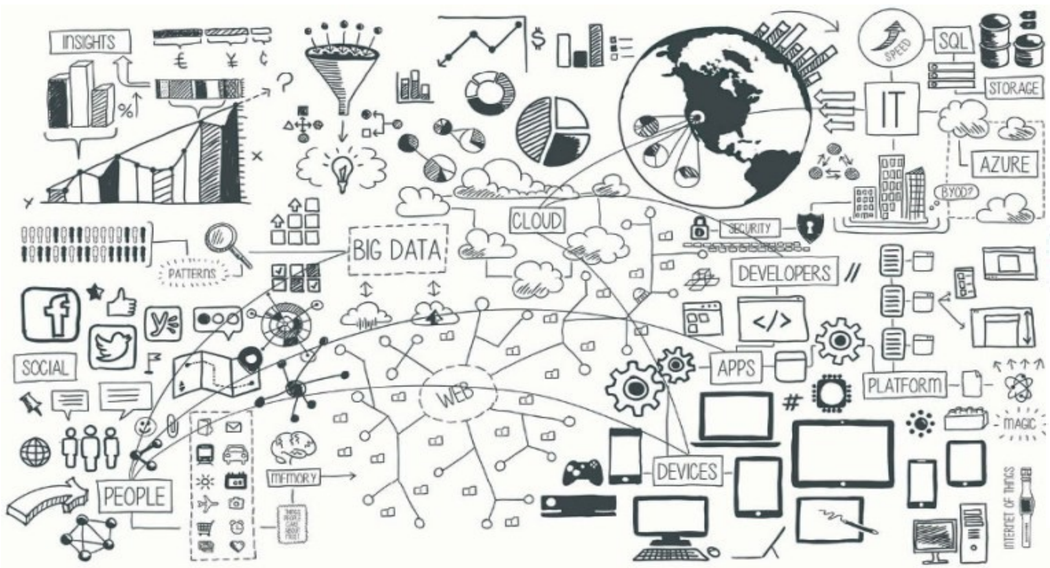
Data Mining (Minería de Datos)

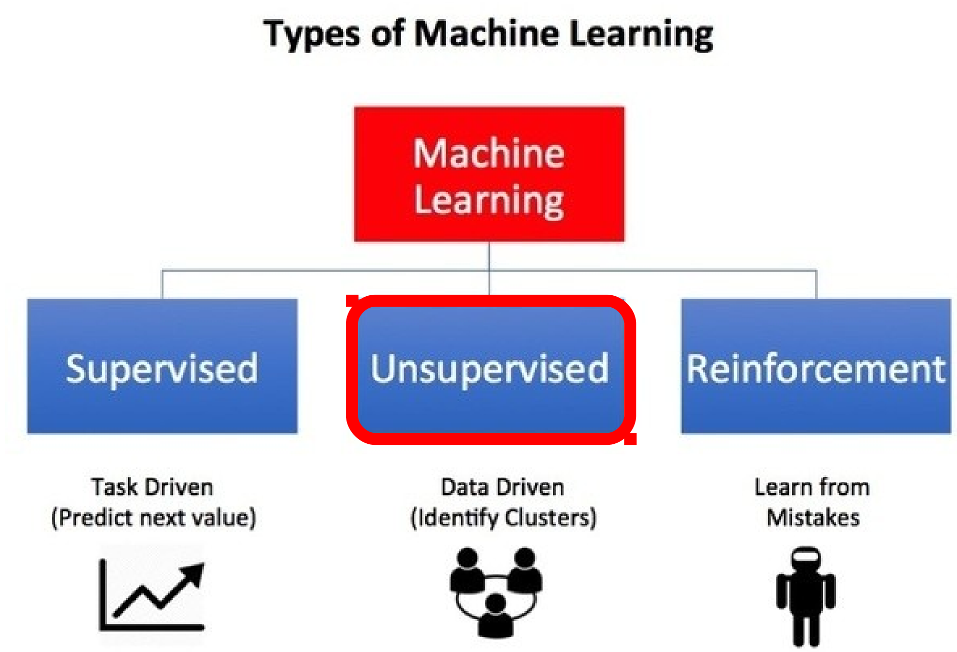
Técnicas de Segmentación: Clustering



Sixto Herrera

Grupo de Meteorología

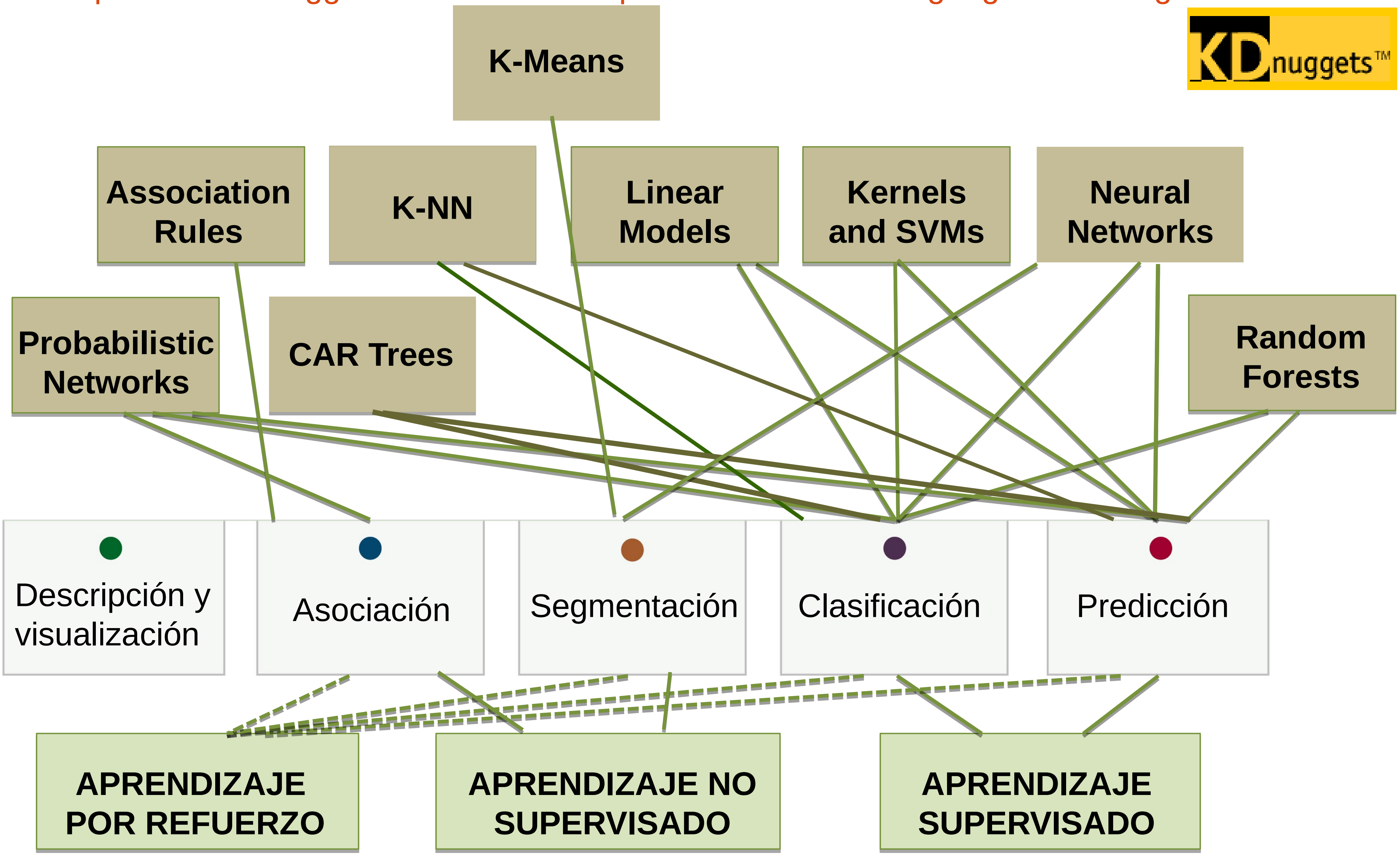
Univ. de Cantabria – CSIC
MACC / IFCA



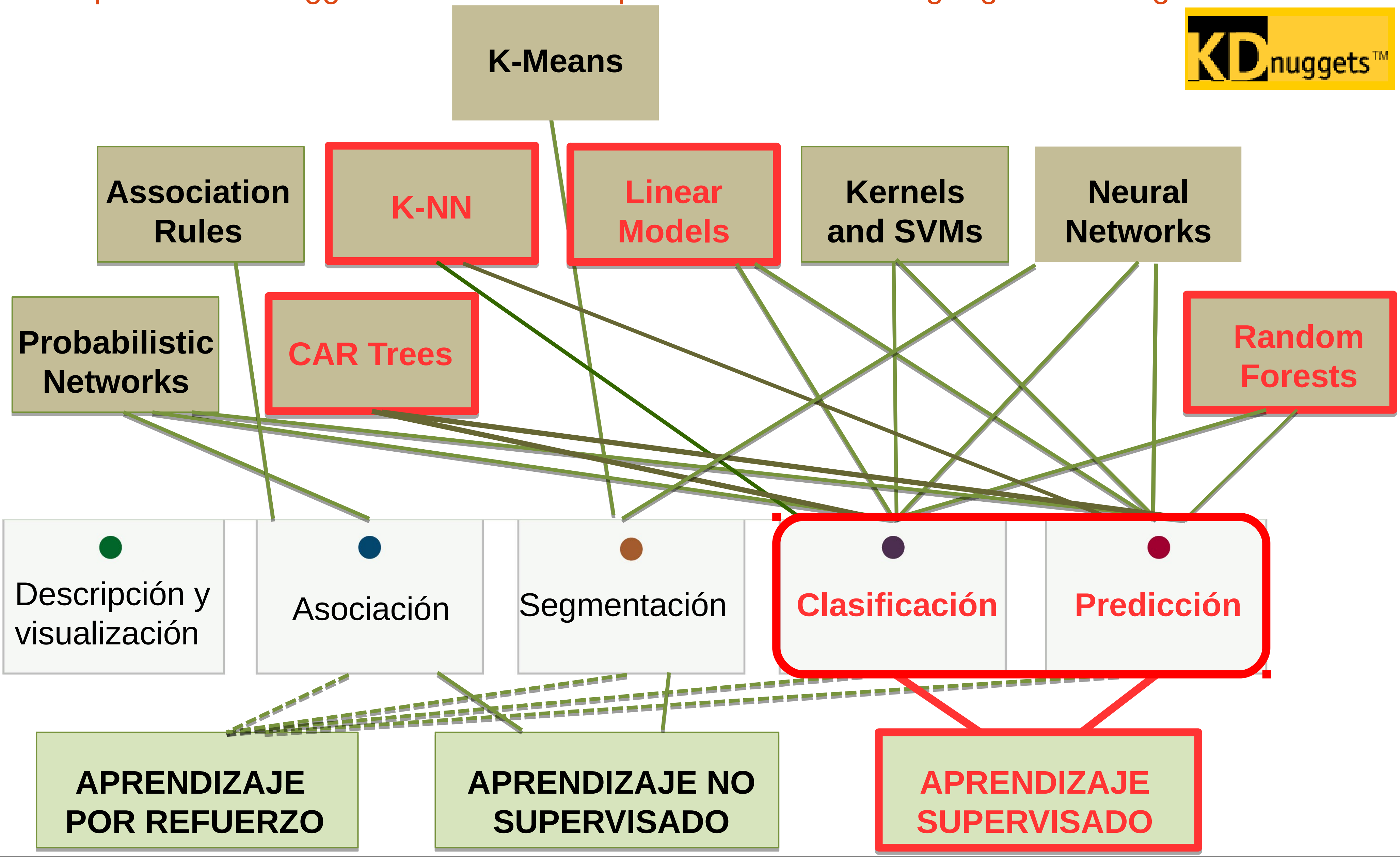
NOTA: Las líneas de código de R en esta presentación se muestran sobre un fondo gris

Oct	30	Aplazada (sesión de refuezo)
Nov	6	Presentación, introducción y perspectiva histórica
	8	Paradigmas, problemas canonicos y data challenges
	13	Reglas de asociación
	15	Practica: Reglas de asociación
	20	Evaluación, sobrejuste y crossvalidacion
	22	Practica: Crossvalidacion
	27	Arboles de clasificacion y decision
	29	Practica: Arboles de clasificación
		T01. Datos discretos
Dic	4	Técnicas de vecinos cercano (k-NN)
	11	Práctica: Vecinos cercanos
	13	Reducción de dimensión lineal
	18	Practica: LDA y PCA
	20	Reducción no lineal
		T02. Clasificación
Ene	8	Árboles de decisión: Regresión (CART)
	10	Practica: CART
	15	Ensembles: Bagging and Boosting
	17	Practica Random Forests
		T03. Prediccion
	22	Practica Gradient boosting
	24a	**Técnicas de agrupamiento**
	24b	**Practica: Técnicas de agrupamiento**
	29a	Practica: El paquete CARET
	29b	Examen

