



# Instituto Politécnico Nacional

## *Escuela Superior de Cómputo*

### **Unidad de Aprendizaje**

Data Mining

### **Proyecto**

Final

### **Grupo**

3CV11

### **Profesora**

Ocampo Botello Fabiola

### **Fecha de entrega**

14 de junio de 2021

**Palabras clave:** arritmia, clasificación, regresión, árboles, clúster, RA, limpieza

### **Integrantes:**

- Aguilar Martínez Oswaldo
- Arévalo Andrade Miguel Ángel
- Guerrero Espinosa Ximena Mariana

# Índice

Introducción.....	6
Descripción del conjunto de datos.....	6
Diccionario de datos .....	7
Tratamiento de datos .....	12
Después de la limpieza.....	16
Anexos.....	20
Configuración del nodo File Reader.....	21
Configuración del nodo Statistics .....	22
Configuración del nodo Missing Value .....	23
Configuración del nodo Missing Value Column Filter .....	24
Configuración del nodo Histogram .....	25
Técnicas de minería de datos.....	26
Enlace del video.....	27
Árbol de clasificación ID3.....	29
Marco teórico .....	29
Elección de atributos ID3.....	30
Conceptos importantes .....	31
Información.....	31
Entropía.....	31
Información y aprendizaje .....	31
Criterio de división .....	31
Reglas de clasificación. ....	31
Atributos .....	31
Descripción del trabajo .....	32
Fragmento del diccionario de datos utilizado .....	32
Árbol ID3 .....	33
Diagrama general generado por la herramienta .....	34
Medidas .....	35

Matriz de confusión .....	35
Sensibilidad.....	35
Especificidad.....	35
Precisión.....	36
Tasa de error .....	36
Exactitud .....	36
Explicación de los positivos verdaderos y positivos falsos .....	36
Evaluación de resultados.....	37
Anexos.....	39
Configuración del nodo Column Filter.....	40
Configuración del nodo Duplicate Row Filter .....	41
Configuración del nodo Number To String .....	42
Configuración del nodo Partitioning .....	43
Configuración del nodo Decision Tree Learner .....	44
Configuración del nodo Decision Tree View .....	45
Configuración del nodo PMML Writer y Reader.....	46
Configuración del nodo Decision Tree to Ruleset.....	47
Configuración del nodo Data to Report.....	47
Configuración del nodo Image to Report .....	48
Configuración del nodo Decision Tree Predictor .....	49
Configuración del nodo Scorer .....	49
Árbol CART .....	51
Marco teórico .....	51
Atributos.....	52
Descripción del trabajo .....	52
Fragmento del diccionario de datos utilizado .....	52
Árbol CART.....	53
Diagrama general generado por la herramienta .....	54
Medidas .....	55
AUC .....	55
CA .....	56

F1 .....	56
PRECISIÓN .....	56
RECALL .....	56
Evaluación de resultados .....	57
Anexos .....	58
Configuración del nodo File .....	59
Configuración del nodo Impute .....	60
Configuración del nodo Data Table .....	61
Configuración del nodo Data Sampler .....	62
Configuración del nodo Tree .....	63
Configuración del nodo Tree Viewer .....	64
Configuración del nodo Predictions .....	65
Boosting .....	67
Marco teórico .....	67
Descripción del trabajo .....	68
Fragmento del diccionario de datos utilizado .....	69
Gradient Boost .....	70
Diagrama general generado por la herramienta .....	71
Medidas .....	72
Matriz de confusión .....	72
Sensibilidad .....	72
Especificidad .....	72
Precisión .....	73
Tasa de error .....	73
Exactitud .....	73
Explicación de los positivos verdaderos y positivos falsos .....	73
Evaluación de resultados .....	74
Anexos .....	75
Configuración del nodo Gradient Boosted Trees Learner .....	76
Configuración del nodo Gradient Boosted Trees Predictor .....	77
Random Forest .....	79

Marco teórico .....	79
Descripción del trabajo .....	80
Fragmento del diccionario de datos utilizado .....	80
Random Forest .....	81
Diagrama general generado por la herramienta .....	82
Medidas .....	83
Matriz de confusión .....	83
Sensibilidad.....	83
Especificidad.....	83
Precisión.....	84
Tasa de error .....	84
Exactitud .....	84
Explicación de los positivos verdaderos y positivos falsos .....	84
Evaluación de resultados .....	85
Anexos.....	86
Configuración del nodo Random Forest Learner .....	87
Configuración del nodo Random Forest Predictor .....	88
Reglas de Asociación .....	90
Marco teórico .....	90
Descripción del trabajo .....	91
Fragmento del diccionario de datos utilizado .....	92
Pretratamiento .....	92
Reglas de asociación .....	94
Medidas .....	96
Soporte .....	97
Confianza .....	97
Evaluación de resultados.....	98
Anexos.....	103
Regresión lineal .....	105
Enunciado del Problema .....	105
Diccionario de Datos.....	105

Desarrollo: Proceso KDD .....	105
Limpieza de los Datos .....	106
Aplicación .....	107
Significancia Estadística .....	117
Regresión Múltiple .....	119
Enunciado del Problema .....	119
Diccionario de Datos .....	119
Desarrollo: Proceso KDD .....	120
Limpieza de los Datos .....	120
Aplicación .....	122
Significancia Estadística .....	132
Clúster .....	134
Tabla comparativa .....	134
Diccionario de Datos .....	137
Desarrollo: Proceso KDD Tratamiento de los Datos .....	140
Aplicación .....	141
Anexos Flujo de trabajo de KNIME .....	144
Referencias .....	145

## Introducción

Una arritmia es un problema relacionado con la frecuencia o el ritmo del latido cardíaco. Durante una arritmia, el corazón puede latir demasiado rápido, demasiado lento o con un ritmo irregular. Cuando un corazón late demasiado rápido, la afección se llama taquicardia. Cuando un corazón late demasiado lento, la afección se llama bradicardia.

La arritmia es causada por cambios en el tejido y la actividad del corazón o en las señales eléctricas que controlan los latidos del corazón. Estos cambios pueden deberse a daños causados por enfermedades, lesiones o la genética. A menudo no se presentan síntomas, pero algunas personas sienten latidos cardíacos irregulares. Puede sentirse mareado o tener dificultad para respirar.

La prueba más común utilizada para diagnosticar una arritmia es un electrocardiograma (EKG o ECG). Su doctor realizará otras pruebas según sea necesario. Es posible que él o ella le recomiende medicinas, la colocación de un dispositivo que pueda corregir un latido cardíaco irregular o una cirugía para reparar los nervios que sobre estimulan el corazón. Si la arritmia no se trata, es posible que el corazón no pueda bombear suficiente sangre al cuerpo. Esto puede dañar el corazón, el cerebro u otros órganos.

## Descripción del conjunto de datos

<b>Nombre:</b>	Arrhythmia
<b>Objetivo:</b>	El objetivo es aplicar todas las técnicas de minería de datos vistas en el curso de Data Mining, tales como limpieza de datos, árbol de clasificación, árboles de regresión, reglas de asociación, regresión lineal y múltiple y clúster.
<b>Créditos:</b>	Original Owners of Database:  1. H. Altay Guvenir, PhD., Bilkent University, Department of Computer Engineering and Information Science, 06533 Ankara, Turkey Phone: +90 (312) 266 4133 Email: guvenir '@' cs.bilkent.edu.tr  2. Burak Acar, M.S., Bilkent University, EE Eng. Dept. 06533 Ankara, Turkey Email: buraka '@' ee.bilkent.edu.tr  3. Haldun Muderrisoglu, M.D., Ph.D.,

	<p>Baskent University, School of Medicine Ankara, Turkey</p> <p>Donor:</p> <p>H. Altay Guvenir Bilkent University, Department of Computer Engineering and Information Science, 06533 Ankara, Turkey Phone: +90 (312) 266 4133 Email: guvenir '@' cs.bilkent.edu.tr</p>
<b>Enlace de acceso:</b>	<a href="https://archive.ics.uci.edu/ml/datasets/Arrhythmia">https://archive.ics.uci.edu/ml/datasets/Arrhythmia</a>

## Diccionario de datos

#	Nombre	Significado	Tipo	Dominio
<b>1</b>	Age	Edad en años	Lineal	0-89
<b>2</b>	Sex	Género	Nominal	0 = Masculino; 1 = Femenino
<b>3</b>	Height	Estatura en cm	Lineal	132-190
<b>4</b>	Weight	Peso en kg	Lineal	10-104
<b>5</b>	QRS duration	Promedio de la duración del QRS en mseg	Lineal	61-138
<b>6</b>	P-R interval	Duración media entre el inicio de las ondas P y Q en mseg	Lineal	0-524
<b>7</b>	Q-T interval	Duración media entre el inicio de Q y el desplazamiento de ondas T en mseg	Lineal	241-509
<b>8</b>	T interval	Duración media de la onda T en mseg	Lineal	0-205
<b>9</b>	P interval	Duración media de la onda P en mseg	Lineal	-172-169
<b>10</b>	QRS	Ángulo vectorial en grados en el plano frontal de QRS	Lineal	-144-177
<b>11</b>	T	Ángulo vectorial en grados en el plano frontal de T	Lineal	-93-170
<b>12</b>	P	Ángulo vectorial en grados en el plano frontal de P	Lineal	-170-180
<b>13</b>	QRST	Ángulo vectorial en grados en el plano frontal de QRST	Lineal	-170-180

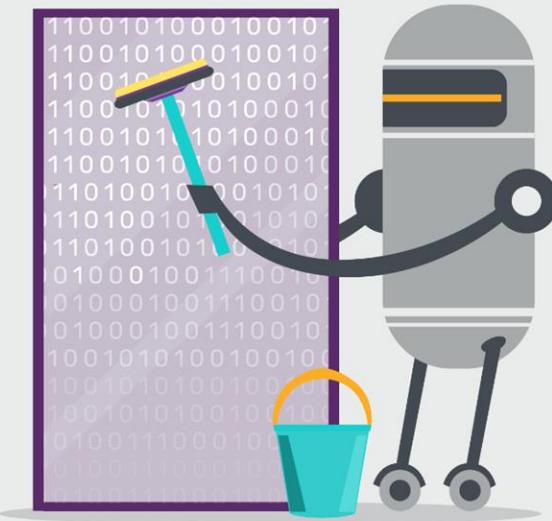
<b>14</b>	J	Ángulo vectorial en grados en el plano frontal de J	Lineal	0-112
<b>15</b>	Heart rate	Número de latidos del corazón por minuto	Lineal	0-88
<b>16</b>	Q	Anchura media, en mseg. de la onda Q	Lineal	0-156
<b>17</b>	R	Anchura media, en mseg. de la onda R	Lineal	0-88
<b>18</b>	S	Anchura media, en mseg. de la onda S	Lineal	0-24
<b>19</b>	R'	Anchura media, en mseg. de la onda R'	Lineal	0
<b>20</b>	S'	Anchura media, en mseg. de la onda S'	Lineal	0-100
<b>21</b>	Number of intrinsic deflections	Número de deflexiones intrínsecas	Lineal	0-1
<b>22</b>	Existence of ragged R wave	Existencia de onda R irregular	Nominal	0-1
<b>23</b>	Existence of diphasic derivation of R wave	Existencia de derivación difásica de la onda R	Nominal	0-1
<b>24</b>	Existence of ragged P wave	Existencia de onda P irregular	Nominal	0-1
<b>25</b>	Existence of diphasic derivation of P wave	Existencia de derivación difásica de la onda P	Nominal	0-1
<b>26</b>	Existence of ragged T wave	Existencia de onda T irregular	Nominal	0-1
<b>27</b>	Existence of diphasic derivation of T wave, nominal	Existencia de derivación difásica de la onda T	Nominal	0-1
<b>28 a 39</b>	channel DII similar (16 a 27)	...	Nominal y Lineal	0-76
<b>40 a 51</b>	channel DIII	...	Nominal y Lineal	0-116
<b>52 a 63</b>	channel AVR	...	Nominal y Lineal	0-80
<b>64 a 75</b>	channel AVL	...	Nominal y Lineal	0-148
<b>76 a 87</b>	channel AVF	...	Nominal y Lineal	0128
<b>88 a 99</b>	channel V1	...	Nominal y Lineal	0-216
<b>100 a 111</b>	channel V2	...	Nominal y Lineal	0-108

<b>112 a 123</b>	channel V3	...	Nominal y Lineal	0-132
<b>124 a 135</b>	channel V4	...	Nominal y Lineal	0-92
<b>136 a 147</b>	channel V5	...	Nominal y Lineal	0-136
<b>148 a 159</b>	channel V6	...	Nominal y Lineal	0-148
<b>160</b>	JJ wave (channel DI)	Amplitud, * 0,1 milivoltios, de onda JJ	Lineal	-2.7-0
<b>161</b>	Q wave	Amplitud, * 0,1 milivoltios, de onda Q	Lineal	0-2.80
<b>162</b>	R wave	Amplitud, * 0,1 milivoltios, de onda R	Lineal	0-1.9
<b>163</b>	S wave	Amplitud, * 0,1 milivoltios, de onda SS	Lineal	0-0.095
<b>164</b>	R' wave	Amplitud, * 0,1 milivoltios, de onda R'	Lineal	0
<b>165</b>	S' wave	Amplitud, * 0,1 milivoltios, de onda S'	Lineal	-1.5-1.7
<b>166</b>	P wave	Amplitud, * 0,1 milivoltios, de onda P	Lineal	-8.7-3.7
<b>167</b>	T wave	Amplitud, * 0,1 milivoltios, de onda T	Lineal	-33.3-155.2
<b>168</b>	QRSA	Suma de áreas de todos los segmentos dividida por 10, (Área = ancho * alto / 2)	Lineal	-38.8-74.3
<b>169</b>	QRSTA	QRSA + 0.5 * ancho de onda T * 0.1 * altura de T onda. (Si T es difásico, entonces el segmento más grande es considerado)	Lineal	-3.9-1.9
<b>170 a 179</b>	channel DII	...	Lineal	-3.4-0
<b>180 a 189</b>	channel DIII	...	Lineal	0-19.2
<b>190 a 199</b>	channel AVR	...	Lineal	-16.5-0
<b>200 a 209</b>	channel AVL	...	Lineal	0-3.2
<b>210 a 219</b>	channel AVF	...	Lineal	-1.5-0
<b>220 a 229</b>	channel V1	...	Lineal	-1.5-3.4

<b>230 a 239</b>	channel V2	...	Lineal	-30.3-0
<b>240 a 249</b>	channel V3	...	Lineal	0-28.5
<b>250 a 259</b>	channel V4	...	Lineal	-43.3-0
<b>260 a 269</b>	channel V5	...	Lineal	0-14.9
<b>270 a 279</b>	channel V6	...	Lineal	-4-0
<b>280</b>	class	Clasificación de la arritmia	Nominal	[1,16]

Tabla 1. Diccionario de datos

## Tratamiento de datos



## Tratamiento de datos

La limpieza de datos es el proceso de preparar datos para nuestro análisis mediante la eliminación o modificación de datos incorrectos, incompletos, irrelevantes, duplicados o con formato incorrecto. Por lo general, estos datos no son necesarios ni útiles cuando se trata de analizar datos porque pueden dificultar el proceso o proporcionar resultados inexactos. Existen varios métodos para limpiar los datos, dependiendo de cómo se almacenen junto con las respuestas que se buscan. La limpieza de datos no se trata simplemente de borrar información para hacer espacio para nuevos datos, sino de encontrar una manera de maximizar la precisión de un conjunto de datos sin necesariamente eliminar información. Por un lado, la limpieza de datos incluye más acciones que eliminar datos, como corregir errores de ortografía y sintaxis, estandarizar conjuntos de datos y corregir errores como campos vacíos, códigos faltantes e identificar puntos de datos duplicados. La limpieza de datos se considera un elemento fundamental de los conceptos básicos de la ciencia de datos, ya que juega un papel importante en el proceso analítico y en el descubrimiento de respuestas confiables.

Es por ello que se identificaron los atributos que poseen datos faltantes en nuestro conjunto de datos de Arrhythmia,

Col10
25
23
64
62
46
10
?
45
43
52
31
19
45
19
50
68
-164
-56
68
15
-2
52
46
87

### Columna 10: Atributo T Ángulo vectorial en grados en el plano frontal de T

Para el atributo T haremos uso de la técnica de imputación, considerando que se tratan de valores numéricos reemplazaremos los valores faltantes por la media.

En el siguiente histograma, se muestran las frecuencias de los intervalos de valores y en la última barra de la derecha, mostramos el número de valores faltantes.

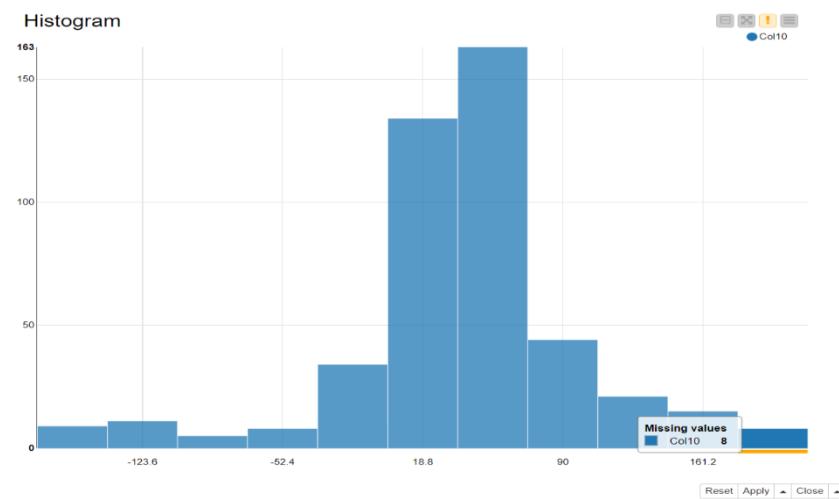


Fig. 1. Columna 10 con datos faltantes

Fig. 2. Histograma columna 10 con datos faltantes

### Columna 11: Atributo P Ángulo vectorial en grados en el plano frontal de P

Col11
57
70
36
77
60
63
55
55
62
?
36
55
61
66
?
60
65
70
68
-14
62
62
15
41
...

Para el atributo P, de igual manera haremos uso de la técnica de imputación, considerando que se tratan de valores numéricos reemplazaremos los valores faltantes por la media.

En el siguiente histograma, se muestran las frecuencias de los intervalos de valores y en la última barra de la derecha, mostramos el número de valores faltantes.

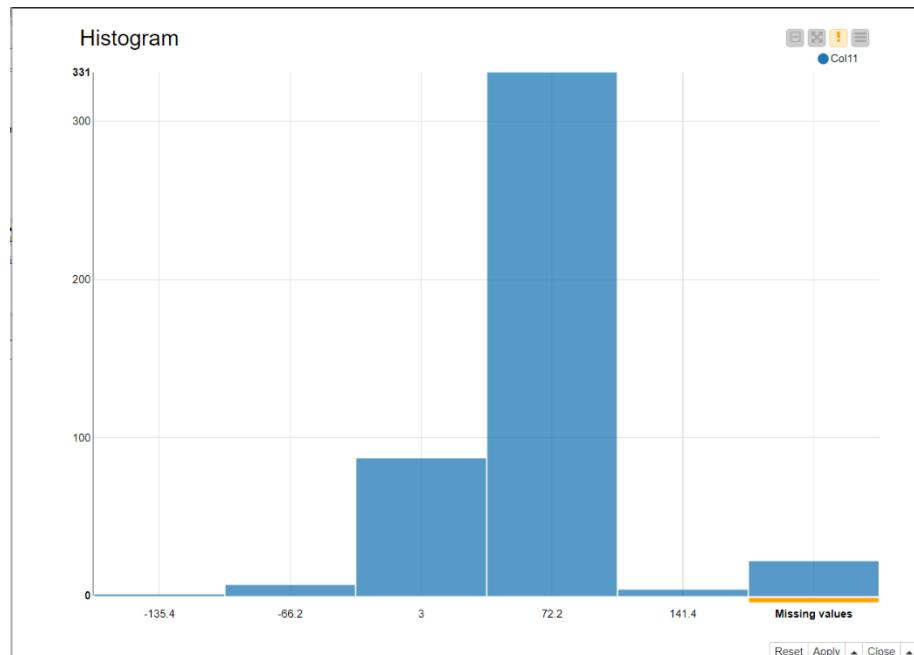


Fig. 3. Columna 11 con datos faltantes

Columna 12: Atributo QRST **Ángulo vectorial en grados en el plano frontal de QRST**

Col12
4
65
36
38
61
65
73
49
66
101
20
26
65
?
12
66
53
-43
55
40
1
13
0
-28

Para el atributo QRST haremos uso de la técnica de imputación, considerando que se tratan de valores numéricos reemplazaremos los valores faltantes por la media.

En el siguiente histograma, se muestran las frecuencias de los intervalos de valores y en la última barra de la derecha, mostramos el número de valores faltantes.

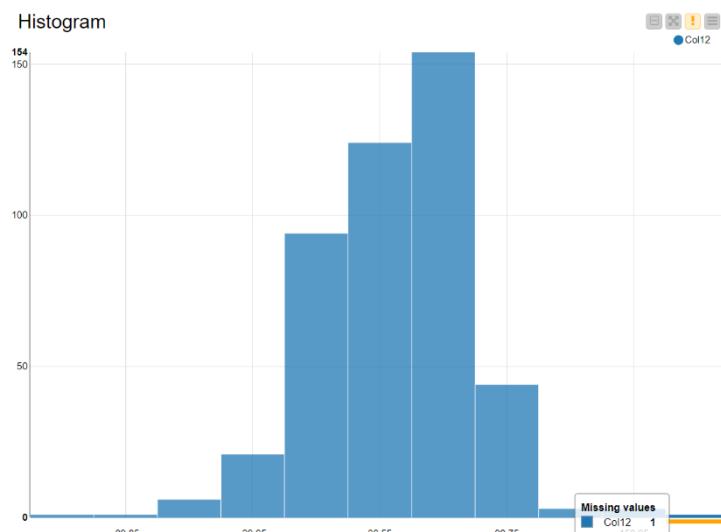


Fig. 6. Histograma columna 12 con datos faltantes

Fig. 5. Columna 12 con datos faltantes

Columna 13: Atributo **J Ángulo vectorial en grados en el plano frontal de J**

Para el atributo J, de igual manera haremos uso de la técnica de imputación, en este atributo, dado que son bastantes los valores faltantes, se les imputó una constante, y esa constante fue obtenida de la moda, para evitar descartar toda la columna.

En el siguiente histograma, se muestran las frecuencias de los intervalos de valores y en la última barra de la derecha, mostramos el número de valores faltantes.

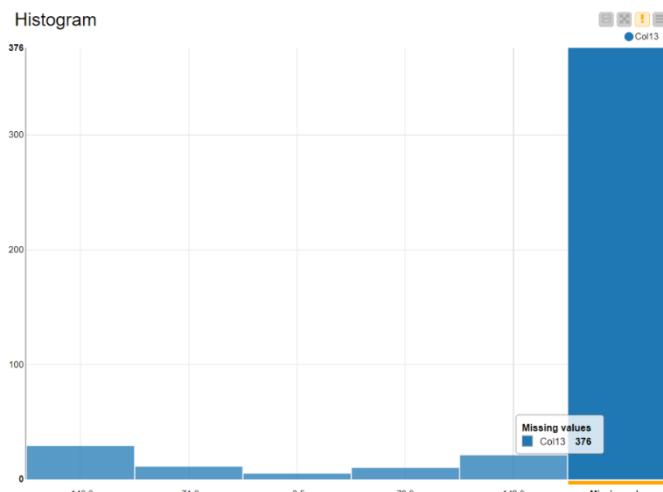


Fig. 8. Histograma columna 13 con datos faltantes

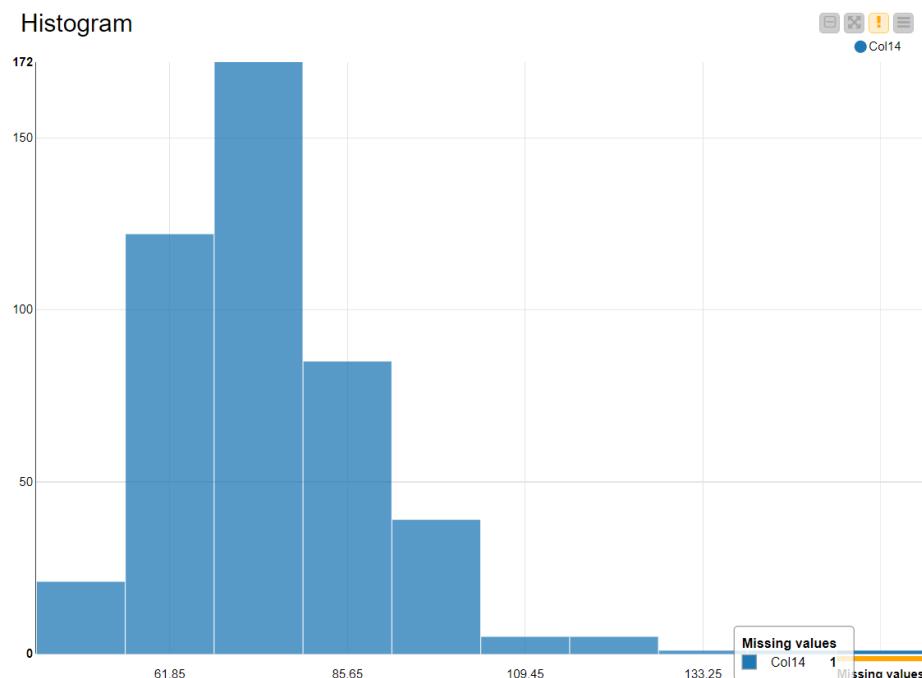
Fig. 7. Columna 13 con datos faltantes

#### Columna 14: Atributo Heart rate Número de latidos del corazón por minuto

Col14
63
53
75
71
?
84
70
67
64
63
70
72
73
56
72
76
67
70
66
66
76
66
77
69

Para el atributo Heart rate haremos uso de la técnica de imputación, considerando que se tratan de valores numéricos reemplazaremos los valores faltantes por la media.

En el siguiente histograma, se muestran las frecuencias de los intervalos de valores y en la última barra de la derecha, mostramos el número de valores faltantes.



## Después de la limpieza

### Columna 10: Atributo T Ángulo vectorial en grados en el plano frontal de T

Col10
13
37
34
11
13
66
49
7
69
34
71
37
42
51
20
45
75
49
-24
28
39
78
56
10
17
112
52
48
153
172
16
20
32
56
46
23
13

En las siguientes imágenes se muestra la columna 10 después de la limpieza realizada y el histograma correspondiente, mostrando las frecuencias de los datos, sin valores faltantes.

