Proyecto Análisis Multivariado

Aaron Mauricio Gómez Jimenéz y Francisca Vilca Sánchez

2023-05-29

Introducción

En el siguiente reporte, se hará el análisis de la base de datos `salud_fetos.csv` que contiene la información sobre 2126 registros de fetos con las características extraídas de exámenes de cardiotocograma, que luego fueron clasificados por obstetras expertos en el tema en 3 clases: Normal, Sospechar y Patológico. El objetivo principal del proyecto es generar un modelo que sea capaz de predecir el grupo en el que se debería agrupar un nuevo feto dados las características que este presenta.

Para lograr esto, primero revisaremos los datos para comprender como funcionan, es decir, un análisis exploratorio, luego revisaremos que variables presentan mayor correlación para que con ellas se creé el modelo a utilizar, una vez seleccionada las variables, mediante algún algoritmo de análisis de conglomerados se crearán los grupos y luego se revisará que tan bien funcionan en el modelo.

Análisis Exploratorio

Lo primero ha hacer es cargar la base de datos y se harña un resumen para ver que variables y de que tipo son:

```
##        LB              AC                  FM                 UC          
##  Min.   :106.0   Min.   :0.000000   Min.   :0.000000   Min.   :0.000000  
##  1st Qu.:126.0   1st Qu.:0.000000   1st Qu.:0.000000   1st Qu.:0.002000  
##  Median :133.0   Median :0.002000   Median :0.000000   Median :0.004000  
##  Mean   :133.3   Mean   :0.003178   Mean   :0.009481   Mean   :0.004366  
##  3rd Qu.:140.0   3rd Qu.:0.006000   3rd Qu.:0.003000   3rd Qu.:0.007000  
##  Max.   :160.0   Max.   :0.019000   Max.   :0.481000   Max.   :0.015000  
##        DL                 DS                  DP                 ASTV      
##  Min.   :0.000000   Min.   :0.000e+00   Min.   :0.0000000   Min.   :12.00  
##  1st Qu.:0.000000   1st Qu.:0.000e+00   1st Qu.:0.0000000   1st Qu.:32.00  
##  Median :0.000000   Median :0.000e+00   Median :0.0000000   Median :49.00  
##  Mean   :0.001889   Mean   :3.293e-06   Mean   :0.0001585   Mean   :46.99  
##  3rd Qu.:0.003000   3rd Qu.:0.000e+00   3rd Qu.:0.0000000   3rd Qu.:61.00  
##  Max.   :0.015000   Max.   :1.000e-03   Max.   :0.0050000   Max.   :87.00  
##       MSTV             ALTV              MLTV            Width       
##  Min.   :0.200   Min.   : 0.000   Min.   : 0.000   Min.   :  3.00  
##  1st Qu.:0.700   1st Qu.: 0.000   1st Qu.: 4.600   1st Qu.: 37.00  
##  Median :1.200   Median : 0.000   Median : 7.400   Median : 67.50  
##  Mean   :1.333   Mean   : 9.847   Mean   : 8.188   Mean   : 70.45  
##  3rd Qu.:1.700   3rd Qu.:11.000   3rd Qu.:10.800   3rd Qu.:100.00  
##  Max.   :7.000   Max.   :91.000   Max.   :50.700   Max.   :180.00  
##       Min              Max            NMax            Nzeros       
```

```
##  Min.   : 50.00   Min.   :122   Min.   : 0.000   Min.   : 0.0000
##  1st Qu.: 67.00   1st Qu.:152   1st Qu.: 2.000   1st Qu.: 0.0000
##  Median : 93.00   Median :162   Median : 3.000   Median : 0.0000
##  Mean   : 93.58   Mean   :164   Mean   : 4.068   Mean   : 0.3236
##  3rd Qu.:120.00   3rd Qu.:174   3rd Qu.: 6.000   3rd Qu.: 0.0000
##  Max.   :159.00   Max.   :238   Max.   :18.000   Max.   :10.0000
##       Mode            Mean           Median         Variance
##  Min.   : 60.0   Min.   : 73.0   Min.   : 77.0   Min.   :  0.00
##  1st Qu.:129.0   1st Qu.:125.0   1st Qu.:129.0   1st Qu.:  2.00
##  Median :139.0   Median :136.0   Median :139.0   Median :  7.00
##  Mean   :137.5   Mean   :134.6   Mean   :138.1   Mean   : 18.81
##  3rd Qu.:148.0   3rd Qu.:145.0   3rd Qu.:148.0   3rd Qu.: 24.00
##  Max.   :187.0   Max.   :182.0   Max.   :186.0   Max.   :269.00
##     Tendency          Health
##  Min.   :-1.0000   Min.   :1.000
##  1st Qu.: 0.0000   1st Qu.:1.000
##  Median : 0.0000   Median :1.000
##  Mean   : 0.3203   Mean   :1.304
##  3rd Qu.: 1.0000   3rd Qu.:1.000
##  Max.   : 1.0000   Max.   :3.000
```

