Análisis No Supervisado

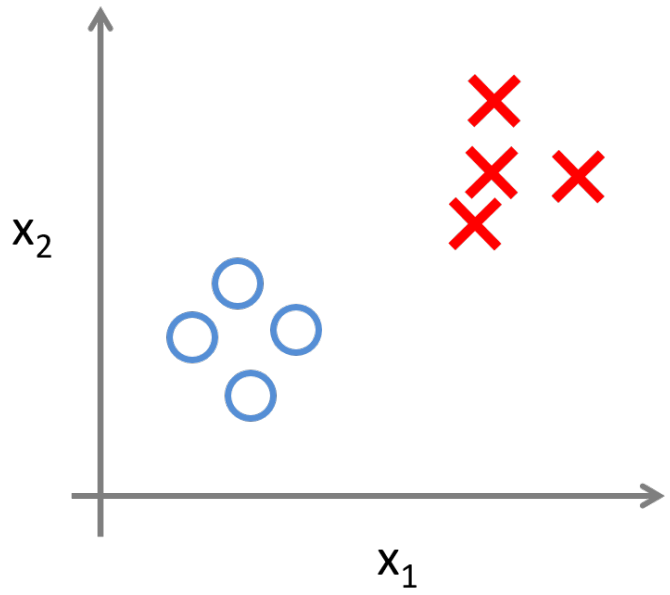
Alvaro J. Riascos Villegas
Universidad de los Andes y Quantil

Junio de 2021

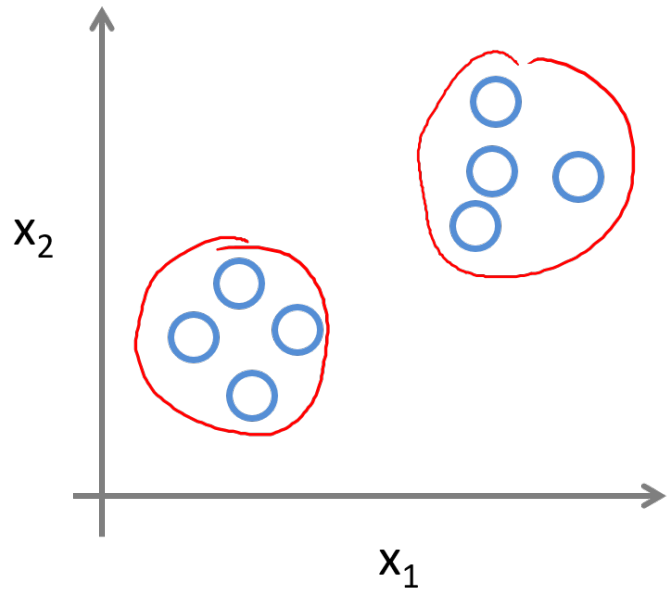
Contenido

Análisis Supervisado vs. No Supervisado

Supervised Learning



Unsupervised Learning



Análisis no supervisado

- En análisis no supervisado no tenemos "marcas".
- Tratamos de descubrir patrones o estructura entre los datos.
- Agrupar datos bajo algún criterio.
- Crear variables nuevas que "resuman" las variables.
- No tenemos como supervisar qué tan bueno es el análisis.

Análisis no supervisado

- En análisis no supervisado no tenemos "marcas".
- Tratamos de descubrir patrones o estructura entre los datos.
- Agrupar datos bajo algún criterio.
- Crear variables nuevas que "resuman" las variables.
- No tenemos como supervisar qué tan bueno es el análisis.

Análisis no supervisado

- En análisis no supervisado no tenemos "marcas".
- Tratamos de descubrir patrones o estructura entre los datos.
- Agrupar datos bajo algún criterio.
- Crear variables nuevas que "resuman" las variables.
- No tenemos como supervisar qué tan bueno es el análisis.

Análisis no supervisado

- En análisis no supervisado no tenemos "marcas".
- Tratamos de descubrir patrones o estructura entre los datos.
- Agrupar datos bajo algún criterio.
- Crear variables nuevas que "resuman" las variables.
- No tenemos como supervisar qué tan bueno es el análisis.

Análisis no supervisado

- En análisis no supervisado no tenemos "marcas".
- Tratamos de descubrir patrones o estructura entre los datos.
- Agrupar datos bajo algún criterio.
- Crear variables nuevas que "resuman" las variables.
- No tenemos como supervisar qué tan bueno es el análisis.

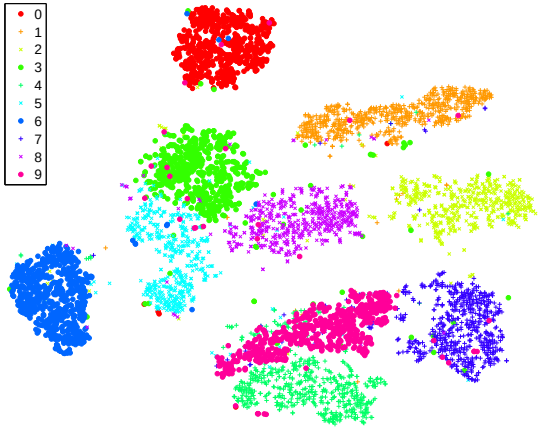
Aplicaciones

Aplicaciones

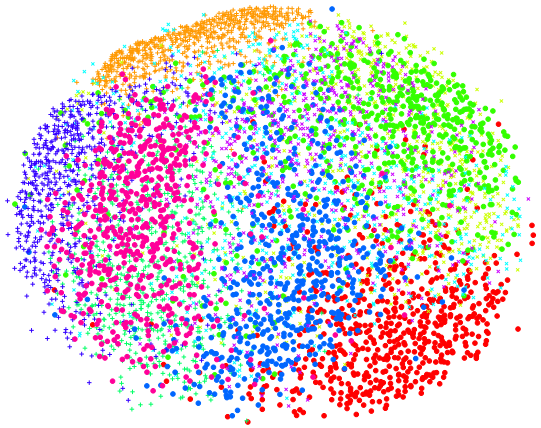
- Segmentar clientes para ofrecerles productos diferentes.

Aplicaciones

- Reducir dimensionalidad.



(a) Visualization by t-SNE.



(b) Visualization by Sammon mapping.

Figure 2: Visualizations of 6,000 handwritten digits from the MNIST data set.

Aplicaciones

- Reducir dimensionalidad (UMAP).

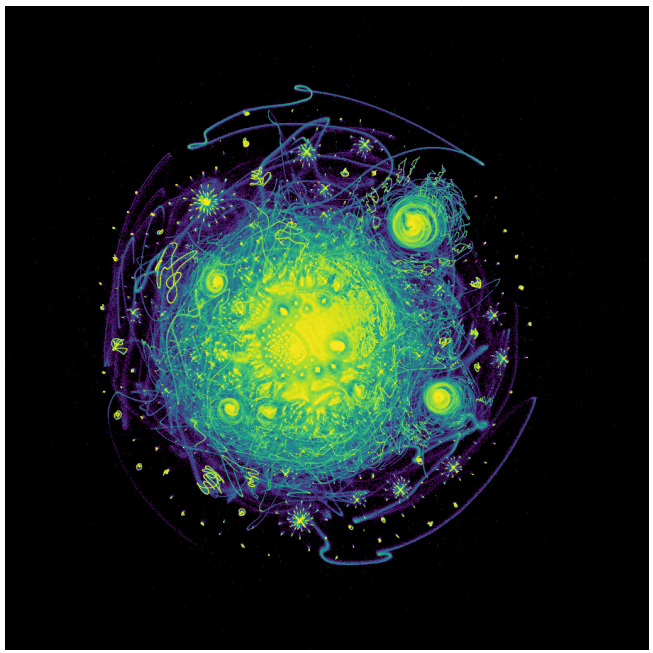
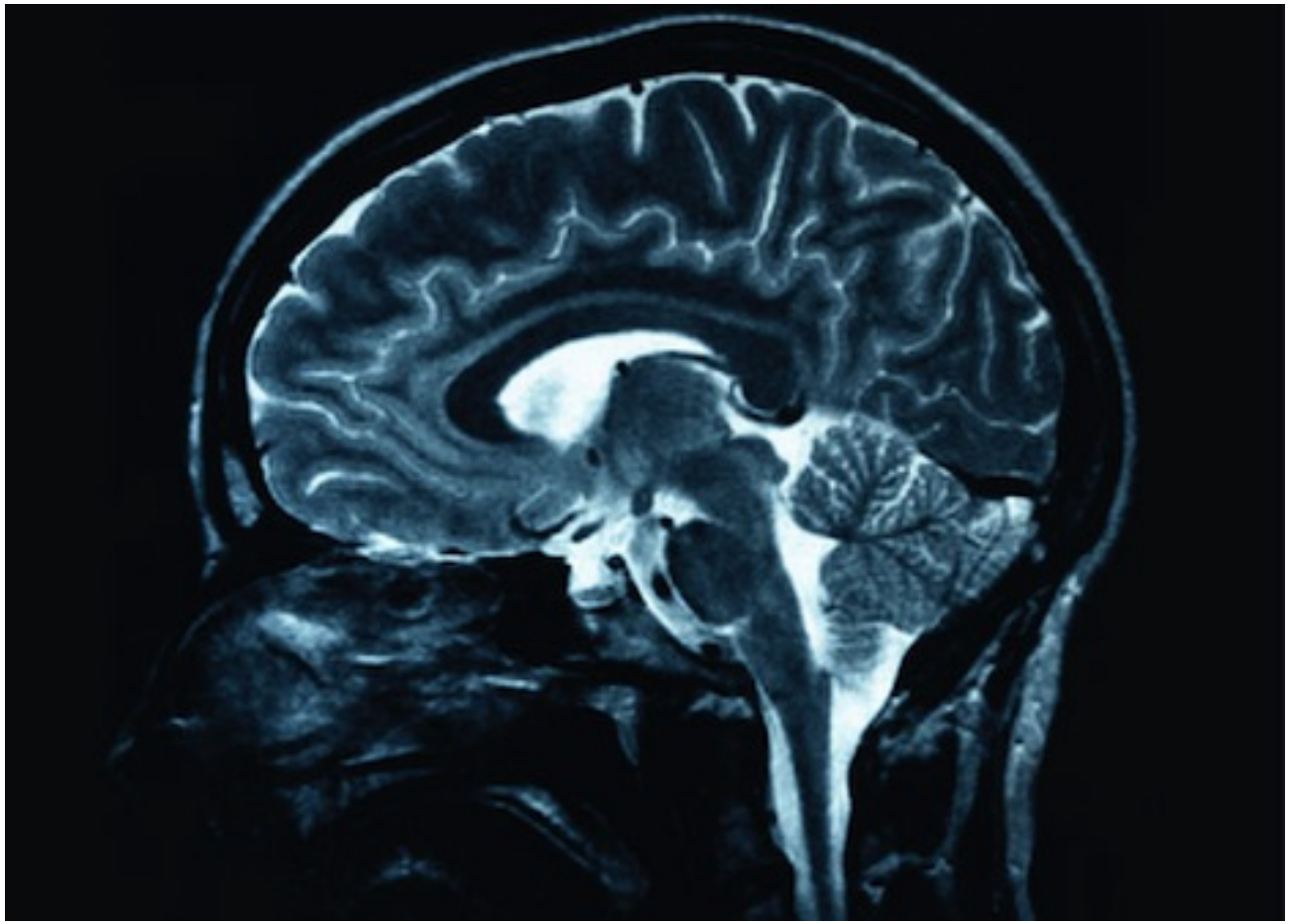


Figure 9: Visualization of 30,000,000 integers as represented by binary vectors of prime divisibility, colored by density of points.

Aplicaciones

- Segmentacion de imágenes.



¿Supervisado o No Supervisado?

1. Predecir el clima en una ciudad dadas unas variables históricas.
2. Definir categorías entre diferentes clientes para ofrecer un servicio.
3. Detectar anomalías en flujos de datos.
4. Agrupar noticias en temas similares.
5. Detectar tumores cerebrales.

¿Supervisado o No Supervisado?

1. Predecir el clima en una ciudad dadas unas variables históricas.
2. Definir categorías entre diferentes clientes para ofrecer un servicio.
3. Detectar anomalías en flujos de datos.
4. Agrupar noticias en temas similares.
5. Detectar tumores cerebrales.