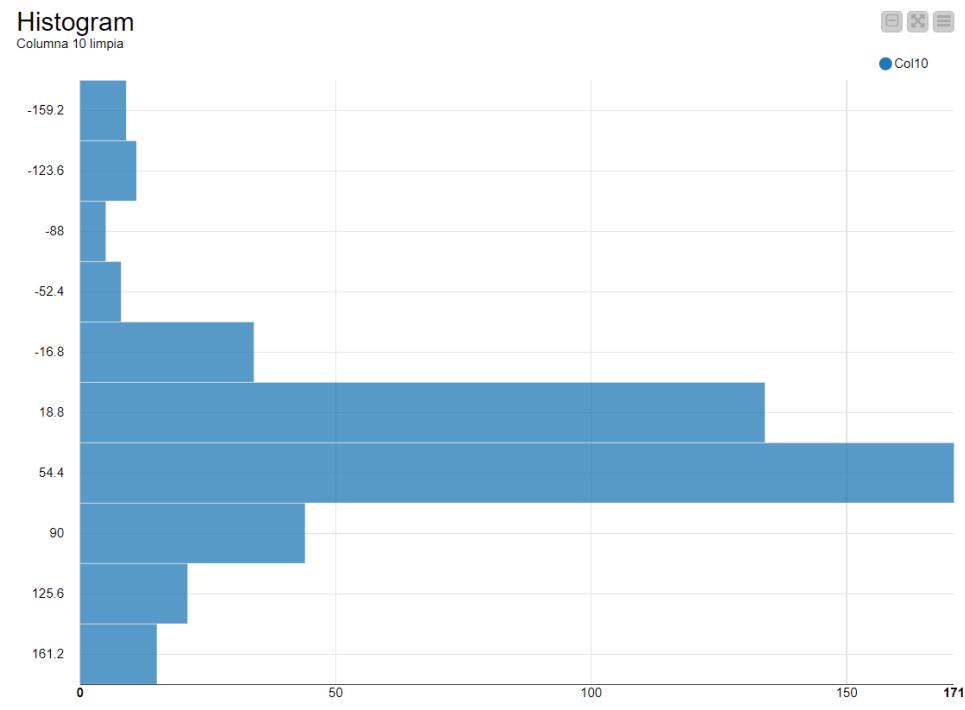


Fig. 11. Columna 10 tratada

Fig. 12. Histograma columna 10 tratada

### Columna 11: Atributo P Ángulo vectorial en grados en el plano frontal de P

D	Col11
57	
70	
36	
77	
60	
63	
55	
55	
62	
56	
36	
55	
61	
66	
56	
60	
65	
70	
68	
-14	
62	
62	
15	
41	
60	
46	
34	

En las siguientes imágenes se muestra la columna 11 después de la limpieza realizada y el histograma correspondiente, mostrando las frecuencias de los datos, sin valores faltantes.

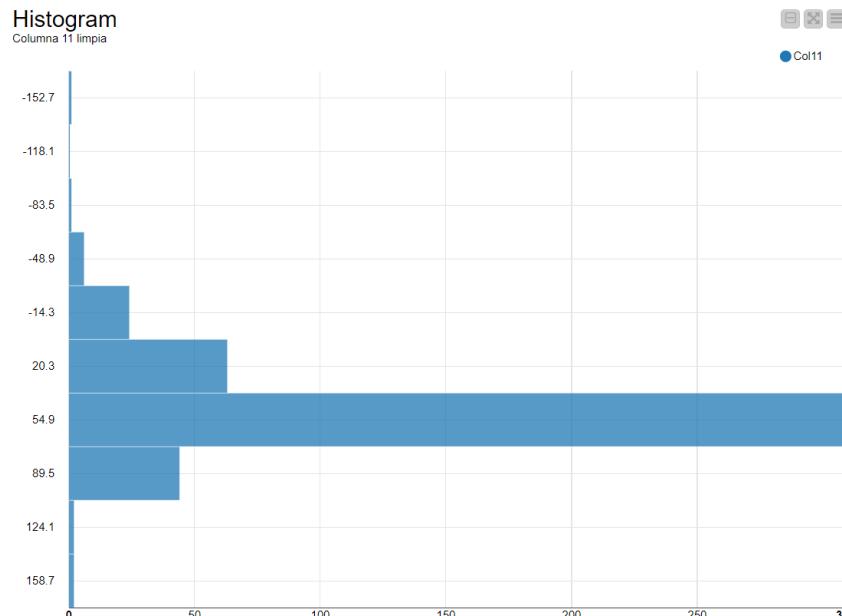


Fig. 14. Histograma columna 11 tratada

Fig. 13. Columna 11 tratada

### Columna 12: Atributo QRST Ángulo vectorial en grados en el plano frontal de QRST

D	Col12
-1	
14	
20	
68	
-10	
27	
33	
9	
59	
62	
31	
26	
3	
33	
76	
104	
30	
68	
62	
67	
31	
39	
57	
58	
3	
32	
26	
56	

En las siguientes imágenes se muestra la columna 12 después de la limpieza realizada y el histograma correspondiente, mostrando las frecuencias de los datos, sin valores faltantes.

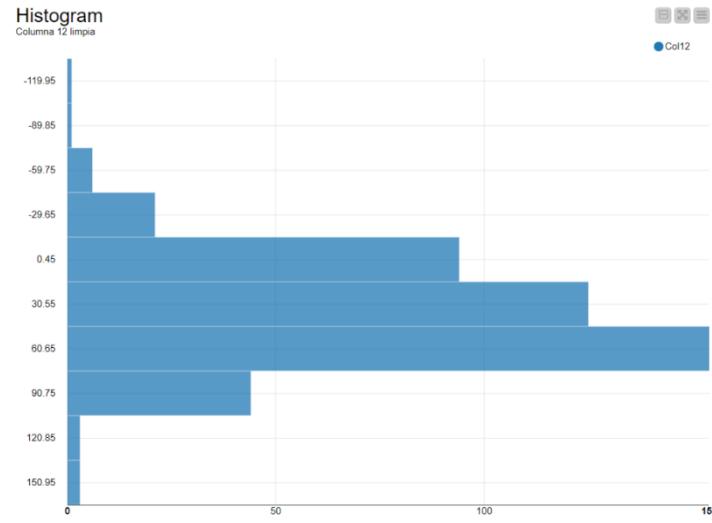


Fig. 16. Histograma columna 12 tratada

Fig. 15. Columna 12 tratada

Columna 13: Atributo **J Ángulo vectorial en grados en el plano frontal de J**

En las siguientes imágenes se muestra la columna 13 después de la limpieza realizada y el histograma correspondiente, mostrando las frecuencias de los datos, sin valores faltantes.

El valor que más se repetía fue el 84, por consiguiente, fue la constante introducida en los valores faltantes.

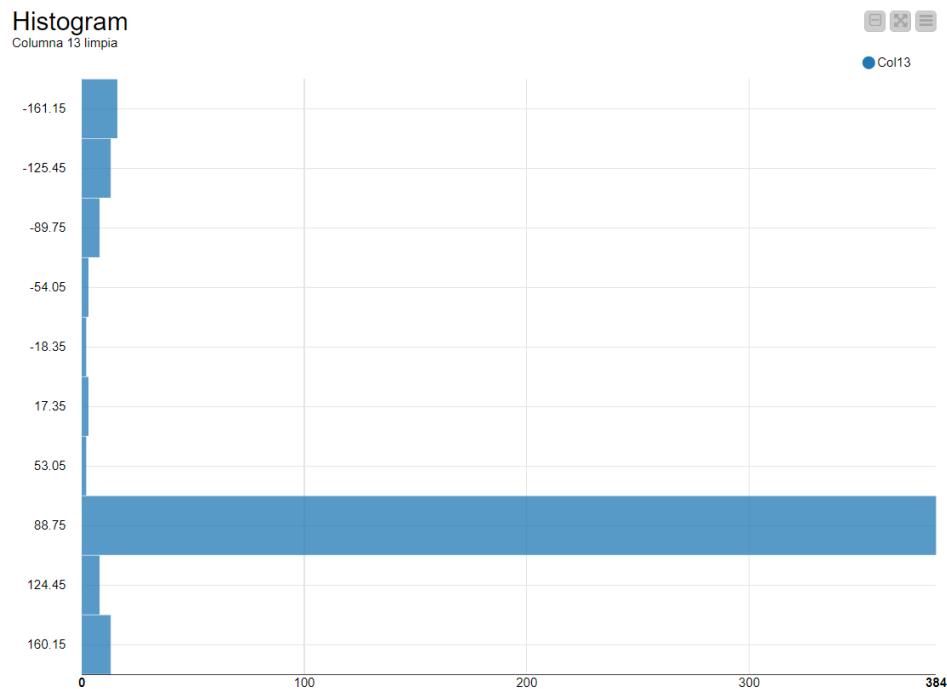


Fig. 17. Columna 13 tratada

Fig. 18. Histograma columna 13  
tratada

**Columna 14: Atributo Heart rate Número de latidos del corazón por minuto**

	Col14
	89
	98
	67
	53
	78
	65
	73
	73
	64
	59
	57
	70
	71
	56
	66
	104
	92
	65
	70
	67
	73
	54
	70
	56
	65
	53
	69
	80

En las siguientes imágenes se muestra la columna 14 después de la limpieza realizada y el histograma correspondiente, mostrando las frecuencias de los datos, sin valores faltantes.

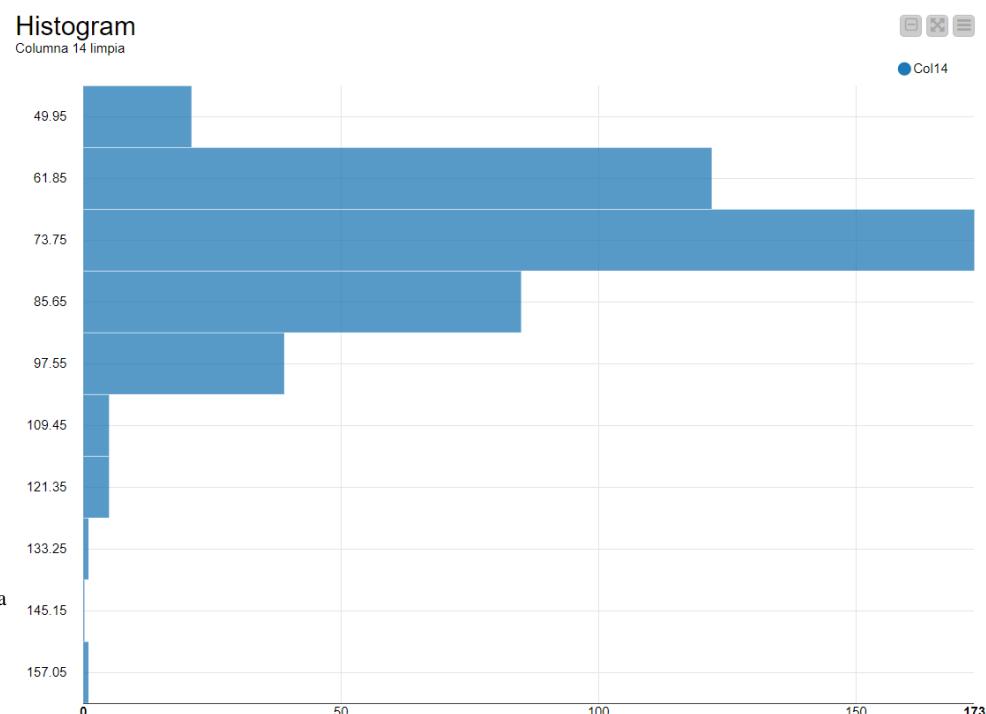


Fig. 20. Histograma column 14  
tratada

## Anexos

Configuración del flujo de trabajo para el tratamiento de datos, usando la herramienta KNIME Analytics Platform

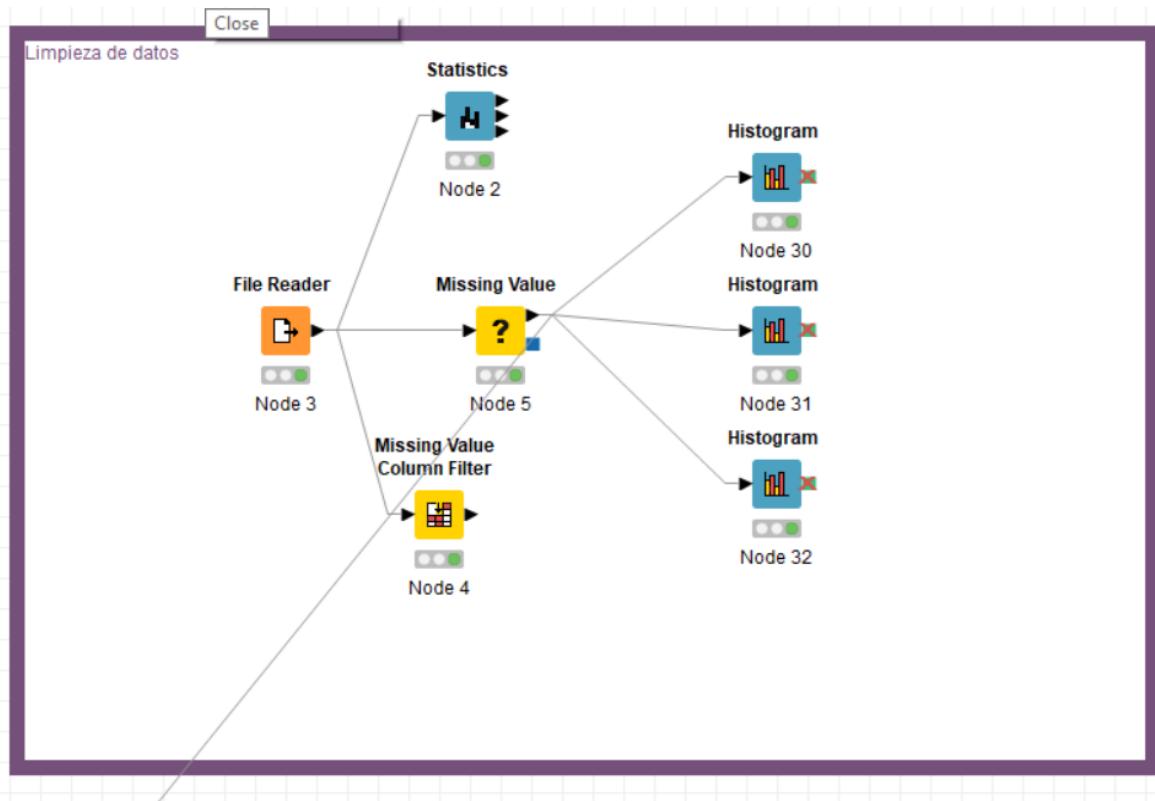


Fig. 21 Flujo de trabajo KNIME

## Configuración del nodo File Reader

Nuestro conjunto de datos no poseía ID de fila ni cabeceras de columna, es un archivo de extensión .data es por ello que no se pudo leer con el nodo CSV Reader u algún otro.

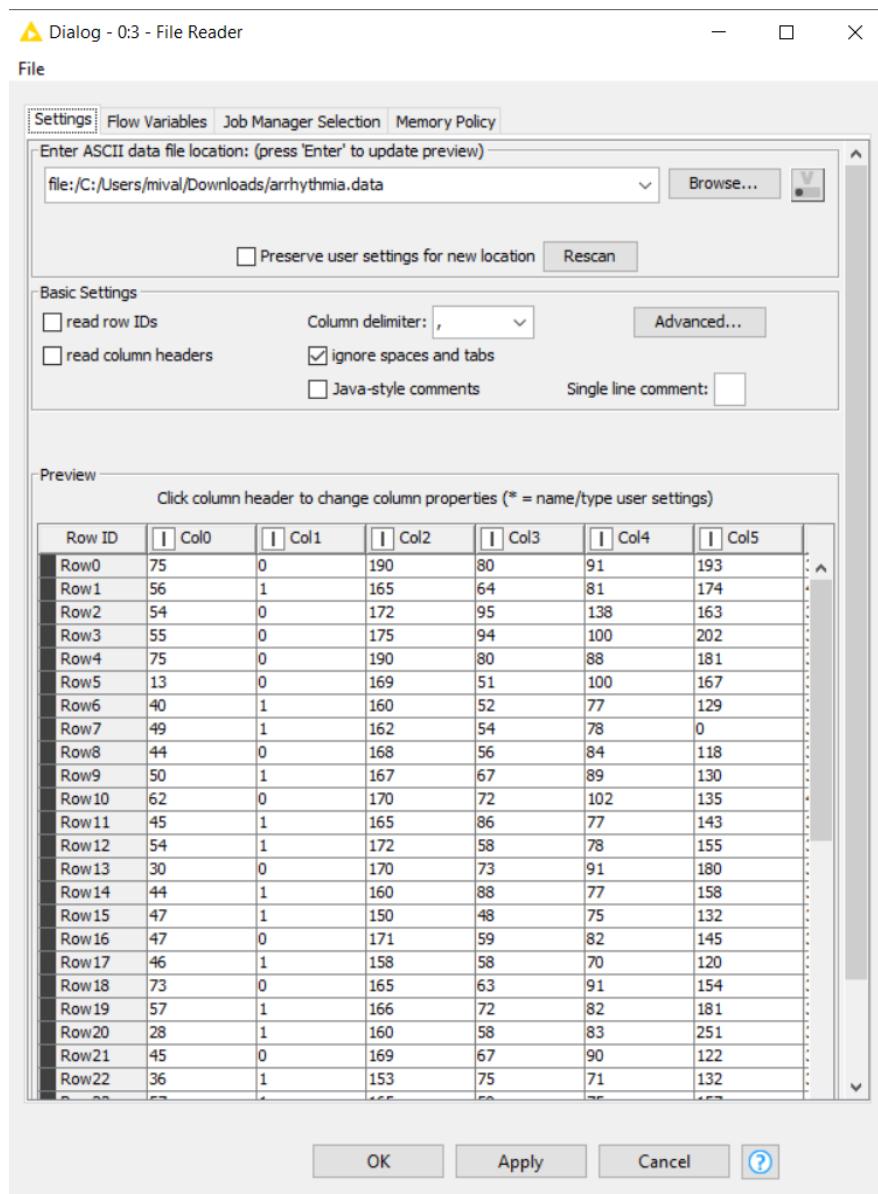


Fig. 22 Configuración nodo File Reader

## Configuración del nodo Statistics

Nodo utilizado para facilitar el cálculo de los rangos en cada atributo.

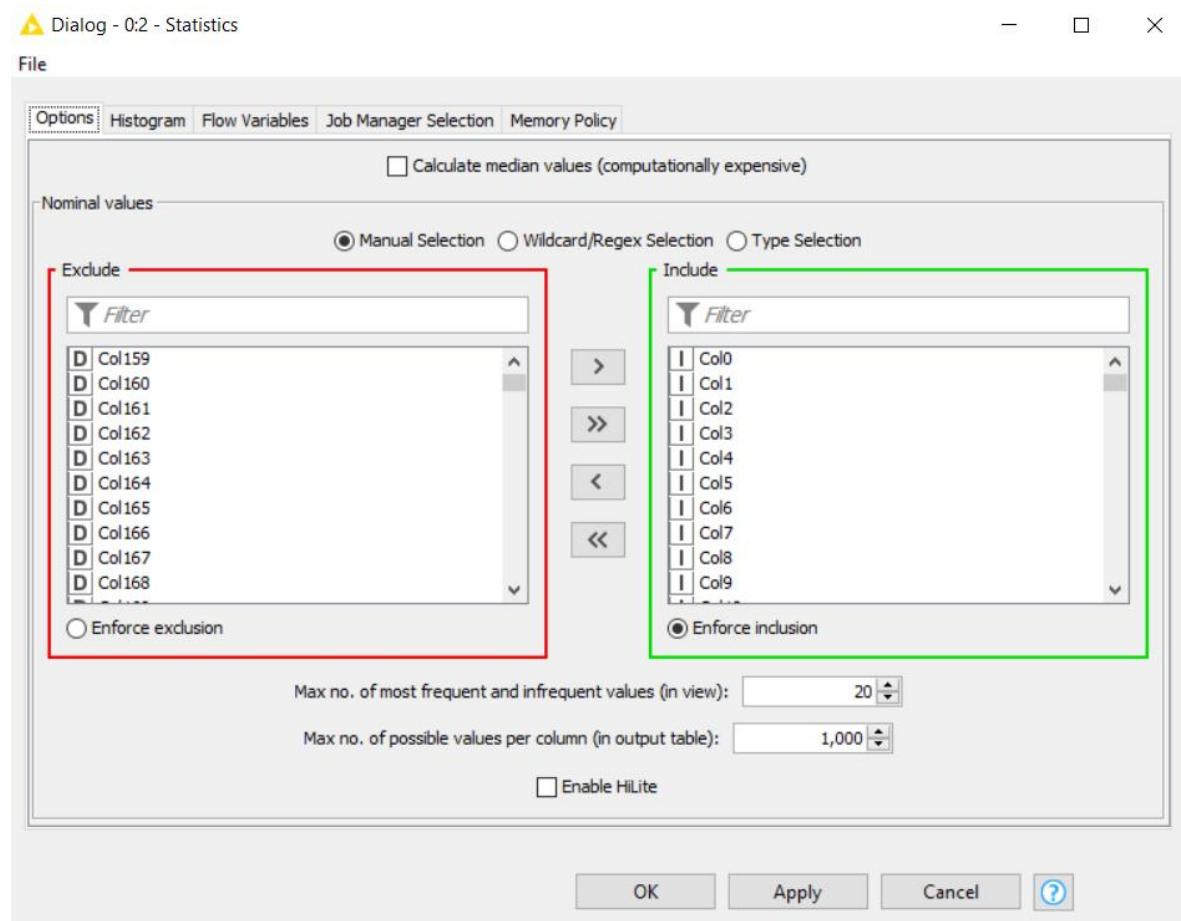


Fig. 23 Configuración nodo Statistics

## Configuración del nodo Missing Value

Como se mencionó anteriormente, para la mayoría de los atributos se le hizo el tratamiento reemplazando los valores faltantes por la media del atributo.

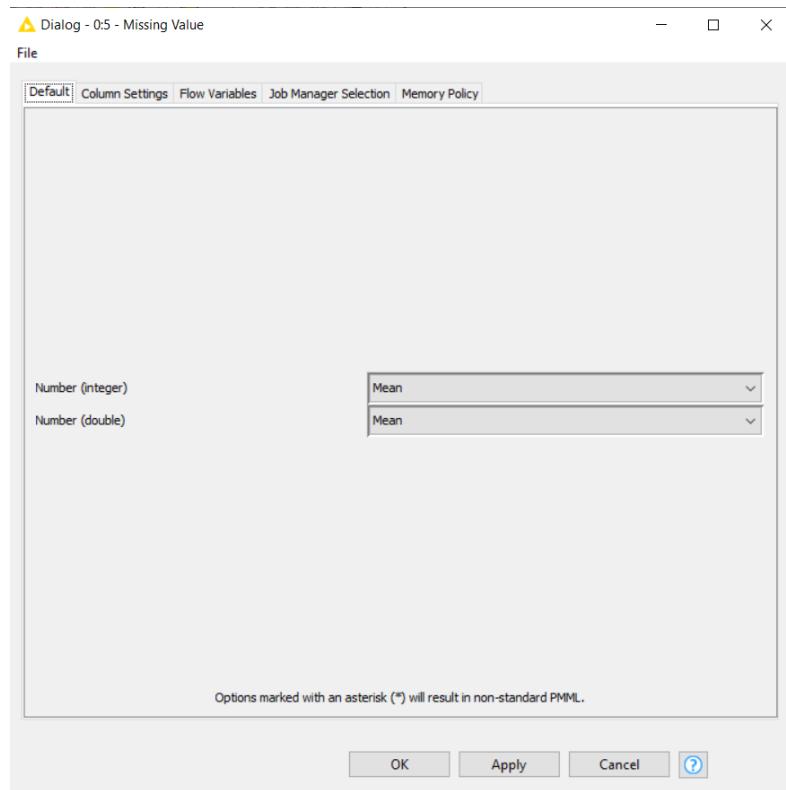


Fig. 24 Configuración nodo Missing Value

En la columna 13 que era la que más atributos faltantes tenía, se hizo la especificación en ella que la imputación fuera por el valor más frecuente.

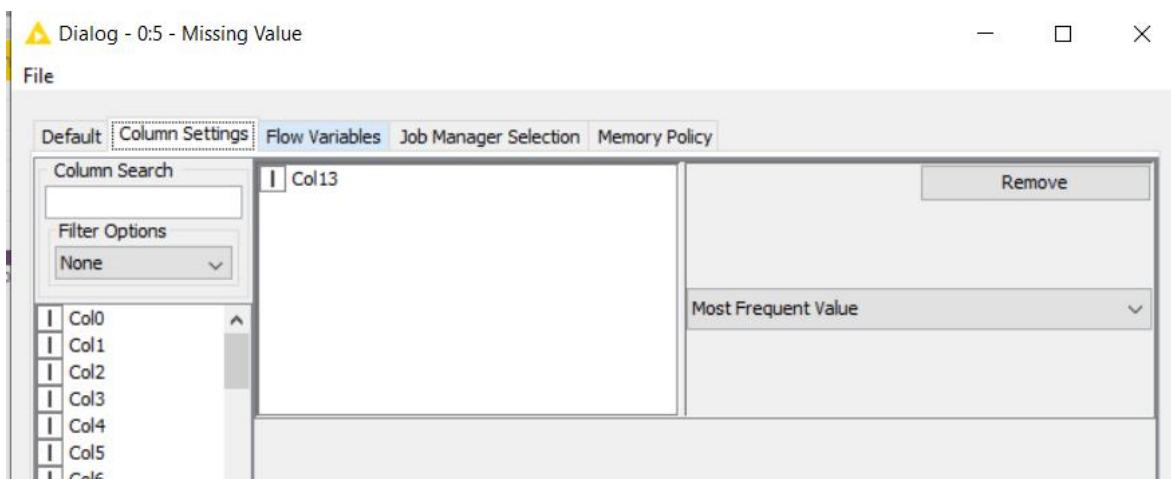


Fig. 25 Configuración nodo Missing Value

## Configuración del nodo Missing Value Column Filter

Con el objetivo de identificar los atributos o columnas con valores faltantes, se hizo un filtrado de estas, para identificar de manera más fácil a cuáles había que hacerles una limpieza de datos, usando un porcentaje de 0.1 % como mínimo de datos faltantes fue posible identificarlas correctamente.

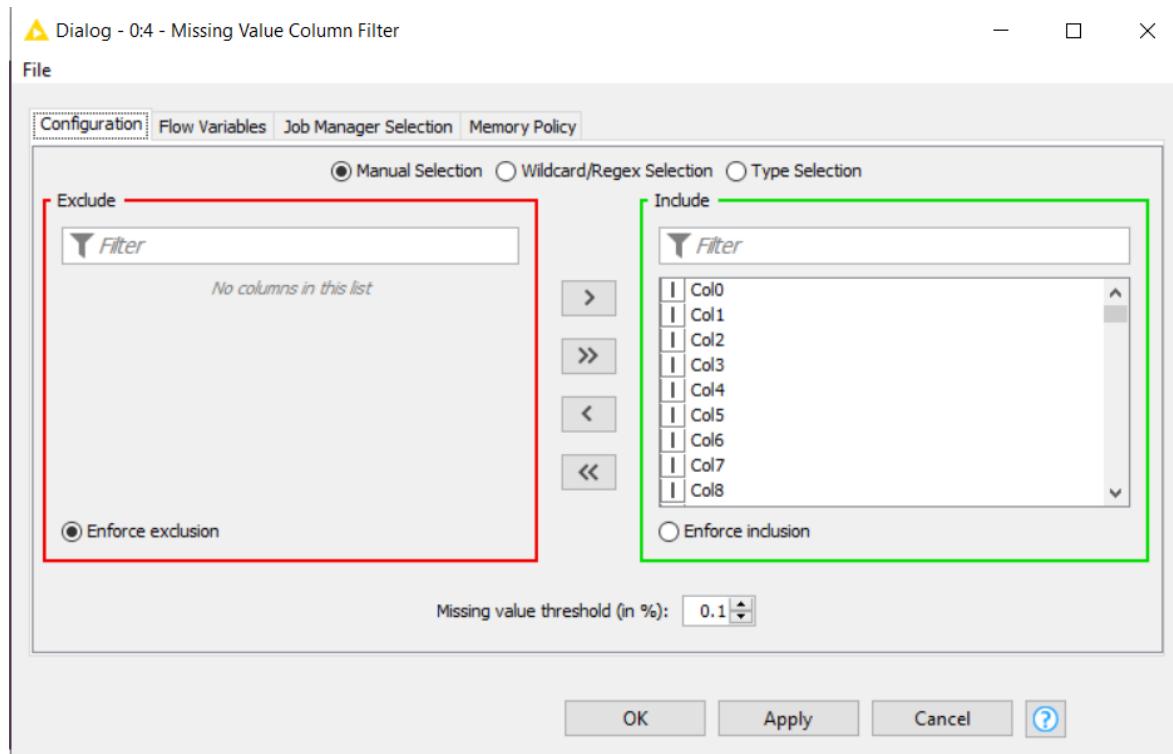


Fig. 26 Configuración nodo Missing Value Column Filter

## Configuración del nodo Histogram

La configuración era dejada por default, únicamente se seleccionaba la columna objetivo a hacerle el histograma, y en caso de que fuese un atributo con datos faltantes, se marcaban con  las casillas de abajo.