Como se puede ver, se tienen 22 variables, todas númericas, sin embargo, la variable `Health` aquella que queremos predecir, es una variable categórica,

Realizando el análisis explotario de los datos para ver si se encuentra relación entre las variables, o alguna información estadística que nos ayude a encontrar relaciones, dependencias o grupos que se puedan formar con las variables de la muestra.

Se comenzará haciendo un correlograma de los datos, como se ve en la Figura 1:

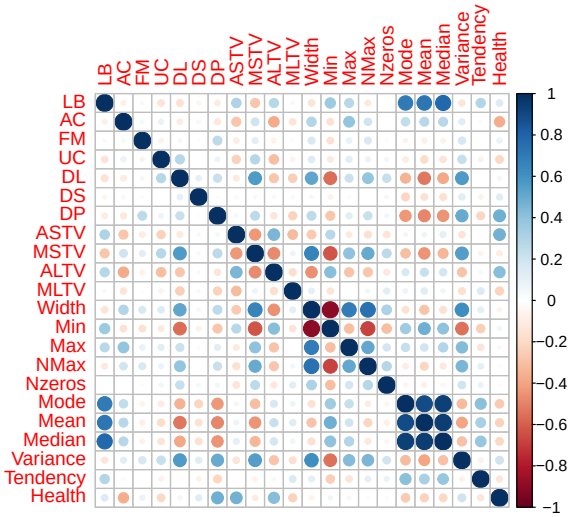

Figure 1: Figura 1: Correlograma de la salud de los fetos

Es posible notar que la variable LB (Frecuencia Cardiaca del Feto) esta relacionada linealmente con la media, moda y mediana. Las otra correlaciones como MSTV (Valor promedio de variación)y el peso tiene una relación lineal positiva baja-moderada. El número de desaceleraciones ligeras (DL) y el minímo de la frecuencia cárdiaca tiene una relación lineal inversa ligera, aunque tambien DL tiene una relación positiva con el peso.

Al analizar los datos se ve que en muchas variables no existe relación, por lo que se filtrarán las variables más significativas en términos de correlación.

```
datos_2<- datos[,c(which(correlaciones[,22]> 0.2 | correlaciones[,22] < -0.2))]
str(datos_2)
```

```
## 'data.frame':    2126 obs. of  11 variables:
##  $ AC      : num  0 0.006 0.003 0.003 0.007 0.001 0.001 0 0 0 ...
##  $ UC      : num  0 0.006 0.008 0.008 0.008 0.01 0.013 0 0.002 0.003 ...
##  $ DP      : num  0 0 0 0 0 0.002 0.003 0 0 0 ...
##  $ ASTV    : num  73 17 16 16 16 26 29 83 84 86 ...
##  $ ALTV    : num  43 0 0 0 0 0 0 6 5 6 ...
##  $ MLTV    : num  2.4 10.4 13.4 23 19.9 0 0 15.6 13.6 10.6 ...
##  $ Mode    : num  120 141 141 137 137 76 71 122 122 122 ...
##  $ Mean    : num  137 136 135 134 136 107 107 122 122 122 ...
##  $ Median  : num  121 140 138 137 138 107 106 123 123 123 ...
##  $ Variance: num  73 12 13 13 11 170 215 3 3 1 ...
##  $ Health  : num  2 1 1 1 1 3 3 3 3 3 ...
```

Al realizar el filtrado, solo 11 variables cumplen con las condiciones, es decir, se redujo a la mitad las variables iniciales.

```
##      AC    UC    DP ASTV ALTV MLTV Mode Mean Median Variance Health
## 1 0.000 0.000 0.000   73   43  2.4  120  137    121       73      2
## 2 0.006 0.006 0.000   17    0 10.4  141  136    140       12      1
## 3 0.003 0.008 0.000   16    0 13.4  141  135    138       13      1
## 4 0.003 0.008 0.000   16    0 23.0  137  134    137       13      1
## 5 0.007 0.008 0.000   16    0 19.9  137  136    138       11      1
## 6 0.001 0.010 0.002   26    0  0.0   76  107    107      170      3
```