¿Supervisado o No Supervisado?

1. Predecir el clima en una ciudad dadas unas variables históricas.
2. Definir categorías entre diferentes clientes para ofrecer un servicio.
3. Detectar anomalías en flujos de datos.
4. Agrupar noticias en temas similares.
5. Detectar tumores cerebrales.

¿Supervisado o No Supervisado?

1. Predecir el clima en una ciudad dadas unas variables históricas.
2. Definir categorías entre diferentes clientes para ofrecer un servicio.
3. Detectar anomalías en flujos de datos.
4. Agrupar noticias en temas similares.
5. Detectar tumores cerebrales.

¿Supervisado o No Supervisado?

1. Predecir el clima en una ciudad dadas unas variables históricas.
2. Definir categorías entre diferentes clientes para ofrecer un servicio.
3. Detectar anomalías en flujos de datos.
4. Agrupar noticias en temas similares.
5. Detectar tumores cerebrales.

Algoritmos de Análisis no Supervisado

- Reglas de asociación.
- K−medias.
- Modelos de mixturas.
- *Isolation Forest*.
- Local outlier factor (LOF).
- SVM de una clase.
- Componentes Principales.
- Autoencoders.
- Aprendizaje en variedades.
- Métodos para series de tiempo.

Contenido

Reglas de Asociación

- La idea es caracterizar la distribución de X.
- Para esto buscamos regiones del espacio en el que X toma valores donde la densidad sea alta.
- Sea X un vector de p variables (estas pueden ser continuas o categóricas) y S_j el soporte de X_j y $s_j \subseteq S_j$. La idea es encontrar valores $(s_1, s_2, ..., s_p)$ tales que

$$P[\bigcap_{j=1}^{p}(X_j \in s_j)] > s, \tag{1}$$

  donde s es un nivel de soporte deseado. En otras palabras, se buscan valores de variables que tiendan a darse simultáneamente.
- Un estimador natural es la fracción de observaciones para las cuales X_i es mayor que s_i para todo i.

Reglas de Asociación

- La idea es caracterizar la distribución de X.
- Para esto buscamos regiones del espacio en el que X toma valores donde la densidad sea alta.
- Sea X un vector de p variables (estas pueden ser continuas o categóricas) y S_j el soporte de X_j y $s_j \subseteq S_j$. La idea es encontrar valores $(s_1, s_2, ..., s_p)$ tales que

$$P[\bigcap_{j=1}^{p}(X_j \in s_j)] > s, \tag{1}$$

  donde s es un nivel de soporte deseado. En otras palabras, se buscan valores de variables que tiendan a darse simultáneamente.
- Un estimador natural es la fracción de observaciones para las cuales X_i es mayor que s_i para todo i.

Reglas de Asociación

- La idea es caracterizar la distribución de X.
- Para esto buscamos regiones del espacio en el que X toma valores donde la densidad sea alta.
- Sea X un vector de p variables (estas pueden ser continuas o categóricas) y S_j el soporte de X_j y $s_j \subseteq S_j$. La idea es encontrar valores $(s_1, s_2, ..., s_p)$ tales que

$$P[\bigcap_{j=1}^{p}(X_j \in s_j)] > s, \tag{1}$$

  donde s es un nivel de soporte deseado. En otras palabras, se buscan valores de variables que tiendan a darse simultáneamente.
- Un estimador natural es la fracción de observaciones para las cuales X_i es mayor que s_i para todo i.

Reglas de Asociación: Simplificación

- Para simplificar el problema vamos a permitir sólo regiones del tipo: un solo valor de la variable X_j, $v_{0,j}$ o la totalidad del soporte de esa variable S_j.
- Esto simplifica el problema a encontrar $\Gamma \subset \{1, ..., p\}$ y valores $\{v_{0,j}\}_{j \in \Gamma}$ tal que:

$$P[\bigcap_{j \in \Gamma} (X_j = v_{0,j})] > s, \tag{2}$$

donde $v_{0,j}$ es uno de los valores posibles de la variable j.

Reglas de Asociación: Simplificación

- Para simplificar el problema vamos a permitir sólo regiones del tipo: un solo valor de la variable X_j, $v_{0,j}$ o la totalidad del soporte de esa variable S_j.
- Esto simplifica el problema a encontrar $\Gamma \subset \{1, ..., p\}$ y valores $\{v_{0,j}\}_{j \in \Gamma}$ tal que:

$$P[\bigcap_{j \in \Gamma}(X_j = v_{0,j})] > s, \tag{2}$$

donde $v_{0,j}$ es uno de los valores posibles de la variable j.

Reglas de Asociación: Simplificación

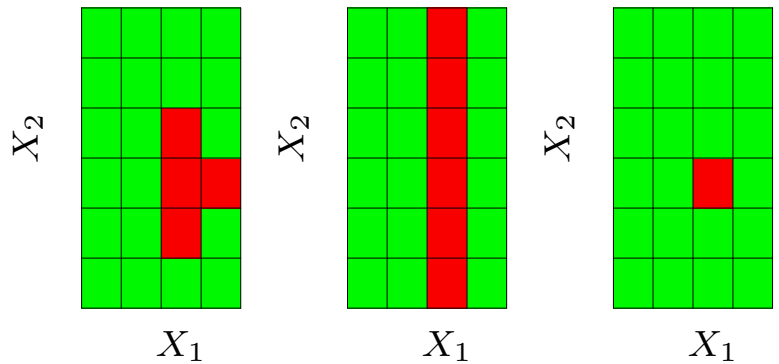


FIGURE 14.1. *Simplifications for association rules. Here there are two inputs X_1 and X_2, taking four and six distinct values, respectively. The red squares indicate areas of high density. To simplify the computations, we assume that the derived subset corresponds to either a single value of an input or all values. With this assumption we could find either the middle or right pattern, but not the left one.*

Reglas de Asociación: Representación con variables dummy

- Si además suponemos que cada variable X_j puede tomar sólo un número finito de variables, podemos definir K como el número total de dummis para describir los diferentes valores de las variables X_j:

$$K = \sum_{j=1}^{p} \mid S_j \mid$$

  y creamos Z_k variables dummy por cada una de las variables K.

- El problema ahora es encontrar: $\Upsilon \subset \{1, 2, 3, ..., K\}$ tal que:

$$P[\bigcap_{j \in \Upsilon} (Z_j = 1)] > s \tag{3}$$

Reglas de Asociación: Representación con variables dummy

- Si además suponemos que cada variable X_j puede tomar sólo un número finito de variables, podemos definir K como el número total de dummis para describir los diferentes valores de las variables X_j:

$$K = \sum_{j=1}^{p} \mid S_j \mid$$

  y creamos Z_k variables dummy por cada una de las variables K.

- El problema ahora es encontrar: $\Upsilon \subset \{1, 2, 3, ..., K\}$ tal que:

$$P[\bigcap_{j \in \Upsilon} (Z_j = 1)] > s \tag{3}$$

Reglas de Asociación

- Todo Υ tal que $P[\bigcap_{j\in\Upsilon}(Z_j = 1)] > s$ representa un grupo de items (canasta) que aparecen con una frecuencia deseada (el soporte mínimo).
- El estimador natural es la frecuencia:

$$Pr[\prod_{j\in\Upsilon}(Z^j = 1)] = \frac{|\{z : \forall j \in \Upsilon\, z_j = 1\}|}{N} = \frac{1}{N}\sum_{i=1}^{N}\prod_{j\in\Upsilon} z_{ij} \quad (4)$$

donde z_i es la observación i (canasta) y z_{ij} denota si el item j está en esa canasta.

Reglas de Asociación

- Todo Υ tal que $P[\bigcap_{j\in\Upsilon}(Z_j = 1)] > s$ representa un grupo de items (canasta) que aparecen con una frecuencia deseada (el soporte mínimo).
- El estimador natural es la frecuencia:

$$Pr[\prod_{j\in\Upsilon}(Z^j = 1)] = \frac{|\{z : \forall j \in \Upsilon\, z_j = 1\}|}{N} = \frac{1}{N}\sum_{i=1}^{N}\prod_{j\in\Upsilon} z_{ij} \quad (4)$$

donde z_i es la observación i (canasta) y z_{ij} denota si el item j está en esa canasta.

Reglas de Asociación

- Las preguntas fundamentales que nos hacemos son:
  1. Dado un conjunto de índices $\Upsilon \subset \{1, ..., K\}$ de alto soporte se pueden particionar de muchas formas en dos subconjuntos A, B. Cada uno define una regla de la forma $A \Rightarrow B$, A se llama el antecedente y B el consecuente. El soporte de la regla es el soporte de Υ. Representa la probabilidad de observar los items A y B en una canasta (i.e., la unión de los items A y los items B).
  2. La confianza en la regla $A \Rightarrow B$, $C(A \Rightarrow B)$ se define como $C(A \Rightarrow B) = \frac{Supp(A \Rightarrow B)}{Supp(A)}$. Es un estimador de $P(B \mid A)$.
  3. El *Lift* de la regla se $A \Rightarrow B$ se define como: $L(A \Rightarrow B) = \frac{C(A \Rightarrow B)}{Supp(B)}$. Es una medida del grado de asociación: $\frac{P(A \wedge B)}{P(A)P(B)}$.
  4. La confianza de B es $Supp(B) = P(B)$.

Reglas de Asociación

- Las preguntas fundamentales que nos hacemos son:
  1. Dado un conjunto de índices $\Upsilon \subset \{1, ..., K\}$ de alto soporte se pueden particionar de muchas formas en dos subconjuntos A, B. Cada uno define una regla de la forma $A \Rightarrow B$, A se llama el antecedente y B el consecuente. El soporte de la regla es el soporte de Υ. Representa la probabilidad de observar los items A y B en una canasta (i.e., la unión de los items A y los items B).
  2. La confianza en la regla $A \Rightarrow B$, $C(A \Rightarrow B)$ se define como $C(A \Rightarrow B) = \frac{Supp(A \Rightarrow B)}{Supp(A)}$. Es un estimador de $P(B \mid A)$.
  3. El *Lift* de la regla se $A \Rightarrow B$ se define como: $L(A \Rightarrow B) = \frac{C(A \Rightarrow B)}{Supp(B)}$. Es una medida del grado de asociación: $\frac{P(A \wedge B)}{P(A)P(B)}$.
  4. La confianza de B es $Supp(B) = P(B)$.

Reglas de Asociación

- Las preguntas fundamentales que nos hacemos son:
  1. Dado un conjunto de índices $\Upsilon \subset \{1, ..., K\}$ de alto soporte se pueden particionar de muchas formas en dos subconjuntos A, B. Cada uno define una regla de la forma $A \Rightarrow B$, A se llama el antecedente y B el consecuente. El soporte de la regla es el soporte de Υ. Representa la probabilidad de observar los items A y B en una canasta (i.e., la unión de los items A y los items B).
  2. La confianza en la regla $A \Rightarrow B$, $C(A \Rightarrow B)$ se define como $C(A \Rightarrow B) = \frac{Supp(A \Rightarrow B)}{Supp(A)}$. Es un estimador de $P(B \mid A)$.
  3. El *Lift* de la regla se $A \Rightarrow B$ se define como: $L(A \Rightarrow B) = \frac{C(A \Rightarrow B)}{Supp(B)}$. Es una medida del grado de asociación: $\frac{P(A \wedge B)}{P(A)P(B)}$.
  4. La confianza de B es $Supp(B) = P(B)$.

Reglas de Asociación

- Las preguntas fundamentales que nos hacemos son:
  1. Dado un conjunto de índices $\Upsilon \subset \{1, ..., K\}$ de alto soporte se pueden particionar de muchas formas en dos subconjuntos A, B. Cada uno define una regla de la forma $A \Rightarrow B$, A se llama el antecedente y B el consecuente. El soporte de la regla es el soporte de Υ. Representa la probabilidad de observar los items A y B en una canasta (i.e., la unión de los items A y los items B).
  2. La confianza en la regla $A \Rightarrow B$, $C(A \Rightarrow B)$ se define como $C(A \Rightarrow B) = \frac{Supp(A \Rightarrow B)}{Supp(A)}$. Es un estimador de $P(B \mid A)$.
  3. El *Lift* de la regla se $A \Rightarrow B$ se define como: $L(A \Rightarrow B) = \frac{C(A \Rightarrow B)}{Supp(B)}$. Es una medida del grado de asociación: $\frac{P(A \wedge B)}{P(A)P(B)}$.
  4. La confianza de B es $Supp(B) = P(B)$.

