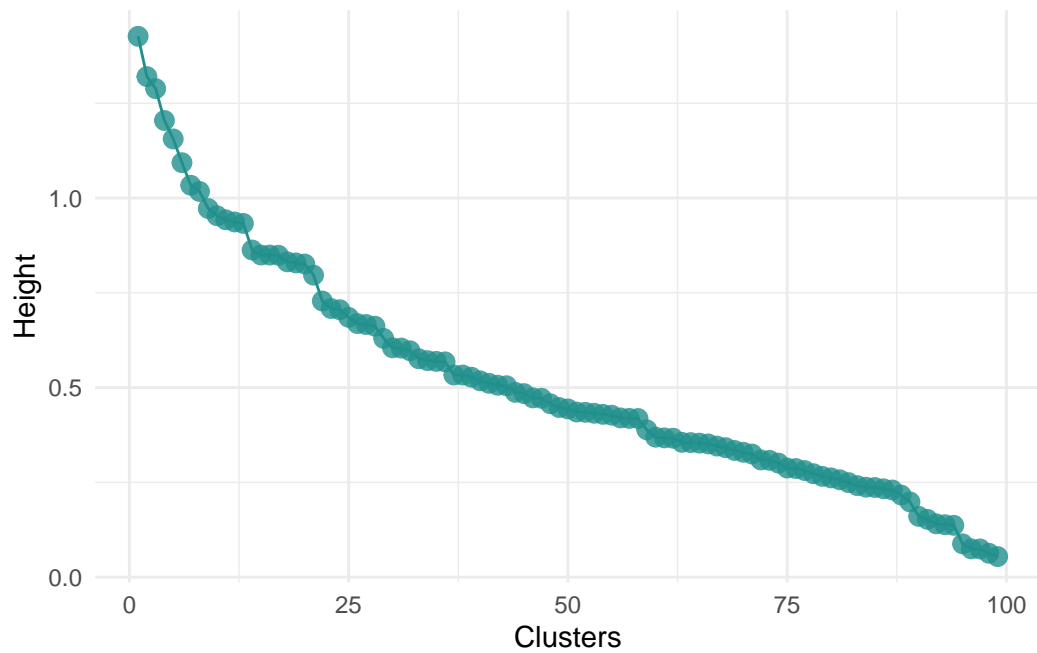
```



```

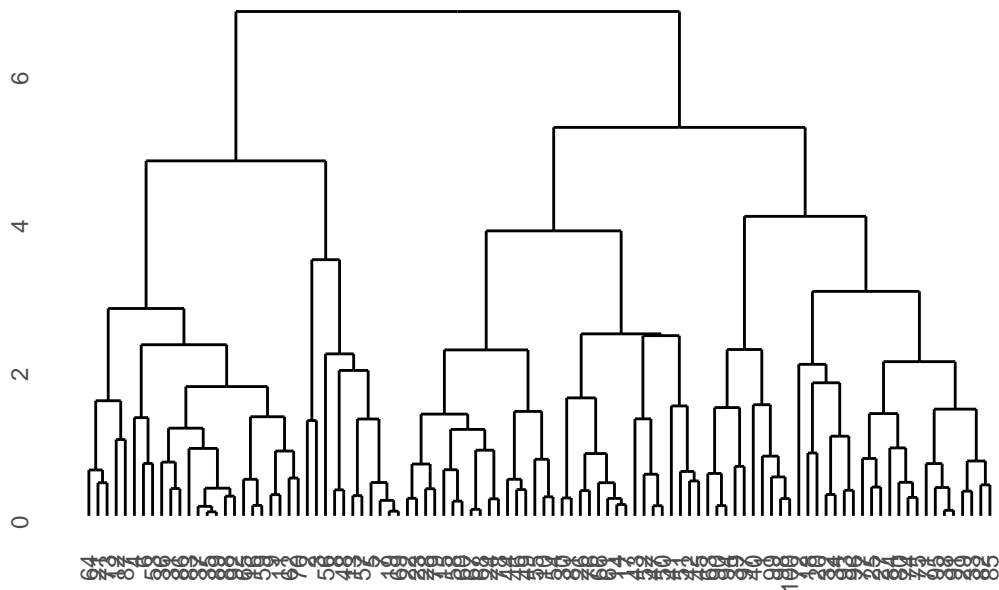
1 # plotando altura x número de clusters
2 autoplot(model, type = "scree")

```

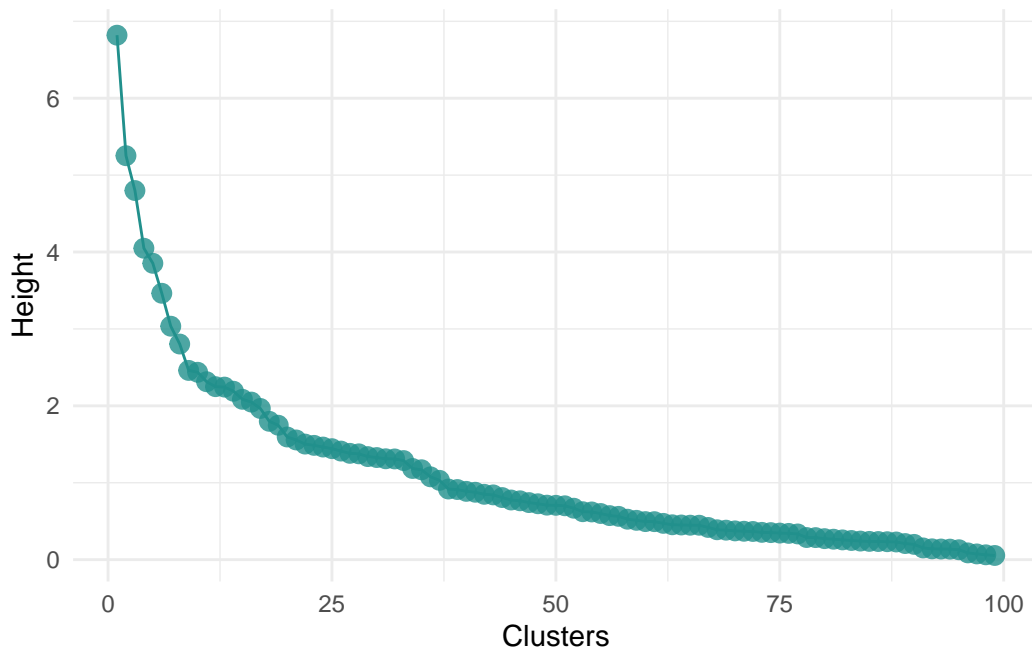


Já com o método *complete*, o dendograma é mais coerente, sugerindo 3 clusters:

```
1 # criando learner
2 learner <- lrn(
3   "clust.hclust",