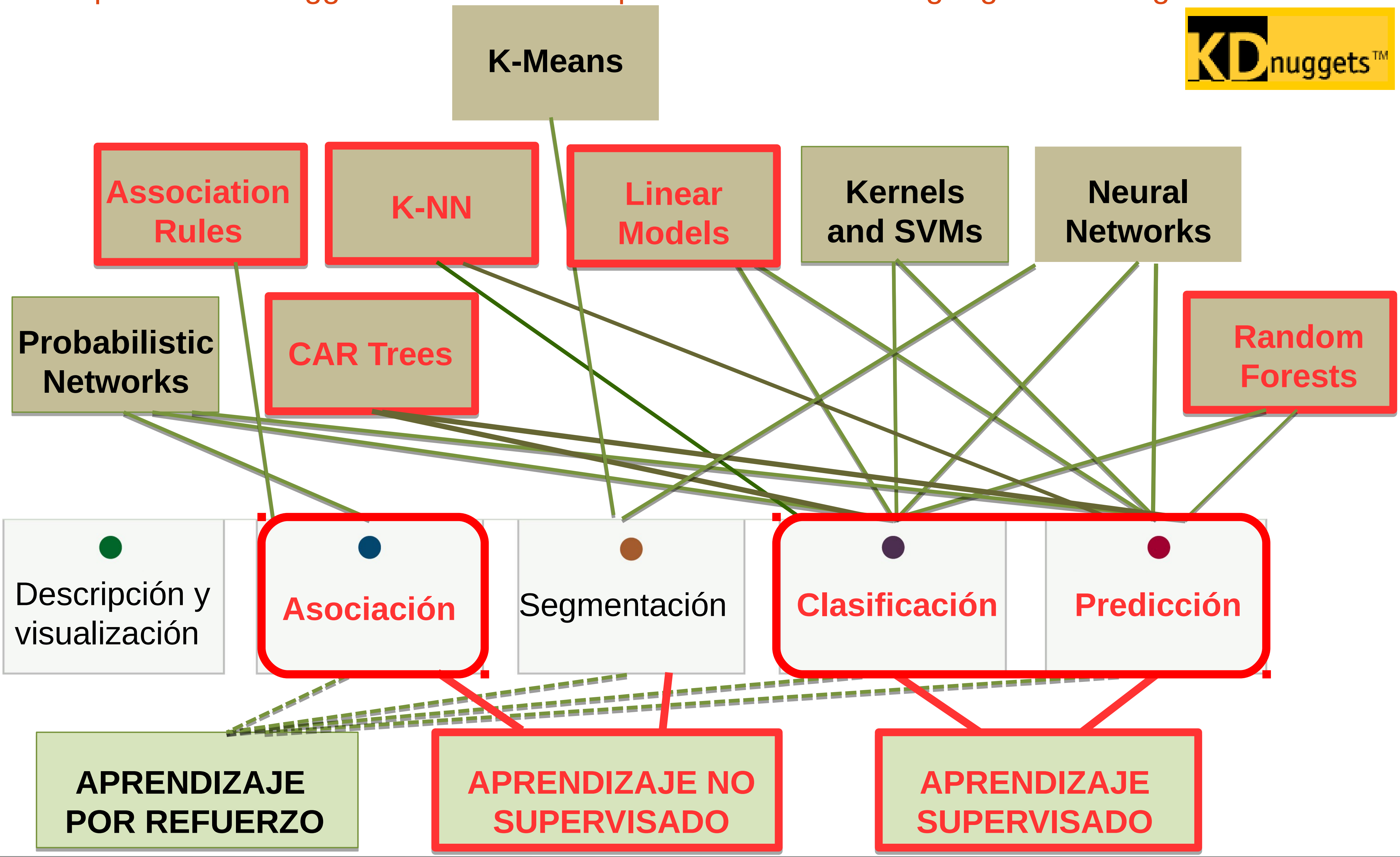
https://www.kdnuggets.com/2015/05/top-10-data-mining-algorithms-explained.html
https://www.kdnuggets.com/2017/10/top-10-machine-learning-algorithms-beginners.html



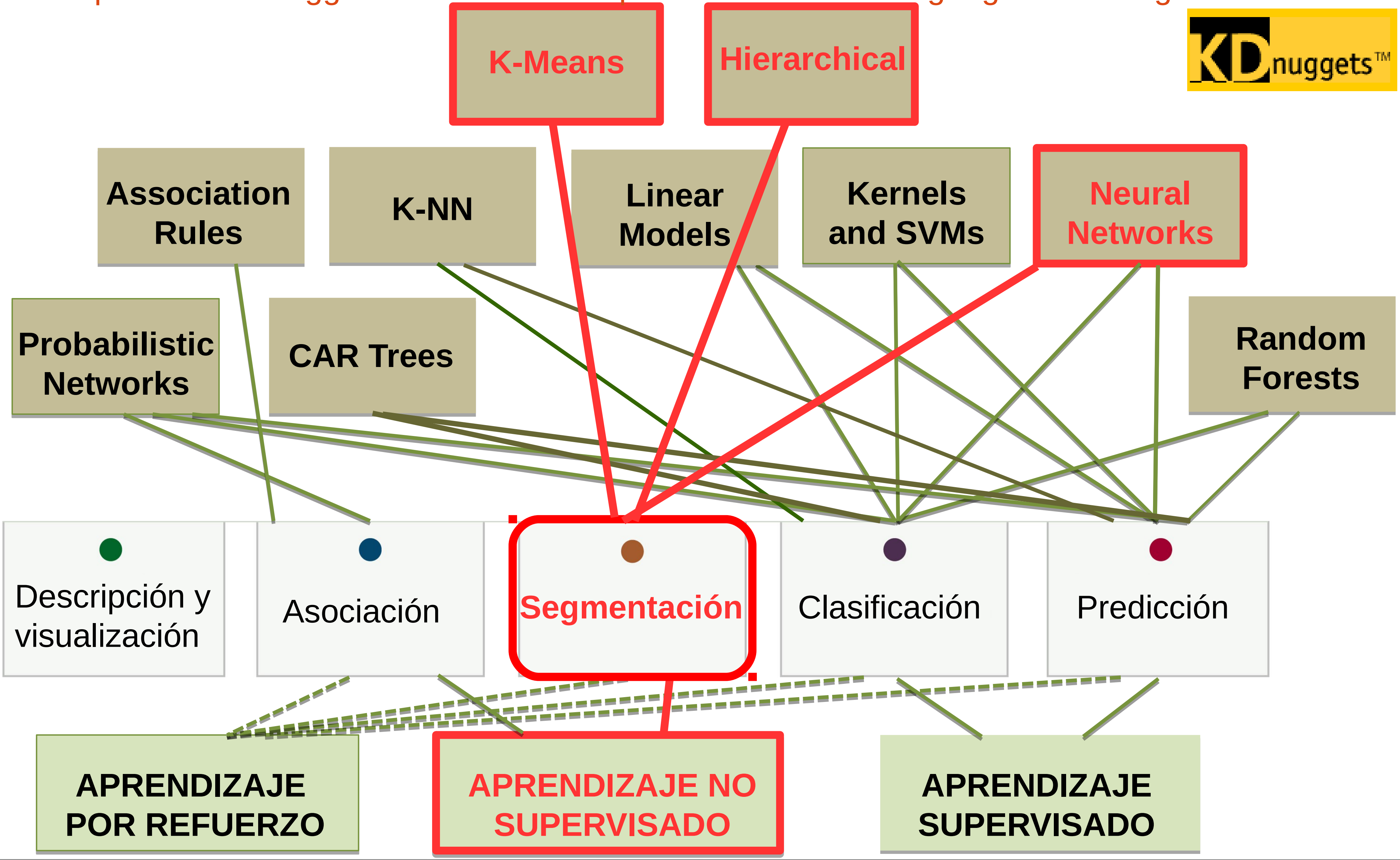
https://www.kdnuggets.com/2015/05/top-10-data-mining-algorithms-explained.html
https://www.kdnuggets.com/2017/10/top-10-machine-learning-algorithms-beginners.html



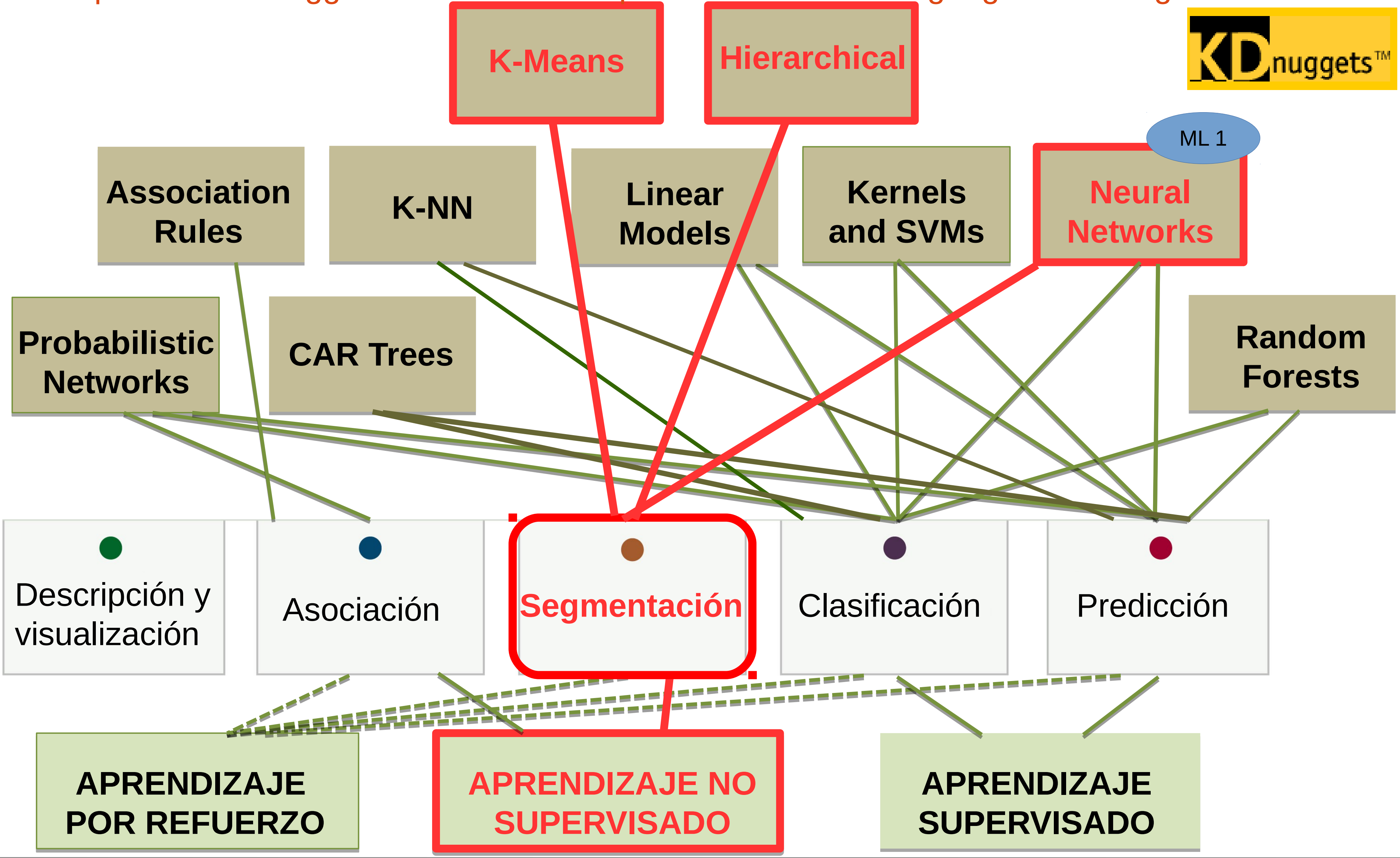
https://www.kdnuggets.com/2015/05/top-10-data-mining-algorithms-explained.html
https://www.kdnuggets.com/2017/10/top-10-machine-learning-algorithms-beginners.html

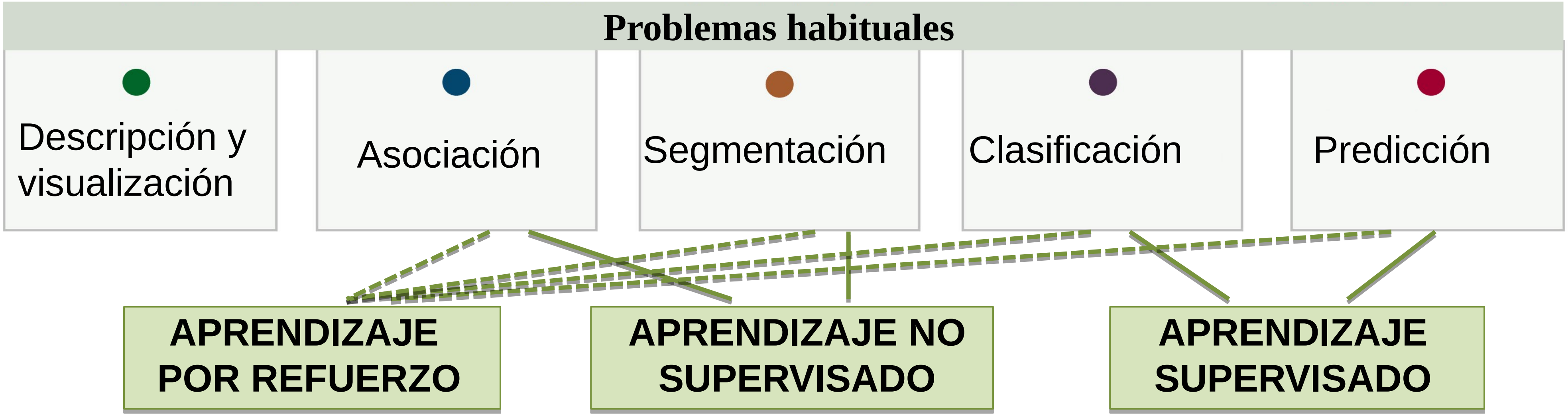


https://www.kdnuggets.com/2015/05/top-10-data-mining-algorithms-explained.html
https://www.kdnuggets.com/2017/10/top-10-machine-learning-algorithms-beginners.html



https://www.kdnuggets.com/2015/05/top-10-data-mining-algorithms-explained.html
https://www.kdnuggets.com/2017/10/top-10-machine-learning-algorithms-beginners.html





Input data: $x_1, x_2, x_3, x_4, \ldots$

Supervised learning: The machine is also given desired outputs $y_1, y_2, \ldots$, and its goal is to learn to produce the correct output given a new input.

Unsupervised learning: The goal of the machine is to build representations of x that can be used for reasoning, decision making, predicting things, communicating etc.