Reglas de Asociación

- Las preguntas fundamentales que nos hacemos son:
  1. Dado un conjunto de índices $\Upsilon \subset \{1, ..., K\}$ de alto soporte se pueden particionar de muchas formas en dos subconjuntos A, B. Cada uno define una regla de la forma $A \Rightarrow B$, A se llama el antecedente y B el consecuente. El soporte de la regla es el soporte de Υ. Representa la probabilidad de observar los items A y B en una canasta (i.e., la unión de los items A y los items B).
  2. La confianza en la regla $A \Rightarrow B$, $C(A \Rightarrow B)$ se define como $C(A \Rightarrow B) = \frac{Supp(A \Rightarrow B)}{Supp(A)}$. Es un estimador de $P(B \mid A)$.
  3. El *Lift* de la regla se $A \Rightarrow B$ se define como: $L(A \Rightarrow B) = \frac{C(A \Rightarrow B)}{Supp(B)}$. Es una medida del grado de asociación: $\frac{P(A \wedge B)}{P(A)P(B)}$.
  4. La confianza de B es $Supp(B) = P(B)$.

Reglas de Asociación: Ejemplo

Association rule 2: Support 13.4%, confidence 80.8%, and lift 2.13.

$$\left[\begin{array}{rcl} \text{language in home} & = & \textit{English} \\ \text{householder status} & = & \textit{own} \\ \text{occupation} & = & \{\textit{professional/managerial}\} \end{array}\right] \Downarrow \text{income} \geq \textit{\$40,000}$$

Association rule 3: Support 26.5%, confidence 82.8% and lift 2.15.

$$\left[\begin{array}{rcl} \text{language in home} & = & \textit{English} \\ \text{income} & < & \textit{\$40,000} \\ \text{marital status} & = & \textit{not married} \\ \text{number of children} & = & \textit{0} \end{array}\right] \Downarrow \text{education} \notin \{\textit{college graduate, graduate study}\}$$

Reglas de Asociación: Algoritmo Apriori

- Obsérvese que $\Upsilon' \subset \Upsilon$ y $supp(\Upsilon) > s$ entonces $supp(\Upsilon') > s$.
- El algoritmo Apriori sigue los siguientes pasos:
  1. Sea $k = 1$. Recorra todos los items individuales que tengan soporte mayor a s. Elimine todas los items que no tengan suficiente soporte (representatividad). Recuérdese la anterior observación).
  2. Para $k + 1$ recorra todas las combinaciones de $k + 1$ items que se puedan construir con los items que han sobrevivido hasta este punto (que no fueron eliminados en la ronda anterior). Conserve solo los conjuntos de $k + 1$ items que tienen soporte mínimo s. Elimine todos los conjuntos de $k + 1$ item que no tiene suficiente representatividad.

Reglas de Asociación: Algoritmo Apriori

- Obsérvese que $\Upsilon' \subset \Upsilon$ y $supp(\Upsilon) > s$ entonces $supp(\Upsilon') > s$.
- El algoritmo Apriori sigue los siguientes pasos:
  1. Sea $k = 1$. Recorra todos los items individuales que tengan soporte mayor a s. Elimine todas los items que no tengan suficiente soporte (representatividad). Recuérdese la anterior observación).
  2. Para $k + 1$ recorra todas las combinaciones de $k + 1$ items que se puedan construir con los items que han sobrevivido hasta este punto (que no fueron eliminados en la ronda anterior). Conserve solo los conjuntos de $k + 1$ items que tienen soporte mínimo s. Elimine todos los conjuntos de $k + 1$ item que no tiene suficiente representatividad.

Reglas de Asociación: Algoritmo Apriori

- Obsérvese que $\Upsilon' \subset \Upsilon$ y $supp(\Upsilon) > s$ entonces $supp(\Upsilon') > s$.
- El algoritmo Apriori sigue los siguientes pasos:
  1. Sea $k = 1$. Recorra todos los items individuales que tengan soporte mayor a s. Elimine todas los items que no tengan suficiente soporte (representatividad). Recuérdese la anterior observación).
  2. Para $k + 1$ recorra todas las combinaciones de $k + 1$ items que se puedan construir con los items que han sobrevivido hasta este punto (que no fueron eliminados en la ronda anterior). Conserve solo los conjuntos de $k + 1$ items que tienen soporte mínimo s. Elimine todos los conjuntos de $k + 1$ item que no tiene suficiente representatividad.

Reglas de Asociación: Algoritmo Apriori

- Obsérvese que $\Upsilon' \subset \Upsilon$ y $supp(\Upsilon) > s$ entonces $supp(\Upsilon') > s$.
- El algoritmo Apriori sigue los siguientes pasos:
  1. Sea $k = 1$. Recorra todos los items individuales que tengan soporte mayor a s. Elimine todas los items que no tengan suficiente soporte (representatividad). Recuérdese la anterior observación).
  2. Para $k + 1$ recorra todas las combinaciones de $k + 1$ items que se puedan construir con los items que han sobrevivido hasta este punto (que no fueron eliminados en la ronda anterior). Conserve solo los conjuntos de $k + 1$ items que tienen soporte mínimo s. Elimine todos los conjuntos de $k + 1$ item que no tiene suficiente representatividad.

Contenido

¿Para qué sirve?

- Es un algoritmo para agrupar puntos "cercanos"
- Se crean K "prototipos" de datos que representan cada clúster.
- Se puede usar para detectar datos anómalos.
- Se puede usar para reducir la dimensionalidad del problema.

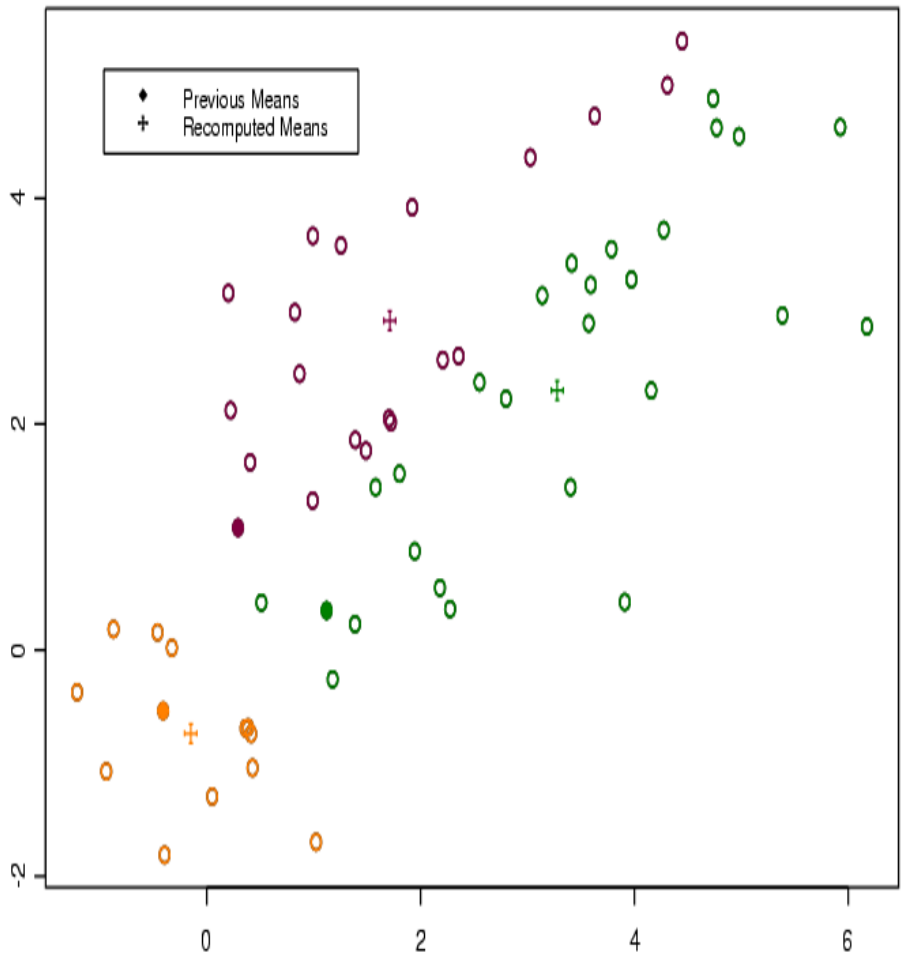
El algoritmo

1. Escoger K centroides iniciales aleatoriamente.
2. Repetir los siguientes pasos hasta convergencia:
   - Asignar cada observación al clúster cuyo centroide sea más cercano.
   - Reasignar el centroide del clúster *i* al promedio de las observaciones de ese clúster.

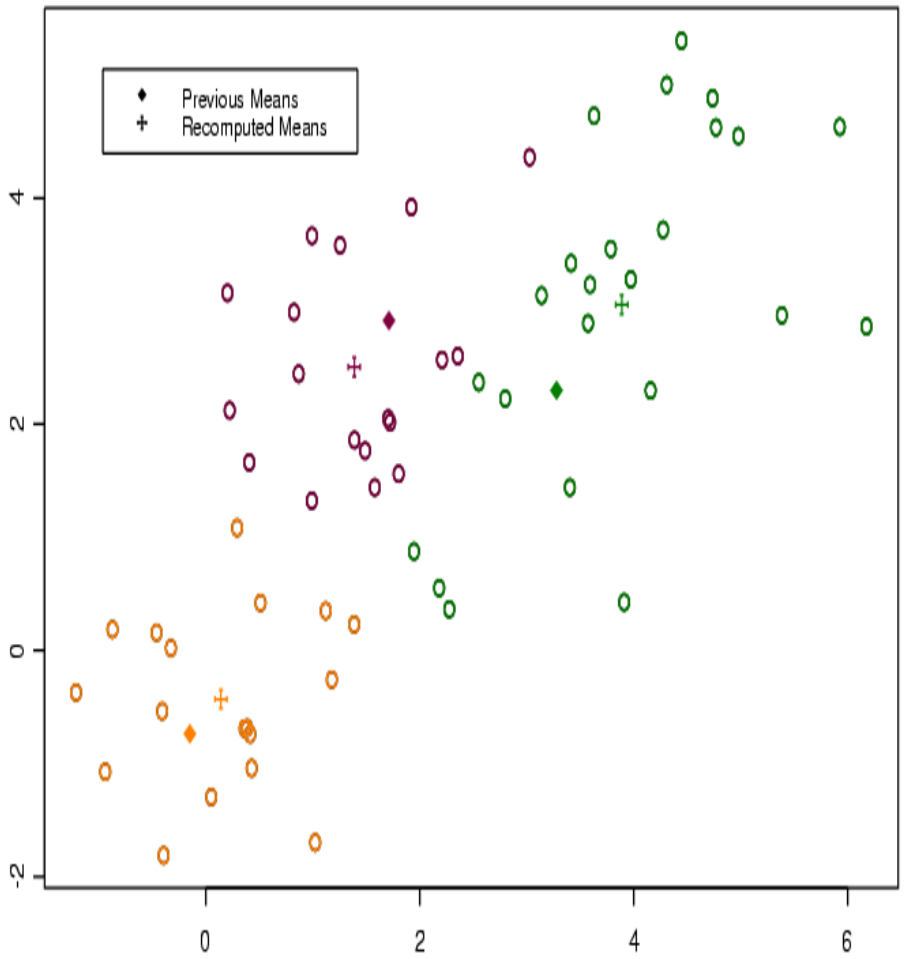
Algoritmo Alternativo

1. Asignar a cada observación aleatoriamente un número entre 1 y K.
2. Repetir los siguientes pasos hasta convergencia:
   - Definir el centroide del clúster i como el promedio de las observaciones de ese clúster.
   - Asignar cada observación al clúster cuyo centroide sea más cercano.