4   distmethod = "euclidean",
5   method = "complete"
6 )
7
8 # treinando o modelo
9 model <- learner$train(task)
10
11 # plotando dendograma
12 autoplot(model)
```



```
1 # plotando altura x número de clusters
2 autoplot(model, type = "scree")
```



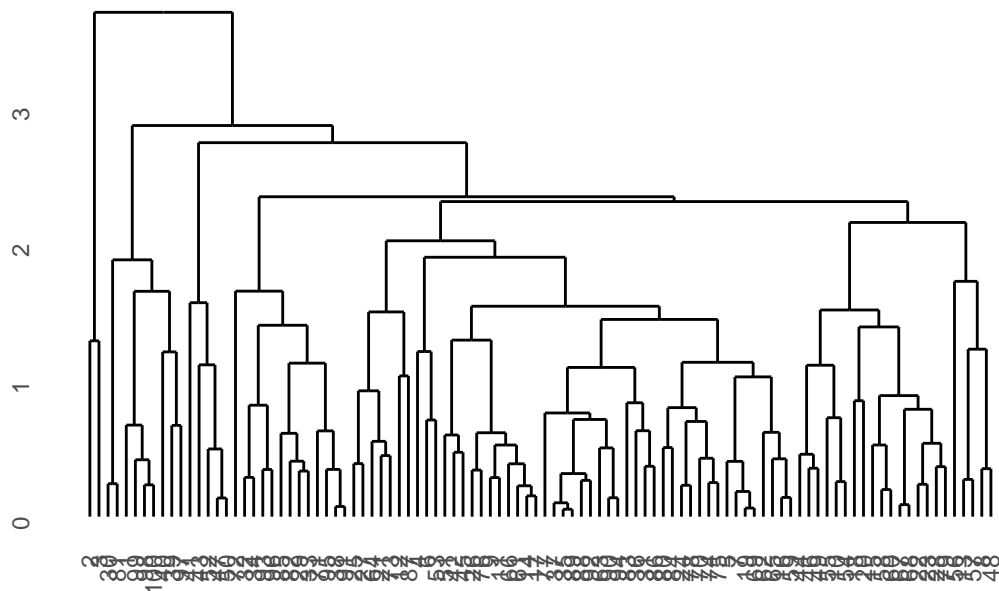
No método *average*, o dendrograma também sugere 3 clusters:



```

1 # criando learner
2 learner <- lrn(
3   "clust.hclust",
4   distmethod = "euclidean",
5   method = "average"
6 )
7
8 # treinando o modelo
9 model <- learner$train(task)
10
11 # plotando dendograma
12 autoplot(model)

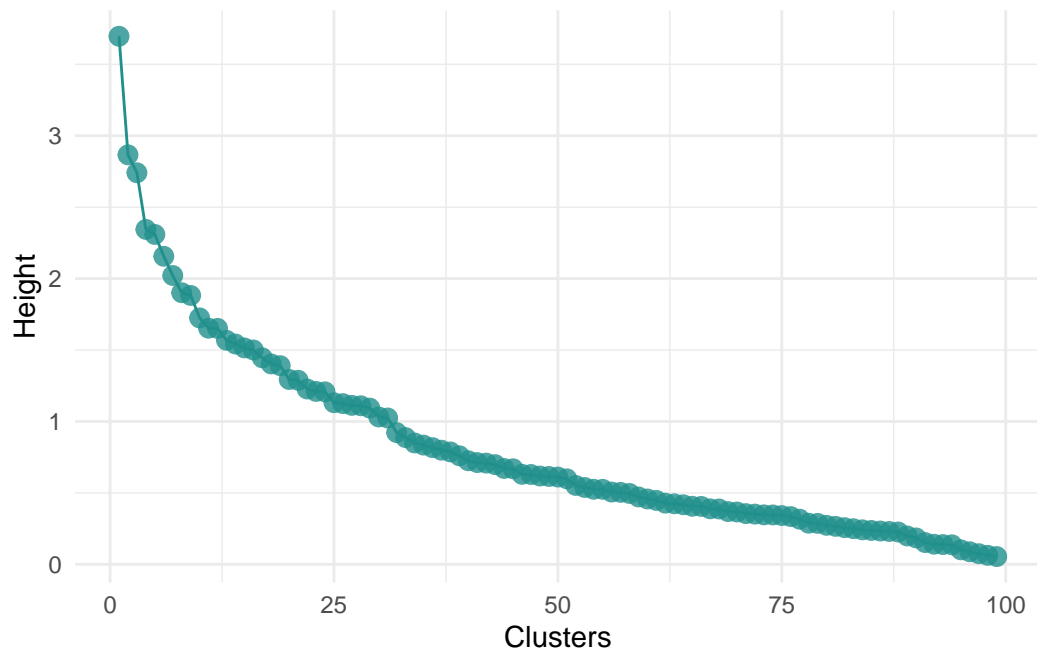
```



```

1 # plotando altura x número de clusters
2 autoplot(model, type = "scree")

```



### 1.3 Anova

```
1 # criando learner
2 learner <- lrn(
3   "clust.hclust",
4   distmethod = "euclidean",
5   method = "complete",
6   k = 3
7 )
8
9 # treinando o modelo
10 model <- learner$train(task)
11
12 # predição
13 pred <- model$predict(task)
```

Warning: Learner 'clust.hclust' doesn't predict on new data and predictions may not make sense on new data.