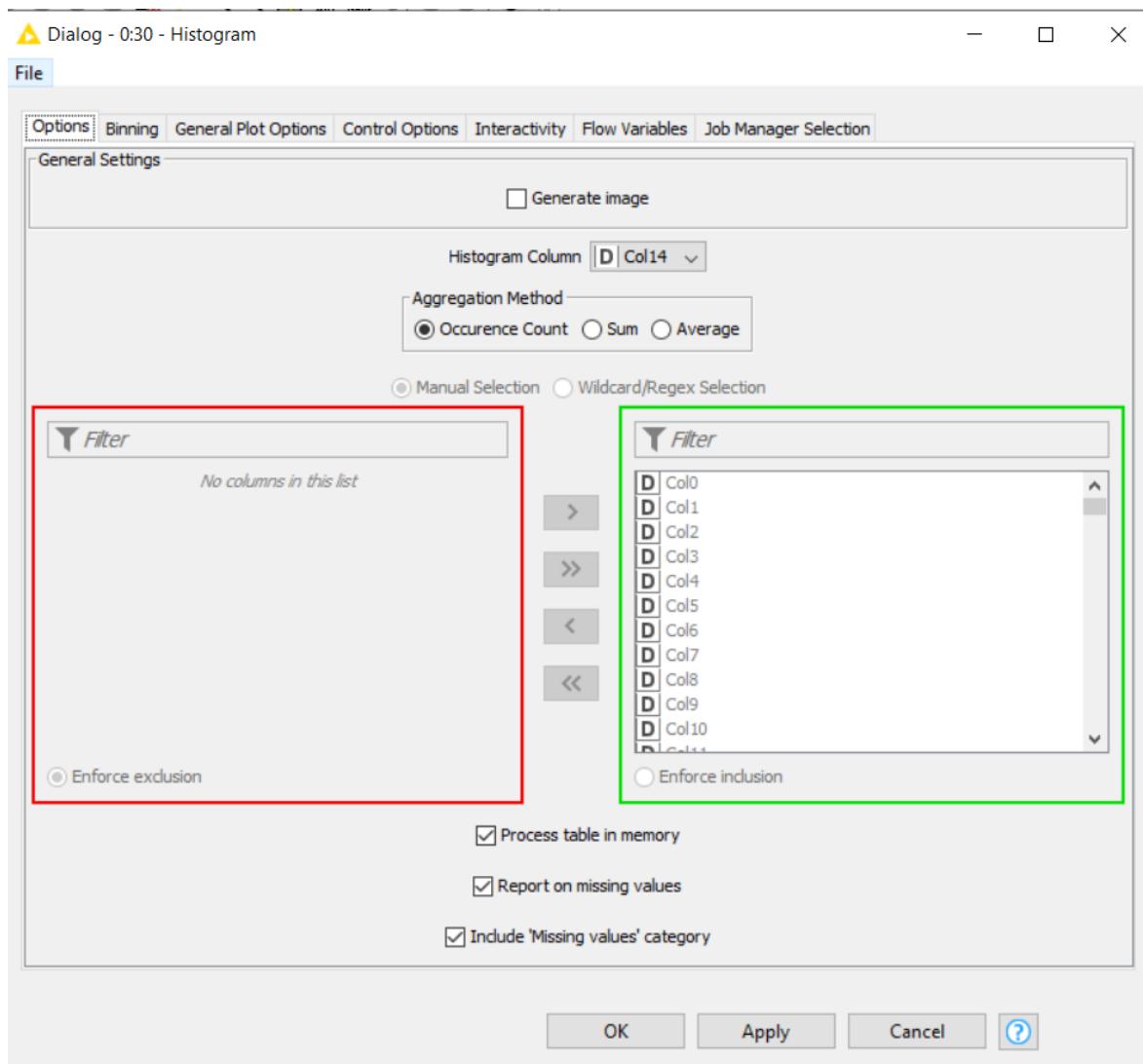


Fig. 26 Configuración nodo Histogram

## Técnicas de minería de datos

Técnica	Objetivo	Atributos
<b>Árbol de clasificación</b>	Generar un árbol de clasificación ID3 con el objetivo de distinguir entre presencia y ausencia de arritmia cardíaca y clasificarla en uno de los 16 grupos. <b>ATRIBUTO OBJETIVO: CLASS</b>	<b>class</b> , nominales de (channel DI, DII, DIII, AVR, AVL, AVF, V1, V2, V3, V4, V5, V6)
<b>Árbol CART</b>	Generar un árbol CART para predecir la probabilidad de que una persona desarrolle una arritmia cardíaca. <b>ATRIBUTO OBJETIVO: CLASS</b>	QRS duration, P-R Interval, P Interval, T, P, QRST, Heart Rate, Channel D1, D2, D3, AVR, AVF, V1, V3
<b>Boosting</b>	Generar un Gradient Boost con el objetivo de predecir la presencia y ausencia de arritmia cardíaca y clasificarla en uno de los 16 grupos. <b>ATRIBUTO OBJETIVO: CLASS</b>	<b>class</b> , nominales de (channel DI, DII, DIII, AVR, AVL, AVF, V1, V2, V3, V4, V5, V6)
<b>Random Forest</b>	Generar un Random Forest de clasificación de árboles ID3 con el objetivo de distinguir entre presencia y ausencia de arritmia cardíaca y clasificarla en uno de los 16 grupos. <b>ATRIBUTO OBJETIVO: CLASS</b>	<b>class</b> , nominales de (channel DI, DII, DIII, AVR, AVL, AVF, V1, V2, V3, V4, V5, V6)
<b>Reglas de asociación</b>	Generar las reglas de asociación para identificar la conexión que tiene la presencia de arritmia con la edad, el sexo, entre otros. <b>ATRIBUTO OBJETIVO: CLASS</b>	Sex, height, weight, heart rate, channel V5, channel V6.
<b>Regresión lineal</b>	Generar un modelo de regresión lineal de predicción, para determinar la relación entre los latidos del paciente y la edad de los pacientes. <b>ATRIBUTO OBJETIVO: HEART RATE</b>	Age, Heart Rate.
<b>Regresión múltiple</b>	Generar un modelo de regresión lineal múltiple con el objetivo de predecir el número de latidos del corazón de los pacientes, en base a la edad, el peso, la duración media entre el inicio de las ondas P y Q en msec, la duración media entre el inicio de Q y el desplazamiento de ondas T en msec, la duración media de la onda T en msec, la duración media de la onda P en msec y el ángulo vectorial en grados en el plano frontal de QRS <b>ATRIBUTO OBJETIVO: HEART RATE</b>	Age, Heart Rate, weight, P-R interval, Q-T interval, T interval, P interval, QRS.
<b>Clúster</b>	Utilizar el algoritmo k-means para generar agrupaciones del conjunto de datos mientras excluimos los elementos que sean diferentes o posean características diferentes.	Todos los atributos.

Tabla 2. Técnicas de Minería de Datos para aplicar

## Enlace del video

[https://drive.google.com/drive/folders/1TWwCLzHifqMahnR1-ewFOj\\_l5GzIMCl0?usp=sharing](https://drive.google.com/drive/folders/1TWwCLzHifqMahnR1-ewFOj_l5GzIMCl0?usp=sharing)

## Árbol de clasificación ID3

# **DECISION TREE (ID3 ALGORITHM)**

## Árbol de clasificación ID3

### Marco teórico

En la historia de la humanidad, las personas habían utilizado diversas tecnologías para modelarse a sí mismas. Hay muchas evidencias de esto desde la antigua China, Egipto y Grecia que da testimonio de la universalidad de esto. Cada nueva tecnología se ha utilizado para construir agentes inteligentes o modelos mentales. Mecanismo de relojería, hidráulica, sistemas de conmutación telefónica, hologramas, analógico computadoras y computadoras digitales se han propuesto como metáforas tecnológicas para la inteligencia y como mecanismos para modelar la mente. Hobbes (1588-1679), quien ha sido descrito por Haugeland (1985), como el "Abuelo de la IA", apoyó la posición de que pensar era el razonamiento simbólico como hablar en voz alta o hacer ejercicio una respuesta con lápiz y papel. Las operaciones simbólicas se hicieron más definidas con el desarrollo de las computadoras. El primero diseñado por computadora de propósito general (pero no construido hasta 1991, en el Museo de Ciencias de Londres) fue el Analítico Motor de Babbage (1792-1871). A principios del siglo XX, se trabajó mucho para comprender cálculo. Se propusieron varios modelos de computación, incluida la máquina de Turing de Alan Turing (1912-1954), una máquina teórica que escribe símbolos en una cinta infinitamente larga, y el cálculo lambda de Church (1903-1995), que es un formalismo matemático para reescribir fórmulas. Hubo una gran cantidad de trabajo sobre sistemas expertos, durante los años setenta y ochenta. El objetivo era captar el conocimiento de un experto en algún ámbito para que una computadora pudiera llevar realizar tareas de expertos. Por ejemplo, DENDRAL [Buchanan y Feigenbaum (1978)], desarrollado de 1965 a 1983 en el campo de la química orgánica, propuestas de estructuras plausibles para nuevos compuestos orgánicos. MYCIN [Buchanan y Shortliffe (1984)], desarrollado de 1972 a 1980, diagnosticó enfermedades infecciosas de la sangre, prescribió terapia antimicrobiana y explicó su razonamiento. Las décadas de 1970 y 1980 también fueron un período en el que el razonamiento de la IA se generalizó en los idiomas, como Prolog [Colmerauer y Roussel (1996)] [Kowalski (1988)]. Durante las décadas de 1990 y 2000 hubo grandes crecimientos en las subdisciplinas de la IA como la percepción, el razonamiento probabilístico y teórico de decisiones, la planificación, la incorporación sistemas, aprendizaje automático y muchos otros campos. Hay muchos de los más trabajos que se han realizado hasta hoy en el avance del trabajo ya realizado, y sobre los nuevos conceptos de la IA. ID3, dicotomizador iterativo 3 es una decisión algoritmo de aprendizaje de árboles que se utiliza para la clasificación de los objetos con el enfoque inductivo iterativo. En este algoritmo se utiliza el enfoque de arriba hacia abajo. El nodo superior se denomina nodo raíz y los demás son nodos hoja. Entonces es un atravesando desde el nodo raíz hasta los nodos hoja. Cada nodo requiere alguna prueba sobre los atributos que deciden el nivel de los nodos hoja. Estos árboles de decisión se utilizan principalmente para la toma de decisiones [8].

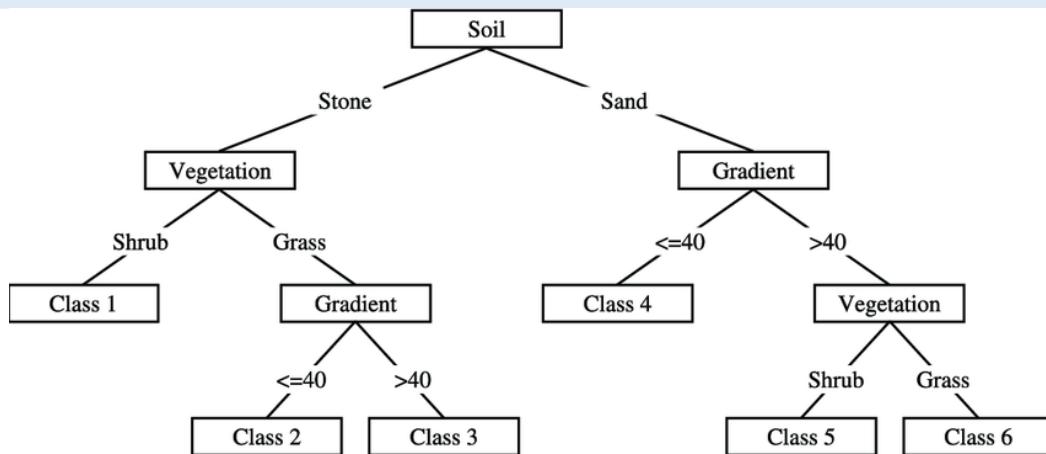


Fig. 27 Árbol ID3

El aprendizaje del árbol de decisiones es un procedimiento para calcular el valor objetivo que tiene una función discreta. La función que ha sido aprendido está simbolizada por un árbol de decisiones. Para la inferencia inductiva, el aprendizaje del árbol de decisiones es uno de los métodos de uso común y generalizado que son de naturaleza práctica [1] [3].

Los algoritmos de aprendizaje del árbol de decisiones se utilizan principalmente por tres razones:

1. El árbol de decisión es una buena inferir de los casos particulares que son instancias no observadas.
2. Los cálculos en estos métodos son eficientes y proporcionales a las instancias que se observan.
3. Al final, el árbol de decisiones que se produce es fácilmente comprensible para el ser humano. [2] [3]

### Elección de atributos ID3

El orden en el que se eligen los atributos determina qué tan complicado es el árbol. ID3 utiliza la teoría de la información para determinar el atributo más informativo. Una medida del contenido de información de un mensaje es la inversa de la probabilidad de recibir el mensaje:

$$\text{información1}(M) = 1 / \text{probabilidad}(M)$$

Tomar logaritmos base dos, hace que la información se corresponda con la cantidad de bits necesarios para codificar un mensaje:

$$\text{información}(M) = -\log_2(\text{probabilidad}(M))$$

## Conceptos importantes

### Información

El contenido de información de un mensaje debe estar relacionado con el grado de sorpresa al recibir el mensaje. Una alta probabilidad de llegada no es tan informativa como los mensajes con baja probabilidad. El aprendizaje tiene como objetivo predecir con precisión, es decir, reducir la sorpresa. Las probabilidades se multiplican para obtener la probabilidad de que sucedan dos o más cosas.

Los logaritmos de las probabilidades permiten agregar información en lugar de multiplicar.

### Entropía

Los diferentes mensajes tienen diferentes probabilidades de llegada. El nivel general de incertidumbre (denominado entropía) es:

$$-\sum_i P_i \log_2 P_i$$

La frecuencia se puede utilizar como una estimación de probabilidad. Si hay 5 ejemplos positivos y 3 ejemplos negativos en un nodo la probabilidad estimada de positivo es  $5/8 = 0,625$ .

### Información y aprendizaje

Podemos pensar en el aprendizaje como la construcción de asignaciones de muchos a uno entre la entrada y la salida. El contenido de información de las entradas mapeándolas a menos salidas. Por lo tanto, tratamos de minimizar la entropía. El mapeo es mapear todo en una salida. Buscamos un compromiso entre precisión y simplicidad.

### Criterio de división

Calcule la entropía basándose en la distribución de clases. Intente dividir en cada atributo. Calcule la ganancia de información esperada. Para cada atributo elija el mejor atributo.

ID3 es un algoritmo de aprendizaje de árbol de decisión simple que utiliza la búsqueda codiciosa de arriba hacia abajo para construir el árbol que decidir las reglas de decisión. Para ello, se requieren algunos conceptos matemáticos. Los dos conceptos que son básicamente involucrados en ID3 son **entropía** y **ganancia de información**.

### Reglas de clasificación.

Si la entropía del atributo es 0, es un nodo homogéneo y no hay necesidad de clasificar más.

Si la entropía del atributo es 1, es un nodo heterogéneo y es necesario clasificarlo más [7].

### Atributos

Para el árbol de clasificación ID3, tanto el atributo objetivo como los atributos predictores (variables dependientes e independientes) deben ser de tipo nominal.

Nuestro atributo objetivo será el último al cuál se le puso el nombre de **class** que es el tipo de arritmia a clasificar (variable dependiente).

Y las variables predictoras (variables independientes) serán los atributos nominales de (**channel DI, DII, DIII, AVR, AVL, AVF, V1, V2, V3, V4, V5, V6**)

## Descripción del trabajo

Algunos tipos de arritmia cardíaca, como la FV (fibrilación ventricular) o la TV (taquicardia ventricular), son potencialmente mortales. Por tanto, la predicción, detección y clasificación de las arritmias son cuestiones muy importantes en la cardiología clínica, tanto para el diagnóstico como para el tratamiento. Recientemente, la investigación se ha concentrado en los dos últimos problemas, a saber, la detección y clasificación de la arritmia, que es un campo maduro [4]. Estos algoritmos se implementan en los desfibriladores cardioversores implantables (ICD) [5], que se utilizan de forma rutinaria para tratar la arritmia cardíaca [6]. Sin embargo, el problema relacionado de la predicción de eventos de arritmia sigue siendo un desafío, es por ello que la aplicación de esta técnica al conjunto de datos es una pequeña contribución a la cardiología clínica, con el fin de enriquecer este campo.

## Fragmento del diccionario de datos utilizado

<b>22</b>	<b>Existence of ragged R wave</b>	<b>Existencia de onda R irregular</b>	<b>Nominal</b>	<b>0-1</b>
<b>23</b>	Existence of diphasic derivation of R wave	Existencia de derivación difásica de la onda R	Nominal	0-1
<b>24</b>	Existence of ragged P wave	Existencia de onda P irregular	Nominal	0-1
<b>25</b>	Existence of diphasic derivation of P wave	Existencia de derivación difásica de la onda P	Nominal	0-1
<b>26</b>	Existence of ragged T wave	Existencia de onda T irregular	Nominal	0-1
<b>27</b>	Existence of diphasic derivation of T wave, nominal	Existencia de derivación difásica de la onda T	Nominal	0-1
<b>33 a 39</b>	channel DII similar (33 a 39)	...	Nominal	0-76
<b>45 a 51</b>	channel DIII	...	Nominal	0-116
<b>57 a 63</b>	channel AVR	...	Nominal	0-80
<b>69 a 75</b>	channel AVL	...	Nominal	0-148
<b>81 a 87</b>	channel AVF	...	Nominal	0128
<b>93 a 99</b>	channel V1	...	Nominal	0-216

<b>105 a 111</b>	channel V2	...	Nominal	0-108
<b>117 a 123</b>	channel V3	...	Nominal	0-132
<b>129 a 135</b>	channel V4	...	Nominal	0-92
<b>141 a 147</b>	channel V5	...	Nominal	0-136
<b>153 a 159</b>	channel V6	...	Nominal	0-148
<b>280</b>	class	Clasificación de la arritmia	Nominal	[1,16]

Tabla 3. Fragmento diccionario de datos para ID3

## Árbol ID3

La tabla de entrada se divide en dos particiones (es decir, en filas), por ejemplo. entrenar y probar datos. Las dos particiones están disponibles en los dos puertos de salida. Para la fase de entrenamiento se tomó una muestra tomada desde arriba, este modo coloca las filas más altas en la primera tabla de salida y el resto en la segunda tabla, con porcentaje relativo del 95%, así como se muestra en la Fig. 28, por otro lado, para la fase de prueba, fue usado el 5% restante configurado similarmente. Su desarrollo fue en la herramienta de software KNIME Analytics Platform.

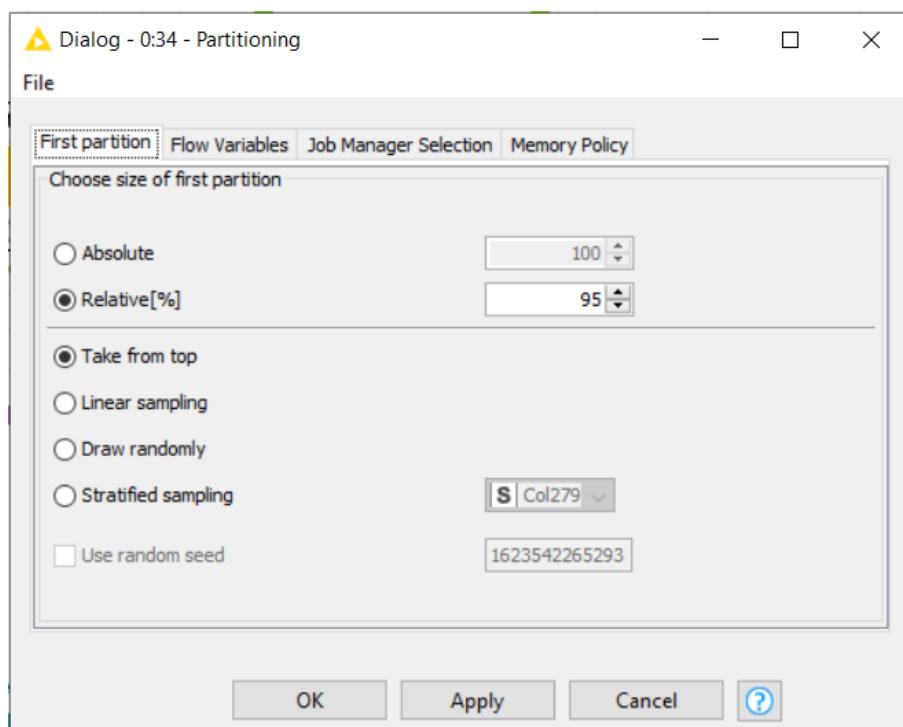


Fig. 28 Particionamiento del conjunto de datos

## Diagrama general generado por la herramienta

En la Fig. 29 se observa el árbol de decisión ID3 que consta de varios nodos. La etiqueta de clase y los recuentos de clases mostrados dentro de los nodos corresponden a los de los datos de entrenamiento. Esto significa especialmente que las frecuencias mostradas no corresponden a las filas utilizadas para la selección (a menos que utilice el conjunto de datos de entrenamiento completo para la selección).

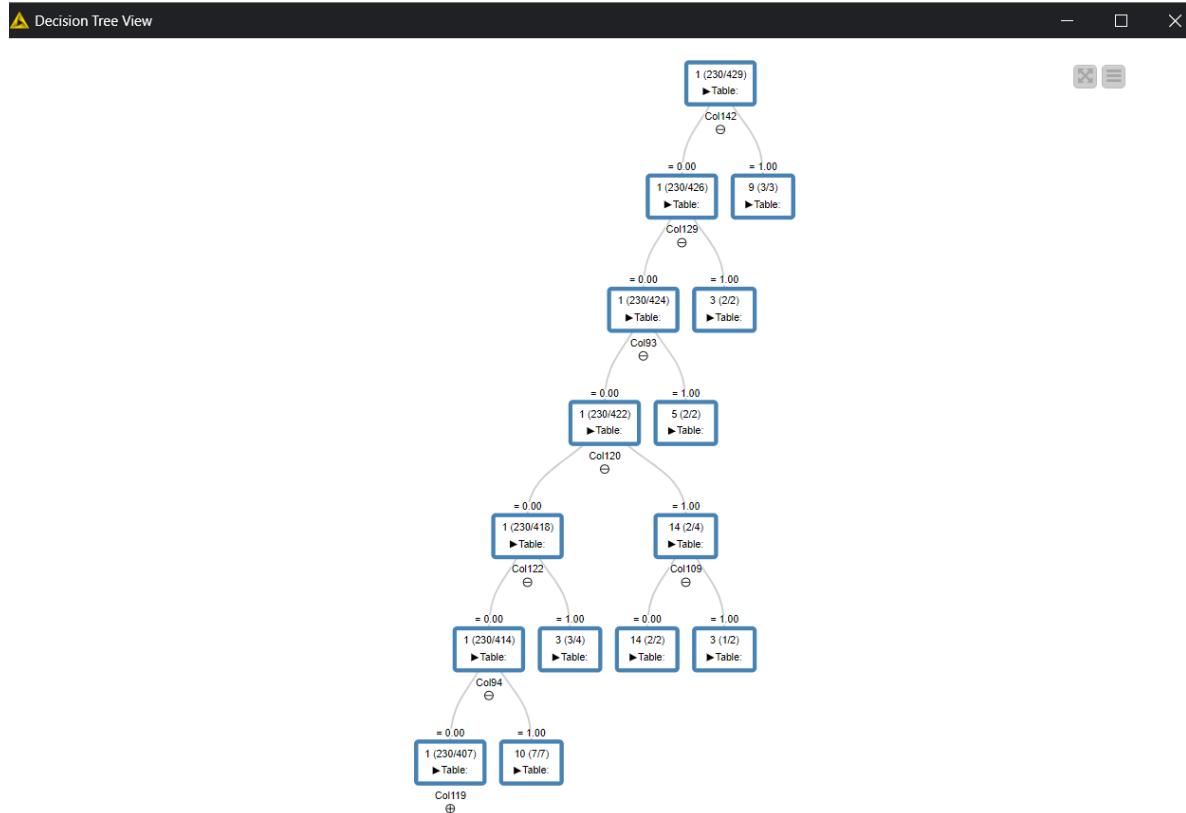


Fig. 29 Árbol ID3 generado por el KNIME Analytics Platform

## Medidas

### Matriz de confusión

La matriz de confusión es una forma tabular de visualizar el rendimiento de su modelo de predicción. Cada entrada en una matriz de confusión denota el número de predicciones realizadas por el modelo donde clasificó las clases correctas o incorrectamente. Cualquier persona que ya esté familiarizada con la matriz de confusión sabe que la mayoría de las veces se explica para un problema de clasificación binaria. En este caso se tiene una matriz de confusión de clase múltiple. A diferencia de la clasificación binaria, no hay clases positivas o negativas aquí, en nuestro caso son los diferentes tipos de arritmia a clasificar.

Fig. 29 Matriz de confusión árbol ID3

### Sensibilidad

La sensibilidad y la especificidad son medidas estadísticas del rendimiento de una prueba de clasificación binaria que se utilizan ampliamente:

La sensibilidad (tasa de verdaderos positivos) mide la proporción de positivos que se identifican correctamente (es decir, la proporción de arritmias clasificadas correctamente).

### Especificidad

La especificidad (tasa de verdaderos negativos) mide la proporción de negativos que se

identifican correctamente (es decir, la proporción de arritmias que no son de un tipo y se clasifican correctamente como arritmias que no son de ese tipo).

## Precisión

La precisión mide qué tan bueno es el modelo para asignar eventos positivos a la clase positiva (tipo de arritmia clasificado en el tipo correcto). Es decir, qué tan precisa es la predicción de tipos de arritmia.

## Tasa de error

### *Error de tipo I*

El primer tipo de error es el rechazo de una verdadera hipótesis nula como resultado de un procedimiento de prueba. Este tipo de error se denomina error de tipo I (falso positivo) y, a veces, se denomina error del primer tipo.

En términos de nuestro trabajo desarrollado, un error de tipo I corresponde a clasificar una arritmia de un tipo, siendo de otro.

### *Error de tipo II*

El segundo tipo de error es no rechazar una hipótesis nula falsa como resultado de un procedimiento de prueba. Este tipo de error se denomina error de tipo II (falso negativo) y también se denomina error de segundo tipo.

En términos de nuestro trabajo desarrollado, un error de tipo II corresponde a clasificar una arritmia de un tipo, en otro.

## Exactitud

En un conjunto de mediciones, la exactitud es la cercanía de las mediciones a un valor específico, mientras que la precisión es la cercanía de las mediciones entre sí.