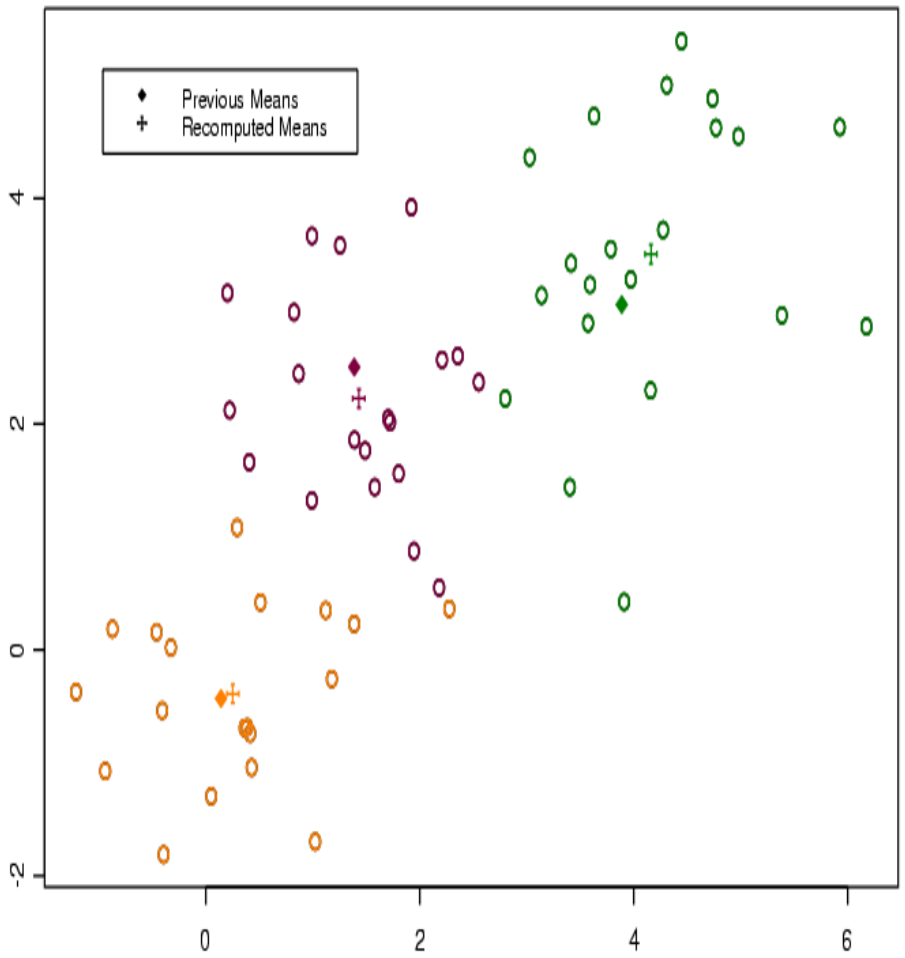
Ejemplo: Algoritmo



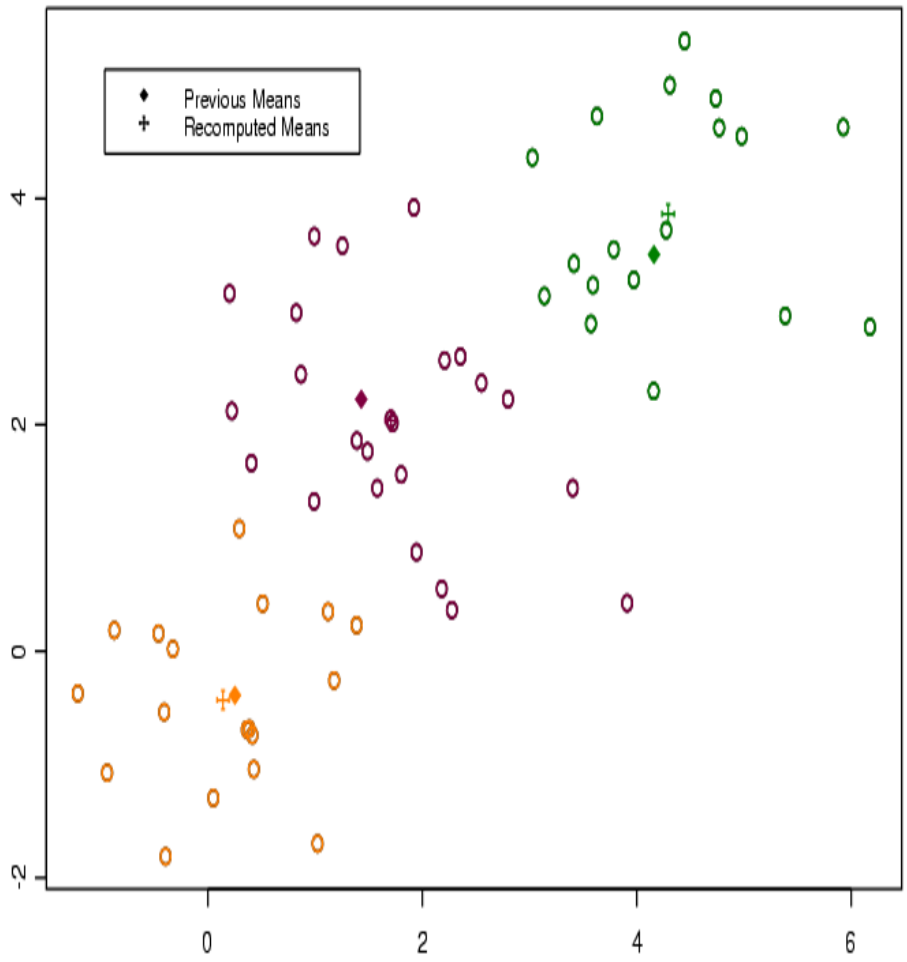
Ejemplo: Algoritmo



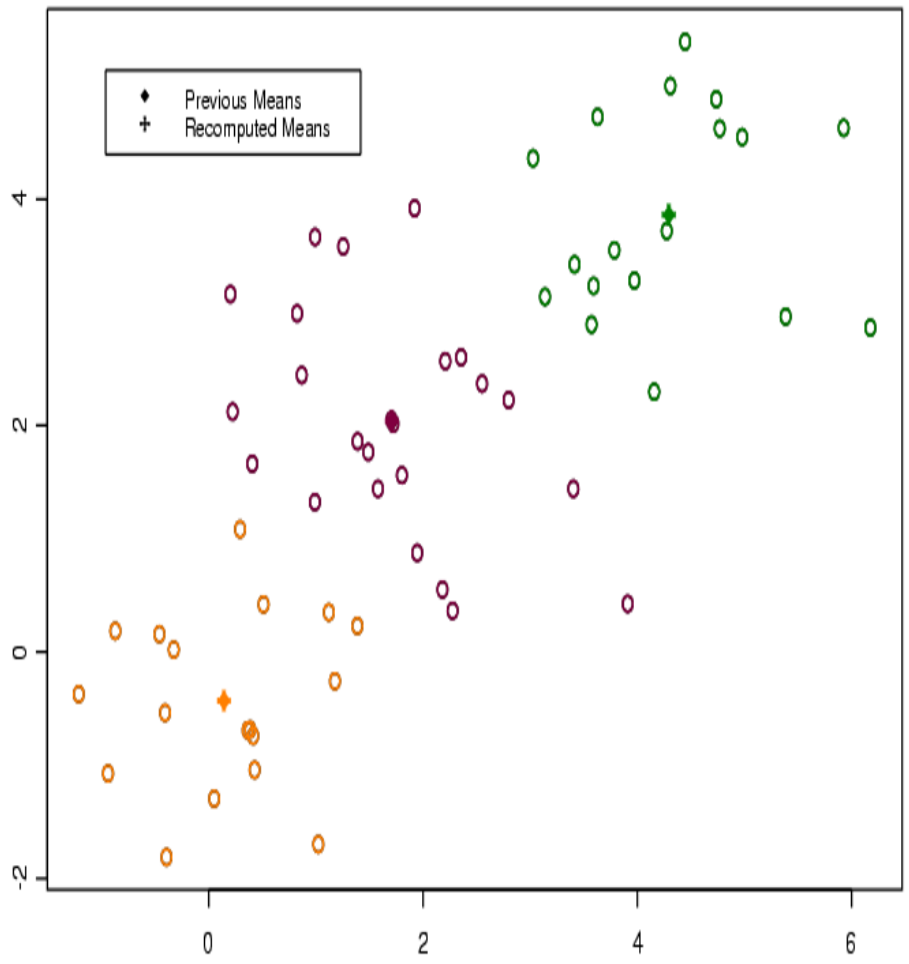
Ejemplo: Algoritmo



Ejemplo: Algoritmo



Ejemplo: Algoritmo



Ejemplo: Particiones

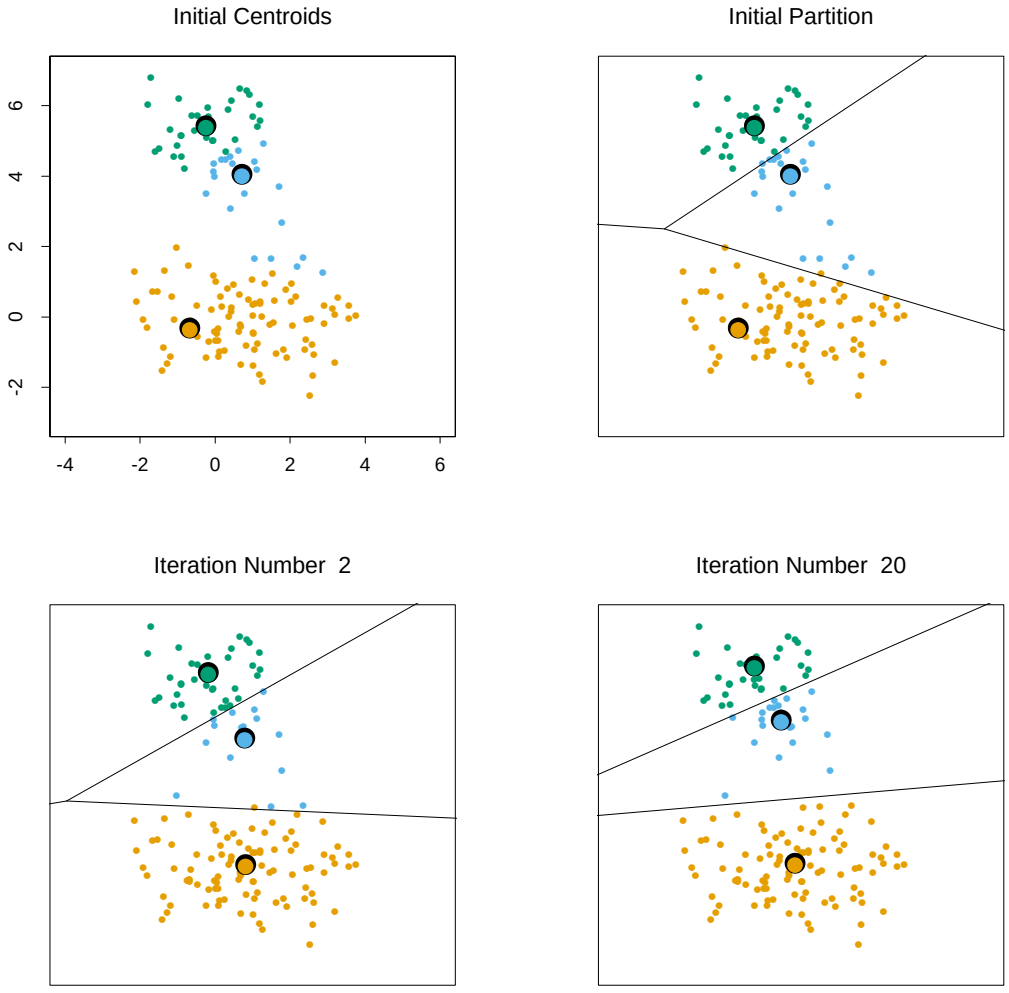
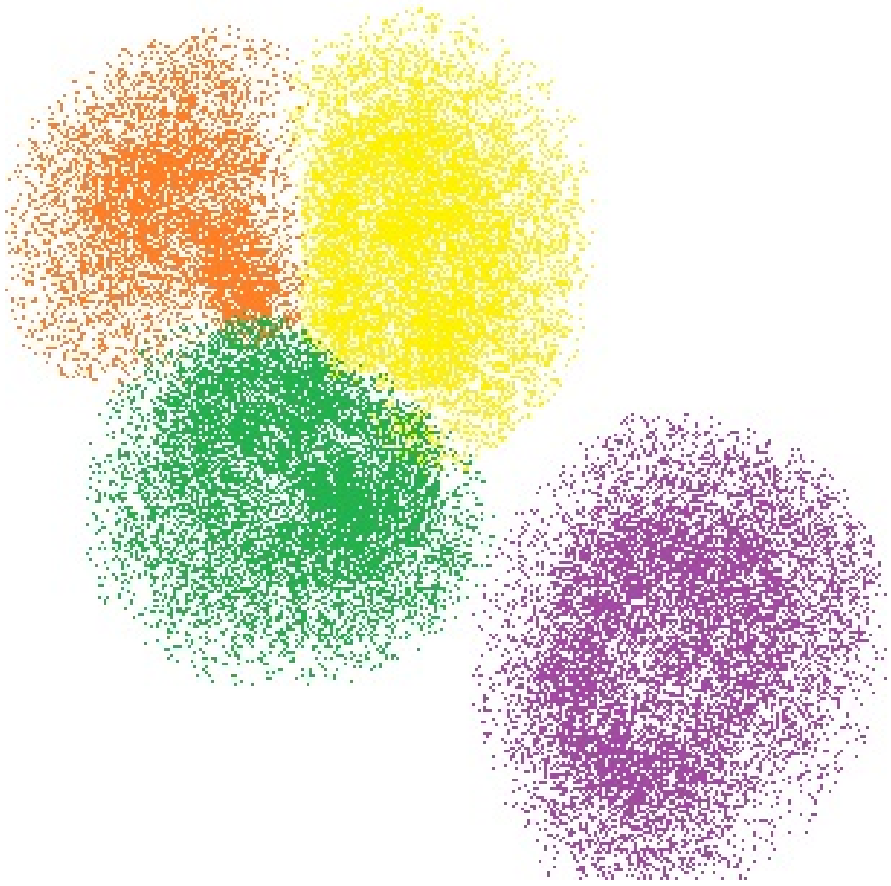