Creación de PCA

En el análisis exploratorio de datos se han descartado algunas variables, pero no significa que se tengan las variables que mejor explican la varianza de los datos, para ello haremos un Análisis de Componentes Principales con las base datos_2, tambien se centrarán y reescalán los datos e imprimimos un resumen para ver con cuantos componentes alcanzamos una variabilidad significante (70%).

```
PCA_cent_res <- prcomp(datos_2, center=TRUE, scale=TRUE)
summary(PCA_cent_res)
```

```
## Importance of components:
##                           PC1    PC2    PC3     PC4     PC5     PC6     PC7
## Standard deviation     1.917 1.5600 1.1935 0.95467 0.90934 0.71767 0.66792
## Proportion of Variance 0.334 0.2212 0.1295 0.08285 0.07517 0.04682 0.04056
## Cumulative Proportion  0.334 0.5552 0.6847 0.76754 0.84271 0.88954 0.93009
##                           PC8    PC9    PC10    PC11
## Standard deviation     0.62726 0.4986 0.30601 0.18231
## Proportion of Variance 0.03577 0.0226 0.00851 0.00302
## Cumulative Proportion  0.96586 0.9885 0.99698 1.00000
```

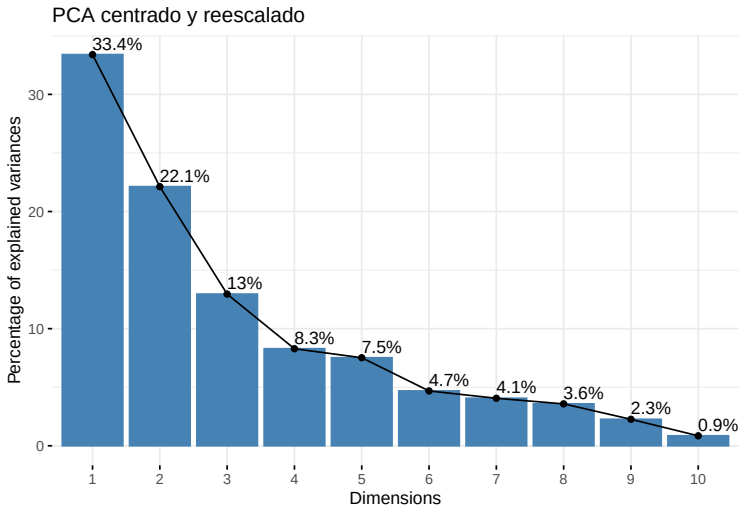

Figure 2: Figura 2: Gráfico de explicación de la varianza

La proporción acumulativa de la varianza dice que con 3 componentes alcanzamos el 68.47% de las varianza, para confirmar se hará tambien un método gráfico para verificar utilizando la regla del codo.

Gráficamente se ve que la curva se dobla en el cuarto componente, ahora dado que dos criterios que se utilizan normalmente dicen que es posible quedarse con 3 y 4 componentes, se usará otro criterio donde elegimos las lambdas mayores a 1.

```
PCA_cent_res$sdev
```

```
##  [1] 1.9166529 1.5599596 1.1935332 0.9546695 0.9093375 0.7176710 0.6679202
##  [8] 0.6272617 0.4986486 0.3060119 0.1823117
```

Es claro notar, que solo hay 3 componentes con lambda mayor a 1, así que se concluyé que los más viable es quedarse con 3 componentes principales, se verá graficamente, en la Figura 3, la dirección de las variables, tomando en cuenta solo dos dimensiones (55%).

Se observa que; La salud del feto y aceleraciones por segundo van en direcciones contrarias, es decir, estas dos variables afectan al segundo componente pero con signo distinto, las variables desaceleraciones prolongadas (DP) y Porcentaje de tiempo con variabilidad a largo plazo anormal (ALTV) no estan correlacionadas, ya que sus vectores son perpendiculares, los vectores que más afectan al primer componente son la media, mediana y moda del histograma de frecuencia fetal.