La exactitud tiene dos definiciones:

1. Más comúnmente, es una descripción de errores sistemáticos, una medida de sesgo estadístico; la baja precisión provoca una diferencia entre un resultado y un valor "verdadero". ISO llama a esto veracidad.
2. Alternativamente, ISO define exactitud como la descripción de una combinación de ambos tipos de error de observación (aleatorio y sistemático), por lo que una alta exactitud requiere tanto alta precisión como veracidad.

## Explicación de los positivos verdaderos y positivos falsos

Un verdadero positivo es un resultado en el que el modelo predice correctamente la clase positiva. De manera similar, un verdadero negativo es un resultado en el que el modelo predice correctamente la clase negativa.

Un falso positivo es un resultado en el que el modelo predice incorrectamente la clase positiva. Y un falso negativo es un resultado en el que el modelo predice incorrectamente la clase negativa.

Row ID	TruePositives	FalsePositives	TrueNegatives	FalseNegatives	Recall	Precision	Sensitivity	Specificity	F-meas...	Accuracy	Cohen's Kappa
8	0	1	22	0	?	0	?	0.957	?	?	?
6	0	2	20	1	0	0	0	0.909	NaN	?	?
10	2	0	19	2	0.5	1	0.5	1	0.667	?	?
1	9	5	3	6	0.6	0.643	0.6	0.375	0.621	?	?
3	1	0	22	0	1	1	1	1	1	?	?
2	0	3	19	1	0	0	0	0.864	NaN	?	?
7	0	0	23	0	?	?	?	1	?	?	?
14	0	0	23	0	?	?	?	1	?	?	?
16	0	0	22	1	0	?	0	1	?	?	?
4	0	0	23	0	?	?	?	1	?	?	?
5	0	0	23	0	?	?	?	1	?	?	?
9	0	0	23	0	?	?	?	1	?	?	?
15	0	0	23	0	?	?	?	1	?	?	?
Overall	?	?	?	?	?	?	?	?	?	0.522	0.17

Fig. 30. Tabla de estadísticas de exactitud

## Evaluación de resultados

Los resultados estadísticos y de predicción no fueron los esperados para el árbol ID3, ni aumentando la partición para el aprendizaje del árbol a 95% fue suficiente, obteniendo precisiones del 52% a lo mucho. Eso nos indica que hay un desbalance de clases o clasificación desequilibrada.

La clasificación desequilibrada se refiere a un problema de modelado predictivo de clasificación en el que el número de ejemplos en el conjunto de datos de entrenamiento para cada etiqueta de clase no está equilibrado. Es decir, donde la distribución de clases no es igual o cercana a la misma y, en cambio, está sesgada.

**Clasificación desequilibrada:** un problema de modelado predictivo de clasificación donde la distribución de ejemplos entre las clases no es igual.

Consultando la distribución de valores para cada tipo de arritmia que debe haber en cada clase, confirmamos el desbalance.

### Distribución de clases:

Código de clase:	Clase:	Número de instancias
01	Normal	245
02	Ischemic changes (Coronary Artery Disease)	44
03	Old Anterior Myocardial Infarction	15
04	Old Inferior Myocardial Infarction	15
05	Sinus tachycardy	13
06	Sinus bradycardy	25
07	Ventricular Premature Contraction (PVC)	3
08	Supraventricular Premature Contraction	2
09	Left bundle branch block	9
10	Right bundle branch block	50
11	1. degree AtrioVentricular block	0

12	2. degree AV block	0
13	3. degree AV block	0
14	Left ventricule hypertrophy	4
15	Atrial Fibrillation or Flutter	5
16	Others	22

**Es por ello que apuntamos a una mejora en la clasificación con el uso de las siguientes dos técnicas Boosting y Random Forest.**

## Anexos

Flujo de trabajo configurado en tres secciones, fase de aprendizaje, fase de prueba y pruebas PMML, observe la Fig. 31.

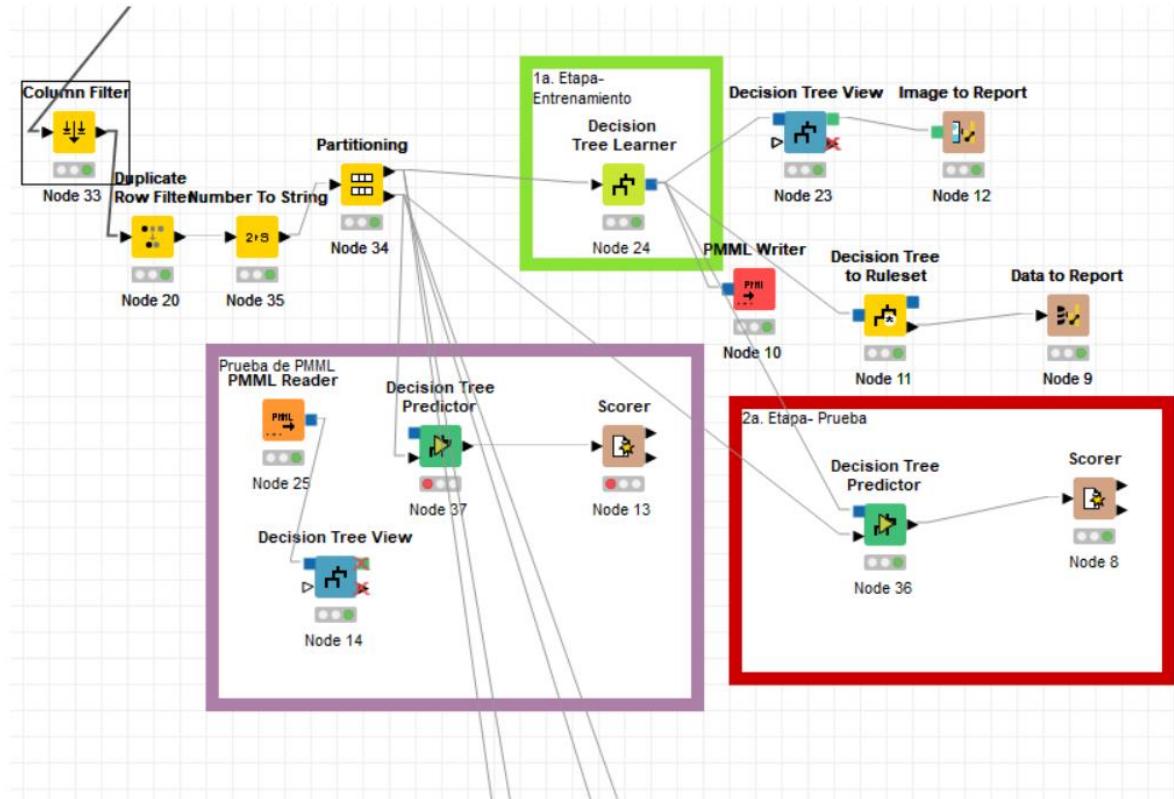


Fig. 31 Flujo de trabajo KNIME

## Configuración del nodo Column Filter

Se excluyeron atributos numéricos, dados los requisitos del algoritmo ID3.

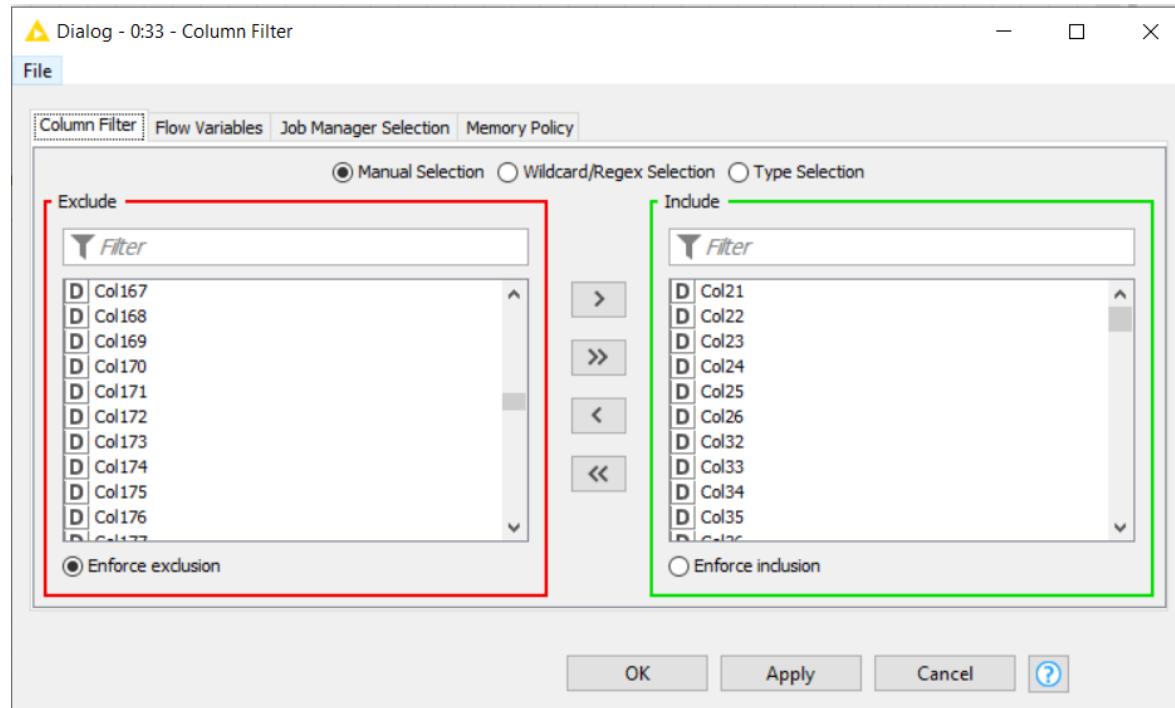


Fig. 32 Configuración nodo Column Filter

## Configuración del nodo Duplicate Row Filter

Se utilizó este nodo con el objetivo de eliminar registros duplicados, para una correcta y mejor fase de aprendizaje, en este caso, no hubo registros duplicados.

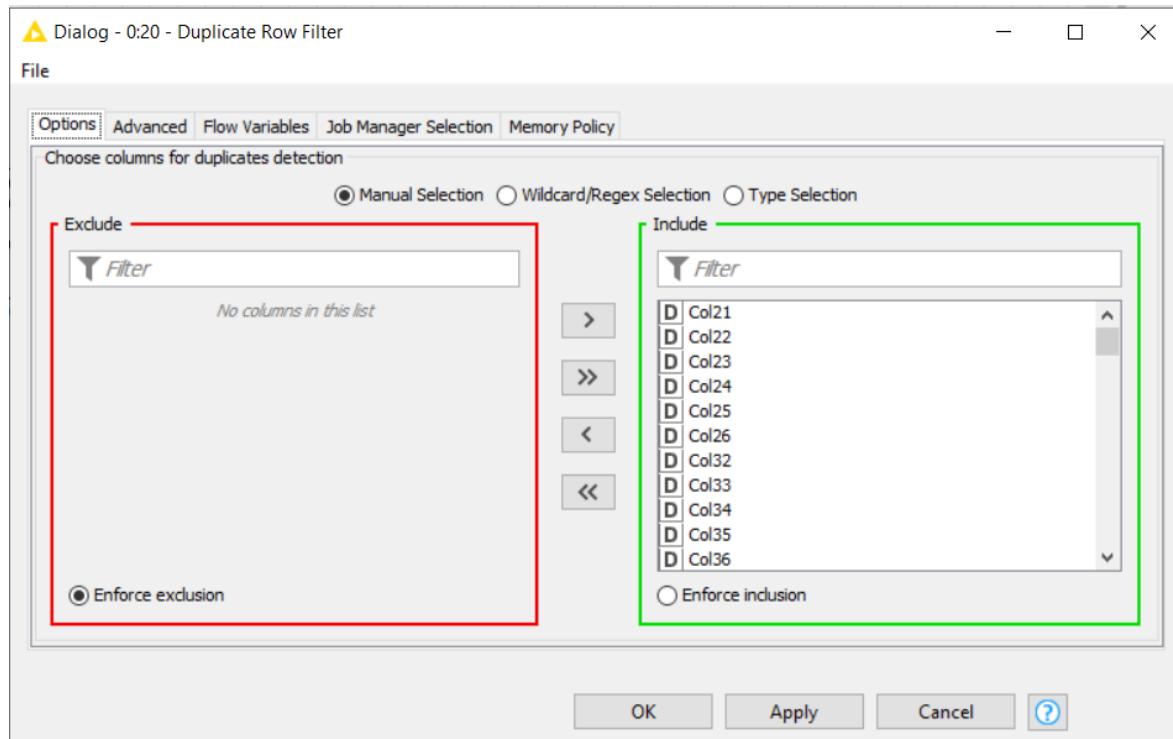


Fig. 33 Configuración nodo Duplicate Row Filter

## Configuración del nodo Number To String

Dado que los requisitos del árbol ID3 exigen atributos nominales, los atributos nominales estaban representados por números y fue necesario convertirlos a string con este nodo.

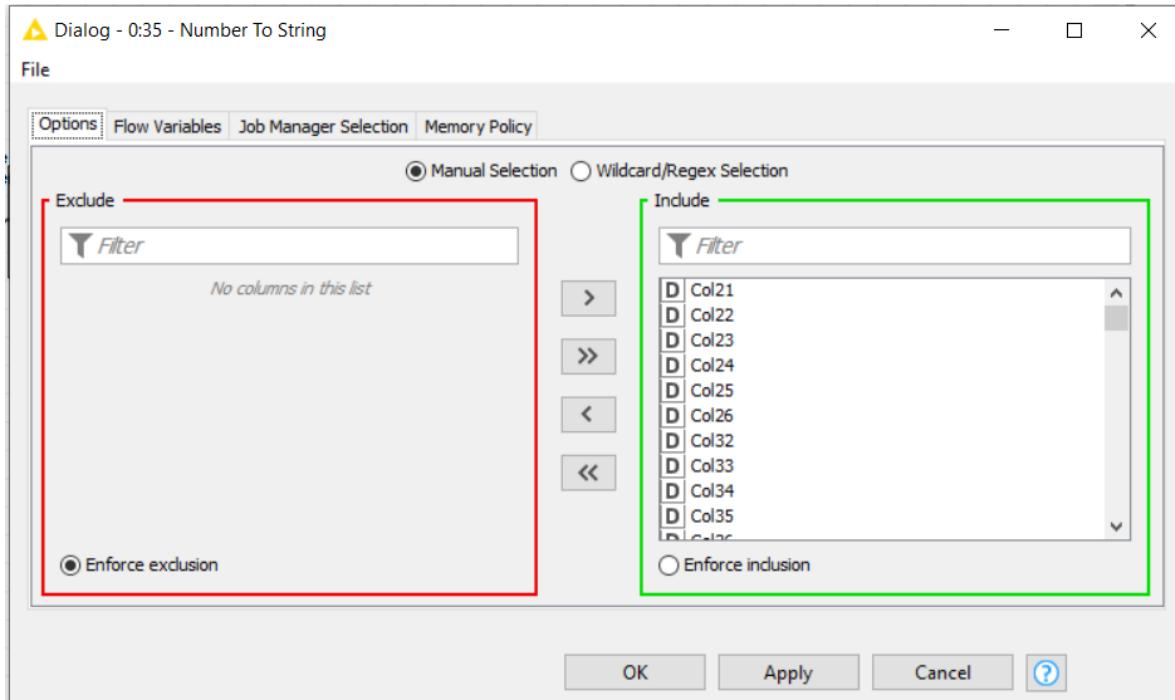


Fig. 34 Configuración nodo Number to String

## Configuración del nodo Partitioning

Configurado al 95% para aprendizaje y 5% de prueba, con la columna objetivo **class** (columna 279), tomando una muestra desde arriba, coloca las filas superiores en la primera tabla de salida y el resto en la segunda tabla.

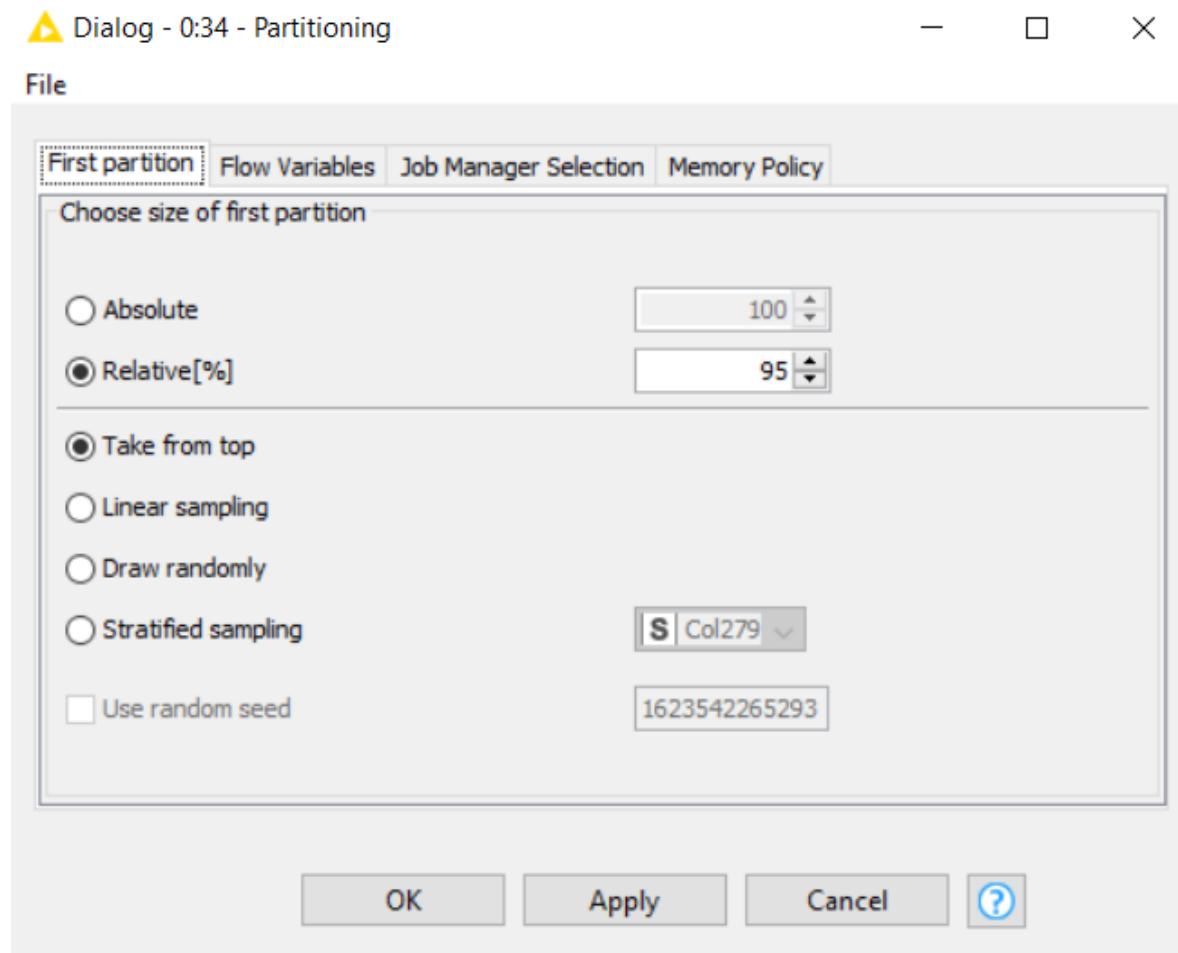


Fig. 35 Configuración nodo Partitioning

## Configuración del nodo Decision Tree Learner

Nuestra clase está en la columna **class** ya que estamos utilizando el algoritmo ID3, nuestra cualidad a medir será el **Gain ratio**; con un número de hilos de 8, lo demás se deja por default como está.

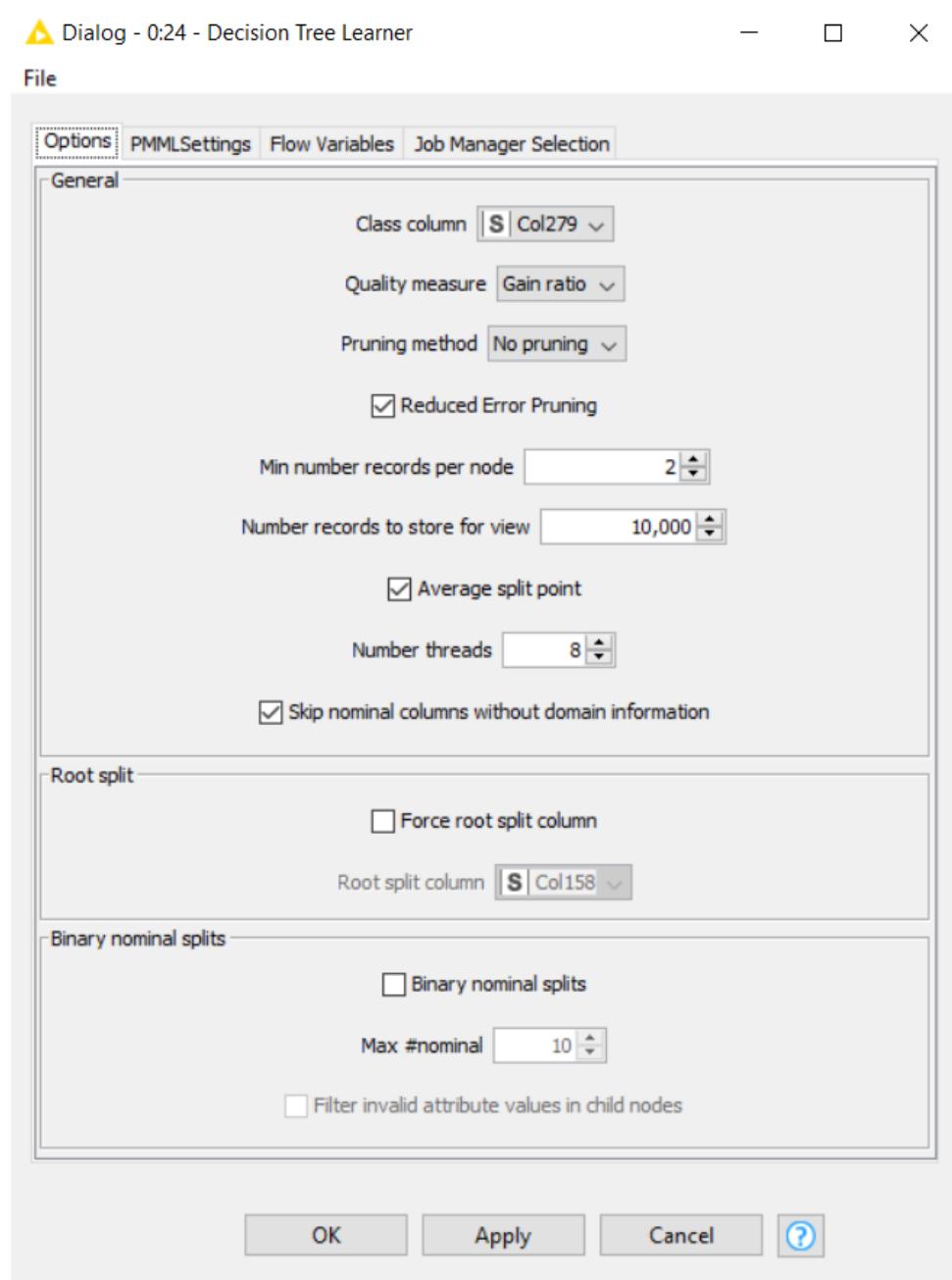


Fig. 36 Configuración nodo Decision Tree Learner

## Configuración del nodo Decision Tree View

Activamos la casilla de crear imagen en el puerto de salida, únicamente.

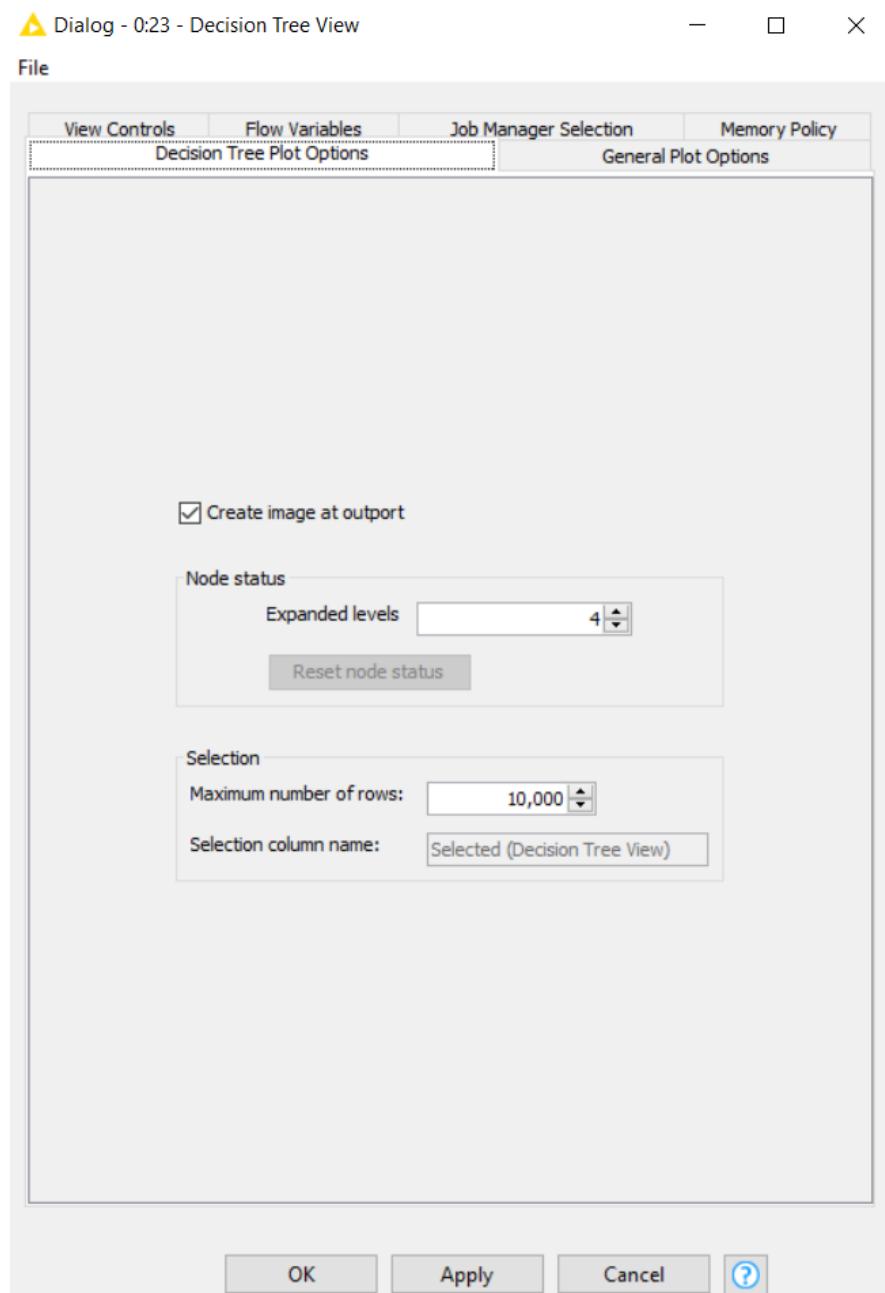


Fig. 37 Configuración nodo Decision Tree View

## Configuración del nodo PMML Writer y Reader

Simplemente se escribe el nombre del archivo PMML a leer y a escribir

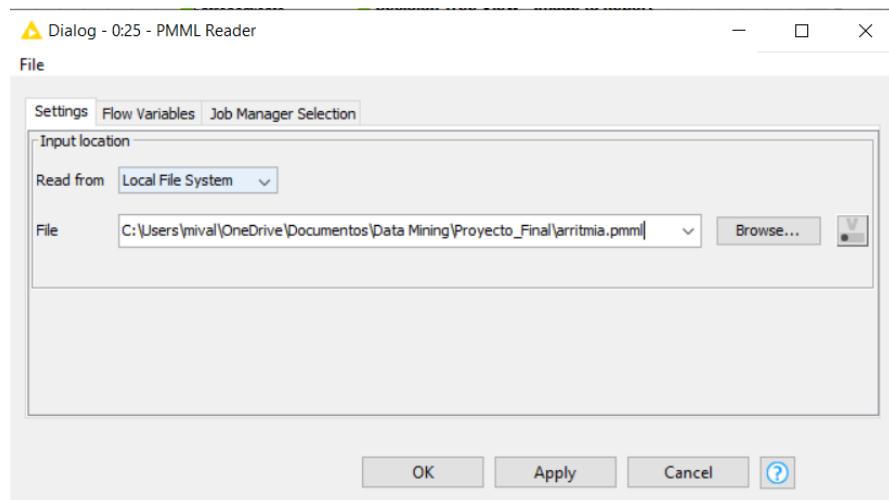
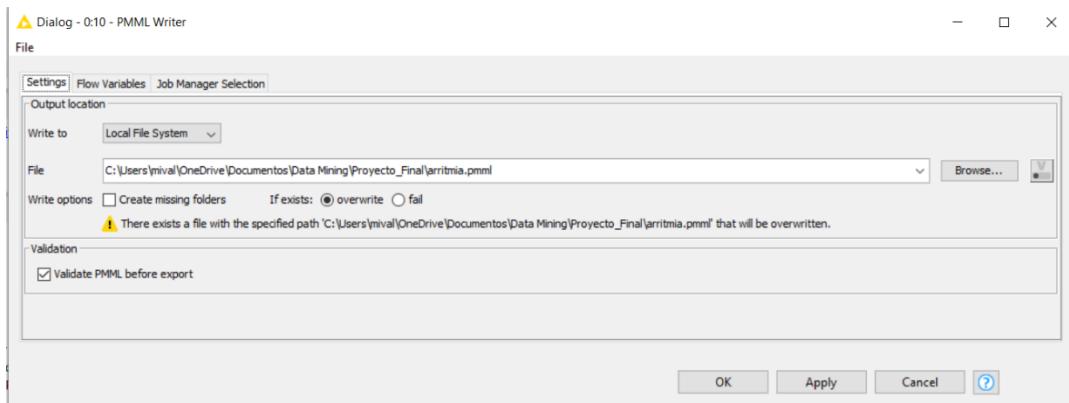


Fig 38 y 39. Ventanas de configuración PMML Writer y PMML Reader

## Configuración del nodo Decision Tree to Ruleset

Se deja la configuración dada por default.