Reinforcement learning: The machine can also produce actions $a_1, a_2, \ldots$ which affect the state of the world, and receives rewards (or punishments) $r_1, r_2, \ldots$. Its goal is to learn to act in a way that maximises rewards in the long term.

Problemas habituales

Descripción y visualización	Asociación	Segmentación	**Clasificación**	**Predicción**

APRENDIZAJE POR REFUERZO | **APRENDIZAJE NO SUPERVISADO** | **APRENDIZAJE SUPERVISADO**

Target Variable: *Y: discrete/factor* or *continuous*

What we are trying to predict.

Predictive Variables: $\{X_1, X_2, \ldots, X_N\}$: *continuous*

"Covariates" used to make predictions.

Predictive Model: $Y = f(X_1, X_2, \ldots, X_N)$

"Learning engine" that estimates the f (or the parameters defining f).

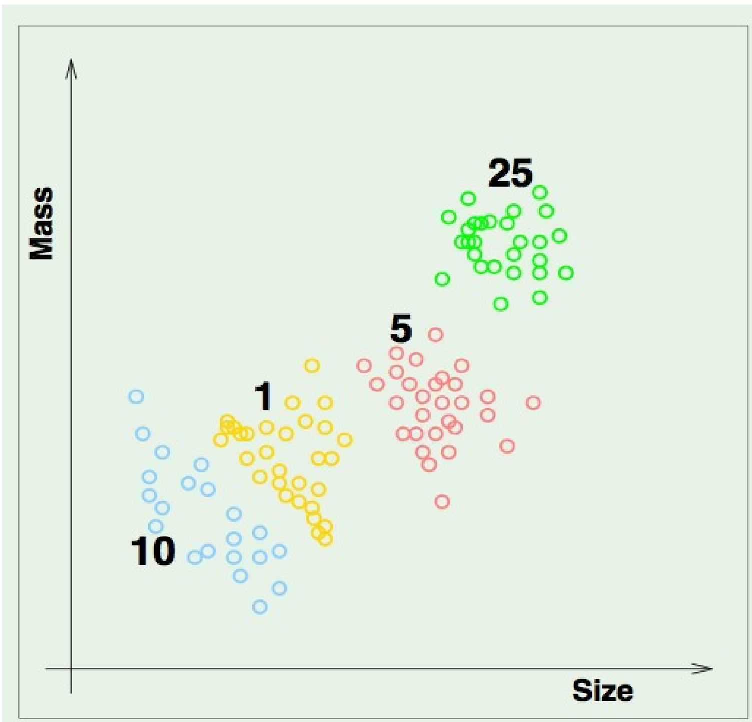
Problemas habituales

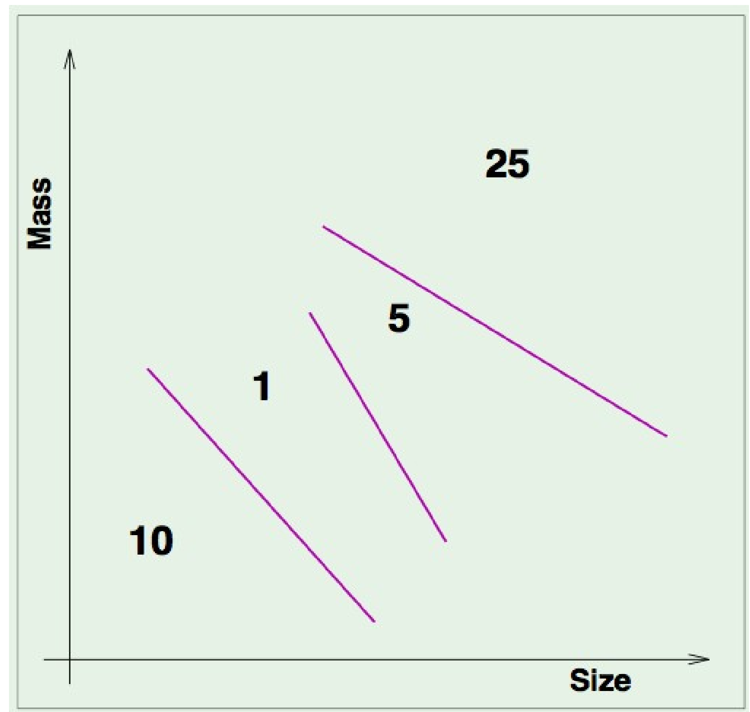
- Descripción y visualización
- Asociación
- Segmentación
- **Clasificación**
- **Predicción**

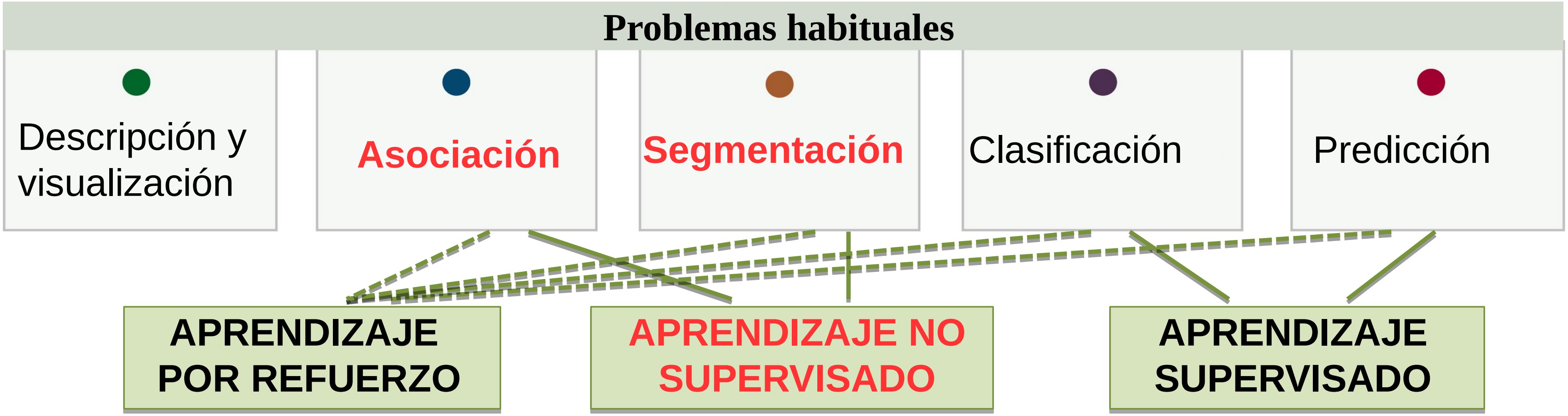
APRENDIZAJE POR REFUERZO

APRENDIZAJE NO SUPERVISADO

APRENDIZAJE SUPERVISADO







Target Variable: *There is no target variable (association)*

K (cluster), discrete: #clusters (segmentation)

Predictive Variables: $\{X_1, X_2, \ldots, X_N\}$: *continuous or discrete*

“Covariates” used to make predictions.

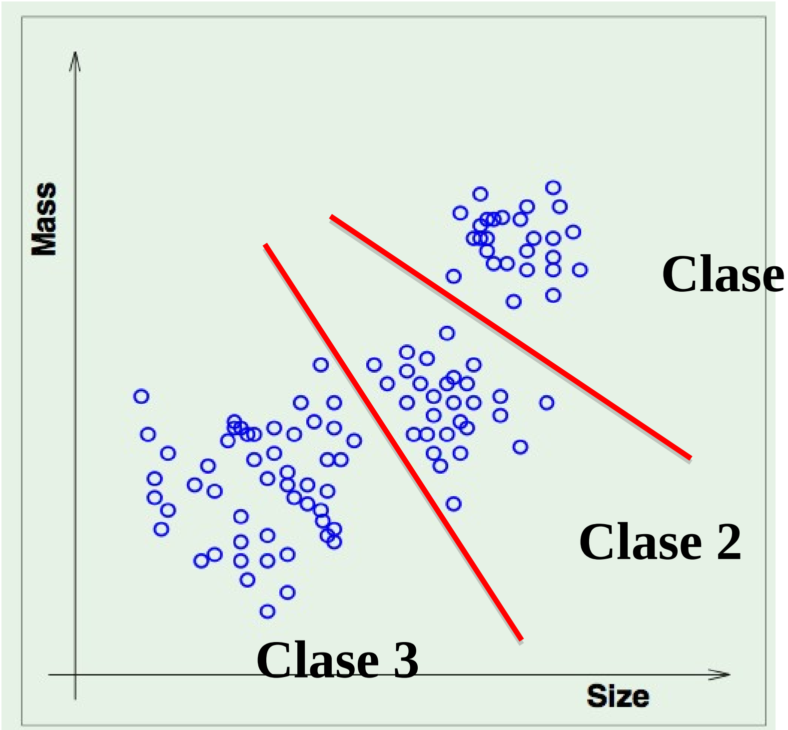
Predictive Model: Algorithmic, based on $(X_1, X_2, \ldots, X_N)$.

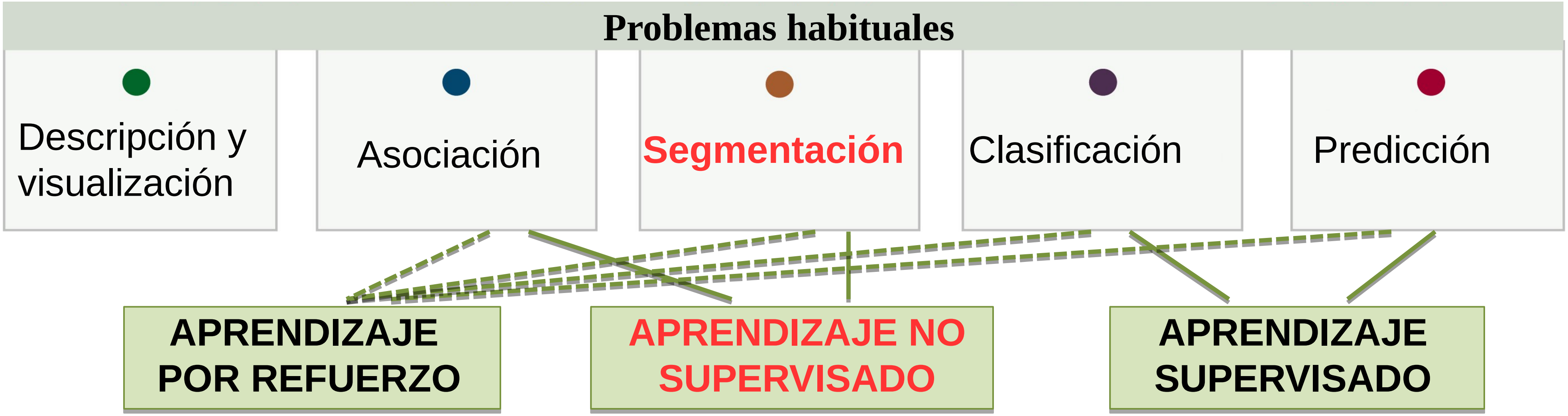
Ad-hoc “learning” and “prediction” engine.

Problemas habituales

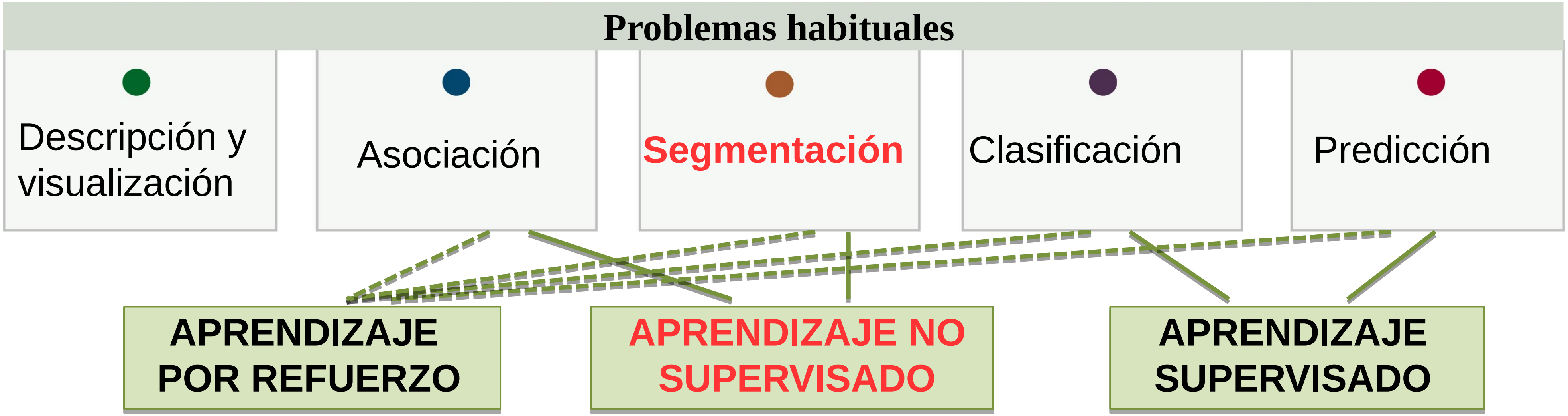
Descripción y visualización	Asociación	Segmentación	Clasificación	Predicción

APRENDIZAJE POR REFUERZO | **APRENDIZAJE NO SUPERVISADO** | **APRENDIZAJE SUPERVISADO**





The main objective of the cluster analysis is to group or segmenting a collection of objects, understood as a set of measurements, into subsets or “clusters,” such that those within each cluster are more closely related to one another than objects assigned to different clusters.



The main objective of the cluster analysis is to group or segmenting a collection of objects, understood as a set of measurements, into subsets or “clusters,” such that those within each cluster are more closely related to one another than objects assigned to different clusters.