```

1 # visualizando os clusters
2 autoplot(pred, task)

```

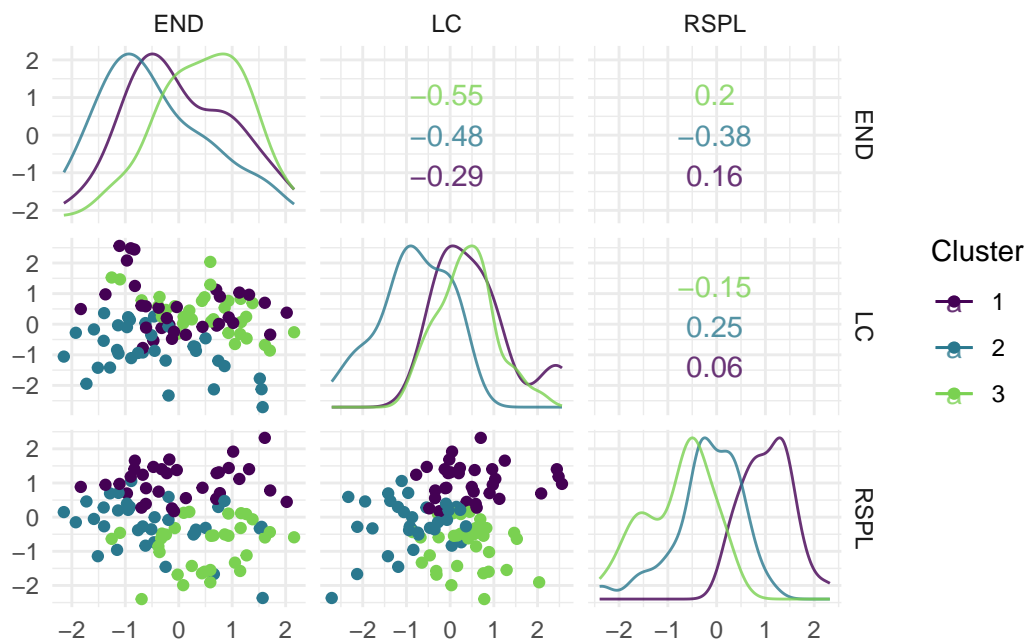
Registered S3 method overwritten by 'GGally':

```

method from
+.gg    ggplot2

```

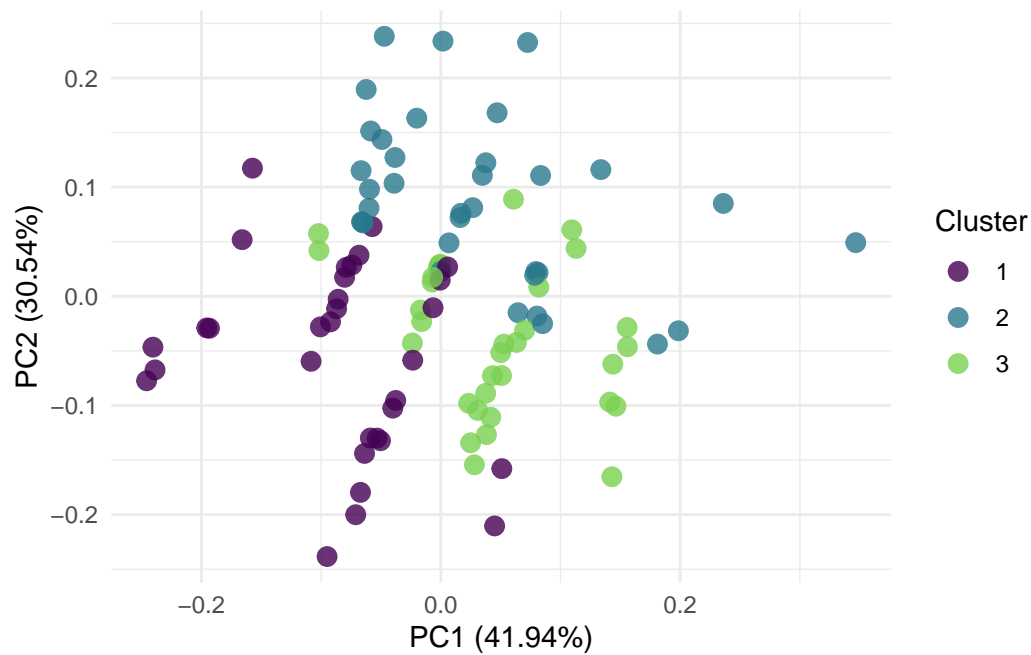
Warning in GGally::ggscatmat(data, color = "partition"): Factor variables are omitted in plot