FIGURE 14.6. *Successive iterations of the K-means clustering algorithm for the simulated data of Figure 14.4.*

K-Medias: Ejemplo



K-Medias: Efecto estándarización variables

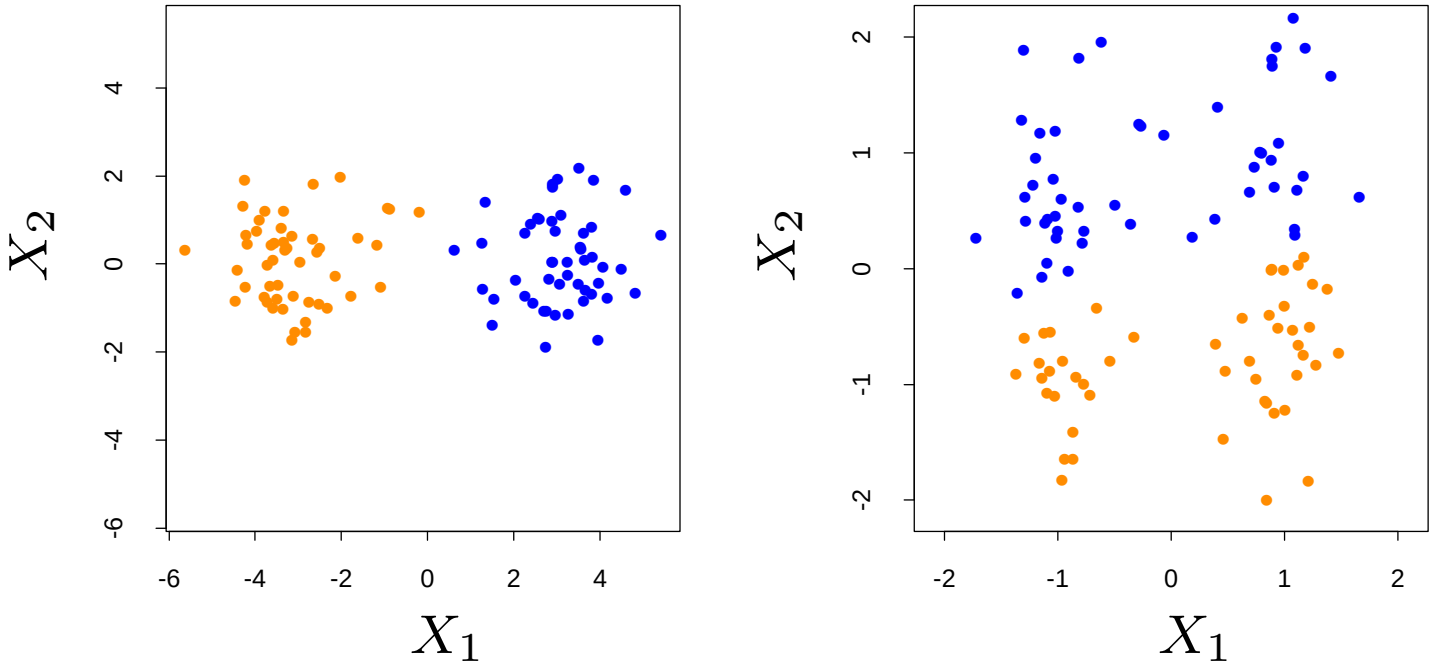
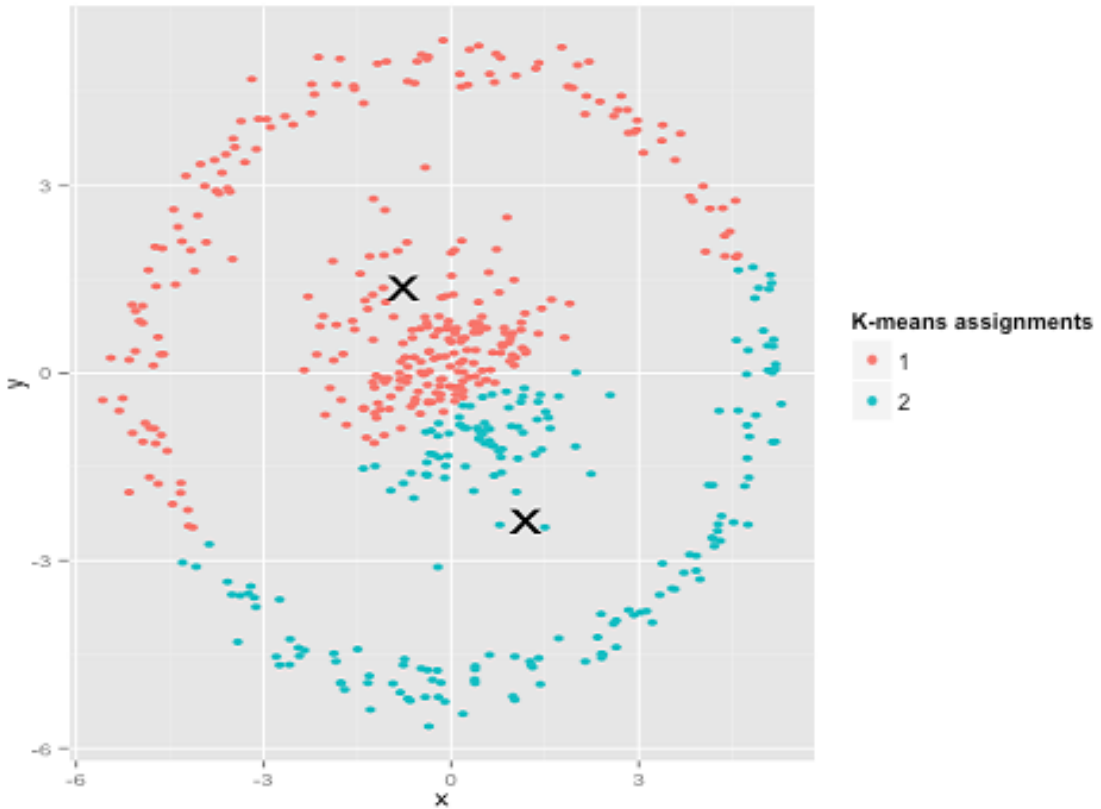


FIGURE 14.5. *Simulated data: on the left, K-means clustering (with K=2) has been applied to the raw data. The two colors indicate the cluster memberships. On the right, the features were first standardized before clustering. This is equivalent to using feature weights* $1/[2 \cdot \mathrm{var}(X_j)]$. *The standardization has obscured the two well-separated groups. Note that each plot uses the same units in the horizontal and vertical axes.*

Casos Problemáticos



Optimización

- K−medias también soluciona el siguiente problema de optimización

$$\arg\min_{\mathbf{S}} \sum_{i=1}^{k} \sum_{\mathbf{x}_j \in S_i} \|x_j - \mu_i\|^2$$

  donde μ_i es la media de puntos en S_i.

- También puede ser:

$$\arg\min_{\mathbf{S}} \sum_{i=1}^{k} \frac{1}{|S_i|} \sum_{x_j, x_{j'} \in S_i} \left\|\mathbf{x}_j - \mathbf{x}_{j'}\right\|^2$$

Problemas con K-Medias

- Es un problema NP.
- Hay varios óptimos locales (pero no infinitos).

Ejemplo: Tumores y expresiones de genes

TABLE 14.2. *Human tumor data: number of cancer cases of each type, in each of the three clusters from K-means clustering.*

Cluster	Breast	CNS	Colon	K562	Leukemia	MCF7
1	3	5	0	0	0	0
2	2	0	0	2	6	2
3	2	0	7	0	0	0
Cluster	Melanoma	NSCLC	Ovarian	Prostate	Renal	Unknown
1	1	7	6	2	9	1
2	7	2	0	0	0	0
3	0	0	0	0	0	0

The data are a 6830×64 matrix of real numbers, each representing an expression measurement for a gene (row) and sample (column). Here we cluster the samples, each of which is a vector of length 6830, corresponding to expression values for the 6830 genes. Each sample has a label such as `breast` (for breast cancer), `melanoma`, and so on; we don't use these labels in the clustering, but will examine *posthoc* which labels fall into which clusters.

Ejemplo: Tumores y expresiones de genes

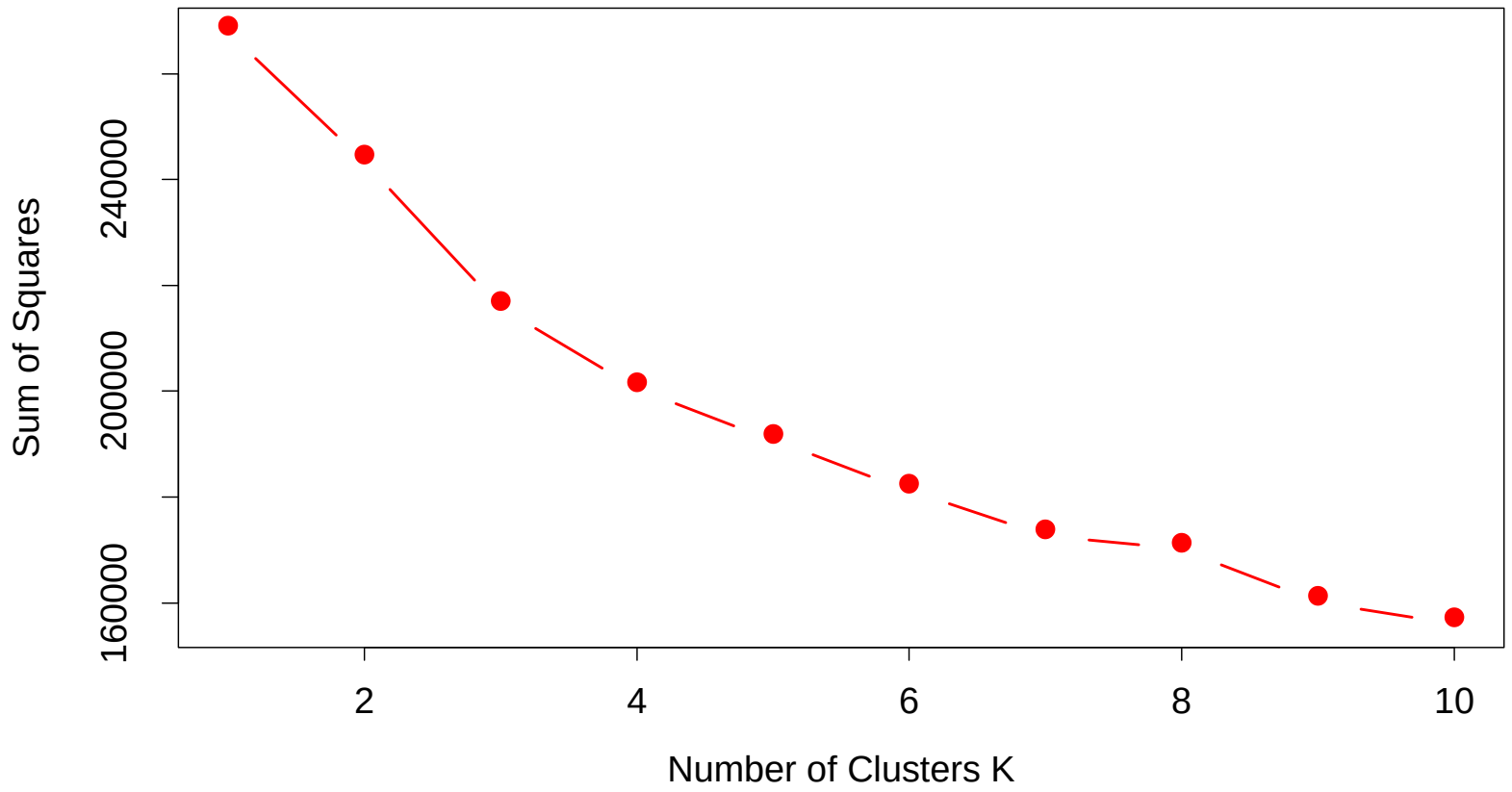


FIGURE 14.8. *Total within-cluster sum of squares for K-means clustering applied to the human tumor microarray data.*