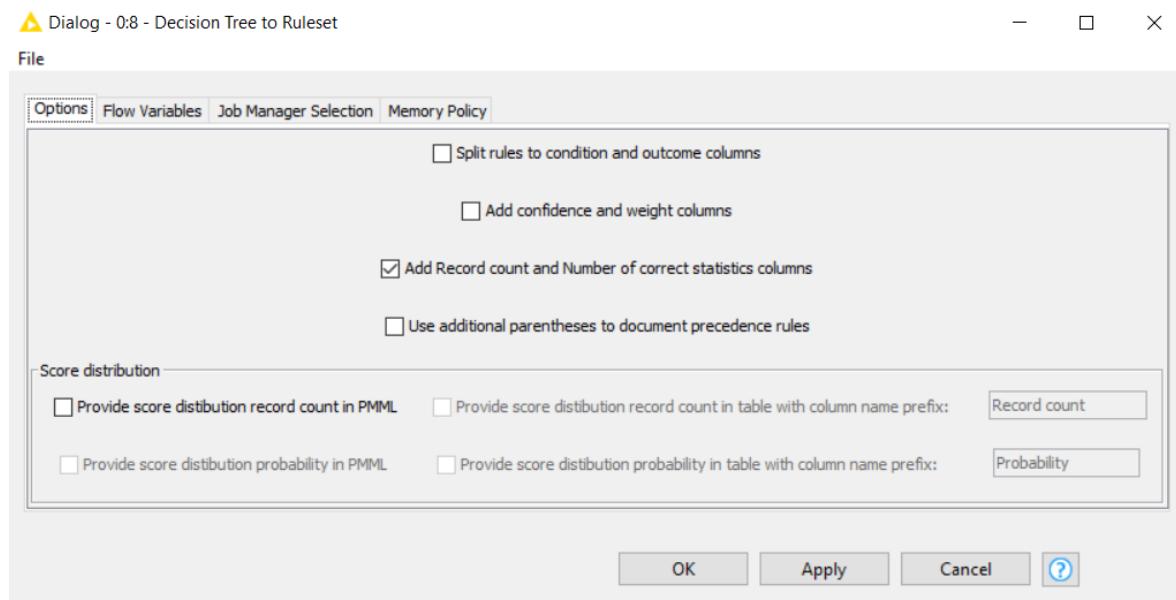


Fig. 40 Ventana de configuración Decision Tree to Ruleset

## Configuración del nodo Data to Report

Se modificó el ancho y alto de la imagen para hacerla más grande.

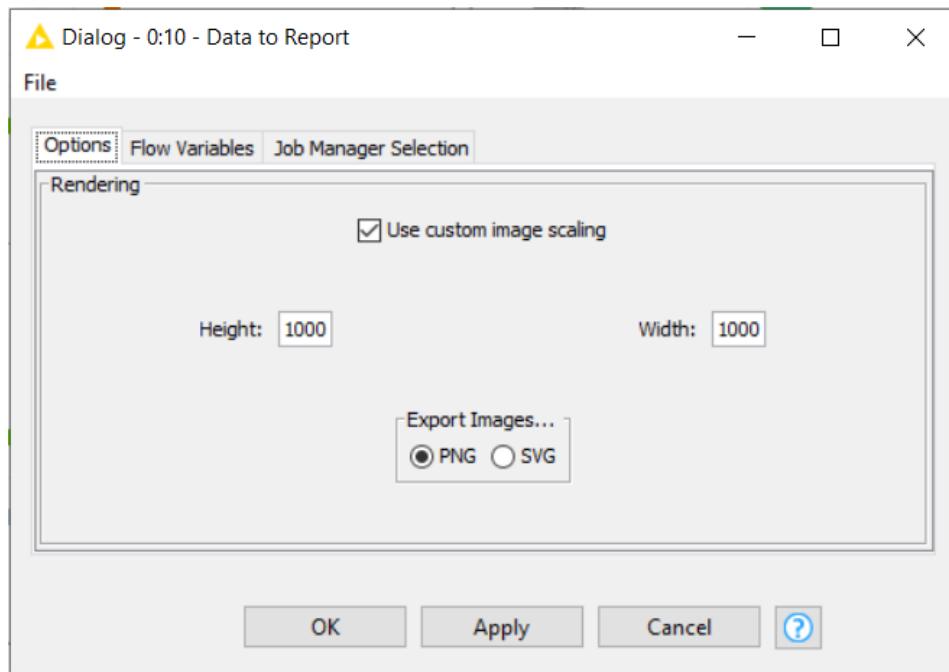


Fig. 41 Ventana de configuración Data to Report

Table View - 0:9 - Data to Report

Fig. 42 Data to report imagen de salida

## Configuración del nodo Image to Report

Configuración dejada por default.

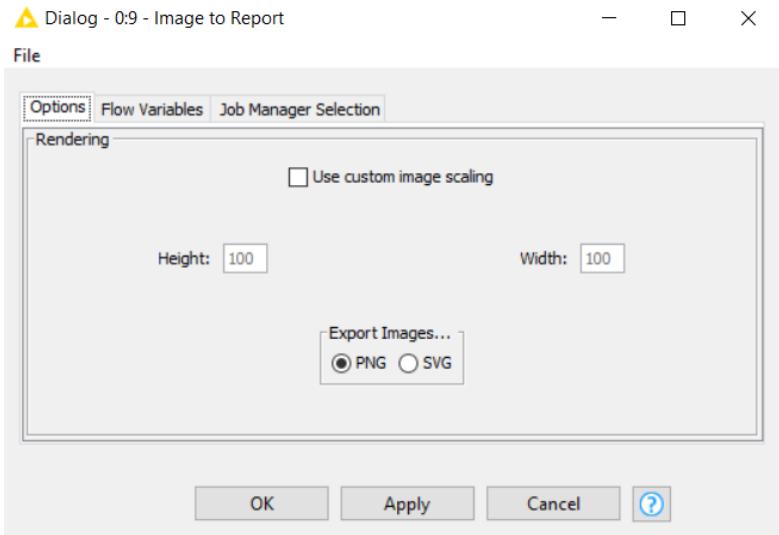


Fig. 43. Ventana de configuración Image to Report

## Configuración del nodo Decision Tree Predictor

Configuración dejada por default.

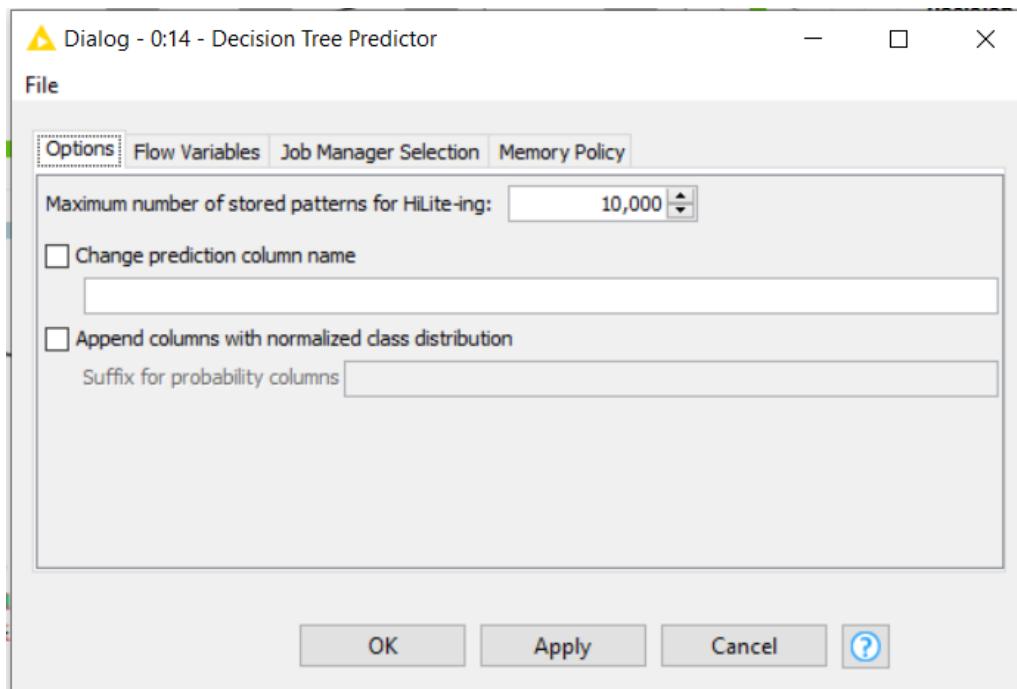


Fig. 44 Ventana de configuración Decision Tree Predictor

## Configuración del nodo Scorer

Como primera columna se elige la columna objetivo **class** (Columna 279) y en la segunda columna, la columna de predicción sobre Columna 279, que añadió el Decision Tree Learner.

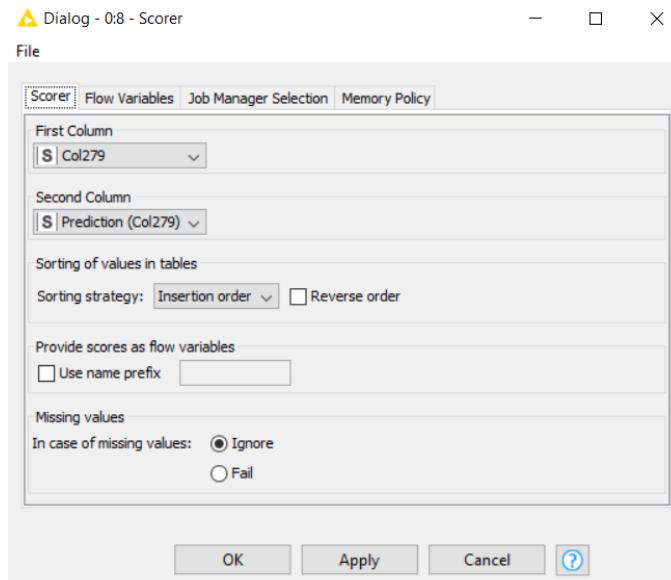


Fig. 45. Ventana de configuración Scorer

## Árbol CART



© dak

## Árbol CART

### Marco teórico

La metodología CART se basa en una destacada teoría matemática introducida en 1984 por cuatro expertos en estadística mundialmente famosos de la Universidad de Stanford y la Universidad de California en Berkeley

Del acrónimo de Classification And Regression Trees, CART, es una de las herramientas más importantes y populares en el campo de la minería de datos. Para quienes no se encuentran familiarizados con CART, en términos generales se puede definir como: un algoritmo basado en árbol que cuyo funcionamiento consiste en examinar muchas maneras de dividir localmente los datos en segmentos más pequeños, basándose en los diferentes valores y combinaciones.

Se selecciona las divisiones con mejor rendimiento, repitiendo el proceso de manera recursiva hasta que se encuentra el conjunto óptimo. Su resultado es un árbol de decisión representado por una serie de divisiones binarias que conducen a nodos terminales que pueden ser descritos por un conjunto específico de reglas.

Si analizamos la construcción del árbol binario detalladamente podemos intuir que su construcción sigue un enfoque de división binaria recursiva, en la algoritmia se le conoce como top-down greedy approach, puesto que analiza la mejor variable para la ramificación solo en el proceso de división actual.

Al mirar el árbol binario resultante es fácil interpretarlo pues tiende a ser bastante intuitivo. Un análisis a este nos puede revelar relaciones importantes existentes en nuestro conjunto, de la misma manera podremos observar de una manera mucho más clara todas aquellas reglas que nos servirán para hacer predicciones con mayor precisión.

Una vez que hemos explorado las diferentes ventajas que nos brindan los árboles de clasificación y regresión también es importante recalcar sus aplicaciones. Al ser una herramienta empleada para la minería de datos es posible afirmar que tiene uso en casi cualquier campo. Su espectro es muy grande, desde situaciones financieras como predecir las dificultades financieras de una empresa, hasta cuestiones médicas como predecir qué tan probable es que un paciente tenga cierto padecimiento.

En el trabajo presente está enfocado en este último punto. Partiendo de un conjunto de datos referente al trastorno de la frecuencia cardiaca, la arritmia, buscamos encontrar modelos predictivos para conocer si no tiene arritmia, o si es que la tiene cuál es su tipo.

Como se definió anteriormente una arritmia es cualquier trastorno en los latidos o en el ritmo del corazón. Con trastorno nos referimos a que el corazón late demasiado rápido, demasiado lento o que tiene un patrón irregular. En el primer caso hablamos de una taquicardia; en el caso en el que va más lento se le conoce como bradicardia.

A grandes rasgos se dice que existen muchos factores que pueden afectar el ritmo cardíaco, como fumar, estrés, defectos cardíacos congénitos, entre otros. Para un diagnóstico el médico

debe realizar pruebas que detecten si se padece una arritmia, es importante detectar esto a tiempo para poder tener más de una opción de tratamiento y que sea efectivo.

Por esta última razón en el documento presente buscaremos un modelo predictivo cuyo objetivo será detectar la no presencia o presencia de una arritmia, de la misma manera, si se encuentra que en efecto existe una arritmia saber a qué tipo pertenece.

## Atributos

Para el árbol CART se admiten variables de entrada y de salida nominales, ordinales y continuas. Bajo esta premisa definimos a nuestro atributo objetivo como el último, al cual se le dio el nombre de **class**, el cual hace referencia al tipo de arritmia, siendo esta nuestra variable dependiente.

Por otro lado, nuestras variables independientes serán los atributos (**QRS duration, P-R Interval, P Interval, T, P, QRST, Heart Rate, Channel D1, D2, D3, AVR, AVF, V1, V3**), de la misma manera usaremos el atributo **age**.

## Descripción del trabajo

Como se ha descrito anteriormente el objetivo principal de esta técnica es el de encontrar un modelo predictivo capaz de arrojar información acerca de la presencia de una arritmia, así como su tipo. Hay que recordar que en medicina un diagnóstico temprano puede marcar la diferencia.

Entre la clasificación de los tipos de arritmia cardiaca brindada por la descripción del conjunto de datos observamos que existen tanto arritmias medianamente inofensivas como arritmias potencialmente mortales como lo son, por ejemplo, la fibrilación o la taquicardia ventriculares.

Por este nivel de peligro antes mencionado consideramos que la creación de este tipo de herramientas puede marcar la diferencia, resultando en una opción accesible para el diagnóstico temprano, apoyando así a la medicina preventiva.

## Fragmento del diccionario de datos utilizado

#	Nombre	Significado	Tipo	Dominio
1	Age	Edad en años	Lineal	0-89
5	QRS duration	Promedio de la duración del QRS en mseg	Lineal	61-138
6	P-R interval	Duración media entre el inicio de las ondas P y Q en mseg	Lineal	0-524
9	P interval	Duración media de la onda P en mseg	Lineal	-172-169
11	T	Ángulo vectorial en grados en el plano frontal de T	Lineal	-93-170
12	P	Ángulo vectorial en grados en el plano frontal de P	Lineal	-170-180

<b>13</b>	QRST	Ángulo vectorial en grados en el plano frontal de QRST	Lineal	-170-180
<b>15</b>	Heart rate	Número de latidos del corazón por minuto	Lineal	0-88
<b>28 a 39</b>	channel DII similar (16 a 27)	...	Nominal y Lineal	0-76
<b>40 a 51</b>	channel DIII	...	Nominal y Lineal	0-116
<b>52 a 63</b>	channel AVR	...	Nominal y Lineal	0-80
<b>76 a 87</b>	channel AVF	...	Nominal y Lineal	0128
<b>88 a 99</b>	channel V1	...	Nominal y Lineal	0-216
<b>100 a 111</b>	channel V2	...	Nominal y Lineal	0-108
<b>112 a 123</b>	channel V3	...	Nominal y Lineal	0-132
<b>280</b>	class	Clasificación de la arritmia	Nominal	[1,16]

Tabla 4. Fragmento diccionario de datos para CART

## Árbol CART

Al ser el primer elemento elaborado en un software diferente a Knime el primer paso fue hacer la imputación ahora en el software Orange en su versión 3.28. En el caso de Orange se debe seleccionar desde el nodo de carga del conjunto de datos cual será nuestro atributo objetivo, así como los atributos independientes.

Cabe mencionar que el conjunto de datos original no contaba con nombre en las columnas, como el conjunto se iba a manipular en muchas ocasiones se optó por antes de iniciar con la herramienta Orange se agregará una nueva fila con los identificadores de los atributos.

Retomando el trabajo realizado en la herramienta Orange es importante decir que el método de imputación que se uso fue con el promedio/más frecuente. La configuración de este nodo se podrá observar en los anexos de esta sección.

Con los dos procesos anteriores obtuvimos un conjunto de datos bien identificado y sin datos faltantes como podremos observar en la Fig. 46.

The figure shows two side-by-side data tables in the Orange data mining interface. Both tables have 452 instances and 13 features. The left table, labeled 'Data Table', shows a row with a missing value in the 'Heart rate' column (row 5, value '?'). The right table, labeled 'Data Table (1)', shows the same data after imputation, where the missing value has been replaced by the mean or mode (row 5, value '74.46'). The configuration panel on the left shows settings like 'Show variable labels (if present)', 'Color by instance classes', and 'Select full rows'.

Fig. 46 Conjunto de datos antes y después de la imputación

En la Fig. 46 la ventana correspondiente al nodo Data Table tiene datos faltantes, esto lo podemos observar en la fila 5 en el atributo Heart Rate. Por otro lado, En la ventana correspondiente al nodo Data Table (1) en la fila 5 en el atributo Heart Rate ahora tenemos un valor, correspondiente a el promedio/más frecuente.

El paso siguiente fue realizar la partición de nuestro conjunto de datos. Esto con el fin de tener una porción para el entrenamiento y otro para las pruebas del modelo resultante. En el caso particular de Orange, el nodo que cumple con este objetivo es aquel con el nombre Data Sampler. Este implementa varios métodos de muestreo yendo desde un muestreo simple hasta el método Bootstrap.

La salida de este nodo es un conjunto de datos muestreado y un conjunto complementario (con instancias del conjunto de entrada que no están incluidas en el conjunto de datos muestreado). Se hicieron pruebas de los diferentes muestreos para comprobar sus resultados, siendo Bootstrap el que mejor desempeño entregó a la hora de la evaluación del modelo.

Conectado al nodo Data Sampler tenemos los nodos Tree y Predictions, donde cumplen con la función de generar nuestro arbol binario y de probar el modelo respectivamente. El ultimo nodo empleado únicamente permite la visualización del arbol del modelo obtenido.

## Diagrama general generado por la herramienta

En la Fig. 47 podemos ver el arbol resultante. Consta de 97 nodos, 49 hojas y 13 niveles. En la raíz podemos encontrar al atributo AVR. Considerando el funcionamiento del algoritmo CART que se explicó en su sección correspondiente, o sea que es un algoritmo del estilo voraz, asumimos que el atributo AVR fue el primer atributo que el algoritmo consideró más relevante.

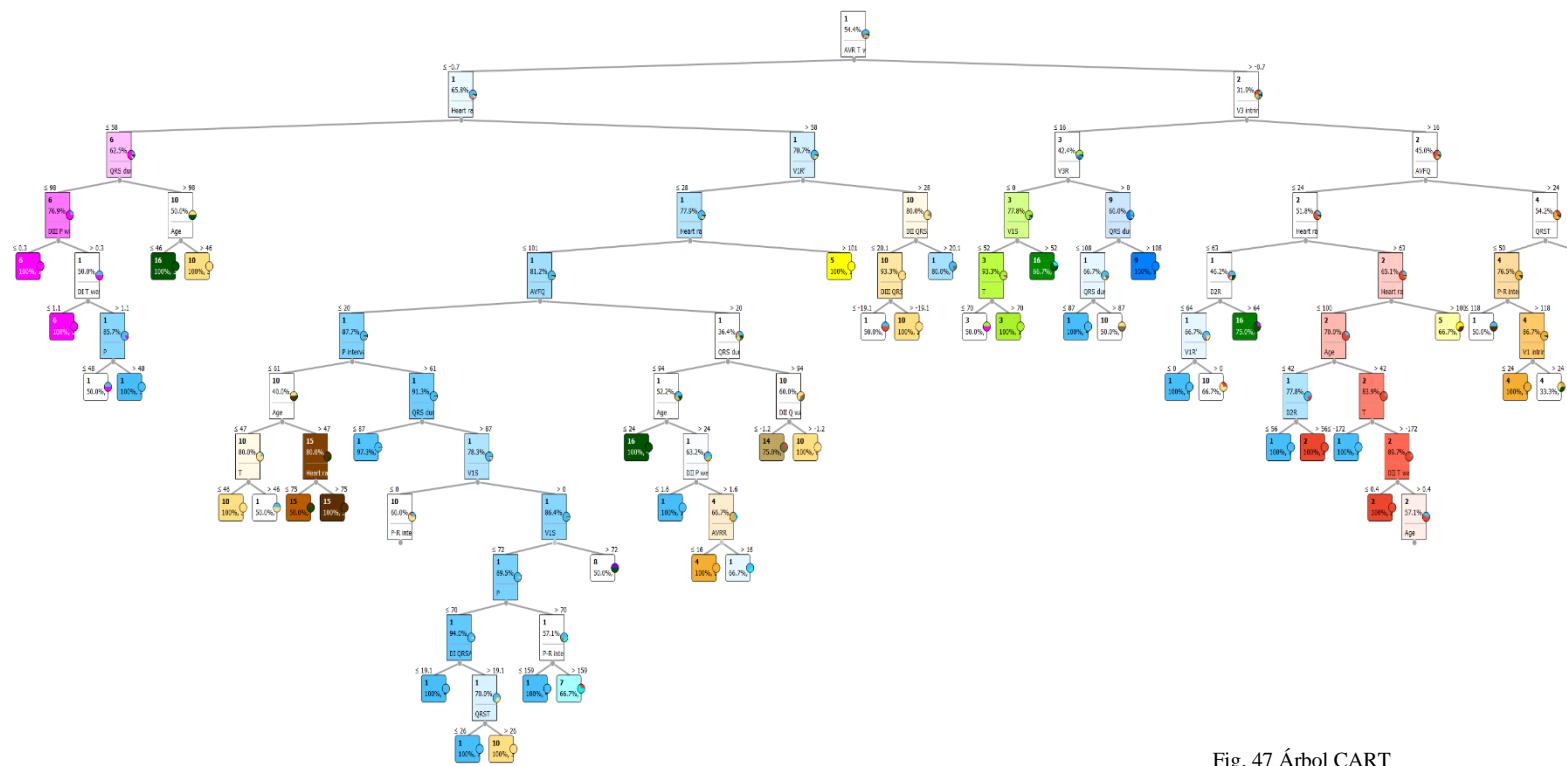


Fig. 47 Árbol CART

## Medidas

Existen muchas formas de medir el desempeño de nuestros modelos. En esta sección enlistaremos y explicaremos cuáles fueron los parámetros importantes para los árboles de regresión y clasificación, con el fin de que en la sección siguiente sea posible desplegar resultados y realizar una evaluación efectiva al modelo.

AUC

Sus siglas significan Area Under the Curve, o área bajo la curva. Es un concepto muy resonado en unidades de aprendizaje como Calculo. En general el AUC mide toda el área bidimensional por debajo de la curva, proporcionando una medición del rendimiento en los umbrales de clasificación. En una de sus muchas definiciones tenemos que representar la probabilidad de que el modelo clasifique un ejemplo positivo aleatorio más algo que un ejemplo negativo aleatorio. El valor de este indicador va del 0 al 1. Si por ejemplo tenemos un modelo que predice todo incorrectamente entonces tiene un AUC de 0.0, si en caso contrario tenemos un modelo que clasifica todo correctamente entonces su valor de AUC será de 1.0.

Es un indicador conveniente puesto que no varía con respecto a la escala, es decir, siempre mide que tan bien se clasifican las predicciones, en lugar de sus valores absolutos. Por otro lado, también se dice que el AUC es invariables con respecto al umbral de clasificación. En sí, mide la calidad de las predicciones del modelo, sin tener en cuenta que umbral de clasificación se elige.

## CA

El parámetro CA refiere al puntaje de clasificación de precisión, se refiere a la dispersión del conjunto de valores obtenidos. Está relacionado con el sesgo de una estimación. Cuanto menos es el sesgo más exacto es una predicción, enfocado a números, entra más cercano sea el valor a 1 podemos concluir que el modelo predictivo es mejor.

En general indica la proximidad de la media de una serie de datos al valor que se acepta como verdadero. Es decir, la proporción de ejemplos correctamente clasificados. Una manera más sencilla de entenderlo es que es un puntaje de clasificación de precisión.

La precisión también indica la reproducibilidad de los resultados y puede definirse como la concordancia entre los valores de dos o más medidas obtenidas de la misma manera y para la misma muestra.