Al gráficar los datos individualmente se nota que existe un grupo central, y dos grupos secundarios a la izquierda y arriba del grupo central, así que se hará una nueva base con los componentes principales seleccionados, ya que con ellos se puede explicar la variación de los datos sin sobre influir en ellos.

```
datos_3 <- PCA_cent_res$x[,c("PC1","PC2","PC3")]
head(datos_3)
```

```
##             PC1        PC2        PC3
## [1,] -1.5076174  2.7534476  0.4854081
## [2,]  0.4744704 -1.8346589 -0.2037212
## [3,]  0.2315457 -1.9124079 -0.7684627
## [4,]  0.2247755 -2.2367381 -1.8021776
## [5,]  0.4372990 -2.4745977 -1.0282294
## [6,] -7.3719586 -0.5019246  2.8567237
```

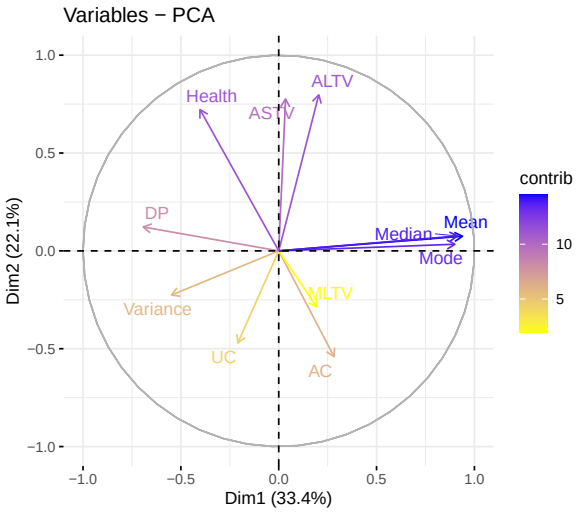

Figure 3: Figura 3: Gráfico de las PC1 y PC2

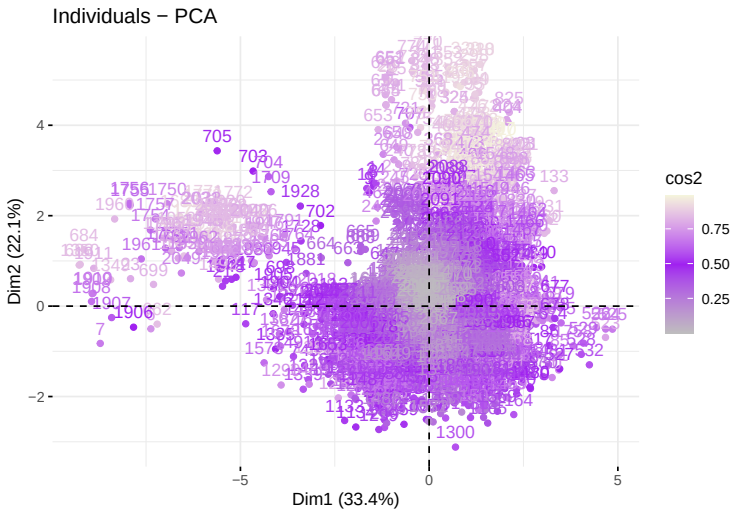

Figure 4: Figura 3: Gráfico de las PC1 y PC2

Creación del modelo

Como se mencionó al inicio de este reporte el objetivo principal, es determinar como será el estado de salud del siguiente feto que se nos presente, para ello, se usará lo que se conoce como análisis de conglomerados para determinar en cual de los tres grupos se debe agrupar:

Método de k-means

Este tipo de metodologías es de las más usadas al crear clusters, como se sabe la cantidad de grupos que hay desde un comienzo, una buena opción sería comenzar con este tipo de algoritmo:

Usando el comando `kmeans()` se puede ver que la matriz de confusión de este método se ve:

```
##        grupo real
## cluster    1    2    3
##       1  202  276   66
##       2   11    7  110
##       3 1442   12    0
```

