Clustering y reglas de asociación

Francisco Charte

En esta sesión nos ocuparemos de:

- Extracción de relaciones entre los datos
  - Minería de reglas de asociación
  - Visualización de las reglas
  - Ejercicios con el paquete `arules`
- Segmentación de los datos
  - Agrupamiento básico y jerárquico
  - Visualización del agrupamiento
  - Ejercicio con el paquete

Extracción de relaciones entre los datos

Minería de reglas de asociación

Extracción de relaciones entre los datos

- Paquete `arules`
  - Clase `transactions` desde `data.frame` o `matrix`
  - Función `apriori()` de minería de reglas de asociación
  - Funciones de utilidad para operar sobre las reglas: `inspect()`, operador `[]`
  - Lectura de archivos de transacciones (`read.transactions`) y exportación de las reglas (`write.csv/write.PMML`)

- Análisis de las reglas obtenidas
  - Obtención de métricas de calidad: `quality()`, `interestMeasure()`
  - Función `plot()` con gráficas específicas de exploración

Minería de reglas de asociación

- Usando parámetros por defecto

```
reglas <- apriori(Adult)
```

- Ajustando soporte y confianza mínimos

```
reglas <- apriori(Adult, parameter =
    list(supp = 0.01, confidence = 0.5))
```

- Seleccionando la forma de las reglas

```
reglas <- apriori(Adult, appearance =
    list(rhs = c("income=large"),
         default = "lhs"))
```

Visualizar reglas y transacciones

- Frecuencia de los *ítems* analizados

```
itemFrequencyPlot(Adult, support = 0.1)
```

- Soporte, confianza y *lift* de las reglas

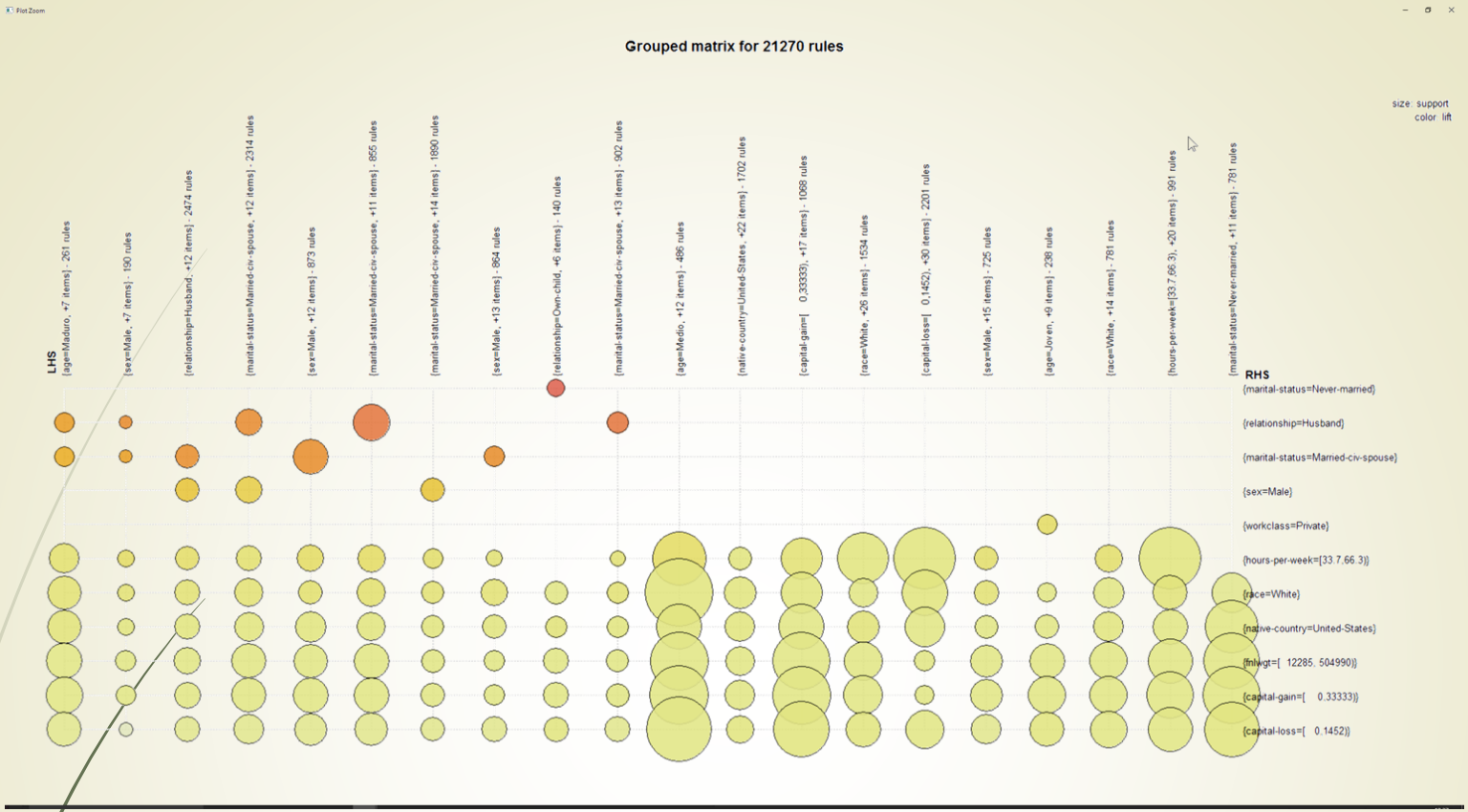
```
plot(reglas)
```

- Antecente vs consecuente de las reglas

```
plot(reglas, method = "grouped")
```