## F1

También conocida como puntuación F equilibrada o medida F, la puntuación F1 se puede interpretar como un promedio ponderado de la precisión y la recuperación, donde una puntuación F1 alcanza su mejor valor en 1 y la peor puntuación en 0. La contribución relativa de precisión y recuperación a la puntuación F1 son iguales. La fórmula para la puntuación F1 es:

$$F1 = 2 * (\text{precision} * \text{recall}) / (\text{precision} + \text{recall})$$

En el caso de varias clases y varias etiquetas, este es el promedio de la puntuación F1 de cada clase con una ponderación en función del promedio.

## PRECISIÓN

La precisión es la razón donde está el número de verdaderos positivos y el número de falsos positivos. La precisión es intuitivamente la capacidad del clasificador de no etiquetar como positiva una muestra que es negativa.

$$\text{tp} / (\text{tp} + \text{fp})$$

El mejor valor es 1 y el peor valor es 0.

## RECALL

La recuperación o recall es la proporción donde está el número de verdaderos positivos y el número de falsos negativos. La recuperación o recall es intuitivamente la capacidad del clasificador de encontrar todas las muestras positivas.

$$\text{tp} / (\text{tp} + \text{fn})$$

El mejor valor es 1 y el peor valor es 0.

## Evaluación de resultados

En la Fig. 48 podemos observar el resultado del nodo Predictions, cuya entrada es la parte del conjunto de datos destinada para las pruebas y el modelo generado en el nodo Tree. En la parte baja de la imagen podemos ver los resultados de la evaluación de la modelo basada en los parámetros previamente enlistados.

Predictions (1)														
Tree	Feature 1	Age	QRS duration	P-R interval	P interval	T	P	QRST	Heart rate	D2R	D3Q	AVRR		
1	0.00 - 6	6	56	81	174	39	37	-17	31	53	64	32	0	0
2	0.00 - 6	6	56	81	174	39	37	-17	31	53	64	32	0	0
3	0.00 - 6	6	56	81	174	39	37	-17	31	53	64	32	0	0
4	0.00 - 1	1	55	100	202	143	11	-5	20	71	64	20	32	0
5	0.00 - 1	7	75	88	181	103	13	61	3	74.46	40	52	28	44
6	0.00 - 14	14	13	100	167	91	66	52	88	84	44	24	36	24
7	0.00 - 1	1	44	84	118	63	69	78	66	64	44	0	12	0
8	0.00 - 1	1	44	84	118	63	69	78	66	64	44	0	12	0
9	0.00 - 4	10	54	78	155	81	42	41	-13	73	44	80	36	72
10	0.00 - 6	6	30	91	180	104	51	60	63	56	48	36	0	32
11	0.00 - 1	1	44	77	158	94	20	45	40	72	72	24	0	0
12	0.00 - 1	1	44	77	158	94	20	45	40	72	72	24	0	0
13	0.00 - 1	10	47	82	145	61	75	77	75	67	52	0	32	0
14	0.00 - 1	1	46	70	120	52	49	-2	54	70	48	0	16	0
15	0.00 - 1	1	28	83	251	183	39	46	43	76	44	0	24	0
16	0.00 - 1	1	28	83	251	183	39	46	43	76	44	0	24	0
17	0.00 - 1	1	45	90	122	78	78	67	80	66	52	0	24	0
18	0.00 - 1	1	45	90	122	78	78	67	80	66	52	0	24	0
19	0.00 - 1	14	34	94	186	125	52	60	77	83	48	24	12	20
20	0.00 - 10	10	31	95	161	83	48	39	30	67	48	0	48	0
21	0.00 - 10	10	31	95	161	83	48	39	30	67	48	0	48	0
22	0.00 - 2	2	56	90	164	99	153	41	0	79	60	0	0	0
23	0.00 - 1	1	50	75	125	63	32	73	35	93	64	0	0	0
24	0.00 - 1	1	50	75	125	63	32	73	35	93	64	0	0	0
25	0.00 - 4	4	69	82	145	101	46	71	47	80	60	36	16	28
26	0.00 - 4	4	69	82	145	101	46	71	47	80	60	36	16	28

Model AUC CA F1 Precision Recall Specificity

Tree 0.995 0.949 0.947 0.949 0.949 0.970

452 1

Fig. 48 Resultado nodo de predicción

A primera vista observamos que hay algunos valores mal clasificados. Un ejemplo concreto lo podemos ver en la fila 9 donde el valor real es 10 y el modelo arroja un cuatro. Esto tiene un sustento y es ninguno de nuestros parámetros es 1 cerrado. Recordemos que en las medidas anteriores uno siempre es el valor máximo, indicando que el modelo clasifica de manera correcta todo.

Este uno es un valor sino imposible muy difícil de alcanzar, pero, entre más cerca se esté al mismo podemos afirmar que nuestro modelo es mejor. Avanzando uno por uno los valores obtenidos empezamos con AUC.

Sabemos que, si por ejemplo tenemos un modelo que predice todo incorrectamente entonces el valor del área bajo la curva será de 0.0, si en caso contrario tenemos un modelo que clasifica todo correctamente entonces su valor de AUC será de 1.0. En el caso particular de nuestro modelo tenemos que el valor es AUC es de 0.995, un valor demasiado cercano a 1. Por este valor podemos concluir que, aunque el modelo no clasifica todos los valores correctamente si lo hace en su gran mayoría.

Continuando con el parámetro CA. Hay que recordar que CA refiere al puntaje de clasificación de precisión. De la misma manera, entre más cercano sea a 1 el valor podemos decir que el modelo es más o menos efectivo. En el caso particular de nuestro modelo tenemos que el valor obtenido es de 0.949. Aunque evidentemente  $1 > 0.949$  el valor si es muy próximo al valor ideal.

Con respecto al valor que tiene  $f1$  hay que recordar en primera instancia que este parámetro en específico es un promedio ponderado de la precisión y la recuperación. Comparado con los dos elementos anteriores entre más cercano sea el valor a uno el desempeño del modelo es mejor. Para este parámetro tenemos un valor de 0.947, valor muy cercano a 1. Se concluye que, aunque no es el mejor modelo, es un modelo bastante competente.

Finalmente, en precisión y en recall tenemos en ambos el valor de 0.949. Como en todos los casos anteriores, entre más cercano sea el valor a 1 asumimos que el modelo cumple mejor con su propósito.

Con el análisis anterior sabemos que a grandes rasgos el modelo generado es eficiente, cumple con su propósito. Es capaz de clasificar de manera correcta, en la mayoría de sus casos, el atributo objetivo.

## Anexos

En la Fig. 49 podemos observar el flujo de trabajo empleado para el arbol CART.

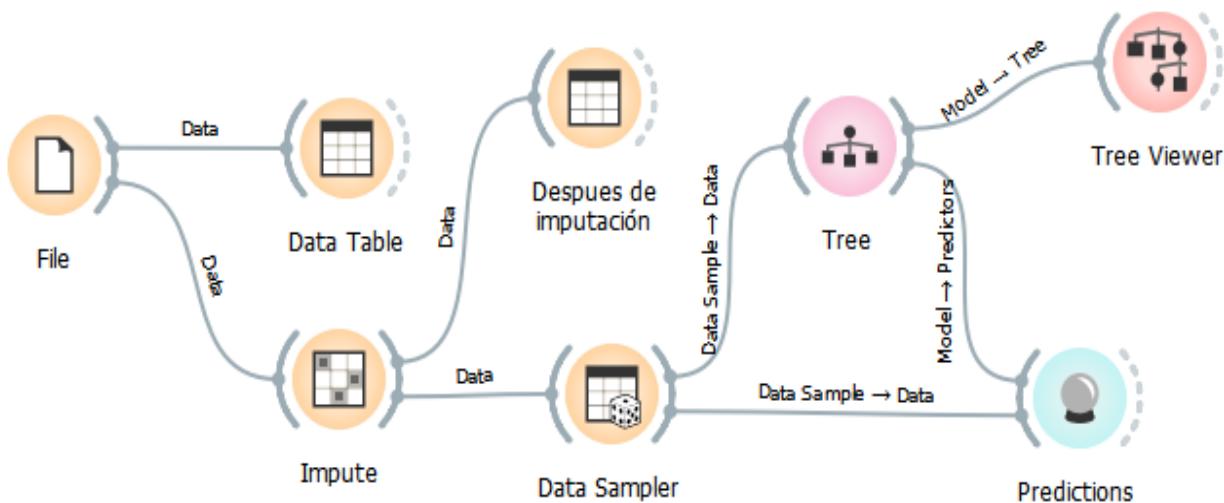


Fig. 49 Workflow CART en Orange

## Configuración del nodo File

En el caso del software Orange, desde que se carga el conjunto de datos es necesario seleccionar que atributos se usarán. En la configuración del nodo también es posible cambiar el tipo de cada atributo, así como definir cuál será el atributo objetivo del flujo de trabajo.

Para el caso particular de este desarrollo se eligieron los atributos preseleccionados y se estableció que la clase cuyo identificador es **CLASS** fuera el atributo objetivo poniendo la opción de este como objetivo, en el caso de los demás atributos según si eran parte de la lista de atributos independientes a usar se dejó como feature. En caso de ser un atributo que no se usaría se estableció la opción en skip.

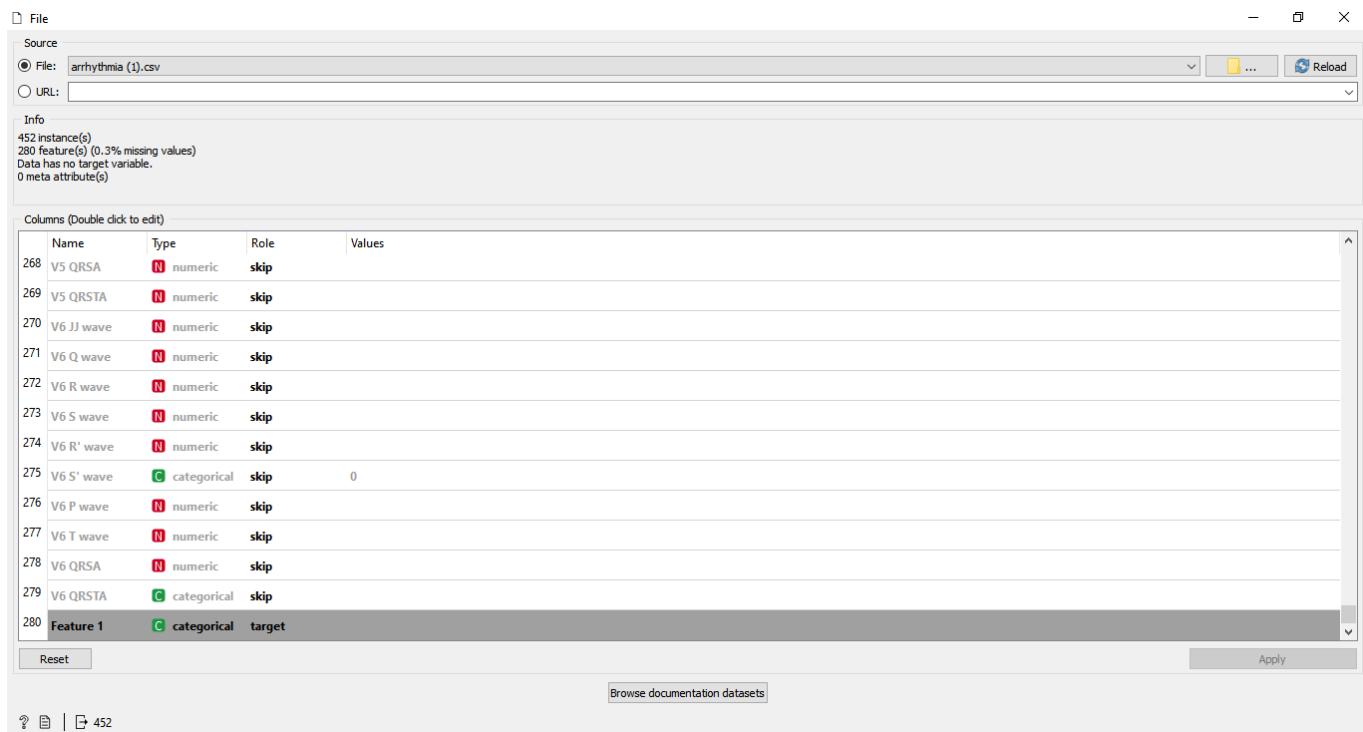


Fig. 50 Configuración nodo File

## Configuración del nodo Impute

El propósito de este nodo es el de, como lo dice su nombre, realizar el proceso de imputación. Como se verá en la Fig. 51 para la imputación se tenían varias opciones como lo son la imputación con el promedio o con el más frecuente, también tenemos la opción de que sea con un valor distinto, o con un valor que nosotros como usuario ingresemos. También se nos daba la opción de eliminar todas las filas en las que existiera un dato faltante.

Esta última se descartó puesto que no nos resultó razonable eliminar todo un registro. Las opciones en las que la imputación se realizaba de manera aleatoria o en las que el usuario ingresaba un valor específico también se descartaron puesto que, para el propósito del ejercicio podría ensuciar los datos. Finalmente se optó por la imputación por medio del promedio/valor más frecuente.

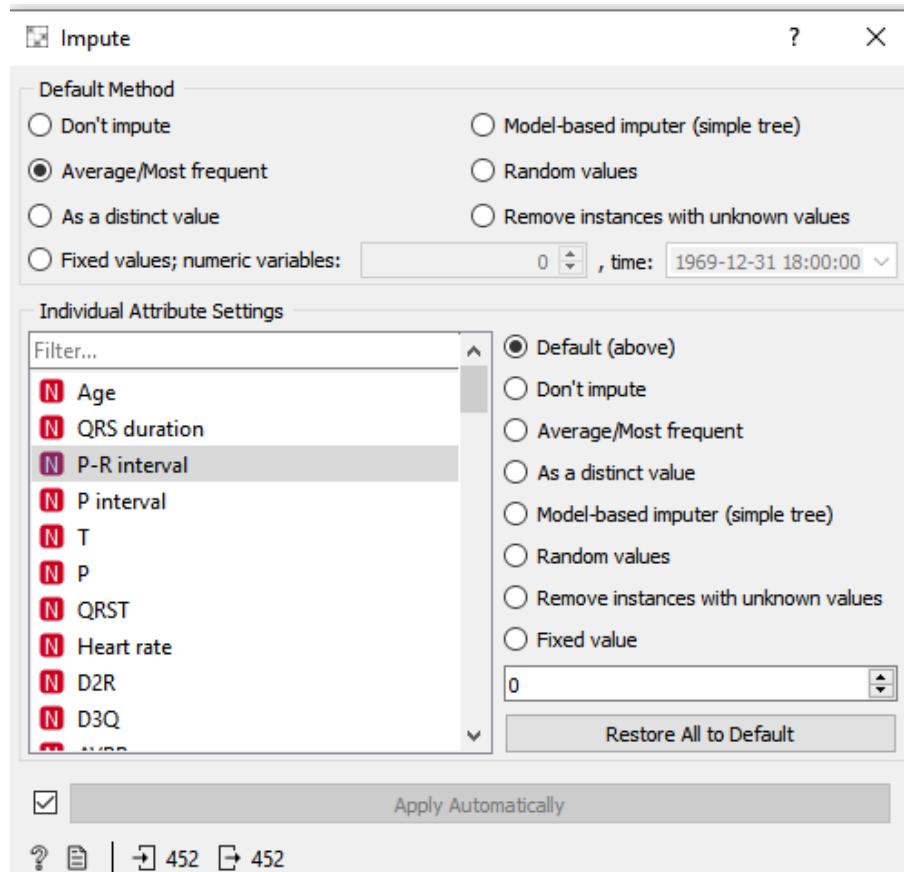
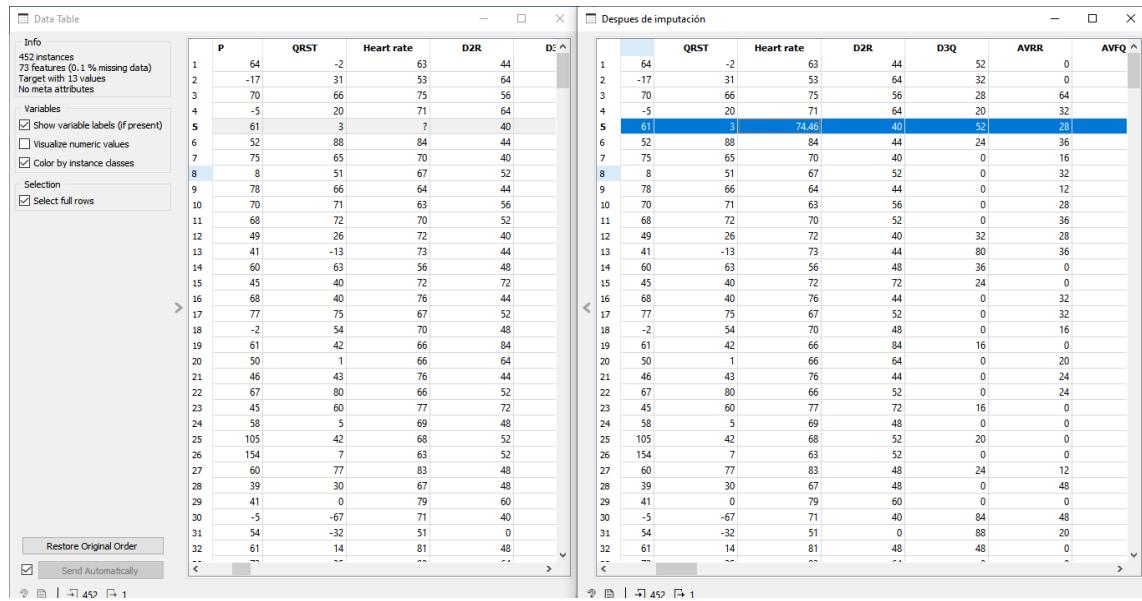


Fig. 51 Configuración nodo Impute

## Configuración del nodo Data Table

Notemos que en todo el flujo de trabajo contamos con dos Data Table. Este nodo en particular es meramente informativo, su función es el mostrar el conjunto de datos en una tabla. El propósito de su uso fue el verificar que la imputación se había hecho de manera correcta. Por sí mismo, el nodo no tiene una configuración puesto que, como se mencionó, únicamente es para desplegar los datos. En la Fig. 52 veremos el resultado o la salida de estos nodos.



The figure displays two Data Table nodes side-by-side. The left node is titled 'Data Table' and shows the original dataset with 452 instances and 73 features. The right node is titled 'Después de imputación' and shows the dataset after imputation, with the 61st row highlighted in blue. Both nodes have a header row with columns: P, QRST, Heart rate, D2R, D3Q, AVRR, and AVFQ.

P	QRST	Heart rate	D2R	D3Q	AVRR	AVFQ
1	64	-2	63	44	52	0
2	-17	31	53	64	32	0
3	70	66	75	56	28	64
4	-5	20	71	64	20	32
5	61	3	?	40	52	28
6	52	88	84	44	24	36
7	75	65	70	40	0	16
8	8	51	67	52	0	32
9	78	66	64	44	0	12
10	70	71	63	56	0	28
11	68	72	70	52	0	36
12	49	26	72	40	32	28
13	41	-13	73	44	80	36
14	60	63	56	48	36	0
15	45	40	72	72	24	0
16	68	40	76	44	0	32
17	77	75	67	52	0	32
18	-2	54	70	48	0	16
19	61	42	66	84	16	0
20	50	1	66	64	0	20
21	46	43	76	44	0	24
22	67	80	66	52	0	24
23	45	60	77	72	16	0
24	58	5	69	48	0	0
25	105	42	68	52	20	0
26	154	7	63	52	0	0
27	60	77	83	48	24	12
28	39	30	67	48	0	48
29	41	0	79	60	0	0
30	-5	-67	71	40	84	48
31	54	-32	51	0	88	20
32	61	14	81	48	48	0
...	...	...	...	...	...	...

Fig. 52 Salida nodos Data Table

## Configuración del nodo Data Sampler

Este es el nodo encargado de seleccionar un subconjunto de instancias de datos de un conjunto de datos de entrada. Por tanto, tiene como entrada nuestro conjunto de datos. A la salida podemos encontrar las instancias de datos muestreadas y los datos fuera de la muestra. Siendo entonces una de estas instancias orientada al entrenamiento del modelo y la otra a las pruebas.

El nodo Data Sampler implementa varios métodos de muestreo de datos. Devuelve un conjunto de datos muestreado y un conjunto complementario (con instancias del conjunto de entrada que no están incluidas en el conjunto de datos muestreado).

En este caso se eligió este método porque resultó ser el que dio mejores resultados para el entrenamiento del árbol deseado. En la Fig. 53 podremos observar a detalle la ventana de configuración de este nodo.

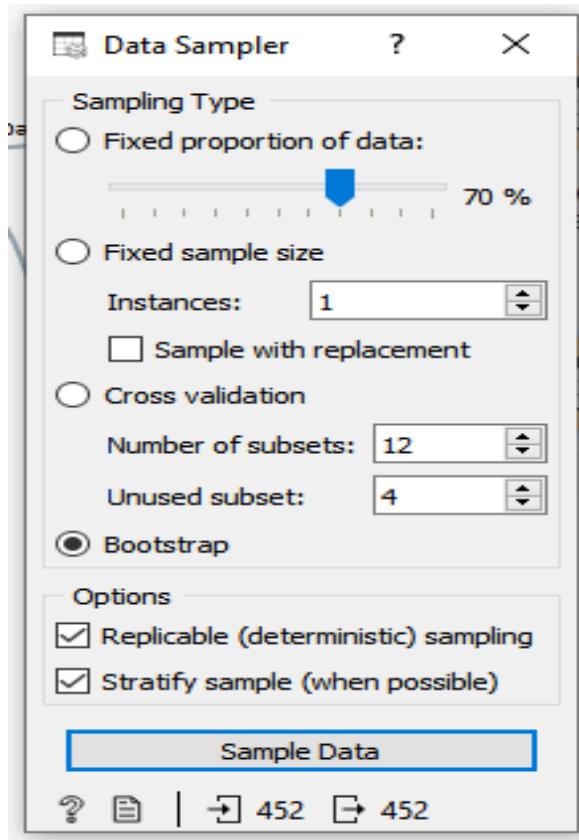


Fig. 53 Configuración Data Sampler

## Configuración del nodo Tree

Para la configuración de este nodo se tomaron en consideración las características del árbol CART, es decir, como salida de este nodo debíamos obtener un árbol binario, es por esto por lo que los campos se encuentran marcados, de la misma manera en esta sección se pueden controlar diferentes aspectos como que tan profundo queremos que sea el árbol resultante. En la Fig. 54 veremos la configuración a detalle de este nodo, en general casi todo se dejó el valor por default.

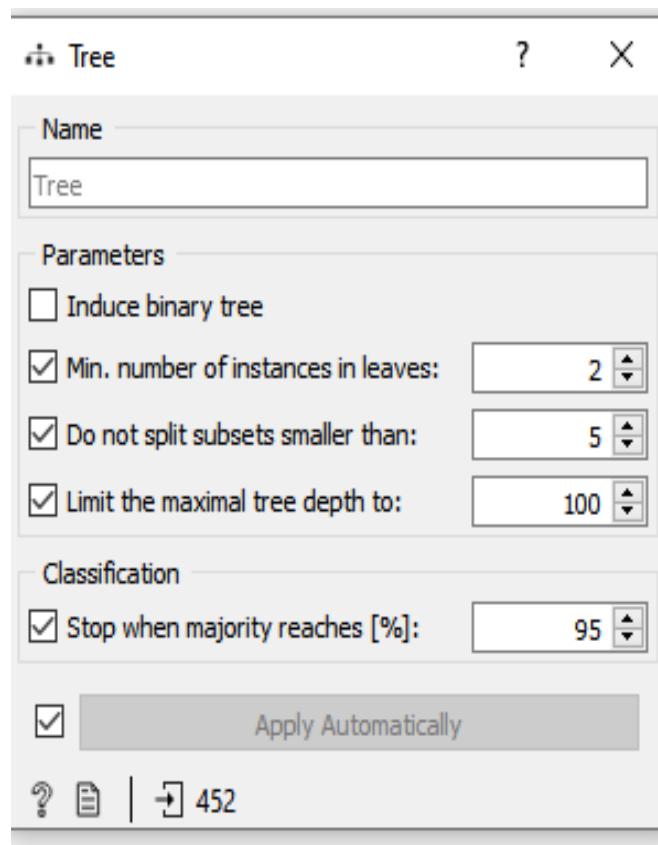


Fig. 54 Configuración nodo Tree

## Configuración del nodo Tree Viewer

Este nodo no necesita ninguna configuración puesto que solo es ilustrativo. En el podremos ver el arbol generado del nodo que lo precede. Una vez que damos doble clic sobre este se desplegará una ventana donde veremos el modelo, también tendremos una sección para ver información relacionada con el arbol e igual las opciones para guardar dicha imagen. En la Fig. 55 podremos ver las secciones antes mencionadas.

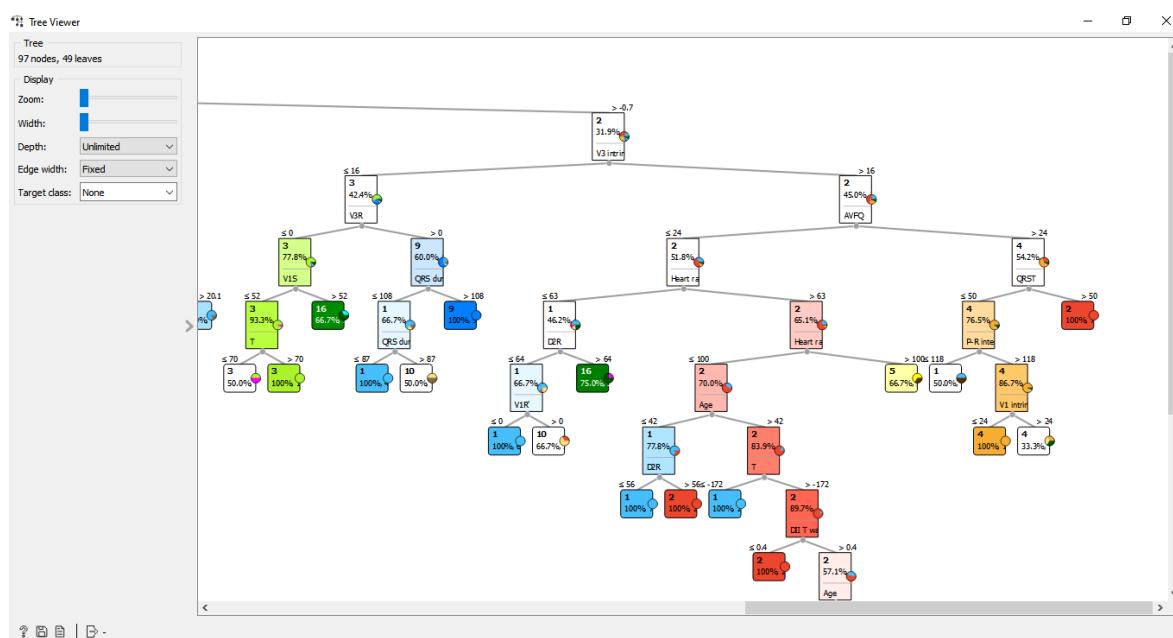


Fig. 55 Salida Tree Viewer

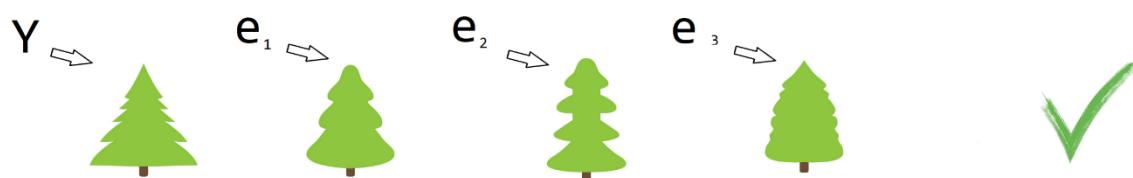
## Configuración del nodo Predictions

El nodo Predictions no tiene una configuración. Su salida depende de la entrada del nodo. En general, su función es la de permitir visualizar los resultados de nuestro modelo. Tiene como entrada la porción del conjunto orientado a las pruebas y el modelo que se probara con estos datos. En la primera columna veremos el valor predicho, la segunda el valor real y las columnas siguientes son los atributos usados para la predicción. En la Fig. 56 veremos las secciones antes mencionadas.