De aquí es posible observar que para el grupo de sanos, el algoritmo los ubica en el cluster 3 para los sopechosos en el cluster 1 y los patológicos en el 2. Sin embargo, se ve que para diferenciar a los sospechosos de si es sano o patológico el algoritmo no funciona muy bien, esto se puede comprobar con algunas medidas de rendimiento:

```
## Confusion Matrix and Statistics
##
##
##        1    2    3
##   1  202  276   66
##   2   11    7  110
##   3 1442   12    0
##
## Overall Statistics
##
##                Accuracy : 0.0983
##                  95% CI : (0.086, 0.1118)
##     No Information Rate : 0.7785
##     P-Value [Acc > NIR] : 1
##
##                   Kappa : -0.2254
##
##  Mcnemar's Test P-Value : <2e-16
##
## Statistics by Class:
##
##                      Class: 1 Class: 2 Class: 3
## Sensitivity           0.12205 0.023729  0.00000
## Specificity           0.27389 0.933916  0.25436
## Pos Pred Value        0.37132 0.054688  0.00000
## Neg Pred Value        0.08154 0.855856  0.73810
## Prevalence            0.77846 0.138758  0.08278
## Detection Rate        0.09501 0.003293  0.00000
## Detection Prevalence  0.25588 0.060207  0.68391
## Balanced Accuracy     0.19797 0.478822  0.12718
```

De esta salida, es posible notar que la `balanced accuarracy` no funciona muy bien y la `specifity` funciona bien solo para algunos casos, si se revisa visualmente, como en la Figura 4, se tiene:

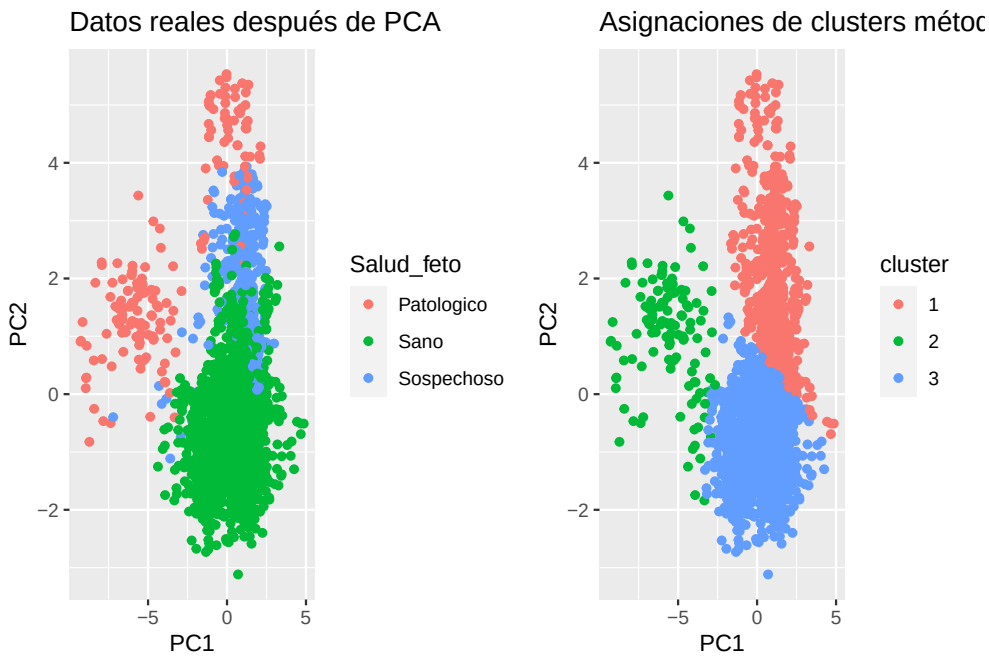

Figure 5: Figura 4: Gráfico de PC1 y PC2, para los datos reales y los del método k-means

De esta formas es mucho más claro ver que el algoritmo k-means le es difícil determinar en que grupo ubicar los casos sospechoso, ya que comete mucho error, debido a ello, quizás usar otro método sea aceptable.

Método qda()

Una posible solución para este problema sea utilizar el análisis de discrimantes, especificamente el qda(), esto debido a que como se muestran en los datos originales, estos forman algunas curvas, dando idea de que un ajuste cuadrático, funcionará mucho mejor, armando el algoritmo se tiene que su matriz de confusión es:

```
##         grupo real
## cluster    1    2    3
##       1 1589   46    2
##       2   64  238   21
##       3    2   11  153
```