```

1 autoplot(pred, task, type = "pca")

```



```

1 # bind com dados
2 dados_pred <- cbind(dados, as.data.table(pred))
3
4 # ANOVA para variável LC
5 aov(LC ~ partition, data = dados_pred) > summary()

```

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
partition	1	0.051	0.05131	0.334	0.565
Residuals	98	15.057	0.15364		

```

1 # ANOVA para variável END
2 aov(END ~ partition, data = dados_pred) > summary()

```

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
partition	1	0.0236	0.023574	3.467	0.0656 .
Residuals	98	0.6663	0.006799		

---

Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

```

1 # ANOVA para variável RSPL
2 aov(RSPL ~ partition, data = dados_pred) > summary()

```

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
partition	1	0.9804	0.9804	126.9	<2e-16 ***
Residuals	98	0.7570	0.0077		

---  
 Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

Para a variável LC e END, a ANOVA sugere que não há diferenças significativas entre os clusters. Já para a variável RSPL, há diferenças significativas entre os clusters.

## 1.4 Método de K-means

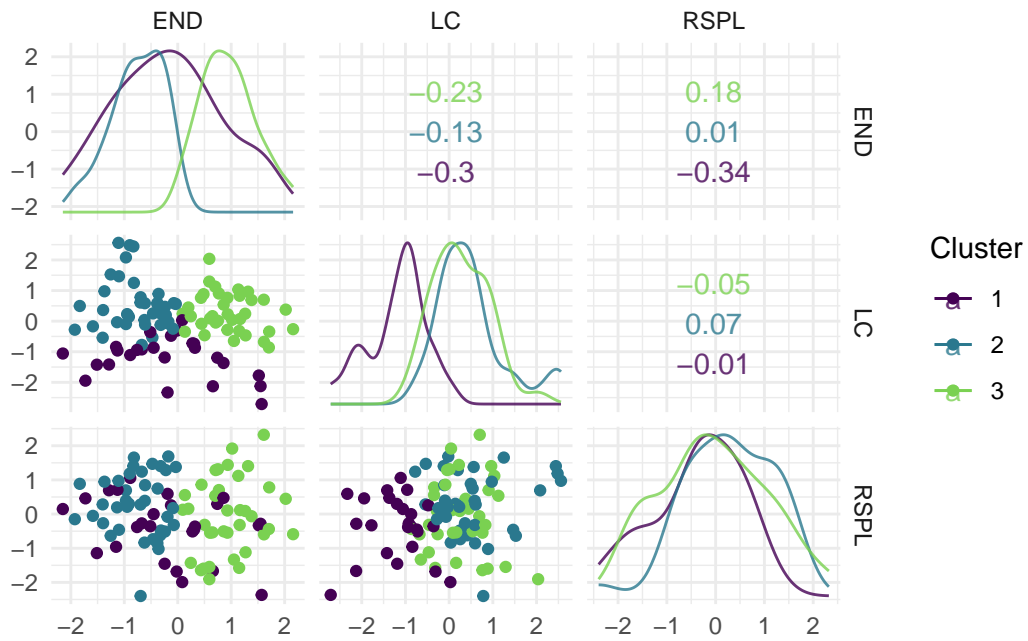
Utilizando agrupamento não hierarquico, podemos testar o método de K-means:

```

1 # criando learner
2 learner <- lrn(
3   "clust.kmeans",
4   centers = 3
5 )
6
7 # treinando o modelo
8 model <- learner$train(task)
9
10 # predição
11 pred <- model$predict(task)
12
13 # visualizando os clusters
14 autoplot(pred, task)

```

Warning in GGally::ggscatmat(data, color = "partition"): Factor variables are omitted in plot