Predictions																
	Tree	Feature 1	Age	QRS duration	P-R interval	P interval	T	P	QRST	Heart rate	D2R	D3Q	AVRR			
1	0.00 - 6	6	56	81	174	39	37	-17	31	53	64	32	0	0	0	0
2	0.00 - 6	6	56	81	174	39	37	-17	31	53	64	32	0	0	0	0
3	0.00 - 6	6	56	81	174	39	37	-17	31	53	64	32	0	0	0	0
4	0.00 - 1	1	55	100	202	143	11	-5	20	71	64	20	32	0	0	0
5	0.00 - 1	7	75	88	181	103	13	61	3	74.46	40	52	28	44	44	44
6	0.00 - 14	14	13	100	167	91	66	52	88	84	44	24	36	24	36	24
7	0.00 - 1	1	44	84	118	63	69	78	66	64	44	0	12	0	0	0
8	0.00 - 1	1	44	84	118	63	69	78	66	64	44	0	12	0	0	0
9	0.00 - 4	10	54	78	155	81	42	41	-13	73	44	80	36	72	36	72
10	0.00 - 6	6	30	91	180	104	51	60	63	56	48	36	0	32	0	32
11	0.00 - 1	1	44	77	158	94	20	45	40	72	72	24	0	0	0	0
12	0.00 - 1	1	44	77	158	94	20	45	40	72	72	24	0	0	0	0
13	0.00 - 1	10	47	82	145	61	75	77	75	67	52	0	32	0	0	0
14	0.00 - 1	1	46	70	120	52	49	-2	54	70	48	0	16	0	0	0
15	0.00 - 1	1	28	83	251	183	39	46	43	76	44	0	24	0	0	0
16	0.00 - 1	1	28	83	251	183	39	46	43	76	44	0	24	0	0	0
17	0.00 - 1	1	45	90	122	78	78	67	80	66	52	0	24	0	0	0
18	0.00 - 1	1	45	90	122	78	78	67	80	66	52	0	24	0	0	0
19	0.00 - 1	14	34	94	186	125	52	60	77	83	48	24	12	20	12	20
20	0.00 - 10	10	31	95	161	83	48	39	30	67	48	0	48	0	0	0
21	0.00 - 10	10	31	95	161	83	48	39	30	67	48	0	48	0	0	0
22	0.00 - 2	2	56	90	164	99	153	41	0	79	60	0	0	0	0	0
23	0.00 - 1	1	50	75	125	63	32	73	35	93	64	0	0	0	0	0
24	0.00 - 1	1	50	75	125	63	32	73	35	93	64	0	0	0	0	0
25	0.00 - 4	4	69	82	145	101	46	71	47	80	60	36	16	28	16	28
26	0.00 - 4	4	69	82	145	101	46	71	47	80	60	36	16	28	16	28

Fig. 56 Salida nodo Predictions

## Boosting



## Boosting

### Marco teórico

Los clasificadores de aumento de gradiente son tipos específicos de algoritmos que se utilizan para tareas de clasificación, como sugiere su nombre.

Las características son las entradas que se le dan al algoritmo de aprendizaje automático, las entradas que se utilizarán para calcular un valor de salida. En un sentido matemático, las características del conjunto de datos son las variables que se utilizan para resolver la ecuación. La otra parte de la ecuación es la etiqueta o el objetivo, que son las clases en las que se categorizarán las instancias. Debido a que las etiquetas contienen los valores objetivo para el clasificador de aprendizaje automático, al entrenar un clasificador, debe dividir los datos en conjuntos de entrenamiento y prueba. El conjunto de entrenamiento tendrá objetivos/etiquetas, mientras que el conjunto de prueba no contendrá estos valores.

La idea detrás del "aumento de gradiente" es tomar una hipótesis débil o un algoritmo de aprendizaje débil y hacer una serie de ajustes que mejorarán la solidez de la hipótesis / aprendiz. Este tipo de Impulso de Hipótesis se basa en la idea de Probabilidad Aproximadamente Aprendizaje Correcto (PAC).

Este método de aprendizaje PAC investiga problemas de aprendizaje automático para interpretar su complejidad, y se aplica un método similar al refuerzo de hipótesis.

En el refuerzo de hipótesis, observa todas las observaciones en las que se entrena el algoritmo de aprendizaje automático y deja solo las observaciones que el método de aprendizaje automático clasificó con éxito, eliminando las otras observaciones. Se crea un nuevo alumno débil y se prueba en el conjunto de datos que se clasificaron de manera deficiente, y luego solo se conservan los ejemplos que se clasificaron correctamente.

Esta idea se realizó en el algoritmo Adaptive Boosting (AdaBoost). Para AdaBoost, muchos aprendices débiles se crean inicializando muchos algoritmos de árbol de decisión que solo tienen una sola división, como el "muñón" en la imagen de abajo.

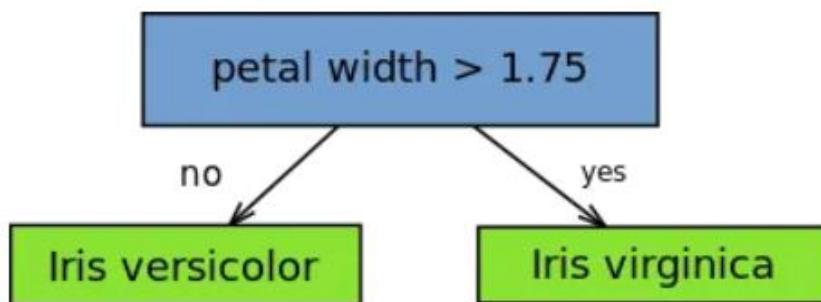


Fig. 57 Árbol decisión con una división.

El algoritmo pondera las instancias/observaciones en el conjunto de entrenamiento y se asigna más peso a las instancias que son difíciles de clasificar. Los aprendices más débiles se agregan al sistema de forma secuencial y se asignan a las instancias de capacitación más difíciles.

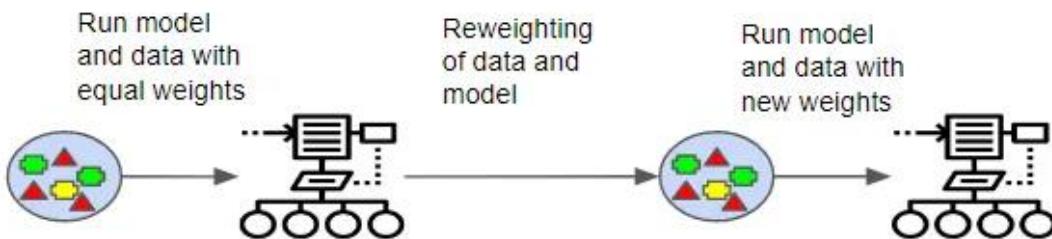


Fig. 58 Funcionamiento del modelo y datos.

## Descripción del trabajo

Si ha pasado algún tiempo en el aprendizaje automático y la ciencia de datos, definitivamente se habría encontrado con una distribución de clases desequilibrada. Este es un escenario en el que el número de observaciones que pertenecen a una clase es significativamente menor que las que pertenecen a las otras clases.

Este problema es predominante en escenarios donde la detección de anomalías es crucial como robo de electricidad, transacciones fraudulentas en bancos, identificación de enfermedades raras, etc. En esta situación, el modelo predictivo desarrollado utilizando algoritmos de aprendizaje automático convencionales podría estar sesgado e inexacto.

Esto sucede porque los algoritmos de aprendizaje automático generalmente están diseñados para mejorar la precisión al reducir el error. Por lo tanto, no tienen en cuenta la distribución / proporción de clases o el equilibrio de clases.

Tal como se acaba de describir, y por consecuencia de los resultados obtenidos en el algoritmo ID3 que dieron una baja precisión y alto margen de error, ocasionado por el desbalance de clases en el conjunto de datos, se optará por esta técnica de **Boosting** en específico **Gradient Boost**, con el objetivo de mejorar la precisión y los resultados obtenidos anteriormente, mediante los modelos de Gradient Boost.

En Gradient Boosting, muchos modelos se entran secuencialmente. Es un algoritmo de optimización numérico donde cada modelo minimiza la función de pérdida,  $y = ax + b + e$ , utilizando el Método de descenso de gradiente.

Los árboles de decisión se utilizan como aprendices débiles en Gradient Boosting.

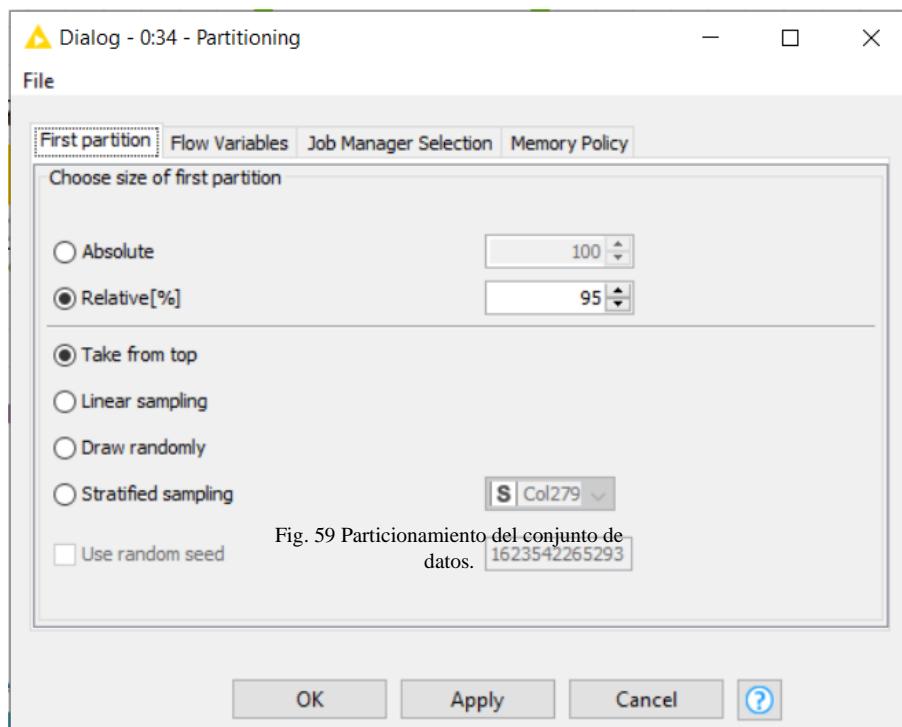
## Fragmento del diccionario de datos utilizado

22	Existence of ragged R wave	Existencia de onda R irregular	Nominal	0-1
23	Existence of diphasic derivation of R wave	Existencia de derivación difásica de la onda R	Nominal	0-1
24	Existence of ragged P wave	Existencia de onda P irregular	Nominal	0-1
25	Existence of diphasic derivation of P wave	Existencia de derivación difásica de la onda P	Nominal	0-1
26	Existence of ragged T wave	Existencia de onda T irregular	Nominal	0-1
27	Existence of diphasic derivation of T wave, nominal	Existencia de derivación difásica de la onda T	Nominal	0-1
33 a 39	channel DII similar (33 a 39)	...	Nominal	0-76
45 a 51	channel DIII	...	Nominal	0-116
57 a 63	channel AVR	...	Nominal	0-80
69 a 75	channel AVL	...	Nominal	0-148
81 a 87	channel AVF	...	Nominal	0128
93 a 99	channel V1	...	Nominal	0-216
105 a 111	channel V2	...	Nominal	0-108
117 a 123	channel V3	...	Nominal	0-132
129 a 135	channel V4	...	Nominal	0-92
141 a 147	channel V5	...	Nominal	0-136
153 a 159	channel V6	...	Nominal	0-148
280	class	Clasificación de la arritmia	Nominal	[1,16]

Tabla 5. Fragmento de diccionario de datos para Boosting

## Gradient Boost

La tabla de entrada se divide en dos particiones (es decir, en filas), por ejemplo. entrenar y probar datos. Las dos particiones están disponibles en los dos puertos de salida. Para la fase de entrenamiento se tomó una muestra tomada desde arriba, este modo coloca las filas más altas en la primera tabla de salida y el resto en la segunda tabla, con porcentaje relativo del 95%, así como se muestra en la Fig. 48, por otro lado, para la fase de prueba, fue usado el 5% restante configurado similarmente. Su desarrollo fue en la herramienta de software KNIME Analytics Platform.



## Diagrama general generado por la herramienta

En la Fig. 49 se observa el primer árbol de decisiones de los 100 modelos generados.

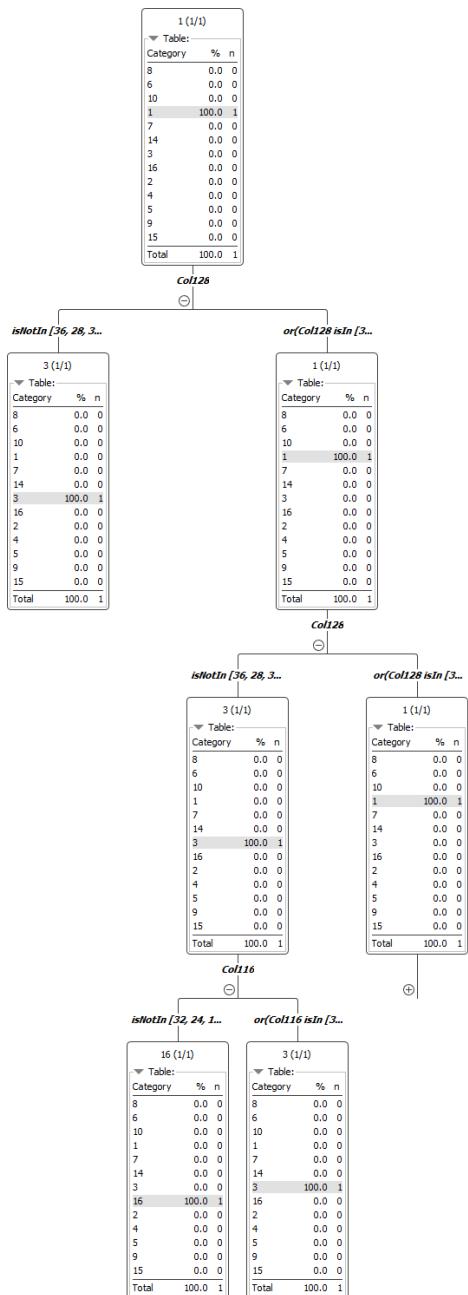


Fig. 60 Primer modelo de los árboles de decisiones generados por el KNIME

## Medidas

### Matriz de confusión

La matriz de confusión es una forma tabular de visualizar el rendimiento de su modelo de predicción. Cada entrada en una matriz de confusión denota el número de predicciones realizadas por el modelo donde clasificó las clases correctas o incorrectamente. Cualquier persona que ya esté familiarizada con la matriz de confusión sabe que la mayoría de las veces se explica para un problema de clasificación binaria. En este caso se tiene una matriz de confusión de clase múltiple. A diferencia de la clasificación binaria, no hay clases positivas o negativas aquí, en nuestro caso son los diferentes tipos de arritmia a clasificar.

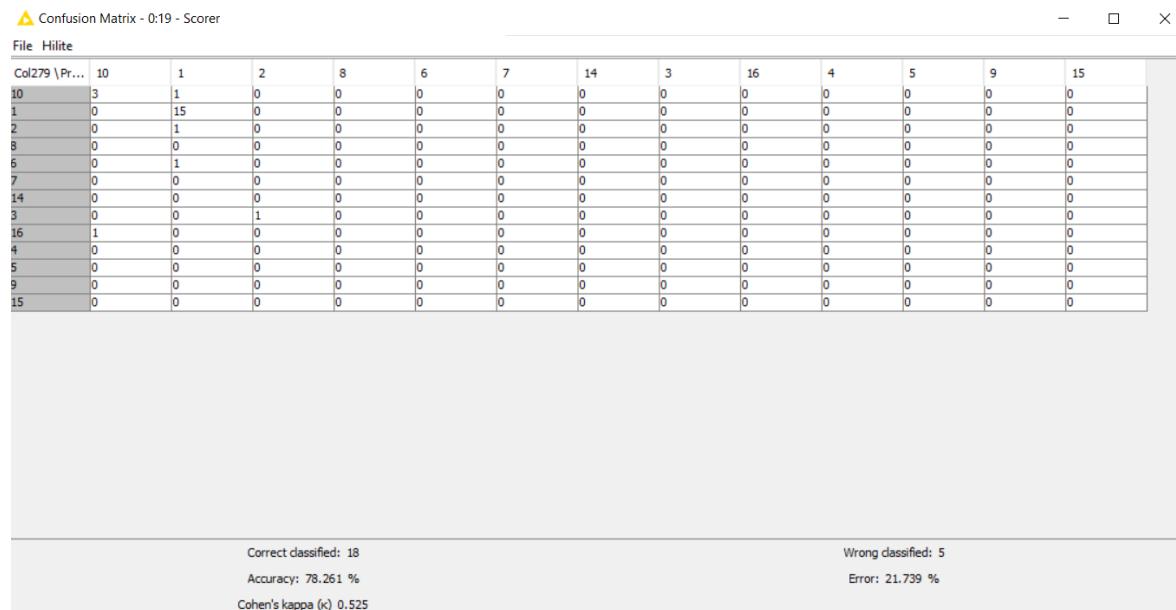


Fig. 61 Matriz de confusión Boosting

### Sensibilidad

La sensibilidad y la especificidad son medidas estadísticas del rendimiento de una prueba de clasificación binaria que se utilizan ampliamente:

La sensibilidad (tasa de verdaderos positivos) mide la proporción de positivos que se identifican correctamente (es decir, la proporción de arritmias clasificadas correctamente).

### Especificidad

La especificidad (tasa de verdaderos negativos) mide la proporción de negativos que se identifican correctamente (es decir, la proporción de arritmias que no son de un tipo y se clasifican correctamente como arritmias que no son de ese tipo).

## Precisión

La precisión mide qué tan bueno es el modelo para asignar eventos positivos a la clase positiva (tipo de arritmia clasificado en el tipo correcto). Es decir, qué tan precisa es la predicción de tipos de arritmia.

## Tasa de error

### *Error de tipo I*

El primer tipo de error es el rechazo de una verdadera hipótesis nula como resultado de un procedimiento de prueba. Este tipo de error se denomina error de tipo I (falso positivo) y, a veces, se denomina error del primer tipo.

En términos de nuestro trabajo desarrollado, un error de tipo I corresponde a clasificar una arritmia de un tipo, siendo de otro.

### *Error de tipo II*

El segundo tipo de error es no rechazar una hipótesis nula falsa como resultado de un procedimiento de prueba. Este tipo de error se denomina error de tipo II (falso negativo) y también se denomina error de segundo tipo.

En términos de nuestro trabajo desarrollado, un error de tipo II corresponde a clasificar una arritmia de un tipo, en otro.

## Exactitud

En un conjunto de mediciones, la exactitud es la cercanía de las mediciones a un valor específico, mientras que la precisión es la cercanía de las mediciones entre sí.

La exactitud tiene dos definiciones:

1. Más comúnmente, es una descripción de errores sistemáticos, una medida de sesgo estadístico; la baja precisión provoca una diferencia entre un resultado y un valor "verdadero". ISO llama a esto veracidad.
2. Alternativamente, ISO define exactitud como la descripción de una combinación de ambos tipos de error de observación (aleatorio y sistemático), por lo que una alta exactitud requiere tanto alta precisión como veracidad.

## Explicación de los positivos verdaderos y positivos falsos

Un verdadero positivo es un resultado en el que el modelo predice correctamente la clase positiva. De manera similar, un verdadero negativo es un resultado en el que el modelo predice correctamente la clase negativa.

Un falso positivo es un resultado en el que el modelo predice incorrectamente la clase positiva. Y un falso negativo es un resultado en el que el modelo predice incorrectamente la clase negativa.

Row ID	TruePositives	FalsePositives	TrueNegatives	FalseNegatives	Recall	Precision	Sensitivity	Specificity	F-meas...	Accuracy	Cohen'...
10	3	1	18	1	0.75	0.75	0.75	0.947	0.75	?	?
1	15	3	5	0	1	0.833	1	0.625	0.909	?	?
2	0	1	21	1	0	0	0	0.955	NaN	?	?
8	0	0	23	0	?	?	?	1	?	?	?
6	0	0	22	1	0	?	0	1	?	?	?
7	0	0	23	0	?	?	?	1	?	?	?
14	0	0	23	0	?	?	?	1	?	?	?
3	0	0	22	1	0	?	0	1	?	?	?
16	0	0	22	1	0	?	0	1	?	?	?
4	0	0	23	0	?	?	?	1	?	?	?
5	0	0	23	0	?	?	?	1	?	?	?
9	0	0	23	0	?	?	?	1	?	?	?
15	0	0	23	0	?	?	?	1	?	?	?
Overall	?	?	?	?	?	?	?	?	?	0.783	0.525

Fig. 62. Tabla de estadísticas de exactitud  
Boosting

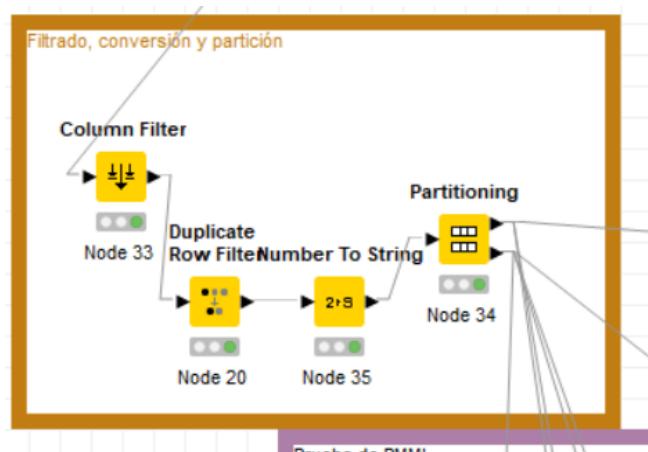
## Evaluación de resultados

Los resultados estadísticos y de predicción mejoraron 52% de precisión con ID3 a un 78% con el uso de la técnica de Gradient Boost, esto consiste en una mejora bastante buena pero no suficiente, el nivel de precisión sigue un poco bajo, esto es debido a que Gradient Boosting crea el primer aprendiz en el conjunto de datos de entrenamiento para predecir las muestras, calcula la pérdida (diferencia entre el valor real y el resultado del primer aprendiz). Y usa esta pérdida para construir un aprendiz mejorado en la segunda etapa. En cada paso, el residual de la función de pérdida se calcula utilizando el método de descenso de gradiente y el nuevo residual se convierte en una variable objetivo para la iteración posterior.

**Esperamos lograr una mejora a la precisión con la técnica de Random Forest.**

## Anexos

Flujo de trabajo en la herramienta de KNIME Analytics Platform, divido en fase de filtrado, conversión y partición, Fig. 52 y sección de Boosting, observe la Fig. 53.



**Nota:** La configuración de estos nodos es similar a la que se usó en ID3, al igual que el Scorer en la Fig. 53.

Fig. 63. Flujo de trabajo KNIME filtrado, conversión y partición

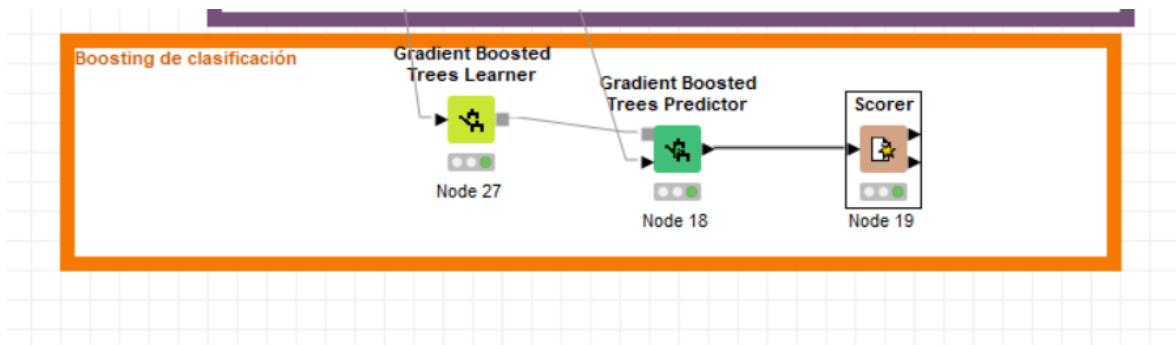


Fig. 64. Flujo de trabajo KNIME fase de Boosting de clasificación

## Configuración del nodo Gradient Boosted Trees Learner

Se utilizaron los mismos atributos que el ID3, y la misma columna objetivo **class** (Columna 279). Generando 100 modelos.

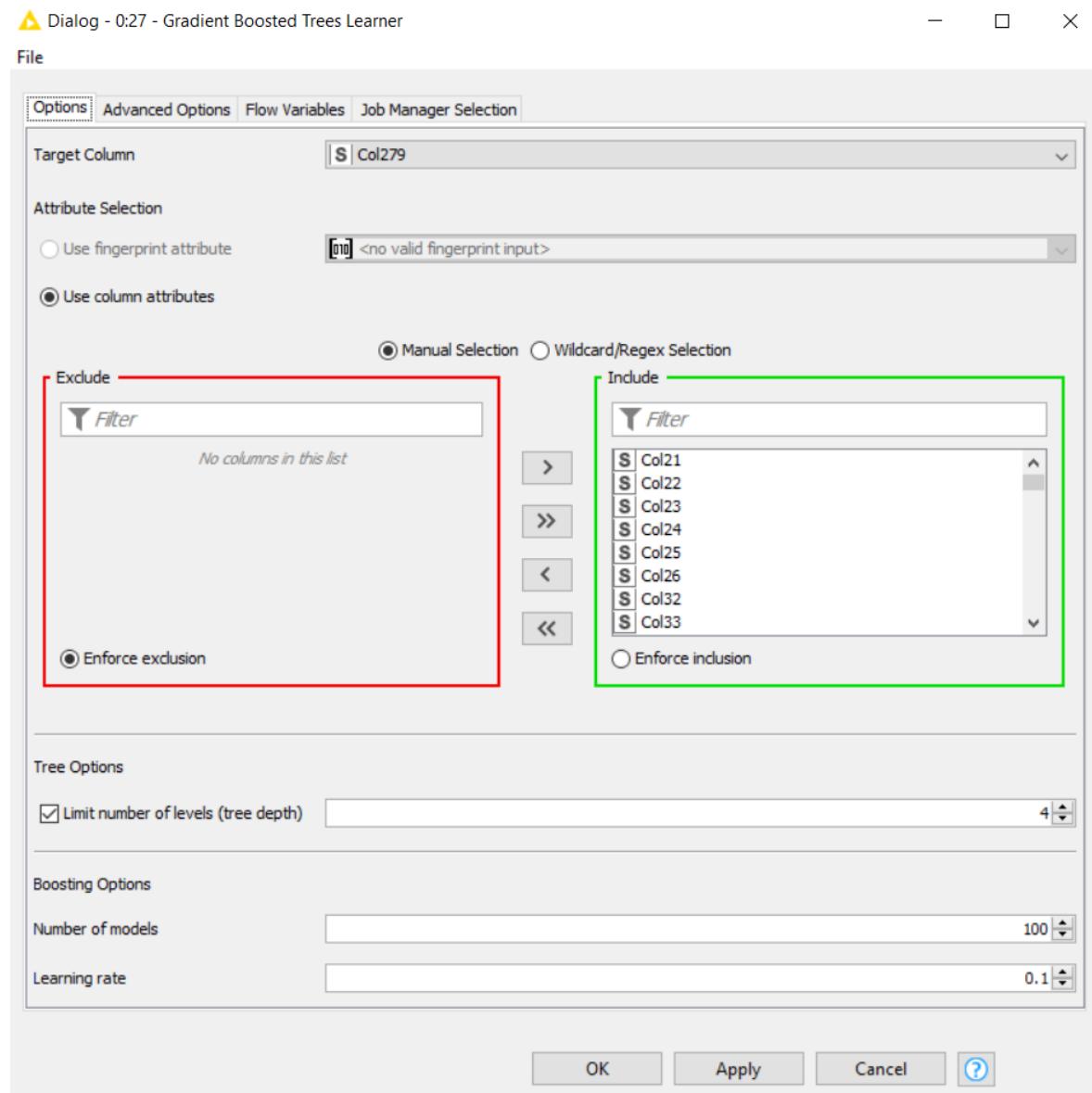


Fig 65. Configuración nodo Gradient Boosted Trees Learner

## Configuración del nodo Gradient Boosted Trees Predictor

Configurado por default.