- Interacción de los *ítems* en las reglas

```
plot(reglas, method = "graph")
plot(reglas, method = "paracoord")
```



Ejercicios prácticos

Minería de reglas de asociación

Extracción de relaciones entre los datos

Segmentación de datos - Clustering

Segmentación de datos - Clustering

- CRAN Task View: Cluster Analysis and Finite Mixture Models

  `https://cran.r-project.org/web/views/Cluster.html`

  - Agrupamiento por particionamiento, jerárquico y basado en estimación de modelos
  - Varias decenas de paquetes
  - Implementaciones de múltiples algoritmos conocidos

- Paquetes que nos interesan
  - `stats`: Funciones básicas `kmeans()` y `hclust()`
  - `fpc`: Comparación de soluciones y estimación nº grupos
  - `cluster`: Funciones de visualización de los clusters
  - `mclust`: Agrupamiento basado en modelos

Clustering por particionamiento

- Paquete `stats`
  - Función `kmeans()` de agrupamiento
  - Precisa el número de grupos o sus centros
  - Minimización de la suma de cuadrados de la distancia de cada punto al centro

- Información devuelta como resultado
  - Número de cluster apara cada muestra de datos - `cluster`
  - Centro de cada cluster - `centers`
  - Suma de cuadrados para cada cluster - `withinss`

Visualización de los grupos

- Partiendo de un particionamiento simple:

```
clusters <- kmeans(iris[, -5], 3)
```

- Con paquete `ggplot`

```
ggplot(iris,
  aes(Petal.Length, Petal.Width)) +
  geom_point(aes(color = clusters$cluster))
```

- Con paquete `cluster`

```
clusplot(iris[, -5], clusters$cluster)
```

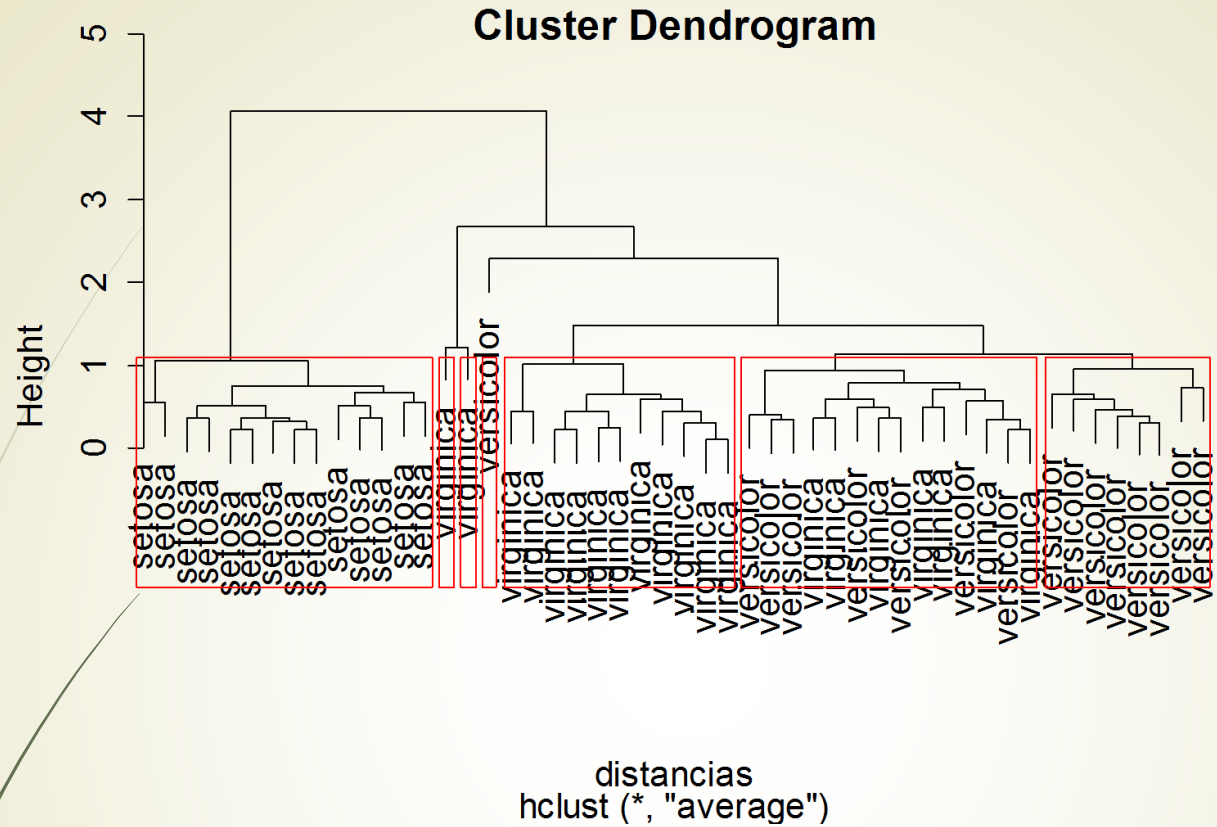
- Con paquete `fpc`

```
plotcluster(iris[, -5], clusters$cluster)
```

Clustering avanzado

- Paquete `stats`
  - Función `hclust()` de agrupamiento jerárquico
  - Precisa distancias entre muestras como entrada
  - Función `plot()` específica para visualiza el dendograma

- Paquete `mclust`
  - Funciones para agrupamiento basado en modelos
  - `clPairs` - Visualización de pares de variables
  - `Mclust` - Basado en información bayesiana
  - `MclustDA` - Basado en análisis discriminante



Ejercicios prácticos

Segmentación de datos - Clustering