K−Medoids

- El algoritmo K-Medias depende de:
  1. Una noción de disimilitud.
  2. Calcula los centros de los grupos en cada iteración como la media de los puntos más cercanos.
- Lo primero puede sustituirse por cualquier medidad de disimilitud $D(x_i, x_{i'})$.
- Lo segundo puede generalizarse a expensas de tiempo computacional.

K−Medoids

- El algoritmo K-Medias depende de:
  1. Una noción de disimilitud.
  2. Calcula los centros de los grupos en cada iteración como la media de los puntos más cercanos.
- Lo primero puede sustituirse por cualquier medidad de disimilitud $D(x_i, x_{i'})$.
- Lo segundo puede generalizarse a expensas de tiempo computacional.

K−Medoids

- El algoritmo K-Medias depende de:
  1. Una noción de disimilitud.
  2. Calcula los centros de los grupos en cada iteración como la media de los puntos más cercanos.
- Lo primero puede sustituirse por cualquier medidad de disimilitud $D(x_i, x_{i'})$.
- Lo segundo puede generalizarse a expensas de tiempo computacional.

K-Medoids

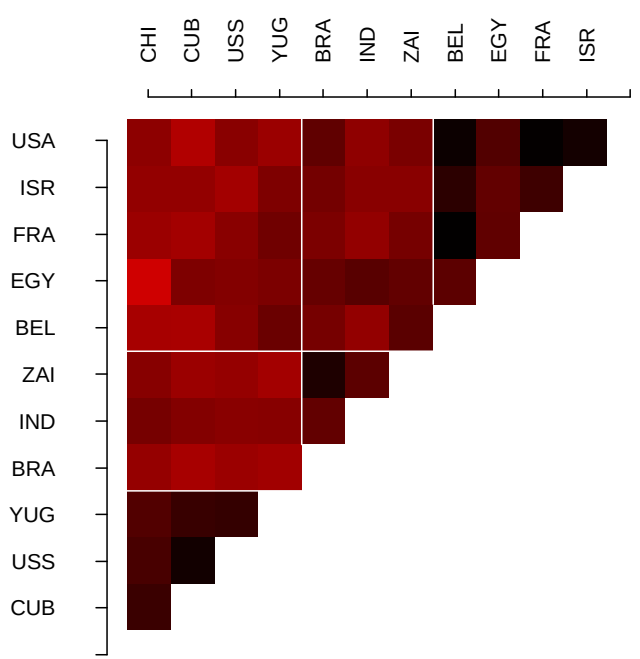
- Dada una asignación de clusters C:
  1. Para cada $C_i \in C$ encontrar el punto x_i^* que minimiza la suma de la disimilitud a los otros puntos del cluster C_i. Defina los centros $m_i = x_i^*, i = 1, ..., K$.
  2. Reasignar los puntos a los centros más cercanos de acuerdo a disimilitud.
  3. Iterar los pasos anteriores hasta que no haya cambios en los clusters.

K-Medoids: Ejemplo

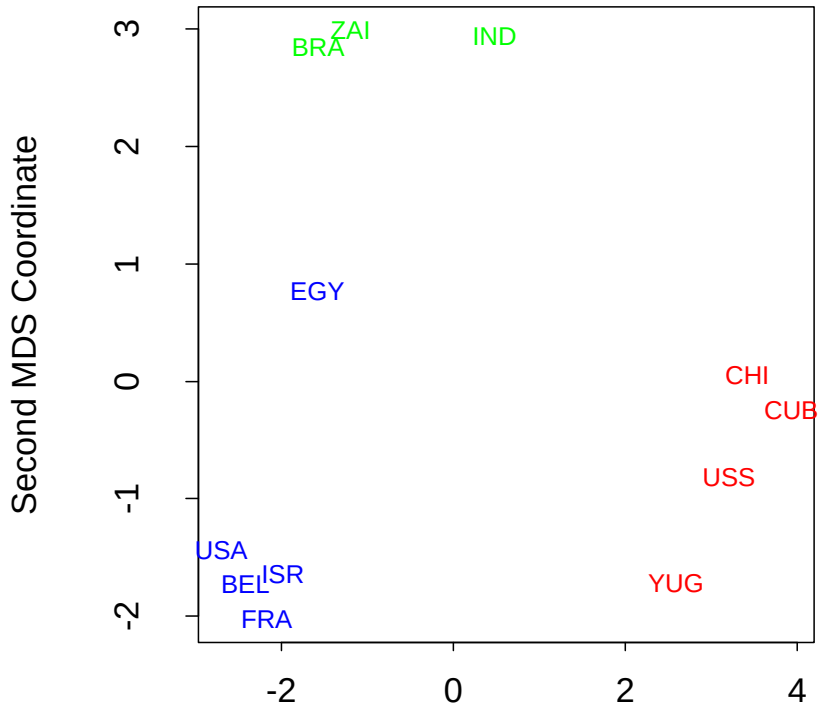
TABLE 14.3. *Data from a political science survey: values are average pairwise dissimilarities of countries from a questionnaire given to political science students.*

	BEL	BRA	CHI	CUB	EGY	FRA	IND	ISR	USA	USS	YUG
BRA	5.58										
CHI	7.00	6.50									
CUB	7.08	7.00	3.83								
EGY	4.83	5.08	8.17	5.83							
FRA	2.17	5.75	6.67	6.92	4.92						
IND	6.42	5.00	5.58	6.00	4.67	6.42					
ISR	3.42	5.50	6.42	6.42	5.00	3.92	6.17				
USA	2.50	4.92	6.25	7.33	4.50	2.25	6.33	2.75			
USS	6.08	6.67	4.25	2.67	6.00	6.17	6.17	6.92	6.17		
YUG	5.25	6.83	4.50	3.75	5.75	5.42	6.08	5.83	6.67	3.67	
ZAI	4.75	3.00	6.08	6.67	5.00	5.58	4.83	6.17	5.67	6.50	6.92

K-Medoids: Ejemplo



Reordered Dissimilarity Matrix



Selección del número de clusters

Dos enfoques:

- Medidas intrísecas: usan información de la muestra.
- Medias extrínsecas: comparan con información externa (agrupaciones conocidas o razonables o agrupaciones supervisadas).

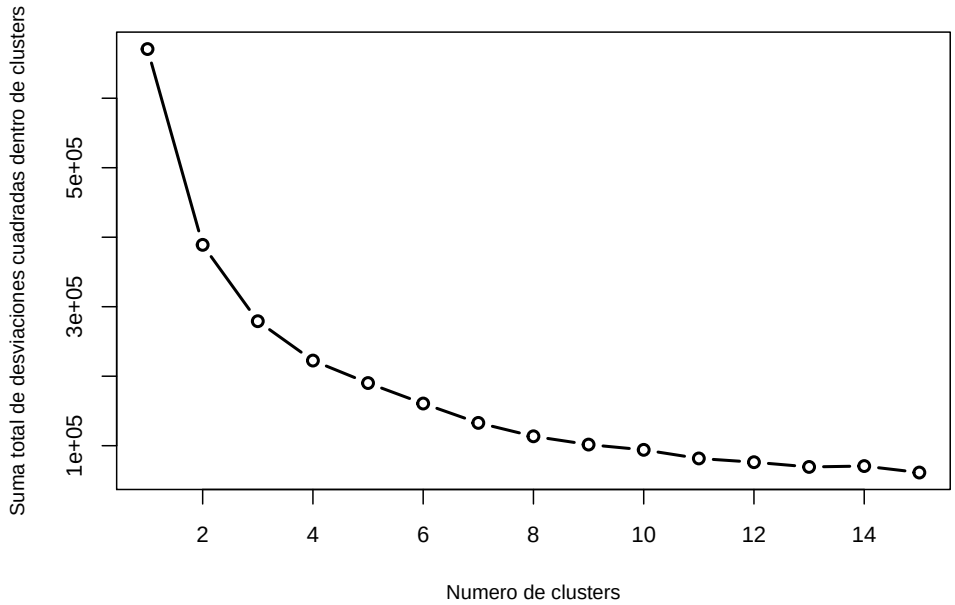
Selección del número de clusters

Medidas intrínsecas:

- El gráfico del "codo".
- Índice de Silueta.
- El índice de Calinski-Harabasz (CH).

Gráfico del "codo"

- Se basa en la suma de todas las desviaciones dentro de los clusters con el número de clusters (i.e., la función objetivo de K-medias).

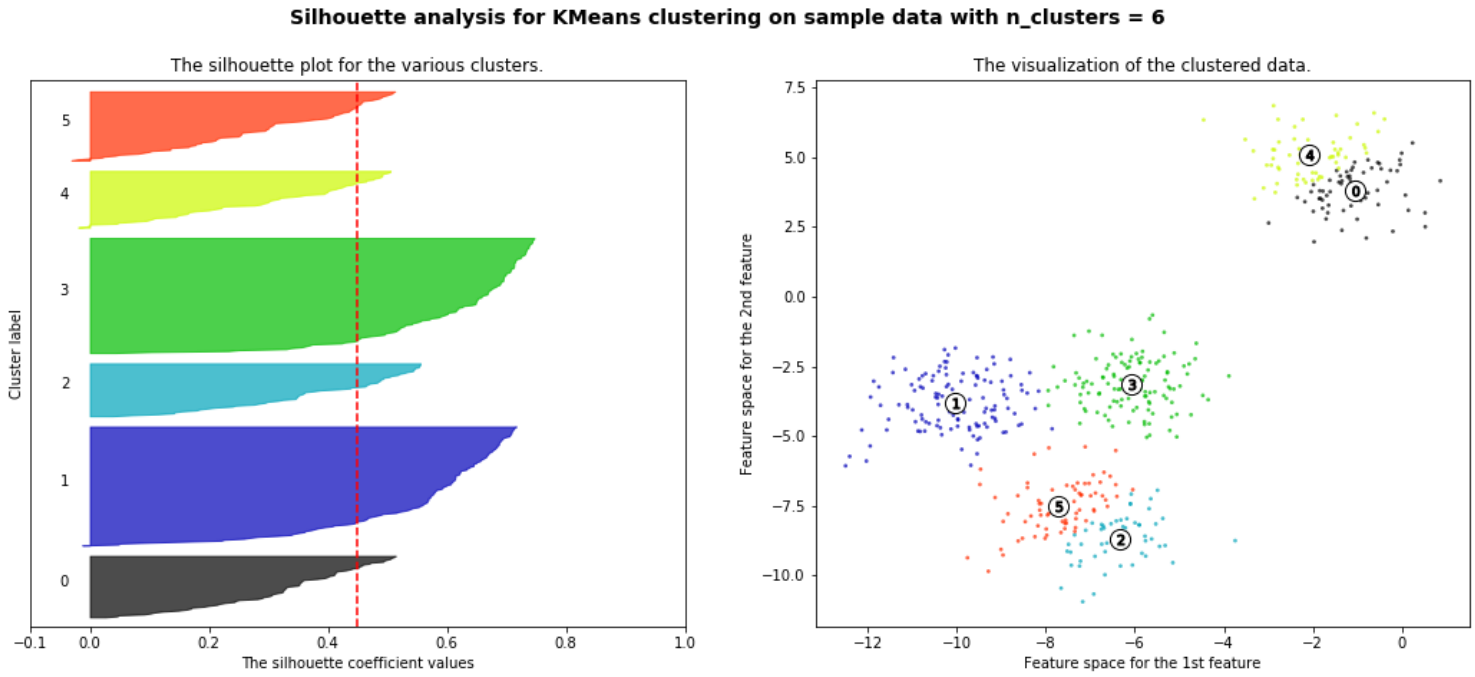


Índice de la Silueta

- Este índice compara qué tan cercano está cada ejemplo del cluster más cercano entre los clusters a los que no pertenece el ejemplo (i.e., este cluster más cercano se conoce como el cluster vecino).
- Para cada observción i calcular a_i, el promedio de la disimilitud con respecto a los demás ejemplos del grupo al que pertenece.
- Para cada uno de los otros grupos calcular la disimilitud de i con cada grupo y sea b_i la menor de estas disimilitudes (i.e., distancia al cluster vecino).
- La silueta S_i de la observación i se define como:

$$S_i = \frac{b_i - a_i}{max\{a_i, b_i\}} \tag{5}$$

Ejemplo Silueta



- La línea roja corresponde al máximo de las siluetas promedio de cada cluster.

Contenido

Componentes Principales

- A veces tenemos muchas variables correlacionadas.
- La información que contienen es **redundante**.
- ¿Podemos construir pocas variables que al representar los datos en esas variables se aproxime bien a los datos observados?
- ¿Pueden estas variables explicar la varianza de los datos observados?

Aproximación Afín

- Sean $x_i \in R^p$ y $q \leq p$.
- La función $f(\lambda) = \mu + V_q \lambda$ con $\lambda \in R^q$ es una transformación afín.
- El problema que queremos resolver es:

$$\min_{\mu, \lambda_i, V_q} \sum_{i=1}^{n} || \, x_i - f(\lambda_i) \, || \tag{6}$$

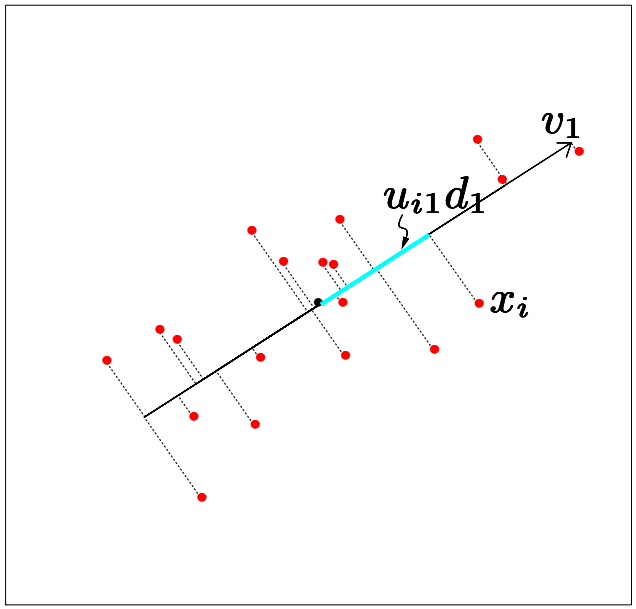


FIGURE 14.20. *The first linear principal component of a set of data. The line minimizes the total squared distance from each point to its orthogonal projection onto the line.*

Aproximación Afín

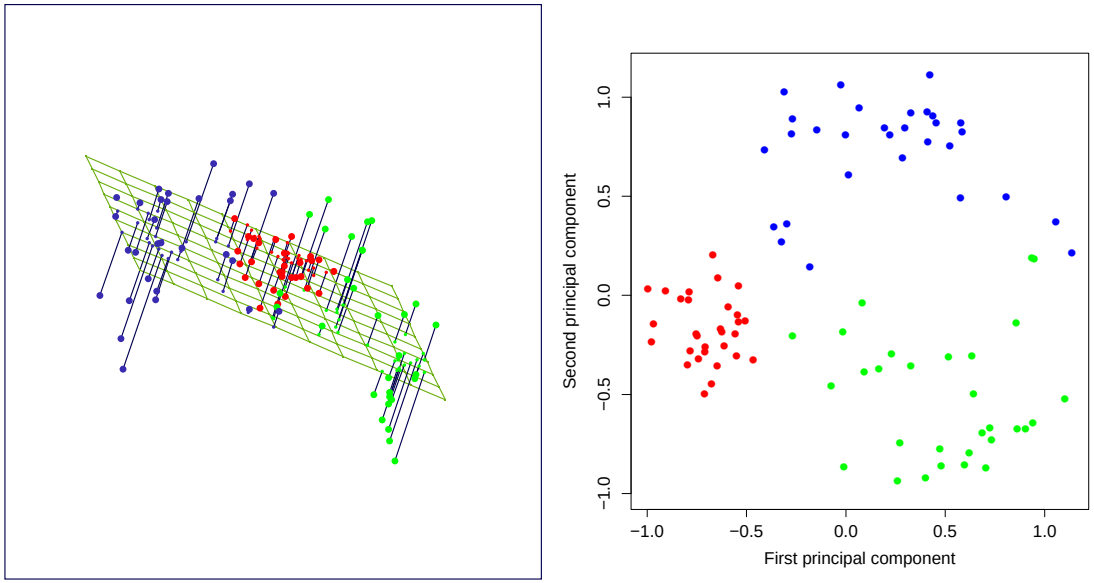


FIGURE 14.21. *The best rank-two linear approximation to the half-sphere data. The right panel shows the projected points with coordinates given by* $\mathbf{U}_2\mathbf{D}_2$*, the first two principal components of the data.*

Aproximación Afín

- Supongamos sin perdida de genralidad que la media de los datos es cero.
- La solución a este problema es equivalente a: $\mu = 0, \lambda_i = V_q^T x_i$ donde:

$$\min_{V_q} \sum_{i=1}^{n} || \, x_i - V_q V_q^T x_i \, || \tag{7}$$

- La matriz $H_q = V_q V_q^T$ es una matriz que proyecta de forma ortogonal cada observación $(H_q x_i)$ en el subespacio generado por las columnas de V_q.

Descomposición SVM

- Para resolver este problema usamos el teorema: *Singular Value Decomposition*.
- Sea $X, n \times p$ entonces:

$$X = UDV^T \tag{8}$$

donde U, V son ortogonales y D es diagonal $D = [d_1, ..., d_p]$, $d_1 \geq ...d_p \geq 0$.

- Las columnas de V se llaman las componentes principales de X.
- Para cada q, la solución del problema de optimización V_q son las primeras q componentes principales (columnas de V).
- Obsérvese que la matriz de varianza covarianza, $\Sigma = X^T X = VD^2 V^T$.

Componentes Principales

- Las componentes principales son variables nuevas.
- Son combinaciones lineales de las variables originales.
- La primera componente es la combinación lineal que maximiza la varianza.
- La segunda componente maximiza la varianza condicional a ser ortogonal a la primera.
- Así sucesivamente podemos construir p componentes.
- Comenzamos discutiendo primero el caso en el que queremos reducir la dimensionalidad de un vector aleatorio X.
- Sea Σ la matriz de varianza covarianza de X.

Componentes Principales: Las CP maximizan varianza

- La idea es encontrar una combinación lineal, $z = Xv$ que disperse al máximo la muestra en el espacio de características.
- Para forzar que el resultado se concentre en la dirección y no en la magnitud de la transformación, se impone que el vector de transformación tenga norma uno, $v^T v = 1$.
- El problema que se quiere resolver es:

$$\max_{\{v : v^T v = 1\}} v^T \Sigma v$$

Se puede demostrar que el vector v^1 que resuelve este problema es el vector propio de X con el mayor valor propio asociado.

Componentes Principales: Las CP maximizan varianza

- Ahora la idea es econtrar una segunda combinación lineal, tal que el vector que la define sea ortogonal al anterior (v^1), tenga norma uno y se maximice la dispersión.
- El problema que se quiere resolver ahora es:

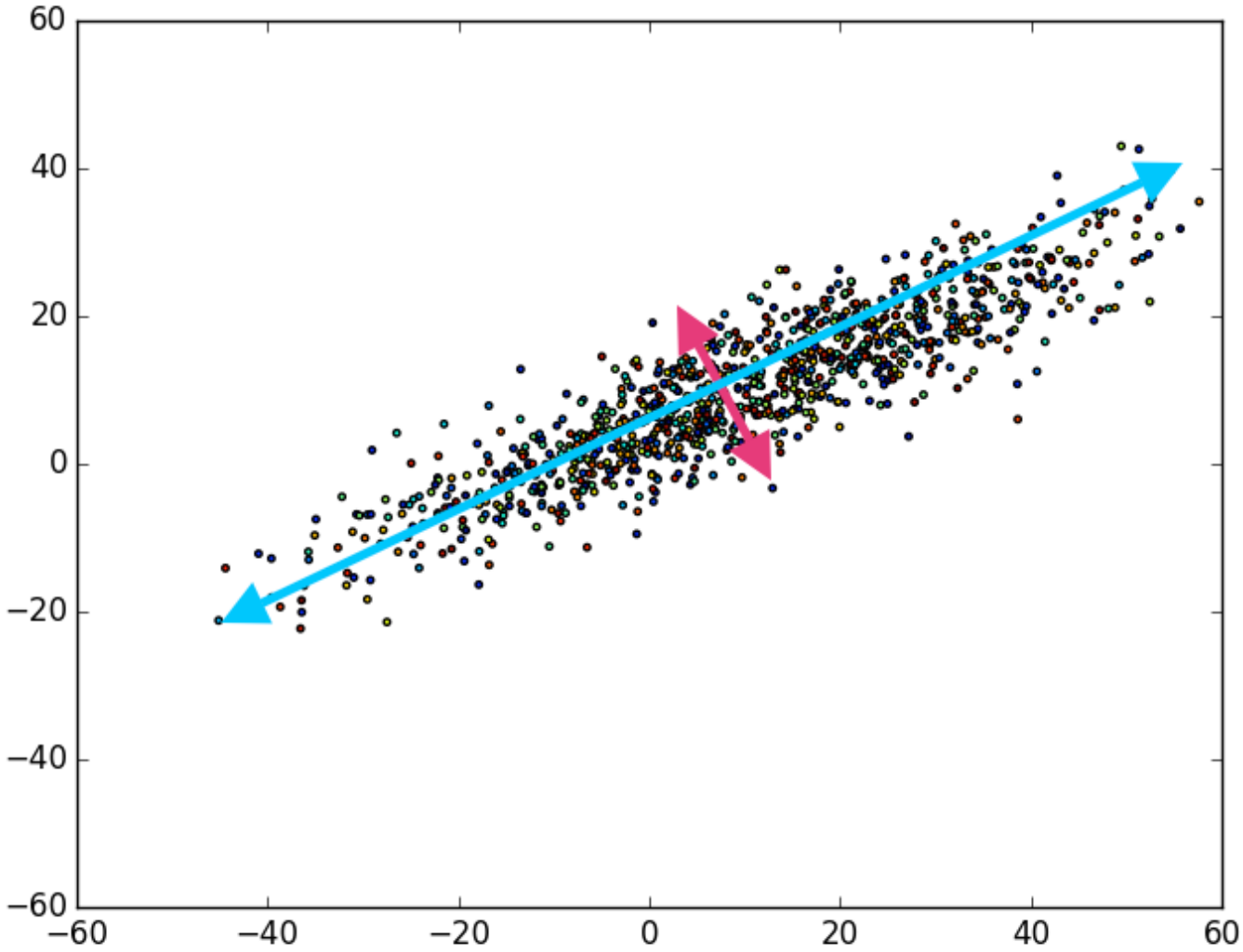
$$\max_{\{v: v^T v = 1, v^T v^1 = 0\}} v^T \Sigma v$$

El vector v^2 que resuelve este problema es un vector propio de X con el segundo mayor valor propio asociado.