```

1 # bind com dados
2 dados_pred <- cbind(dados, as.data.table(pred))
3
4 # ANOVA para variável LC
5 aov(LC ~ partition, data = dados_pred) > summary()

```

```

      Df Sum Sq Mean Sq F value    Pr(>F)
partition    1  3.991    3.991    35.19 4.51e-08 ***
Residuals   98 11.117     0.113
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```

1 # ANOVA para variável END
2 aov(END ~ partition, data = dados_pred) > summary()

```

```

      Df Sum Sq Mean Sq F value    Pr(>F)
partition    1 0.1793  0.17933    34.42 6.03e-08 ***
Residuals   98 0.5106  0.00521
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```

1 # ANOVA para variável RSPL
2 aov(RSPL ~ partition, data = dados_pred) > summary()

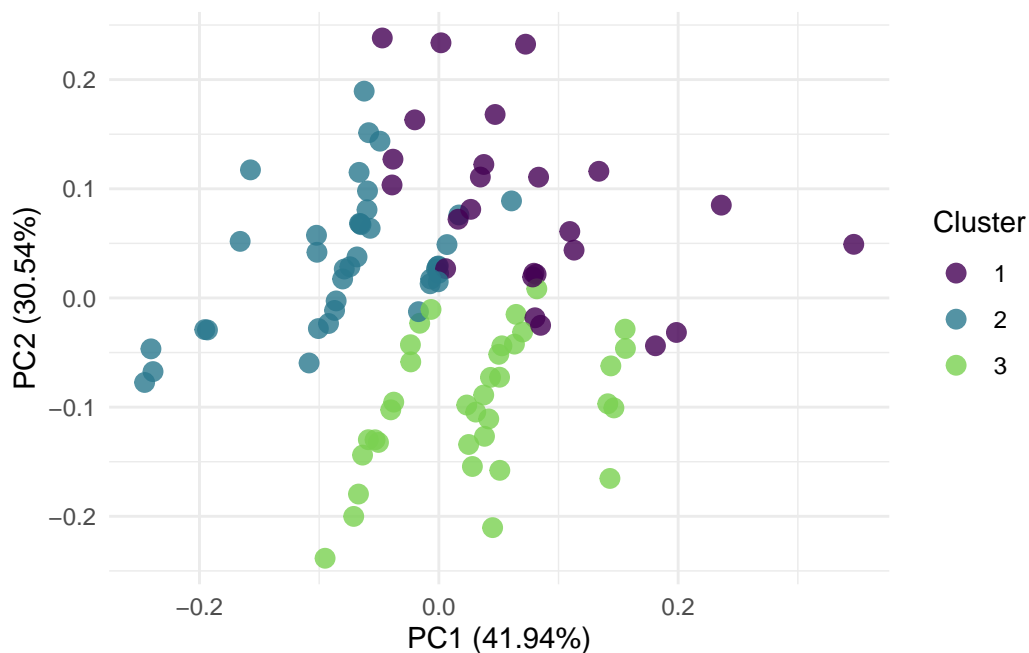
```

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
partition	1	0.0197	0.01973	1.125	0.291
Residuals	98	1.7177	0.01753		

```

1 # visualizando os clusters
2 autoplot(pred, task, type = "pca")

```



Usando método K-Means, a relação de significância da ANOVA se inverteu: para LC e END, há diferenças significativas entre os clusters, enquanto para RSPL, não há diferenças significativas.