Input

Similarity-based → NxN distance matrix D

Feature-based → NxD feature matrix X

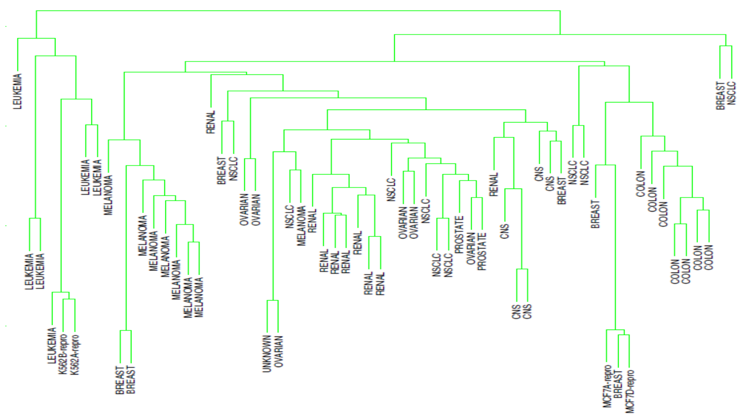
Problemas habituales

Descripción y visualización	Asociación	Segmentación	Clasificación	Predicción

APRENDIZAJE POR REFUERZO | **APRENDIZAJE NO SUPERVISADO** | **APRENDIZAJE SUPERVISADO**

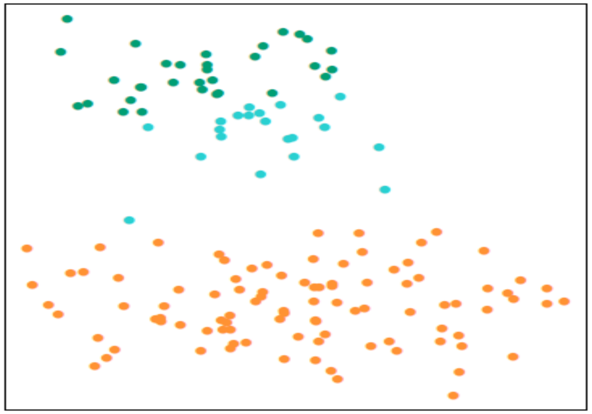
The main objective of the cluster analysis is to group or segmenting a collection of objects, understood as a set of measurements, into subsets or "clusters," such that those within each cluster are more closely related to one another than objects assigned to different clusters.

Output



Partitional/Flat

Hierarchical



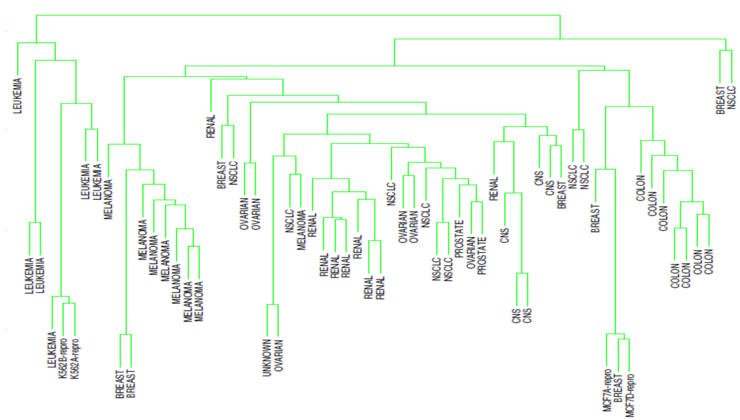
Problemas habituales

Descripción y visualización | Asociación | Segmentación | Clasificación | Predicción

APRENDIZAJE POR REFUERZO | APRENDIZAJE NO SUPERVISADO | APRENDIZAJE SUPERVISADO

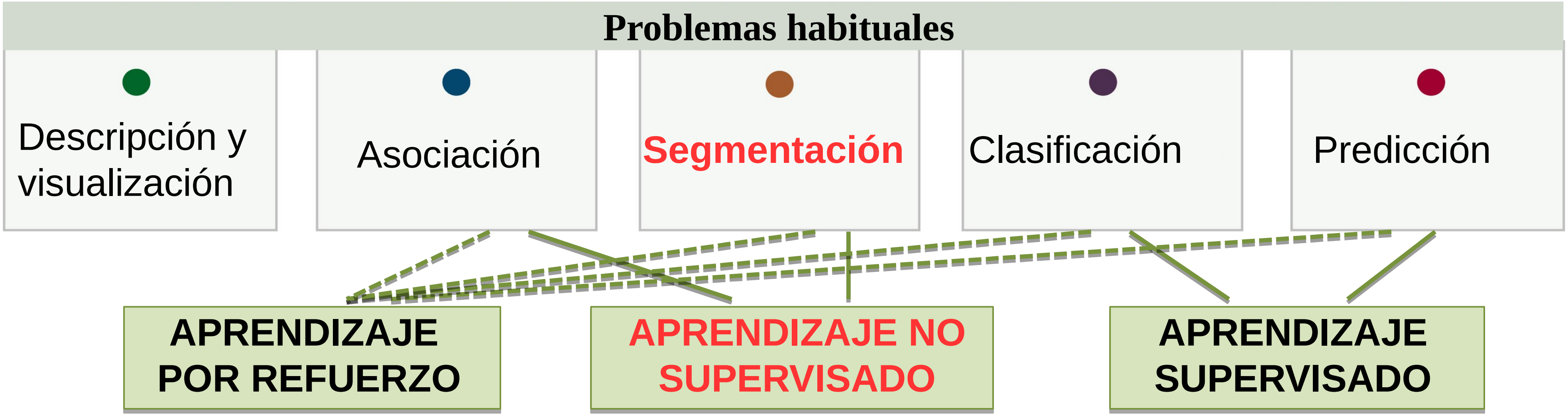
The main objective of the cluster analysis is to group or segmenting a collection of objects, understood as a set of measurements, into subsets or “clusters,” such that those within each cluster are more closely related to one another than objects assigned to different clusters.

Hierarchical



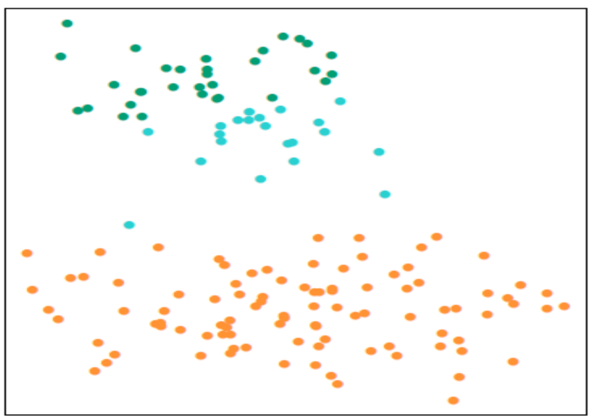
Complejidad: $O\left(N^2 \log N\right)$

Deterministic



The main objective of the cluster analysis is to group or segmenting a collection of objects, understood as a set of measurements, into subsets or “clusters,” such that those within each cluster are more closely related to one another than objects assigned to different clusters.

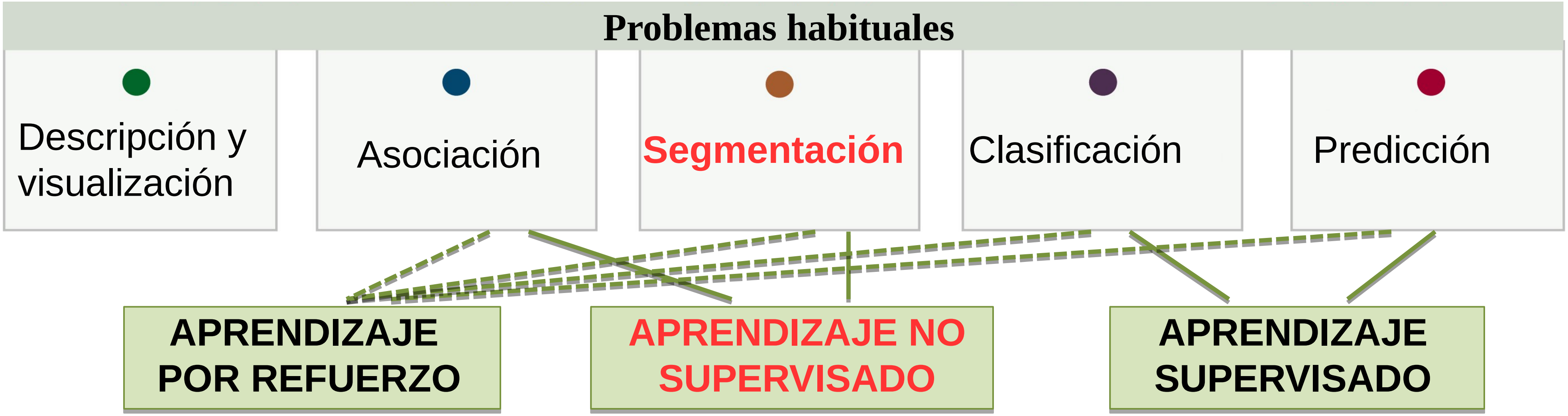
Partitional/Flat



Complejidad: $O(N\,D)$

Depends on the number of clusters

Sensitivity to initial conditions



The main objective of the cluster analysis is to group or segmenting a collection of objects, understood as a set of measurements, into subsets or “clusters,” such that those within each cluster are more closely related to one another than objects assigned to different clusters.

Input	Output
Similiraty-based	**Partitional/Flat**
Feature-based	**Hierarchical**

To this aim, an assessment of the **degree of difference (dissimilarity)** between the objects assigned to the respective clusters is required.

Similarity and distance measures are obtained/defined considering the predictors.
Therefore, strongly depend on the nature of these variables:

Quantitative

Qualitative

Ordinals

etc...

```
? dist
```

Similarity and distance measures are obtained/defined considering the predictors. Therefore, strongly depend on the nature of these variables:

Quantitative

Minkowsky:

$$D(\mathbf{x},\mathbf{y}) = \left(\sum_{i=1}^{m} |x_i - y_i|^r\right)^{1/r}$$

Euclidean:

$$D(\mathbf{x},\mathbf{y}) = \sqrt{\sum_{i=1}^{m} (x_i - y_i)^2}$$

Manhattan / city-block:

$$D(\mathbf{x},\mathbf{y}) = \sum_{i=1}^{m} |x_i - y_i|$$

Camberra:

$$D(\mathbf{x},\mathbf{y}) = \sum_{i=1}^{m} \frac{|x_i - y_i|}{|x_i + y_i|}$$

Chebychev:

$$D(\mathbf{x},\mathbf{y}) = \max_{i=1}^{m} |x_i - y_i|$$

Quadratic:

$$D(\mathbf{x},\mathbf{y}) = (\mathbf{x}-\mathbf{y})^T Q(\mathbf{x}-\mathbf{y}) = \sum_{j=1}^{m}\left(\sum_{i=1}^{m}(x_i - y_i)q_{ji}\right)(x_j - y_j)$$

Q is a problem-specific positive definite $m \times m$ weight matrix

Mahalanobis:

$$D(\mathbf{x},\mathbf{y}) = [\det V]^{1/m}(\mathbf{x}-\mathbf{y})^{\mathrm{T}} V^{-1}(\mathbf{x}-\mathbf{y})$$

V is the covariance matrix of $A_1..A_m$, and A_j is the vector of values for attribute j occuring in the training set instances $1..n$.

Correlation:

$$D(\mathbf{x},\mathbf{y}) = \frac{\sum_{i=1}^{m}(x_i - \overline{x_i})(y_i - \overline{y_i})}{\sqrt{\sum_{i=1}^{m}(x_i - \overline{x_i})^2 \sum_{i=1}^{m}(y_i - \overline{y_i})^2}}$$

$\overline{x_i} = \overline{y_i}$ and is the average value for attribute i occuring in the training set.

Chi-square:

$$D(\mathbf{x},\mathbf{y}) = \sum_{i=1}^{m} \frac{1}{sum_i}\left(\frac{x_i}{size_{\mathbf{x}}} - \frac{y_i}{size_{\mathbf{y}}}\right)^2$$

sum_i is the sum of all values for attribute i occuring in the training set, and $size_x$ is the sum of all values in the vector $\mathbf{x}$.

Kendall's Rank Correlation:

$$D(\mathbf{x},\mathbf{y}) = 1 - \frac{2}{n(n-1)}\sum_{i=1}^{m}\sum_{j=1}^{i-1} \mathrm{sign}(x_i - x_j)\mathrm{sign}(y_i - y_j)$$

sign(x)=-1, 0 or 1 if $x < 0$, $x = 0$, or $x > 0$, respectively.

```
? dist
dEuc<-dist(iris[,-5],method="euclidean")
dMin<-dist(iris[,-5],method="minkowski",p=4)
```

Similarity and distance measures are obtained/defined considering the predictors. Therefore, strongly depend on the nature of these variables:

Quantitative

Ordinals → redefine as the rank or the order (e.g. {low, medium, high} → {1/3,2/3,3/3}).

Minkowsky:

$$D(\boldsymbol{x},\boldsymbol{y}) = \left(\sum_{i=1}^{m} |x_i - y_i|^r\right)^{1/r}$$

Euclidean:

$$D(\boldsymbol{x},\boldsymbol{y}) = \sqrt{\sum_{i=1}^{m} (x_i - y_i)^2}$$

Manhattan / city-block:

$$D(\boldsymbol{x},\boldsymbol{y}) = \sum_{i=1}^{m} |x_i - y_i|$$

Camberra:

$$D(\boldsymbol{x},\boldsymbol{y}) = \sum_{i=1}^{m} \frac{|x_i - y_i|}{|x_i + y_i|}$$

Chebychev:

$$D(\boldsymbol{x},\boldsymbol{y}) = \max_{i=1}^{m} |x_i - y_i|$$

Quadratic:

$$D(\boldsymbol{x},\boldsymbol{y}) = (\boldsymbol{x}-\boldsymbol{y})^T Q(\boldsymbol{x}-\boldsymbol{y}) = \sum_{j=1}^{m}\left(\sum_{i=1}^{m}(x_i - y_i)q_{ji}\right)(x_j - y_j)$$

Q is a problem-specific positive definite $m \times m$ weight matrix

Mahalanobis:

$$D(\boldsymbol{x},\boldsymbol{y}) = [\det V]^{1/m}(\boldsymbol{x}-\boldsymbol{y})^{\mathrm{T}} V^{-1}(\boldsymbol{x}-\boldsymbol{y})$$

V is the covariance matrix of $A_1..A_m$, and A_j is the vector of values for attribute j occuring in the training set instances $1..n$.