A diferencia del método k-means, aquí se asigna a cada grupo el cluster correspondiente. Además, parece que el algoritmo logra diferenciar bastante bien a los sospechosos de las otras clases, esto se puede comprobar con algunas medidas de rendimiento:

```
## Confusion Matrix and Statistics
##
##
## pred_qda    1    2    3
##        1 1589   46    2
##        2   64  238   21
```

```
##        3    2   11  153
##
## Overall Statistics
##
##                Accuracy : 0.9313
##                  95% CI : (0.9197, 0.9417)
##     No Information Rate : 0.7785
##     P-Value [Acc > NIR] : <2e-16
##
##                   Kappa : 0.8159
##
##  Mcnemar's Test P-Value : 0.1082
##
## Statistics by Class:
##
##                      Class: 1 Class: 2 Class: 3
## Sensitivity            0.9601   0.8068  0.86932
## Specificity            0.8981   0.9536  0.99333
## Pos Pred Value         0.9707   0.7368  0.92169
## Neg Pred Value         0.8650   0.9684  0.98827
## Prevalence             0.7785   0.1388  0.08278
## Detection Rate         0.7474   0.1119  0.07197
## Detection Prevalence   0.7700   0.1519  0.07808
## Balanced Accuracy      0.9291   0.8802  0.93133
```

Se puede apreciar, que la `balanced accurracy`, `sensitivity` y `specifity`, funcionan mucho mejor, lo que nos da buenos indicios de que el modelo funcionará, si revisamos visualmente usando la Figura 5, se comprueba lo que las medidas de rendimiento, ya predecián:

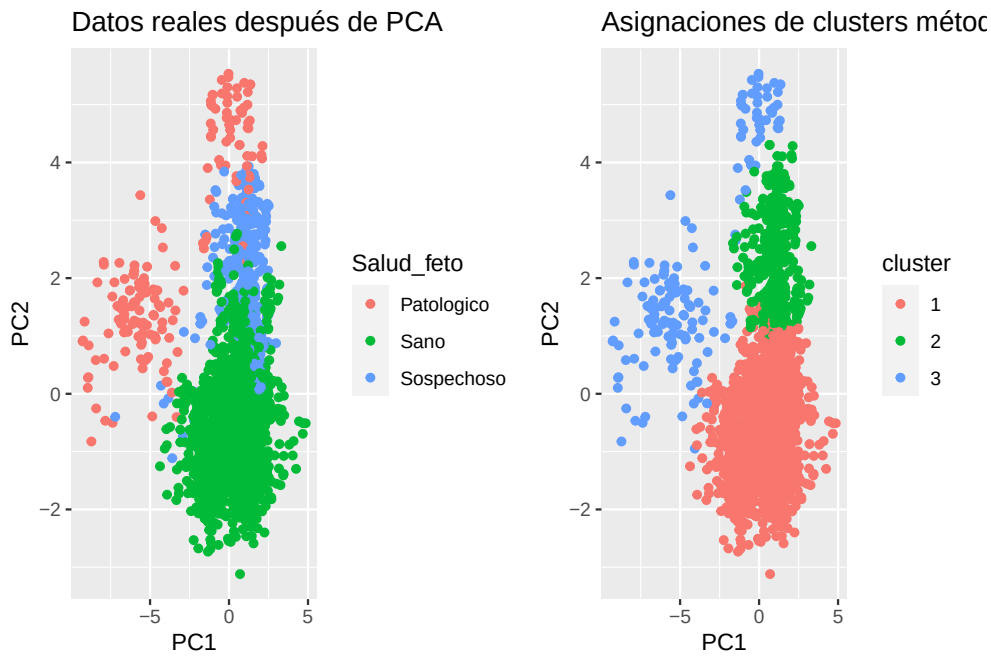

Figure 6: Figura 5: Gráfico de PC1 y PC2, para los datos reales y los del método qda

Con la Figura 5, se confirma la idea de que el método `qda` es una mejor opción.

Conclusión

El análisis de conglomerados es una buena opción, sin embargo, en ocasiones puede no ser la mejor y existen otros algoritmos que son capaces de elaborar mejores predicciones sobre sus datos, por ejemplo para este caso, el hacer un análisis de discriminante ayuda a rescatar de mejor forma la información, para así poder predecir mejor la información.

Con el modelo propuesto de qda(), es posible identificar la salud del próximo feto que se nos presente, haciendo una combinación lineal de las variables originales.

Bibliografía

- Fetal Health Classification. (2020). En *kaggle*. https://www.kaggle.com/datasets/andrewmvd/fetal-health-classification?datasetId=916586&language=R
- Curso Análisis Multivariado. (2022). https://joseperusquia.github.io/multivariate.html#Presentaciones