```

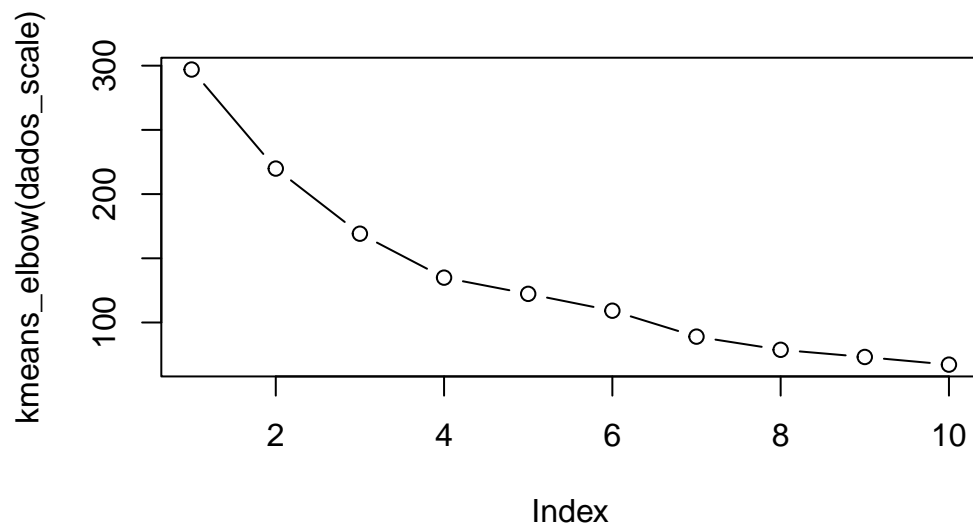
1 # definição de k ótimo via elbow
2 kmeans_elbow <- function(data, k_max = 10) {
3   wss <- numeric(k_max)
4   for (k in 1:k_max) {
5     model <- kmeans(data, centers = k)
6     wss[k] <- model$tot.withinss
7   }
8   return(wss)
9 }
10

```

```

11 # plotando elbow
12 dados_scale > kmeans_elbow() > plot(type = "b")

```

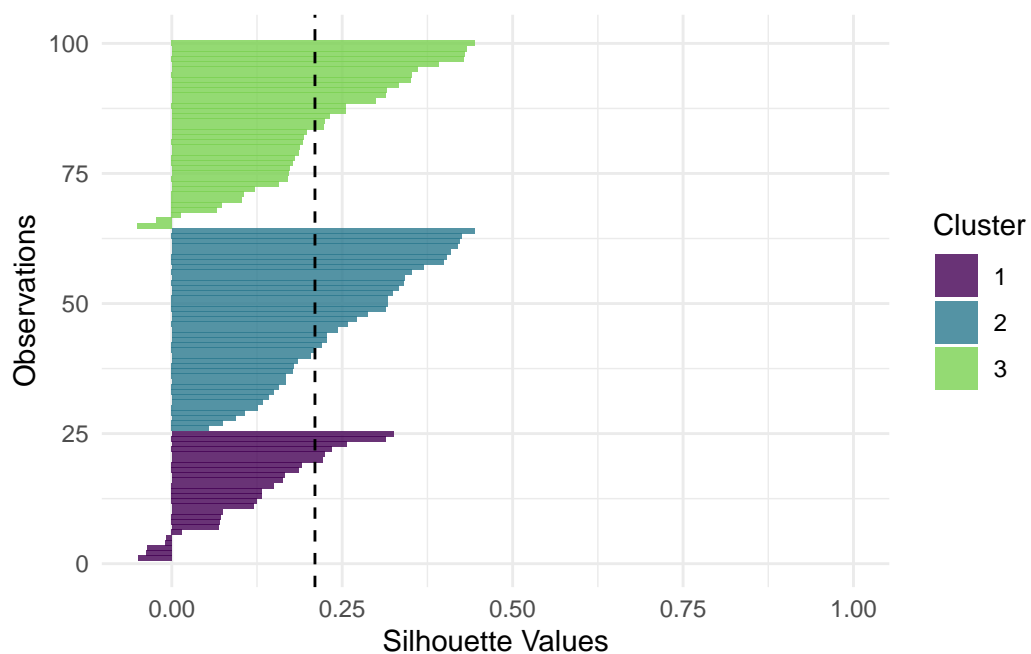


Por esse método, até 4 clusters poderiam ser considerados.

```

1 # silhouette plot
2 autoplot(pred, task, type = "sil")

```



Poucas observações estão negativas, sugerindo que o método obteve separação aceitável.