Correlation:

$$D(\boldsymbol{x},\boldsymbol{y}) = \frac{\sum_{i=1}^{m}(x_i - \overline{x_i})(y_i - \overline{y_i})}{\sqrt{\sum_{i=1}^{m}(x_i - \overline{x_i})^2 \sum_{i=1}^{m}(y_i - \overline{y_i})^2}}$$

$\overline{x_i} = \overline{y_i}$ and is the average value for attribute i occuring in the training set.

Chi-square:

$$D(\boldsymbol{x},\boldsymbol{y}) = \sum_{i=1}^{m} \frac{1}{sum_i}\left(\frac{x_i}{size_{\boldsymbol{x}}} - \frac{y_i}{size_{\boldsymbol{y}}}\right)^2$$

sum_i is the sum of all values for attribute i occuring in the training set, and $size_x$ is the sum of all values in the vector $\boldsymbol{x}$.

Kendall's Rank Correlation:

$$D(\boldsymbol{x},\boldsymbol{y}) = 1 - \frac{2}{n(n-1)}\sum_{i=1}^{m}\sum_{j=1}^{i-1}\operatorname{sign}(x_i - x_j)\operatorname{sign}(y_i - y_j)$$

sign(x)=-1, 0 or 1 if $x<0$, $x=0$, or $x>0$, respectively.

```
? dist
dEuc<-dist(iris[,-5],method="euclidean")
dMin<-dist(iris[,-5],method="minkowski",
```

Similarity and distance measures are obtained/defined considering the predictors. Therefore, strongly depend on the nature of these variables:

Quantitative

Ordinals → redefine as the rank or the order (e.g. {low, medium, high} → {1/3,2/3,3/3}).

Qualitative – Categorical → assign a distance of 1 if the features are different and 0 otherwise.

$$D(x,y)=\sum_{j=1}^{D} I(x_j \neq y_j)$$

```
? dist
dEuc<-dist(iris[,-5],method="euclidean")
dMin<-dist(iris[,-5],method="minkowski",
library(cluster)
?daisy
dNom<-daisy(iris,metric="gower")
```

Minkowsky: $D(\boldsymbol{x},\boldsymbol{y})=\left(\sum_{i=1}^{m}|x_i-y_i|^r\right)^{1/r}$

Euclidean: $D(\boldsymbol{x},\boldsymbol{y})=\sqrt{\sum_{i=1}^{m}(x_i-y_i)^2}$

Manhattan / city-block: $D(\boldsymbol{x},\boldsymbol{y})=\sum_{i=1}^{m}|x_i-y_i|$

Camberra: $D(\boldsymbol{x},\boldsymbol{y})=\sum_{i=1}^{m}\frac{|x_i-y_i|}{|x_i+y_i|}$

Chebychev: $D(\boldsymbol{x},\boldsymbol{y})=\max_{i=1}^{m}|x_i-y_i|$

Quadratic: $D(\boldsymbol{x},\boldsymbol{y})=(\boldsymbol{x}-\boldsymbol{y})^T Q(\boldsymbol{x}-\boldsymbol{y})=\sum_{j=1}^{m}\left(\sum_{i=1}^{m}(x_i-y_i)q_{ji}\right)(x_j-y_j)$

Q is a problem-specific positive definite $m\times m$ weight matrix

Mahalanobis:

$D(\boldsymbol{x},\boldsymbol{y})=[\det V]^{1/m}(\boldsymbol{x}-\boldsymbol{y})^{\mathrm{T}}V^{-1}(\boldsymbol{x}-\boldsymbol{y})$

V is the covariance matrix of $A_1..A_m$, and A_j is the vector of values for attribute j occuring in the training set instances $1..n$.

Correlation:

$$D(\boldsymbol{x},\boldsymbol{y})=\frac{\sum_{i=1}^{m}(x_i-\overline{x_i})(y_i-\overline{y_i})}{\sqrt{\sum_{i=1}^{m}(x_i-\overline{x_i})^2\sum_{i=1}^{m}(y_i-\overline{y_i})^2}}$$

$\overline{x_i}=\overline{y_i}$ and is the average value for attribute i occuring in the training set.

Chi-square: $D(\boldsymbol{x},\boldsymbol{y})=\sum_{i=1}^{m}\frac{1}{sum_i}\left(\frac{x_i}{size_{\boldsymbol{x}}}-\frac{y_i}{size_{\boldsymbol{y}}}\right)^2$

sum_i is the sum of all values for attribute i occuring in the training set, and $size_x$ is the sum of all values in the vector $\boldsymbol{x}$.

Kendall's Rank Correlation: $D(\boldsymbol{x},\boldsymbol{y})=1-\frac{2}{n(n-1)}\sum_{i=1}^{m}\sum_{j=1}^{i-1}\mathrm{sign}(x_i-x_j)\mathrm{sign}(y_i-y_j)$

sign(x)=-1, 0 or 1 if $x<0$, $x=0$, or $x>0$, respectively.

Defined the distance:

$$D(x_i, x_{i'}) = \sum_{j=1}^{p} w_j \cdot d_j(x_{ij}, x_{i'j}); \quad \sum_{j=1}^{p} w_j = 1. \qquad D_I(x_i, x_{i'}) = \sum_{j=1}^{p} w_j \cdot (x_{ij} - x_{i'j})^2$$

The objective of the clustering algorithms is to:

$$T = \frac{1}{2}\sum_{i=1}^{N}\sum_{i'=1}^{N} d_{ii'} = \frac{1}{2}\sum_{k=1}^{K}\sum_{C(i)=k}\left(\sum_{C(i')=k} d_{ii'} + \sum_{C(i')\neq k} d_{ii'}\right),$$

$$T = W(C) + B(C), \begin{cases} W(C) = \frac{1}{2}\sum_{k=1}^{K}\sum_{C(i)=k}\sum_{C(i')=k} d(x_i, x_{i'}). \\ B(C) = \frac{1}{2}\sum_{k=1}^{K}\sum_{C(i)=k}\sum_{C(i')\neq k} d_{ii'} \end{cases}$$

Minimizes the distance intragroup

and

Maximizes the distance between clusters.

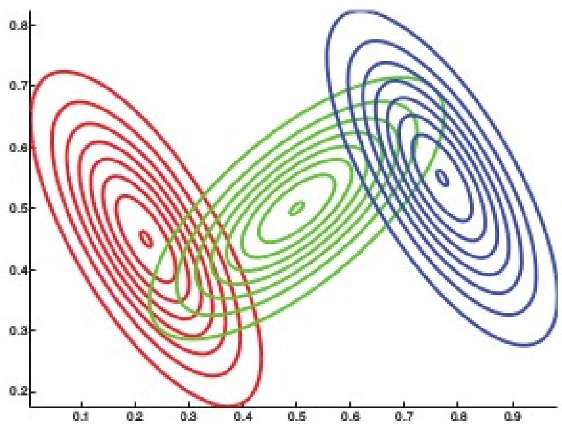
```
? kmeans
kmModel<-kmeans(iris[,-5],3,nstart = 1)
kmModel$withinss ## Vector of within-cluster sum of squares, one component per cluster
kmModel$betweenss ## The between-cluster sum of squares
```

Problemas habituales

Descripción y visualización	Asociación	**Segmentación**	Clasificación	Predicción

APRENDIZAJE POR REFUERZO | **APRENDIZAJE NO SUPERVISADO** | **APRENDIZAJE SUPERVISADO**

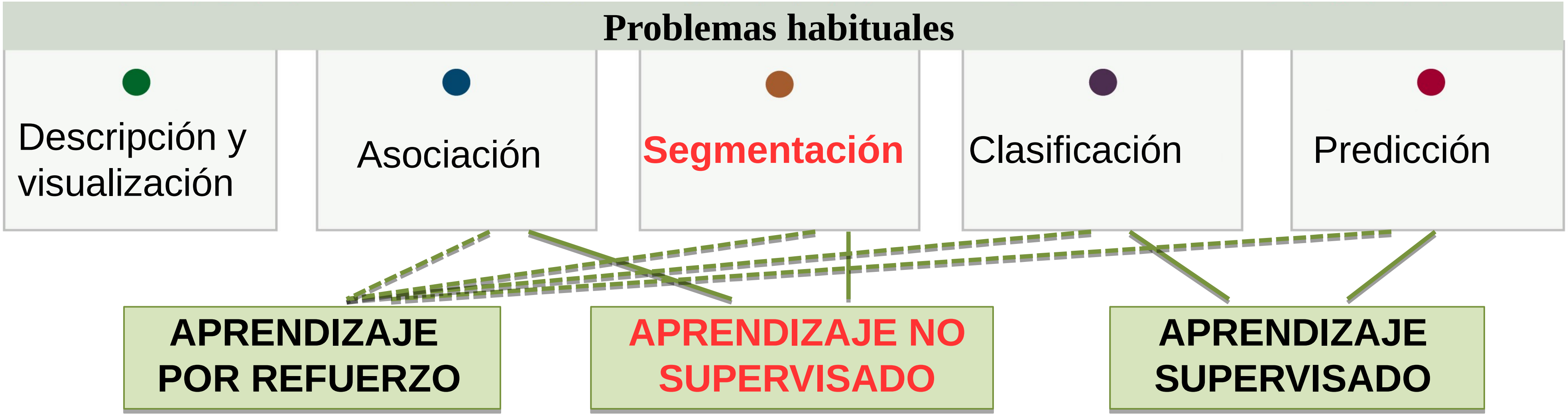
Distribution-based clustering



$$p(\mathbf{x}_i|\boldsymbol{\theta}) = \sum_{k=1}^{K} \pi_k p_k(\mathbf{x}_i|\boldsymbol{\theta})$$

π_k satisfy $0 \leq \pi_k \leq 1$ and $\sum_{k=1}^{K} \pi_k = 1$.

Source: Machine Learning A Probabilistic Perspective, Kevin P. Murphy, The MIT Press, Cambridge, Massachusetts, London, England



Gaussian Mixtures (EM-algorithm)

```
library(MASS)
library(mclust)
? Mclust
```

$$p(\mathbf{x}_i|\boldsymbol{\theta}) = \sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{x}_i|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

π_k satisfy $0 \leq \pi_k \leq 1$ and $\sum_{k=1}^{K} \pi_k = 1$.

- **Expectation step (E):** Calculate the expected value of the log likelihood under the current estimate of the parameters.

$$r_{ik} = \frac{\pi_k p(\mathbf{x}_i|\boldsymbol{\theta}_k^{(t-1)})}{\sum_{k'} \pi_{k'} p(\mathbf{x}_i|\boldsymbol{\theta}_{k'}^{(t-1)})}$$

- **Maximization step (M):** Find the parameters that maximize the log likelihood.

$$\boldsymbol{\mu}_k = \frac{\sum_i r_{ik}\mathbf{x}_i}{r_k}$$

$$\boldsymbol{\Sigma}_k = \frac{\sum_i r_{ik}(\mathbf{x}_i - \boldsymbol{\mu}_k)(\mathbf{x}_i - \boldsymbol{\mu}_k)^T}{r_k} = \frac{\sum_i r_{ik}\mathbf{x}_i\mathbf{x}_i^T}{r_k} - \boldsymbol{\mu}_k\boldsymbol{\mu}_k^T$$

Iris Data Set

Download: Data Folder, Data Set Description

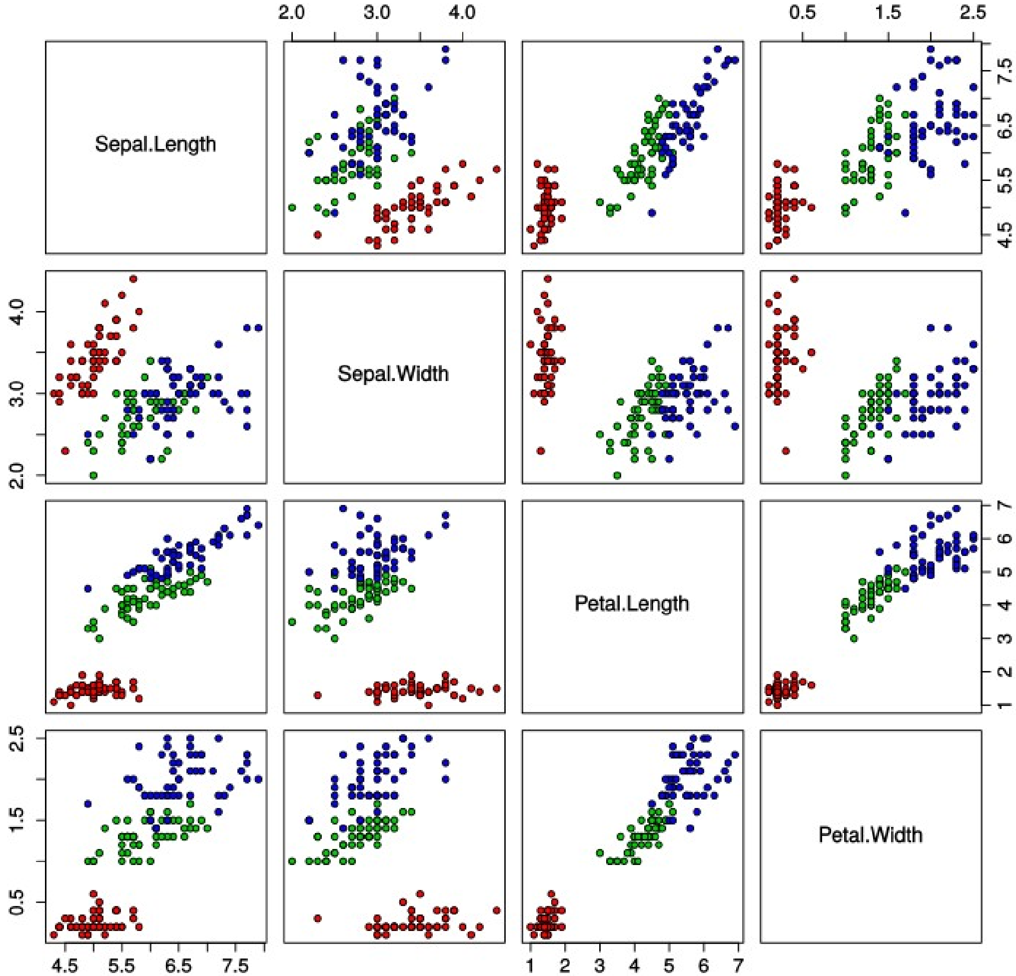
Abstract: Famous database; from Fisher, 1936