- Para medir qué tanto las componentes explican la varianza del vector aleatorio X:

$$\sum_{j=1}^{p} var(X_j) = traza(\Sigma) = \sum_{j=1}^{p} \lambda_j = \sum_{j=1}^{p} var(Y_j)$$

Componentes Principales



PCA en la práctica

1. Aplicar PCA.
2. Escoger primeras componentes que expliquen x % de la varianza.
3. Hacer un modelo de clasificación o regresión basado en esas componentes.
4. Esto puede transformar un modelo de 200 variables en uno igual de bueno de 5 variables.

Aplicaciones: Números escritos a mano

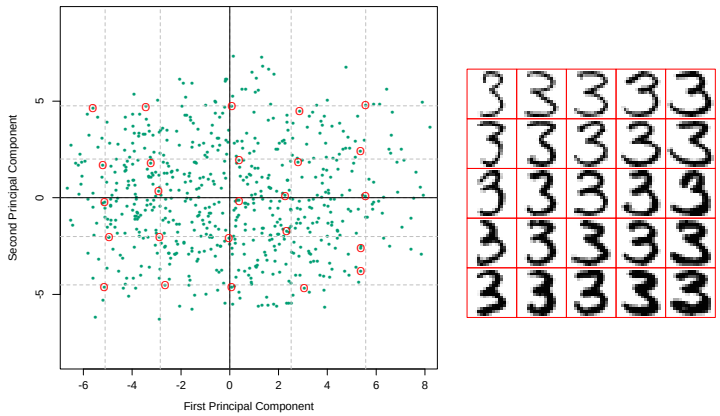


FIGURE 14.23. *(Left panel:) the first two principal components of the hand-written threes. The circled points are the closest projected images to the vertices of a grid, defined by the marginal quantiles of the principal components. (Right panel:) The images corresponding to the circled points. These show the nature of the first two principal components.*

Contenido

Estimación de densidades

- Supongamos que $x_1, ..., x_N$ es una muestra de datos tomada con una distribución con densidad $f_X(x)$.
- Un primer estimador local es:

$$f_X(x) = \frac{\{i : x_i \in N(x)\}}{N\lambda} \tag{9}$$

donde $N(x)$ es una vecindad de tamaño λ.

- Una versión suavizada es:

$$f_X(x) = \frac{1}{N}\sum_{i=1}^{N} K_\lambda(x, x_i) \tag{10}$$

donde por ejemplo $K_\lambda(x, x_i) = \psi_\lambda(x - x_i)$ y ψ_λ es la densidad Gaussiana $(0, \lambda^2)$.

Estimación de densidades

- Supongamos que $x_1, ..., x_N$ es una muestra de datos tomada con una distribución con densidad $f_X(x)$.
- Un primer estimador local es:

$$f_X(x) = \frac{\{i : x_i \in N(x)\}}{N\lambda} \tag{9}$$

donde $N(x)$ es una vecindad de tamaño λ.

- Una versión suavizada es:

$$f_X(x) = \frac{1}{N}\sum_{i=1}^{N} K_\lambda(x, x_i) \tag{10}$$

donde por ejemplo $K_\lambda(x, x_i) = \psi_\lambda(x - x_i)$ y ψ_λ es la densidad Gaussiana $(0, \lambda^2)$.

Estimación de densidades

- Supongamos que $x_1, ..., x_N$ es una muestra de datos tomada con una distribución con densidad $f_X(x)$.
- Un primer estimador local es:

$$f_X(x) = \frac{\{i : x_i \in N(x)\}}{N\lambda} \tag{9}$$

donde $N(x)$ es una vecindad de tamaño λ.

- Una versión suavizada es:

$$f_X(x) = \frac{1}{N} \sum_{i=1}^{N} K_\lambda(x, x_i) \tag{10}$$

donde por ejemplo $K_\lambda(x, x_i) = \psi_\lambda(x - x_i)$ y ψ_λ es la densidad Gaussiana $(0, \lambda^2)$.

Estimación de densidades

- En p dimensiones:

$$f_X(x) = \frac{1}{N(2\lambda^2\pi)^{\frac{p}{2}}} \sum_{i=1}^{N} \exp(-\frac{1}{2}(\frac{|| x - x_i ||}{\lambda})^2) \qquad (11)$$

Estimación de densidades

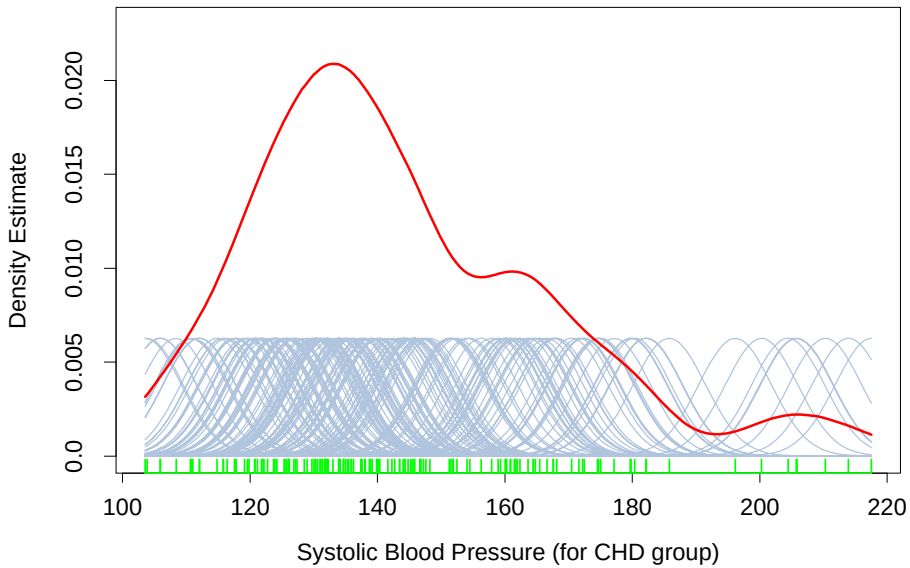


FIGURE 6.13. *A kernel density estimate for systolic blood pressure (for the CHD group). The density estimate at each point is the average contribution from each of the kernels at that point. We have scaled the kernels down by a factor of 10 to make the graph readable.*

Clasificación

- Usando las estimaciones de Kernel de las densidades por clase f_j, el clasificador óptimo se puede escribir (usando la regla de Bayes):

$$P(G = j \mid X = x) = \frac{\pi_j f_j(x)}{\sum_{k=1}^{J} \pi_k f_k(x)} \tag{12}$$

donde π_j son las frecuencias relativas de cada clase.

Kernels para clasificación

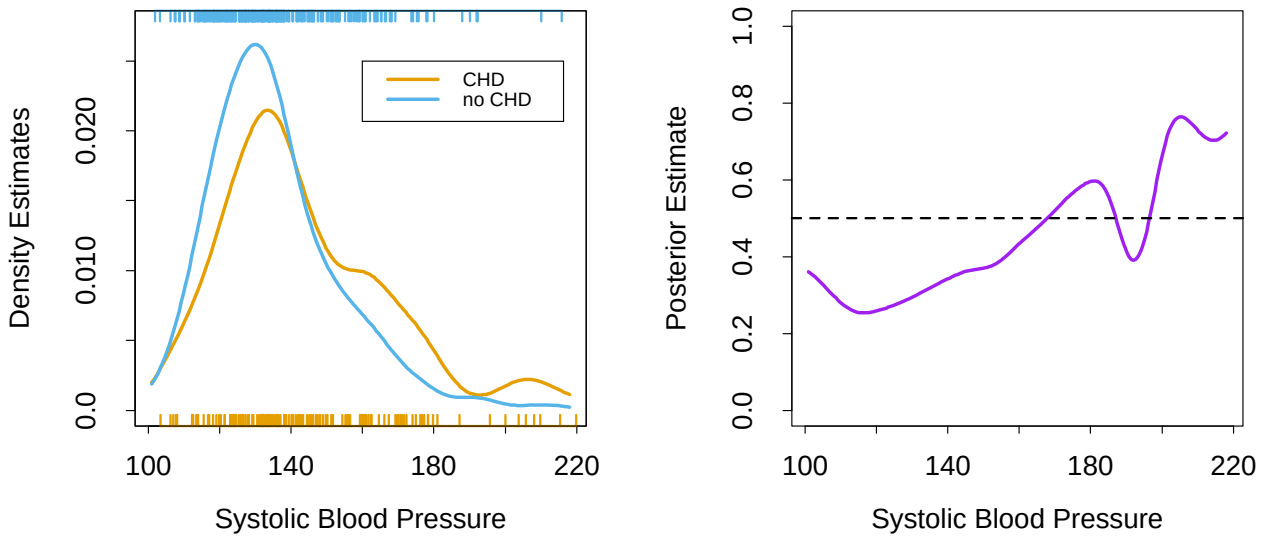


FIGURE 6.14. *The left panel shows the two separate density estimates for systolic blood pressure in the CHD versus no-CHD groups, using a Gaussian kernel density estimate in each. The right panel shows the estimated posterior probabilities for CHD, using (6.25).*

Kernels para clasificación: Naive Bayes

- Cuando el espacio de características es muy grande (p grande) la estimación por kernels tiene mucha varianza.
- El clasificador de Bayer asume independencia de las variables y en ese caso el clasificador de Bayes se reduce a:

$$P(G = j \mid X = x_0) = \frac{\pi_j f_j(x_0)}{\sum_{k=1}^{J} \pi_k f_k(x_0)} \tag{13}$$

donde:

$$f_j(x) = \Pi_{l=1}^{p} f_{j,l}(x_l) \tag{14}$$

- Cada $f_{j,l}$ es un kernel unidimensional. La complejidad del problema se reduce enormemente.
- Si alguna variable es discreta, permite combinar fácilmente variables continuas y categóricas.

Kernels para clasificación: Naive Bayes

- Cuando el espacio de características es muy grande (p grande) la estimación por kernels tiene mucha varianza.
- El clasificador de Bayer asume independencia de las variables y en ese caso el clasificador de Bayes se reduce a:

$$P(G = j \mid X = x_0) = \frac{\pi_j f_j(x_0)}{\sum_{k=1}^{J} \pi_k f_k(x_0)} \tag{13}$$

donde:

$$f_j(x) = \Pi_{l=1}^{p} f_{j,l}(x_l) \tag{14}$$

- Cada $f_{j,l}$ es un kernel unidimensional. La complejidad del problema se reduce enormemente.
- Si alguna variable es discreta, permite combinar fácilmente variables continuas y categóricas.

Kernels para clasificación: Naive Bayes

- Cuando el espacio de características es muy grande (p grande) la estimación por kernels tiene mucha varianza.
- El clasificador de Bayer asume independencia de las variables y en ese caso el clasificador de Bayes se reduce a:

$$P(G = j \mid X = x_0) = \frac{\pi_j f_j(x_0)}{\sum_{k=1}^{J} \pi_k f_k(x_0)} \tag{13}$$

donde:

$$f_j(x) = \Pi_{l=1}^{p} f_{j,l}(x_l) \tag{14}$$

- Cada $f_{j,l}$ es un kernel unidimensional. La complejidad del problema se reduce enormemente.
- Si alguna variable es discreta, permite combinar fácilmente variables continuas y categóricas.

Kernels para clasificación: Naive Bayes

- Cuando el espacio de características es muy grande (p grande) la estimación por kernels tiene mucha varianza.
- El clasificador de Bayer asume independencia de las variables y en ese caso el clasificador de Bayes se reduce a:

$$P(G = j \mid X = x_0) = \frac{\pi_j f_j(x_0)}{\sum_{k=1}^{J} \pi_k f_k(x_0)} \tag{13}$$

donde:

$$f_j(x) = \Pi_{l=1}^{p} f_{j,l}(x_l) \tag{14}$$

- Cada $f_{j,l}$ es un kernel unidimensional. La complejidad del problema se reduce enormemente.
- Si alguna variable es discreta, permite combinar fácilmente variables continuas y categóricas.