Data Set Characteristics:	Multivariate	**Number of Instances:**	150
Attribute Characteristics:	Real	**Number of Attributes:**	4
Associated Tasks:	Classification	**Missing Values?**	No

http://archive.ics.uci.edu/ml/datasets/Iris

```
gmModel<-Mclust(iris[,-5])
summary(gmModel)
Gaussian finite mixture model
fitted by EM algorithm
----------------------------------------
Mclust VEV (ellipsoidal, equal shape)
model with 2 components:
log.likelihood   n df       BIC       ICL
      -215.726 150 26 -561.7285 -561.7289
Clustering table:
  1   2
 50 100
```

Iris Data (red=setosa,green=versicolor,blue=virginica)



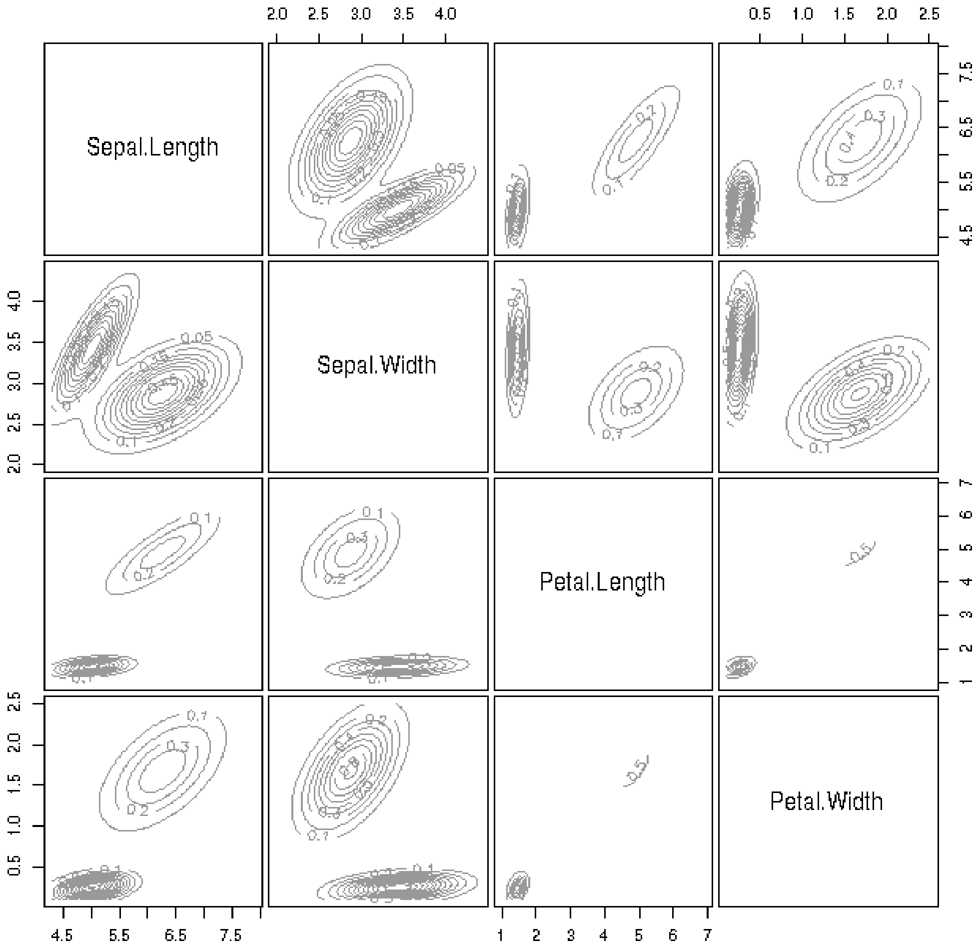
Iris Data Set

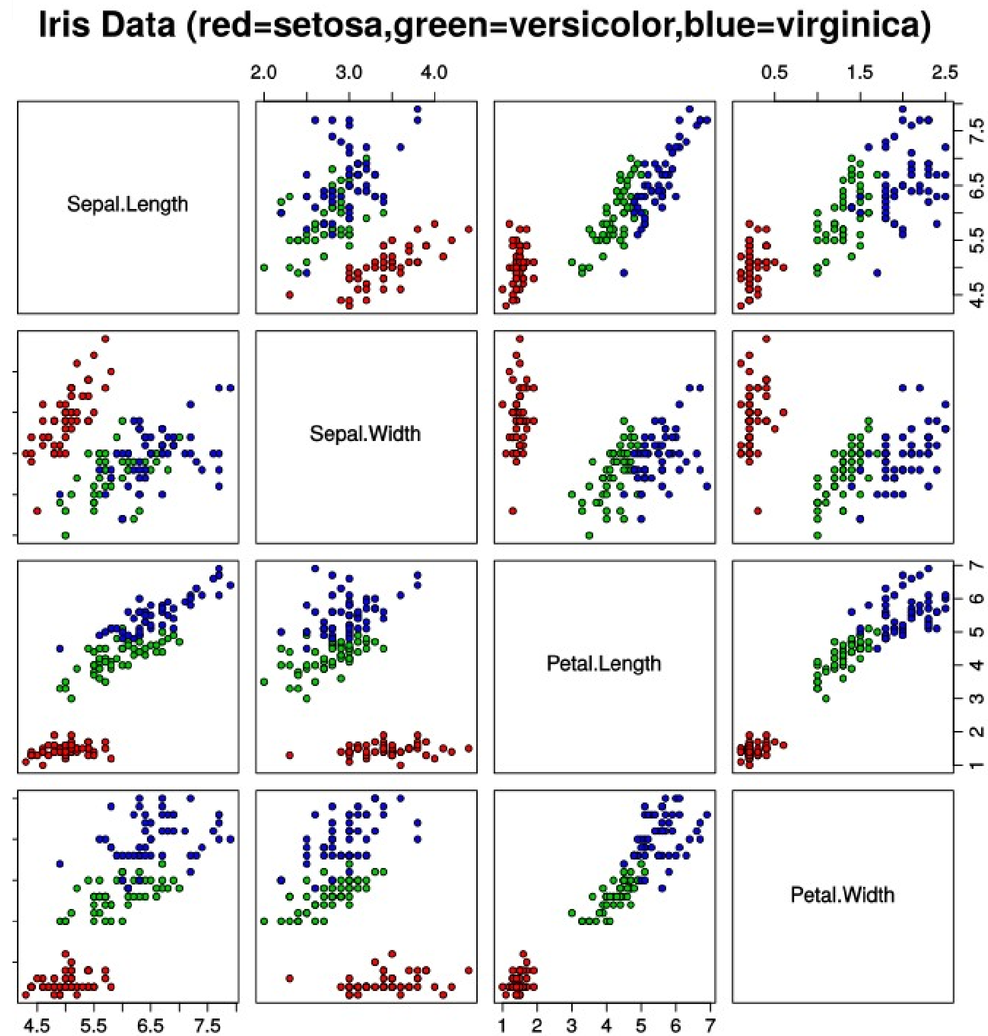
Download: Data Folder, Data Set Description

Abstract: Famous database; from Fisher, 1936

Data Set Characteristics:	Multivariate	**Number of Instances:**	150
Attribute Characteristics:	Real	**Number of Attributes:**	4
Associated Tasks:	Classification	**Missing Values?**	No

```
plot(gmModel,what="density")
```





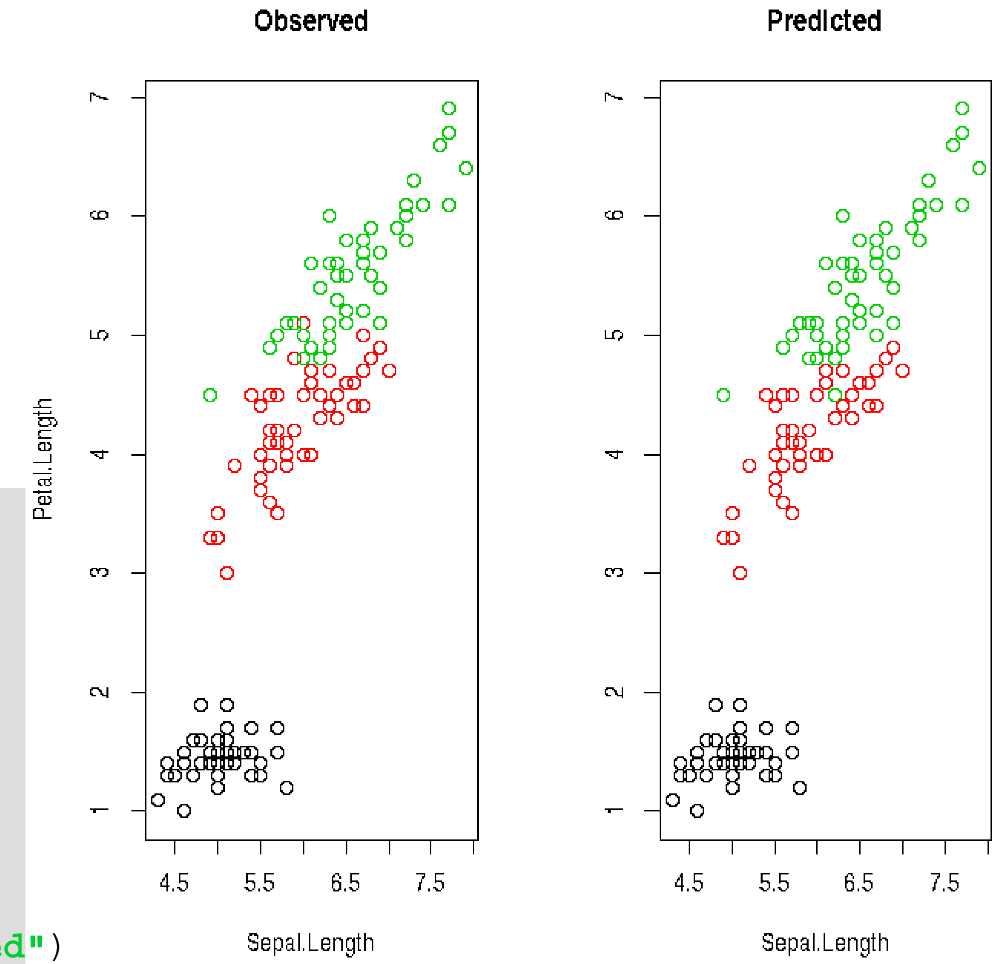
Iris Data Set

Download: Data Folder, Data Set Description

Abstract: Famous database; from Fisher, 1936

Data Set Characteristics:	Multivariate	**Number of Instances:**	150
Attribute Characteristics:	Real	**Number of Attributes:**	4
Associated Tasks:	Classification	**Missing Values?**	No

http://archive.ics.uci.edu/ml/datasets/Iris




```
library(caret)
gmModel<-Mclust(iris[,-5])
summary(gmModel)
plot(gmModel,what="density")
## Defining the number of G-M
gmModel<-Mclust(iris[,-5],G=3)
summary(gmModel)
## Comparing observation and prediction
plot(gmModel,what="density")
par(mfrow=c(1,2))
plot(iris[,c(1,3)],col=iris[,5],main="Observed")
plot(iris[,c(1,3)],col=gmModel$classification,main="Predicted")
confusionMatrix(as.numeric(iris[,5]),gmModel$classification)
```

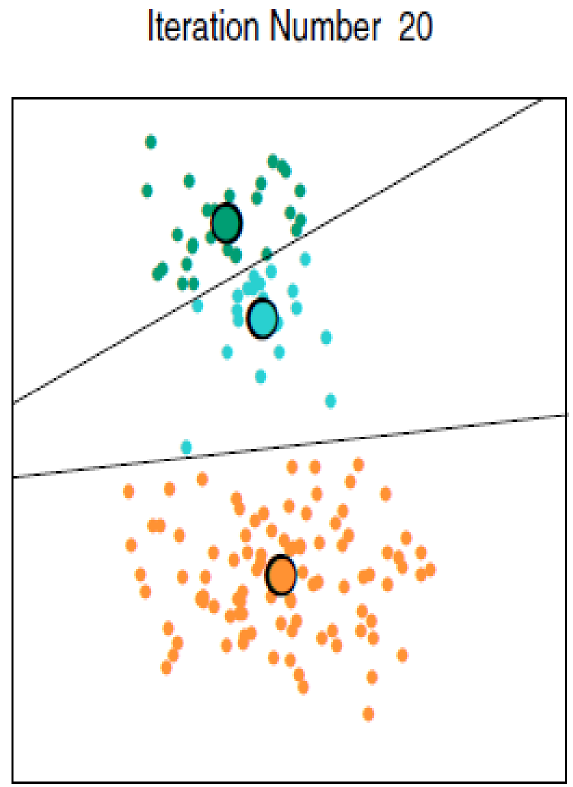
Problemas habituales

- Descripción y visualización
- Asociación
- **Segmentación**
- Clasificación
- Predicción

APRENDIZAJE POR REFUERZO | **APRENDIZAJE NO SUPERVISADO** | **APRENDIZAJE SUPERVISADO**

Distribution-based clustering
Centroid-based clustering: Non-overlapping (e.g. K-Means)

```
library(stats)
? kmeans
kmModel<-kmeans(iris[,-5],3,nstart=1)
summary(kmModel)
## Point center of two attributes
plot(iris[,c(1,3)],col=kmModel$cluster,main="K-Means")
points(kmModel$centers[,c(1,3)],col=1:3,pch=8,cex=2)
confusionMatrix(as.numeric(iris[,5]),kmModel$cluster)
```



K-Means is one of the most used iterative algorithms. It usually considers the Euclidean distance:

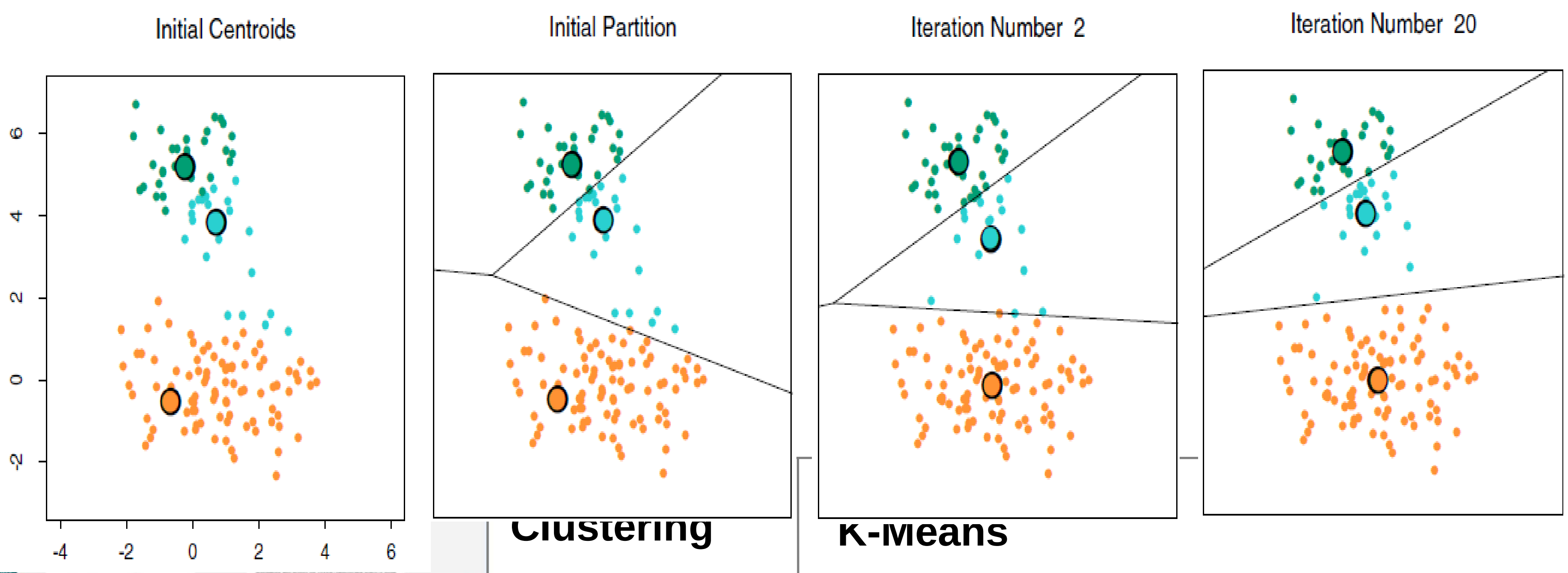
$$d(x_i, x_{i'}) = \sum_{j=1}^{p} (x_{ij} - x_{i'j})^2 = \|x_i - x_{i'}\|^2 \Rightarrow W(C) = \frac{1}{2} \sum_{k=1}^{K} \sum_{C(i)=k} \sum_{C(i')=k} \|x_i - x_{i'}\|^2 = \sum_{k=1}^{K} N_k \sum_{C(i)=k} \|x_i - \bar{x}_k\|^2,$$

The objective is to find **K** centroids solution of the following optimization problem:

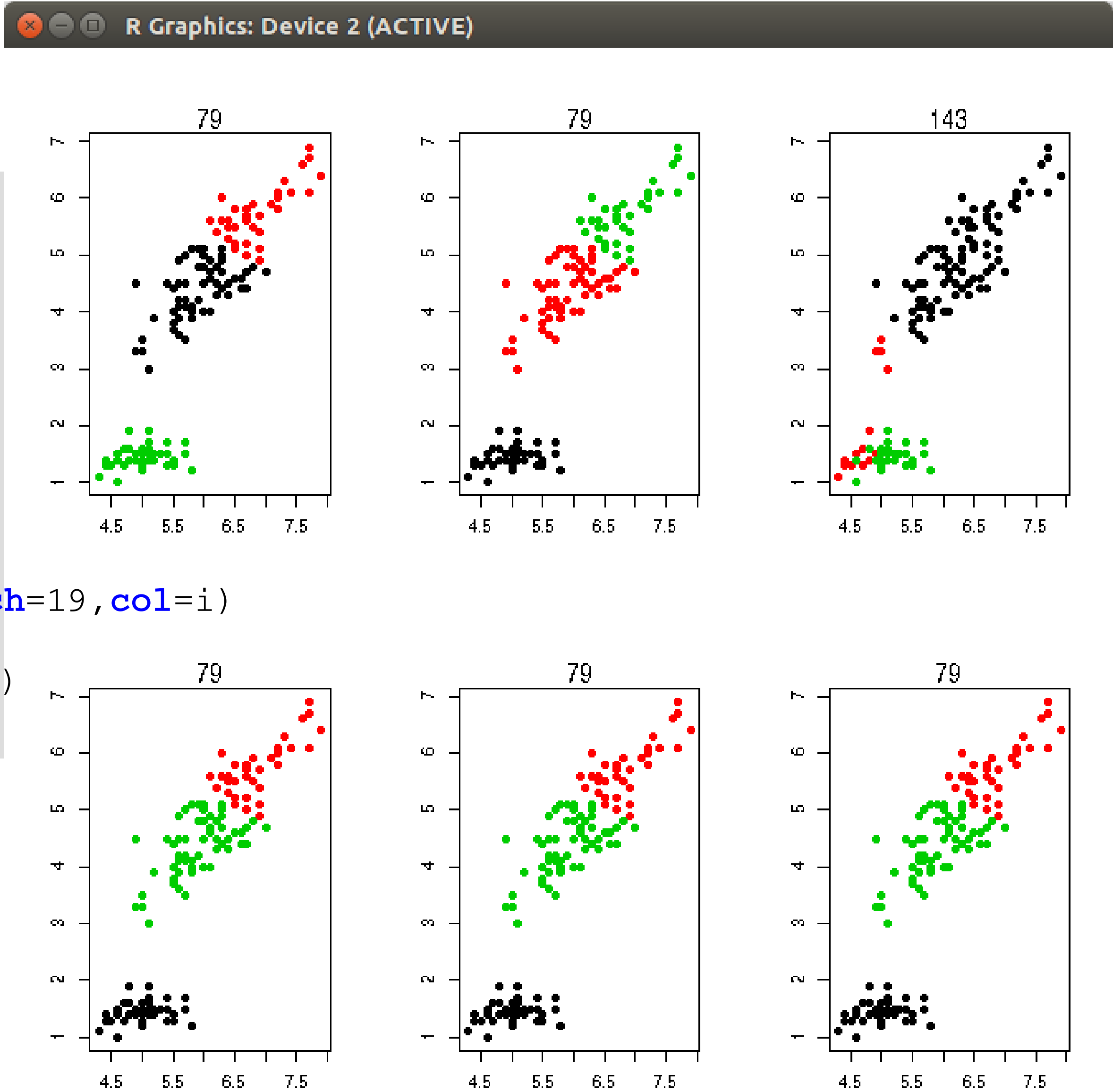
$$\min_{C, \{m_k\}_1^K} \sum_{k=1}^{K} N_k \sum_{C(i)=k} \|x_i - m_k\|^2.$$

Once the parameter **K** is defined:

- First assignation randomly defined.
- Repeat until converge or reach the maximum number of iterations:
  - Estimate the centroid for each cluster.
  - Re-define the clusters considering the new centroids.



```r
k<-3
par(mfrow=c(2,3))
j<-6
while(j>0){
  set.seed(j)
  j<-j-1
  km<-kmeans(iris[,-5],centers=k)
  plot(iris[,c(1,3)],type="n")
  for(i in 1:k){
    points(iris[km$cluster==i,c(1,3)],pch=19,col=i)
  }
  mtext(format(km$tot.withinss,digits=2))
}
```

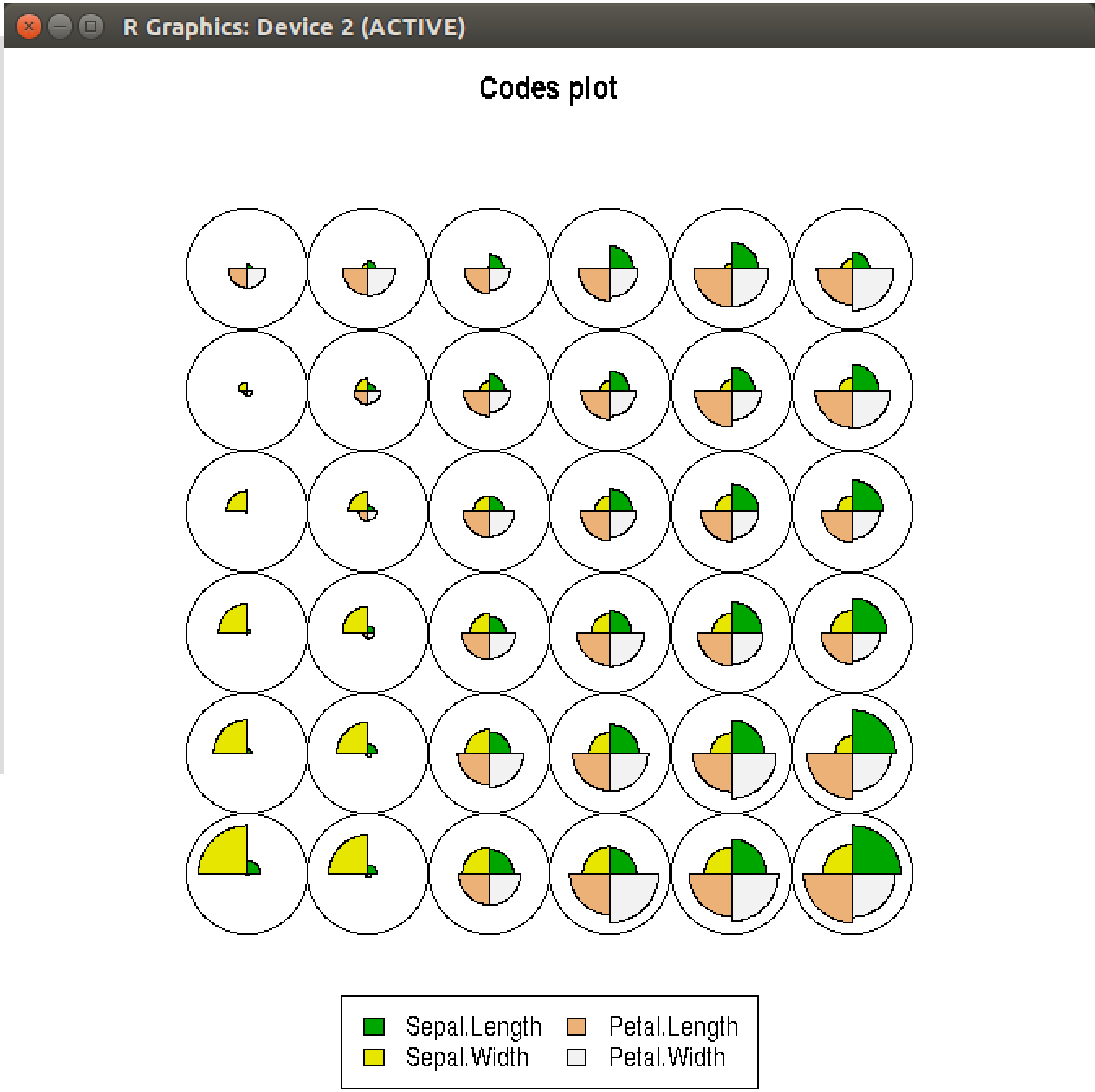


The main problems of this algorithm are:

- How to define the parameter **K**.
- The initialization could lead to different clusters.
- The clusterization is not incremental.

Self-Organising Maps can be considered a modification of the K-means algorithm including **topological constrains**

```
library(kohonen)
? som ## Clustering function.
? somgrid ## Definition of the topology.
som<-som(as.matrix(scale(iris[,-5])),
  somgrid(xdim=6,ydim=6,topo="rectangular"
## Should be used to visualize the data:
plot(som)
## Considering 3 classes
somR<-som(as.matrix(scale(iris[,-5])),
  somgrid(xdim=1,ydim=3,topo="rectangular"
somH<-som(as.matrix(scale(iris[,-5])),
  somgrid(xdim=3,ydim=1,topo="hexagonal"))
plot(somR)
plot(somH)
## We obtain the classification:
confusionMatrix(as.numeric(iris[,5]),
  somH$unit.classif)
```



Problemas habituales

- Descripción y visualización
- Asociación
- **Segmentación**
- Clasificación
- Predicción

APRENDIZAJE POR REFUERZO

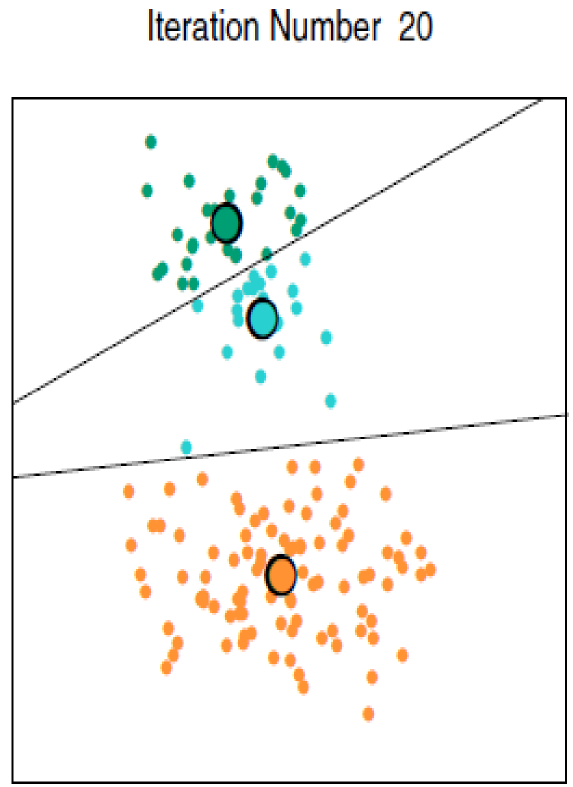
APRENDIZAJE NO SUPERVISADO

APRENDIZAJE SUPERVISADO

Distribution-based clustering

Centroid-based clustering: Non-overlapping (e.g. K-Means)

```
library(stats)
? kmeans
kmModel<-kmeans(iris[,-5],3,nstart=1)
summary(kmModel)
## Point center of two attributes
plot(iris[,c(1,3)],col=kmModel$cluster,main="K-Means")
points(kmModel$centers[,c(1,3)],col=1:3,pch=8,cex=2)
## How much clusters should we use?
totWithinss<-c(1:15)
for(i in 1:15){
  kmModel<-kmeans(iris[,-5],centers=i,nstart=1)
  totWithinss[i]<-kmModel$tot.withinss
}
plot(x=1:15,y=totWithinss,type="b",
  xlab="N. Of Cluster",ylab="Within groups sum of squares")
```



Problemas habituales

- Descripción y visualización
- Asociación
- **Segmentación**
- Clasificación
- Predicción

APRENDIZAJE POR REFUERZO | **APRENDIZAJE NO SUPERVISADO** | **APRENDIZAJE SUPERVISADO**

Distribution-based clustering
Centroid-based clustering: Non-overlapping (e.g. K-Means)
Centroid-based clustering: Fuzzy clustering (e.g. C-Means)

```
library(e1071)
? cmeans
cmModel<-cmeans(iris[,-5],3,iter.max=1,m=2,method="cmeans")
summary(cmModel)
## Point center of two attributes
plot(iris[,c(1,3)],col=cmModel$cluster,main="K-Means")
points(cmModel$centers[,c(1,3)],col=1:3,pch=8,cex=2)
confusionMatrix(as.numeric(iris[,5]),cmModel$cluster)
```

Centroid

$$c_k = \frac{\sum_x w_k(x)^m x}{\sum_x w_k(x)^m}.$$

Weights

$$\arg\min_C \sum_{i=1}^{n} \sum_{j=1}^{c} w_{ij}^m \left\|\mathbf{x}_i - \mathbf{c}_j\right\|^2,$$

$$w_{ij} = \frac{1}{\sum_{k=1}^{c} \left(\frac{\|\mathbf{x}_i - \mathbf{c}_j\|}{\|\mathbf{x}_i - \mathbf{c}_k\|}\right)^{\frac{2}{m-1}}}.$$

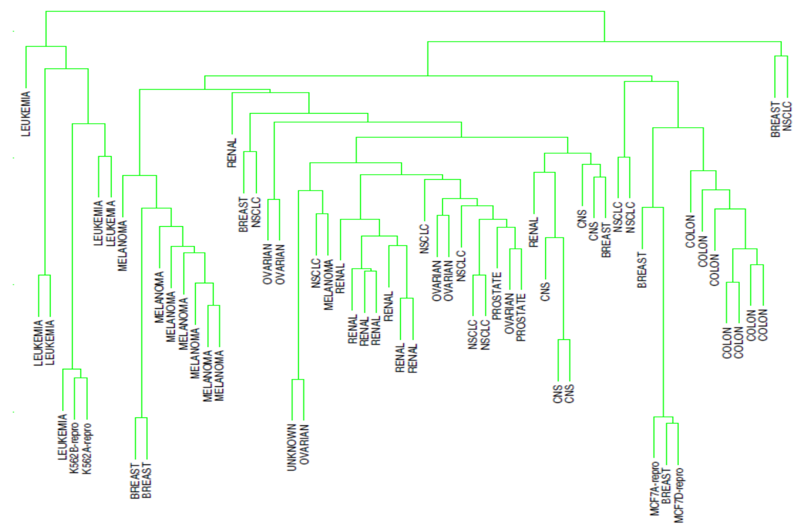
Problemas habituales

Descripción y visualización | Asociación | **Segmentación** | Clasificación | Predicción

APRENDIZAJE POR REFUERZO | **APRENDIZAJE NO SUPERVISADO** | **APRENDIZAJE SUPERVISADO**

Distribution-based clustering
Centroid-based clustering: Non-overlapping (e.g. K-Means)
Centroid-based clustering: Fuzzy clustering (e.g. C-Means)
Hierarchical clustering

```
library(stats)
require(sparcl)## Include colours for the leaves.
? hclust
d<-dist(iris[,-5],method="euclidean")
hcModel<-hclust(d,method="average")
summary(hcModel)
plot(hcModel,main="Dendrogram")
```



Problemas habituales

- Descripción y visualización
- Asociación
- **Segmentación**
- Clasificación
- Predicción

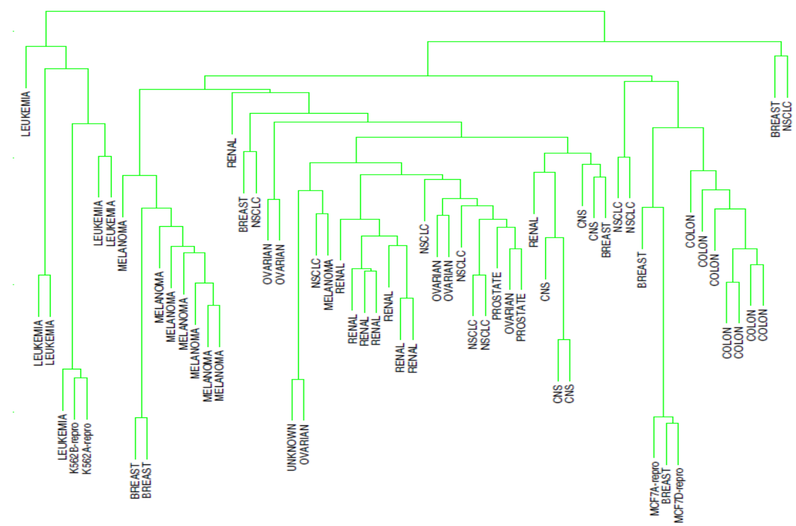
APRENDIZAJE POR REFUERZO

APRENDIZAJE NO SUPERVISADO

APRENDIZAJE SUPERVISADO

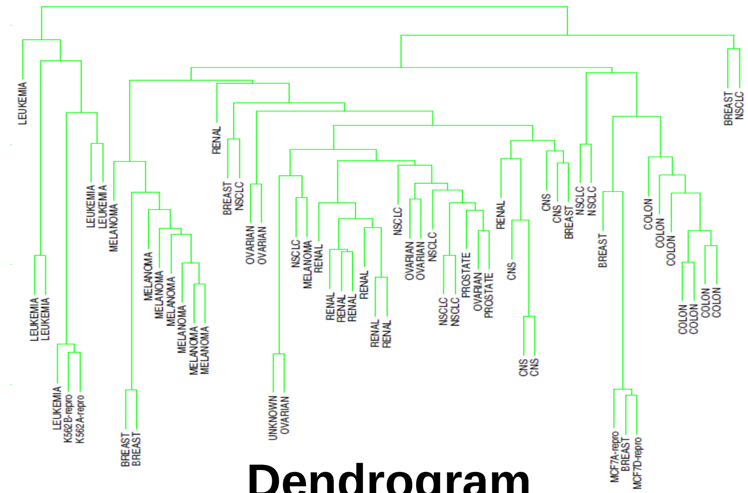
Distribution-based clustering
Centroid-based clustering: Non-overlapping (e.g. K-Means)
Centroid-based clustering: Fuzzy clustering (e.g. C-Means)
Hierarchical clustering

```
library(stats)
require(sparcl)## Include colours for the leaves.
? hclust
d<-dist(iris[,-5],method="euclidean")
hcModel<-hclust(d,method="average")
summary(hcModel)
plot(hcModel,main="Dendrogram")
## To obtain a classification, we should cut the tree
hc3<-cutree(hcModel,3)## 3 classes
ColorDendrogram(hcModel,y=hc3,branchlength=10)
confusionMatrix(as.numeric(iris[,5]),hc3)
```

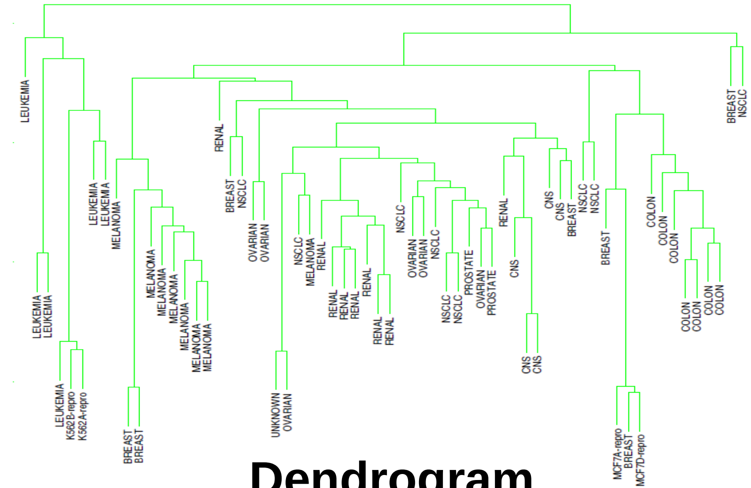


Hierarchical clustering is based on the similarity between members of the different groups/clusters to build the **dendrogram**. There are two approaches:

- **Agglomerative:** bottom up approach.
- **Divisive:** top down approach.



Dendrogram



Dendrogram

Hierarchical clustering is based on the similarity between members of the different groups/clusters to build the **dendrogram**. There are two approaches:

- **Agglomerative:** bottom up approach.
- **Divisive:** top down approach.

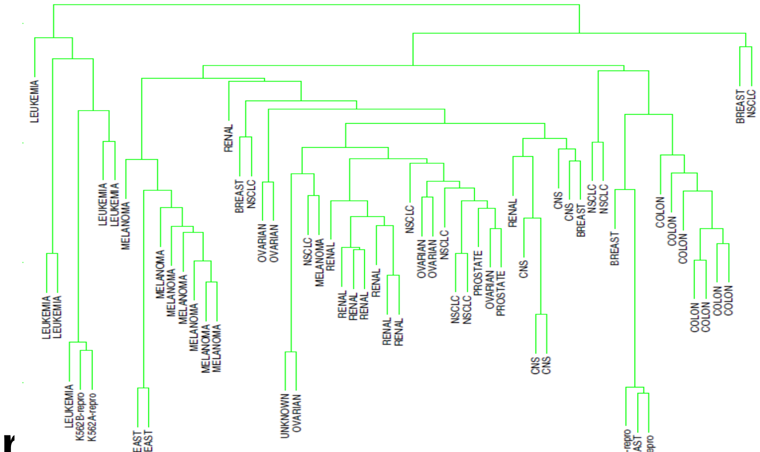
Linkage criterion determines the distance between the clusters and the agglomeration method:

- **Complete-linkage:** maximum of the distances.
- **Single-linkage:** minimum of the distances.
- **Average-linkage (UPGMA):** mean of the distances.
- **Centroid-linkage (UPGMC):** distances between centroids.
- ...

```
d<-dist(iris[,-5],method="euclidean")
## Available linkage criterion: "ward.D","ward.D2","single","complete","average",
## "mcquitty","median" or "centroid".
par(mfrow=c(2,2))
plot(hclust(d,method="complete"),main="Complete-Linkage",col="blue",axes=FALSE)
plot(hclust(d,method="single"),main="Single-Linkage",col="red",axes=FALSE)
plot(hclust(d,method="average"),main="Average-Linkage",col="green",axes=FALSE)
plot(hclust(d,method="centroid"),main="Centroid-Linkage",col="black",axes=FALSE)
```

Hierarchical clustering is based on the similarity between members of the different groups/clusters to build the **dendrogram**. There are two approaches:

- **Agglomerative:** bottom up approach.
- **Divisive:** top down approach.
  - Iteratively apply the K-means algorithm with **K=2**.
  - At each stage, the cluster with the **largest diameter** is selected, where the diameter of a cluster is the **largest dissimilarity** between any two of its observations (Macnaughton Smith et al. (1965), Kaufman and Rousseeuw (1990)).



Dendrogram

```
library(cluster)
? diana ## Divisive clustering algorithm.
d<-dist(iris[,-5],method="euclidean")
diModel<-diana(d,diss=TRUE,metric="euclidean")
plot(diModel)
di3<-cutree(diModel,3)## 3 classes
confusionMatrix(as.numeric(iris[,5]),di3)
confusionMatrix(as.numeric(iris[,5]),hc3)
```