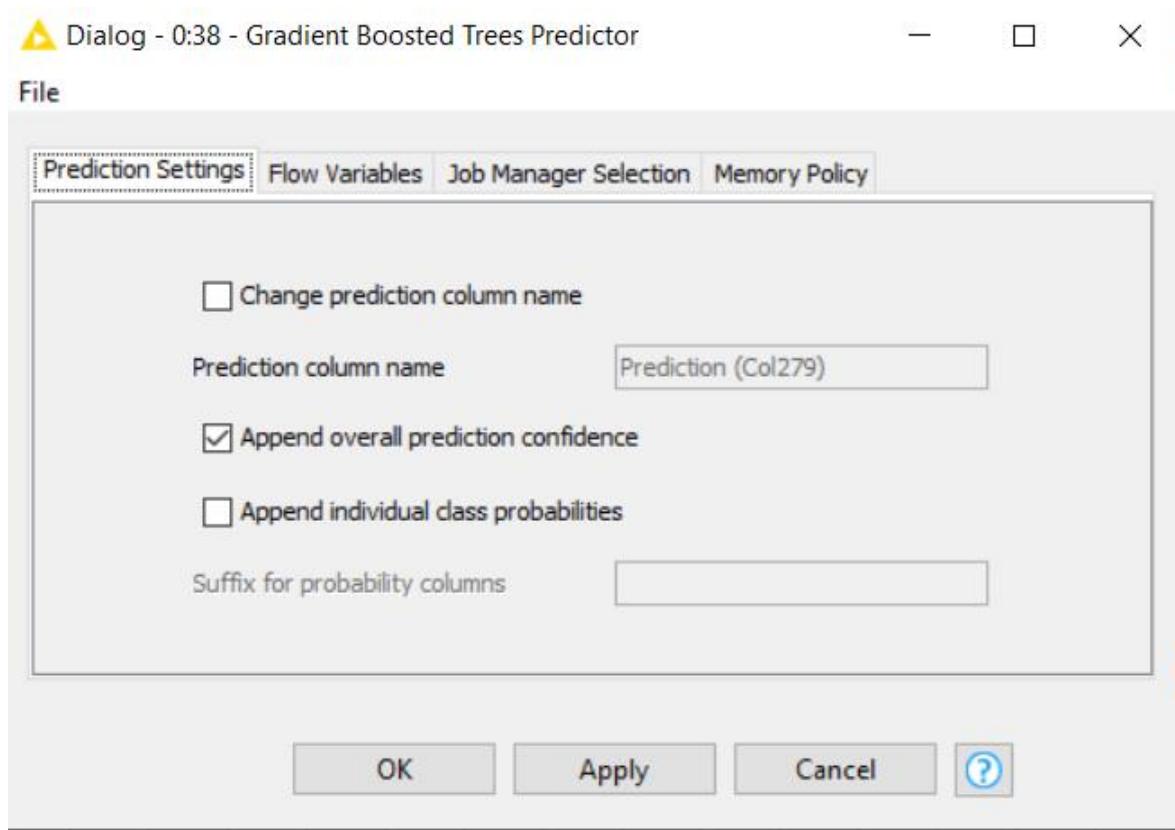
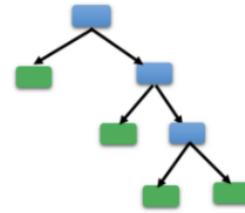
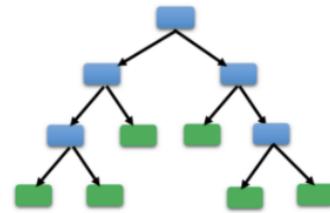
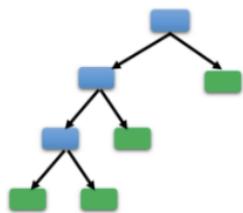
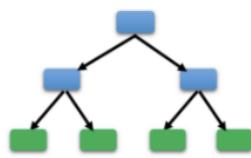
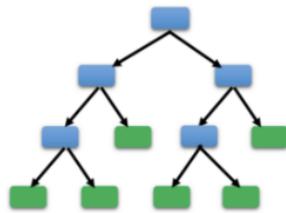
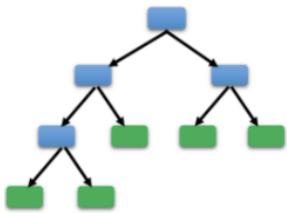


Fig. 66. Configuración nodo Gradient Boosted Trees Predictor

## Random Forest (ID3)



## Random Forest

### Marco teórico

El bosque aleatorio, como su nombre lo indica, consiste en una gran cantidad de árboles de decisión individuales que operan como un conjunto. Cada árbol individual en el bosque aleatorio escoge una predicción de clase y la clase con más votos se convierte en la predicción de nuestro modelo, vea la Fig. 56.

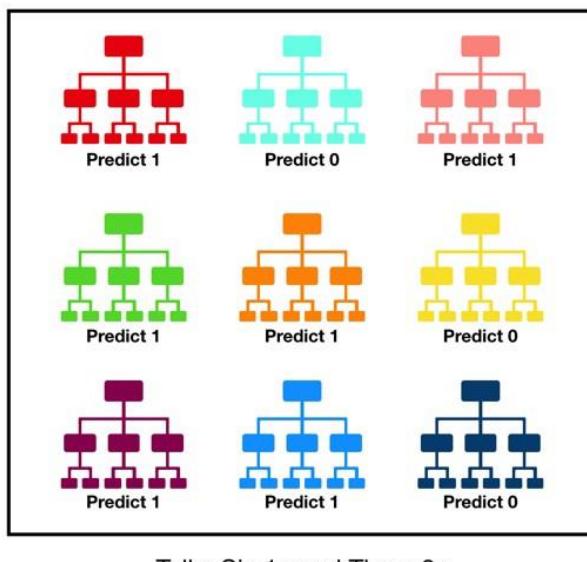


Fig. 67. Predicción Random forest

El concepto fundamental detrás del bosque aleatorio es simple pero poderoso: la sabiduría de las multitudes. En términos de ciencia de datos, la razón por la que el modelo de bosque aleatorio funciona tan bien es:

Un gran número de modelos (árboles) relativamente no correlacionados que operan como un comité superarán a cualquiera de los modelos constituyentes individuales.

La baja correlación entre modelos es la clave. Al igual que las inversiones con correlaciones bajas (como acciones y bonos) se unen para formar una cartera que es mayor que la suma de sus partes, los modelos no correlacionados pueden producir predicciones de conjunto que son más precisas que cualquiera de las predicciones individuales. La razón de este maravilloso efecto es que los árboles se protegen unos a otros de sus errores individuales (siempre que no se equivoquen constantemente en la misma dirección). Si bien algunos árboles pueden estar equivocados, muchos otros árboles serán correctos, por lo que, como grupo, los árboles pueden moverse en la dirección correcta.

## Descripción del trabajo

Bosque aleatorio para clasificación desequilibrada

El bosque aleatorio es otro conjunto de modelos de árboles de decisión y puede considerarse una mejora en el bagging.

Al igual que el bagging, el bosque aleatorio implica seleccionar muestras de arranque del conjunto de datos de entrenamiento y ajustar un árbol de decisiones en cada una. La principal diferencia es que no se utilizan todas las funciones (variables o columnas); en su lugar, se elige un pequeño subconjunto de características (columnas) seleccionadas al azar para cada muestra de arranque. Esto tiene el efecto de des correlacionar los árboles de decisión (haciéndolos más independientes) y, a su vez, mejora la predicción del conjunto.

Es por ello que se aplicará esta técnica con el objetivo de mejorar aún más los resultados obtenidos en ID3 y en Gradient Boost, donde en Gradient Boost, notamos una mejora en la precisión de la clasificación de los tipos de arritmia. Generaremos un Random Forest de árboles ID3 con un total de 200 modelos en este caso.

## Fragmento del diccionario de datos utilizado

<b>22</b>	Existence of ragged R wave	<b>Existencia de onda R irregular</b>	Nominal	<b>0-1</b>
<b>23</b>	Existence of diphasic derivation of R wave	Existencia de derivación difásica de la onda R	Nominal	0-1
<b>24</b>	Existence of ragged P wave	Existencia de onda P irregular	Nominal	0-1
<b>25</b>	Existence of diphasic derivation of P wave	Existencia de derivación difásica de la onda P	Nominal	0-1
<b>26</b>	Existence of ragged T wave	Existencia de onda T irregular	Nominal	0-1
<b>27</b>	Existence of diphasic derivation of T wave, nominal	Existencia de derivación difásica de la onda T	Nominal	0-1
<b>33 a 39</b>	channel DII similar (33 a 39)	...	Nominal	0-76
<b>45 a 51</b>	channel DIII	...	Nominal	0-116
<b>57 a 63</b>	channel AVR	...	Nominal	0-80
<b>69 a 75</b>	channel AVL	...	Nominal	0-148
<b>81 a 87</b>	channel AVF	...	Nominal	0128
<b>93 a 99</b>	channel V1	...	Nominal	0-216

<b>105 a 111</b>	channel V2	...	Nominal	0-108
<b>117 a 123</b>	channel V3	...	Nominal	0-132
<b>129 a 135</b>	channel V4	...	Nominal	0-92
<b>141 a 147</b>	channel V5	...	Nominal	0-136
<b>153 a 159</b>	channel V6	...	Nominal	0-148
<b>280</b>	class	Clasificación de la arritmia	Nominal	[1,16]

Tabla 6. Fragmento diccionario de datos para Random Forest

## Random Forest

La tabla de entrada se divide en dos particiones (es decir, en filas), por ejemplo. entrenar y probar datos. Las dos particiones están disponibles en los dos puertos de salida. Para la fase de entrenamiento se tomó una muestra tomada desde arriba, este modo coloca las filas más altas en la primera tabla de salida y el resto en la segunda tabla, con porcentaje relativo del 95%, así como se muestra en la Fig. 57, por otro lado, para la fase de prueba, fue usado el 5% restante configurado similarmente. Su desarrollo fue en la herramienta de software KNIME Analytics Platform.

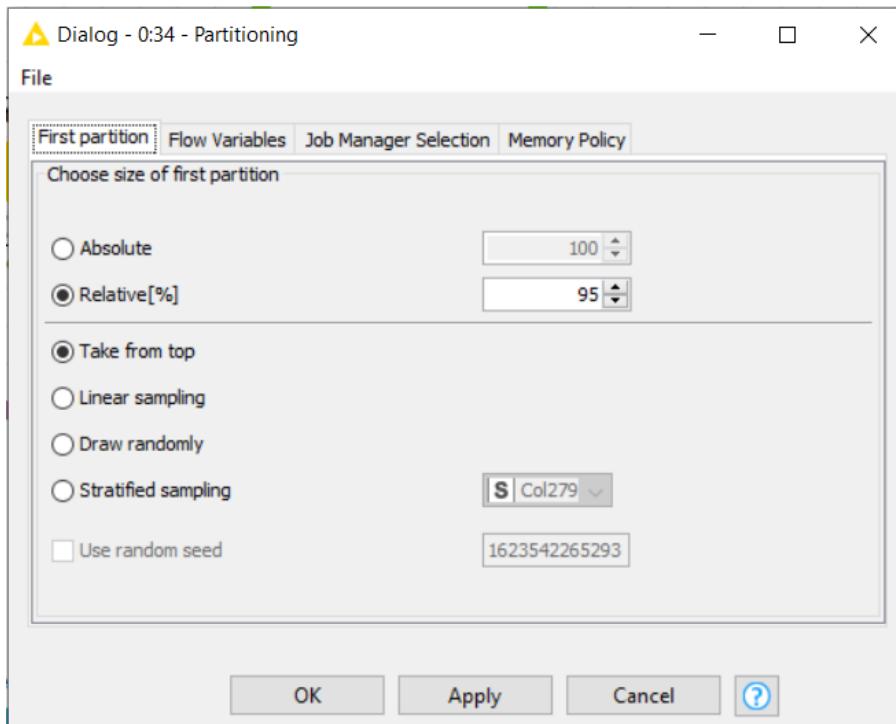


Fig. 68. Particionamiento conjunto de datos

## Diagrama general generado por la herramienta

En la Fig. 58 se observa el primer árbol de decisiones de los 200 modelos generados del Random Forest.

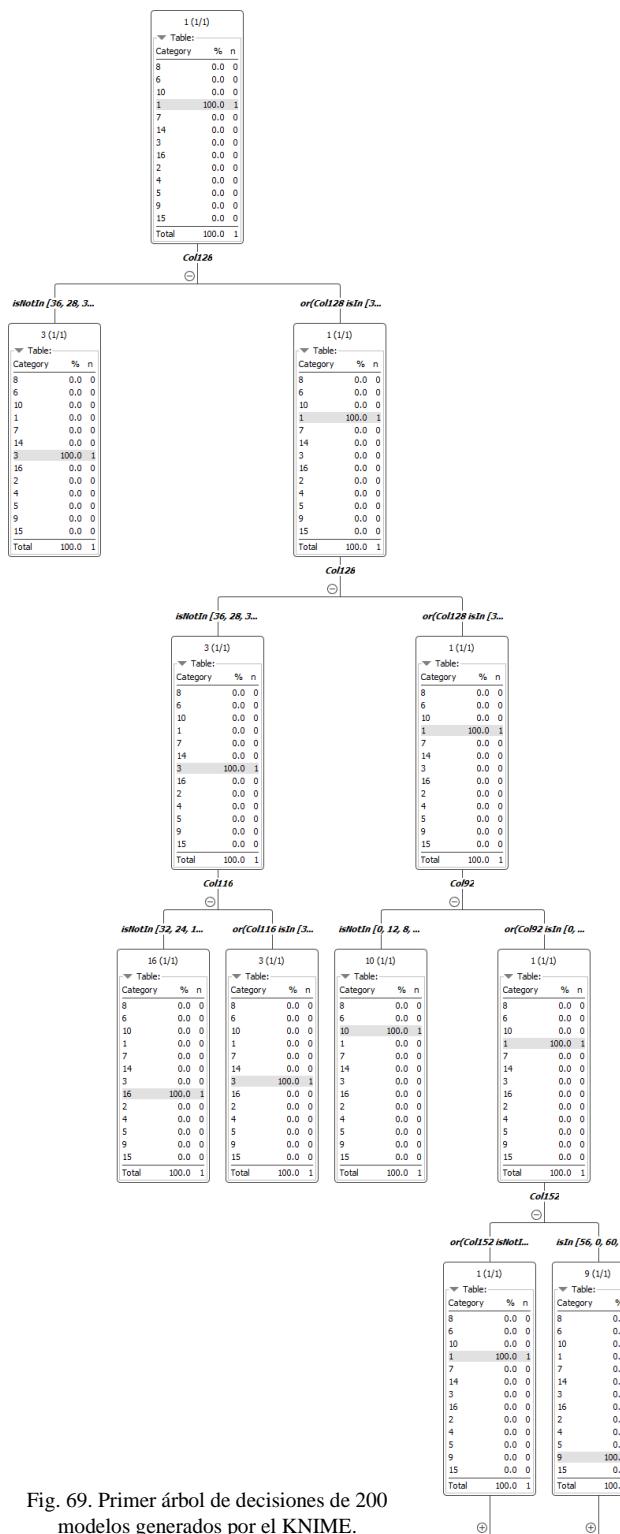


Fig. 69. Primer árbol de decisiones de 200 modelos generados por el KNIME.

## Medidas

### Matriz de confusión

La matriz de confusión es una forma tabular de visualizar el rendimiento de su modelo de predicción. Cada entrada en una matriz de confusión denota el número de predicciones realizadas por el modelo donde clasificó las clases correctas o incorrectamente. Cualquier persona que ya esté familiarizada con la matriz de confusión sabe que la mayoría de las veces se explica para un problema de clasificación binaria. En este caso se tiene una matriz de confusión de clase múltiple. A diferencia de la clasificación binaria, no hay clases positivas o negativas aquí, en nuestro caso son los diferentes tipos de arritmia a clasificar.

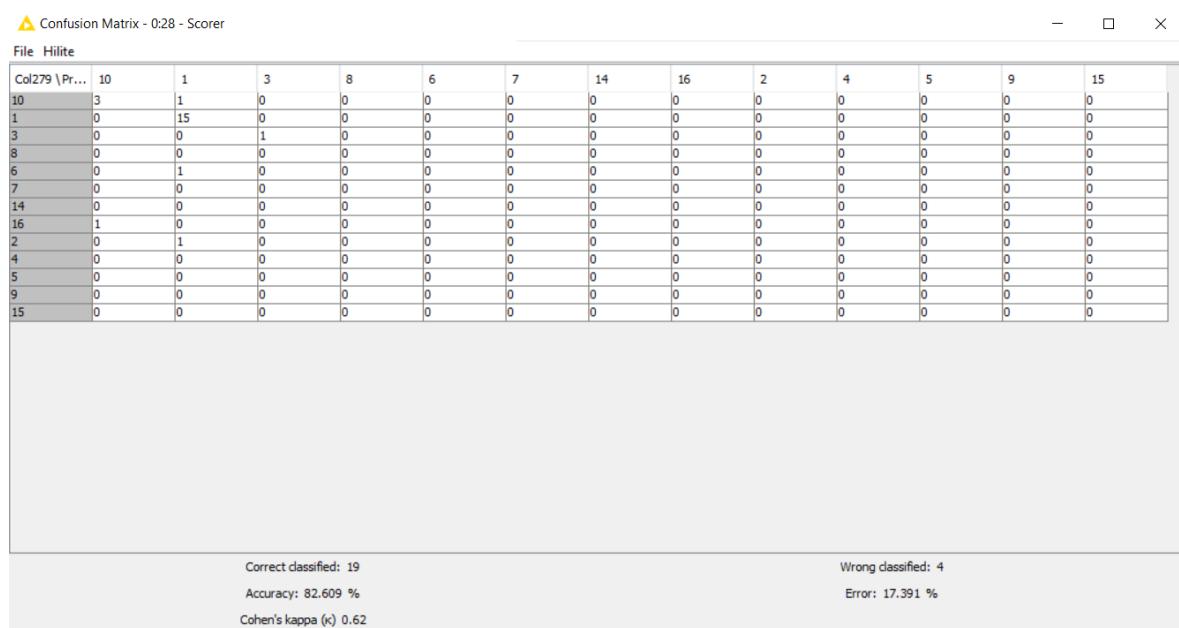


Fig. 70 Matriz de confusión Random Forest

### Sensibilidad

La sensibilidad y la especificidad son medidas estadísticas del rendimiento de una prueba de clasificación binaria que se utilizan ampliamente:

La sensibilidad (tasa de verdaderos positivos) mide la proporción de positivos que se identifican correctamente (es decir, la proporción de arritmias clasificadas correctamente).

### Especificidad

La especificidad (tasa de verdaderos negativos) mide la proporción de negativos que se identifican correctamente (es decir, la proporción de arritmias que no son de un tipo y se clasifican correctamente como arritmias que no son de ese tipo).

## Precisión

La precisión mide qué tan bueno es el modelo para asignar eventos positivos a la clase positiva (tipo de arritmia clasificado en el tipo correcto). Es decir, qué tan precisa es la predicción de tipos de arritmia.

## Tasa de error

### *Error de tipo I*

El primer tipo de error es el rechazo de una verdadera hipótesis nula como resultado de un procedimiento de prueba. Este tipo de error se denomina error de tipo I (falso positivo) y, a veces, se denomina error del primer tipo.

En términos de nuestro trabajo desarrollado, un error de tipo I corresponde a clasificar una arritmia de un tipo, siendo de otro.

### *Error de tipo II*

El segundo tipo de error es no rechazar una hipótesis nula falsa como resultado de un procedimiento de prueba. Este tipo de error se denomina error de tipo II (falso negativo) y también se denomina error de segundo tipo.

En términos de nuestro trabajo desarrollado, un error de tipo II corresponde a clasificar una arritmia de un tipo, en otro.

## Exactitud

En un conjunto de mediciones, la exactitud es la cercanía de las mediciones a un valor específico, mientras que la precisión es la cercanía de las mediciones entre sí.

La exactitud tiene dos definiciones:

1. Más comúnmente, es una descripción de errores sistemáticos, una medida de sesgo estadístico; la baja precisión provoca una diferencia entre un resultado y un valor "verdadero". ISO llama a esto veracidad.
2. Alternativamente, ISO define exactitud como la descripción de una combinación de ambos tipos de error de observación (aleatorio y sistemático), por lo que una alta exactitud requiere tanto alta precisión como alta veracidad.

## Explicación de los positivos verdaderos y positivos falsos

Un verdadero positivo es un resultado en el que el modelo predice correctamente la clase positiva. De manera similar, un verdadero negativo es un resultado en el que el modelo predice correctamente la clase negativa.

Un falso positivo es un resultado en el que el modelo predice incorrectamente la clase positiva. Y un falso negativo es un resultado en el que el modelo predice incorrectamente la clase negativa.

Accuracy statistics - 0:28 - Scorer

File Edit Hilita Navigation View

Table "default" - Rows: 14 Spec - Columns: 11 Properties Flow Variables

Row ID	TruePositives	FalsePositives	TrueNegatives	FalseNegatives	Recall	Precision	Sensitivity	Specificity	F-meas...	Accuracy	Cohen'...
10	3	1	18	1	0.75	0.75	0.75	0.947	0.75	?	?
1	15	3	5	0	1	0.833	1	0.625	0.909	?	?
3	1	0	22	0	1	1	1	1	?	?	?
8	0	0	23	0	?	?	?	1	?	?	?
6	0	0	22	1	0	?	0	1	?	?	?
7	0	0	23	0	?	?	?	1	?	?	?
14	0	0	23	0	?	?	?	1	?	?	?
16	0	0	22	1	0	?	0	1	?	?	?
2	0	0	22	1	0	?	0	1	?	?	?
4	0	0	23	0	?	?	?	1	?	?	?
5	0	0	23	0	?	?	?	1	?	?	?
9	0	0	23	0	?	?	?	1	?	?	?
15	0	0	23	0	?	?	?	1	?	?	?
Overall	?	?	?	?	?	?	?	?	?	0.826	0.62

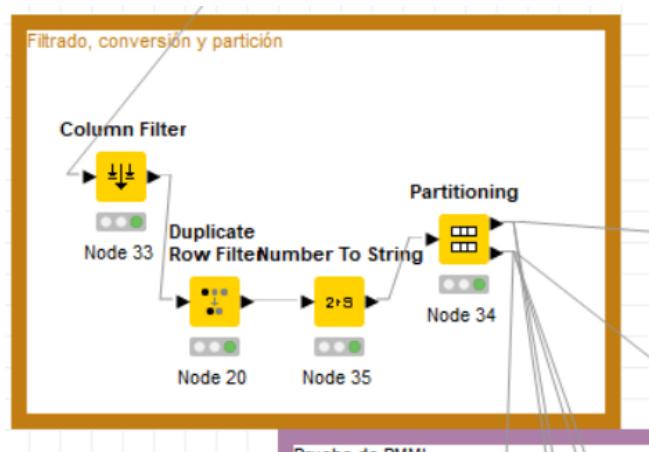
Fig. 71 Tabla de estadísticas de exactitud

## Evaluación de resultados

Los resultados estadísticos y de predicción mejoraron de 78% de precisión con Gradient Boost a un 82% con el uso de la técnica de Random Forest de árboles ID3, esto consiste en un aumento pequeño, no se logró mejorar la precisión ni generando un Random Forest de 1,000 modelos para considerar una probabilidad de al menos un 90% o más. La explicación que damos debido a los resultados obtenidos es que la predicción de los tipos de arritmia puede ser bastante difícil con las herramientas que contamos para la elaboración de este proyecto, no me cierro a que haya técnicas más avanzadas y que logren obtener precisiones altas.

## Anexos

Flujo de trabajo en la herramienta de KNIME Analytics Platform, divido en fase de filtrado, conversión y partición, Fig. 61 y sección de Random Forest Clasificación, observe la Fig. 62.



**Nota:** La configuración de estos nodos es similar a la que se usó en ID3, al igual que el Scorer en la Fig. 62.

Fig 72. Flujo de trabajo KNIME, fase de filtrado, conversión y partición.

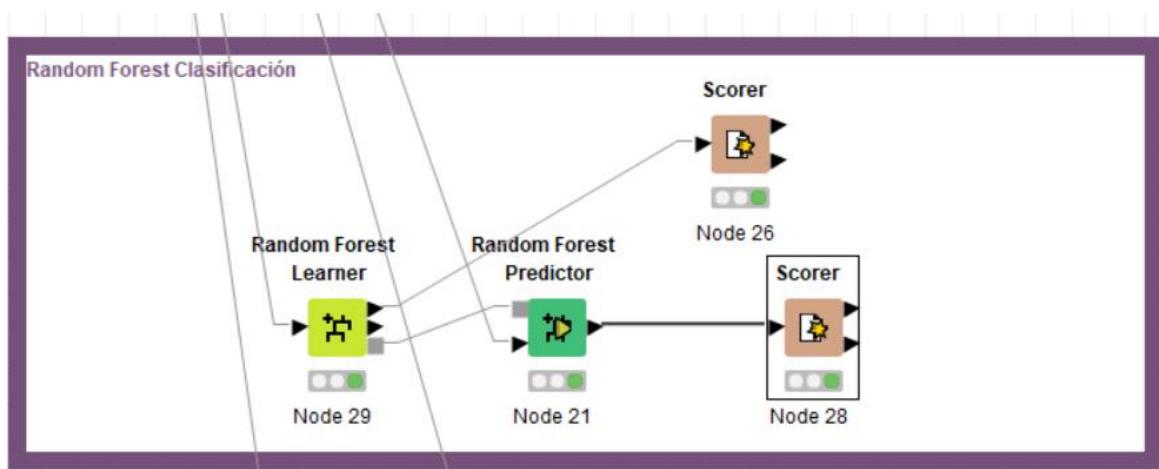


Fig 73. Flujo de trabajo KNIME, fase de Random Forest de clasificación.

## Configuración del nodo Random Forest Learner

Configuración dejada por default, cerciorarse que la columna objetivo sea la Columna 279, se generaron 200 modelos.

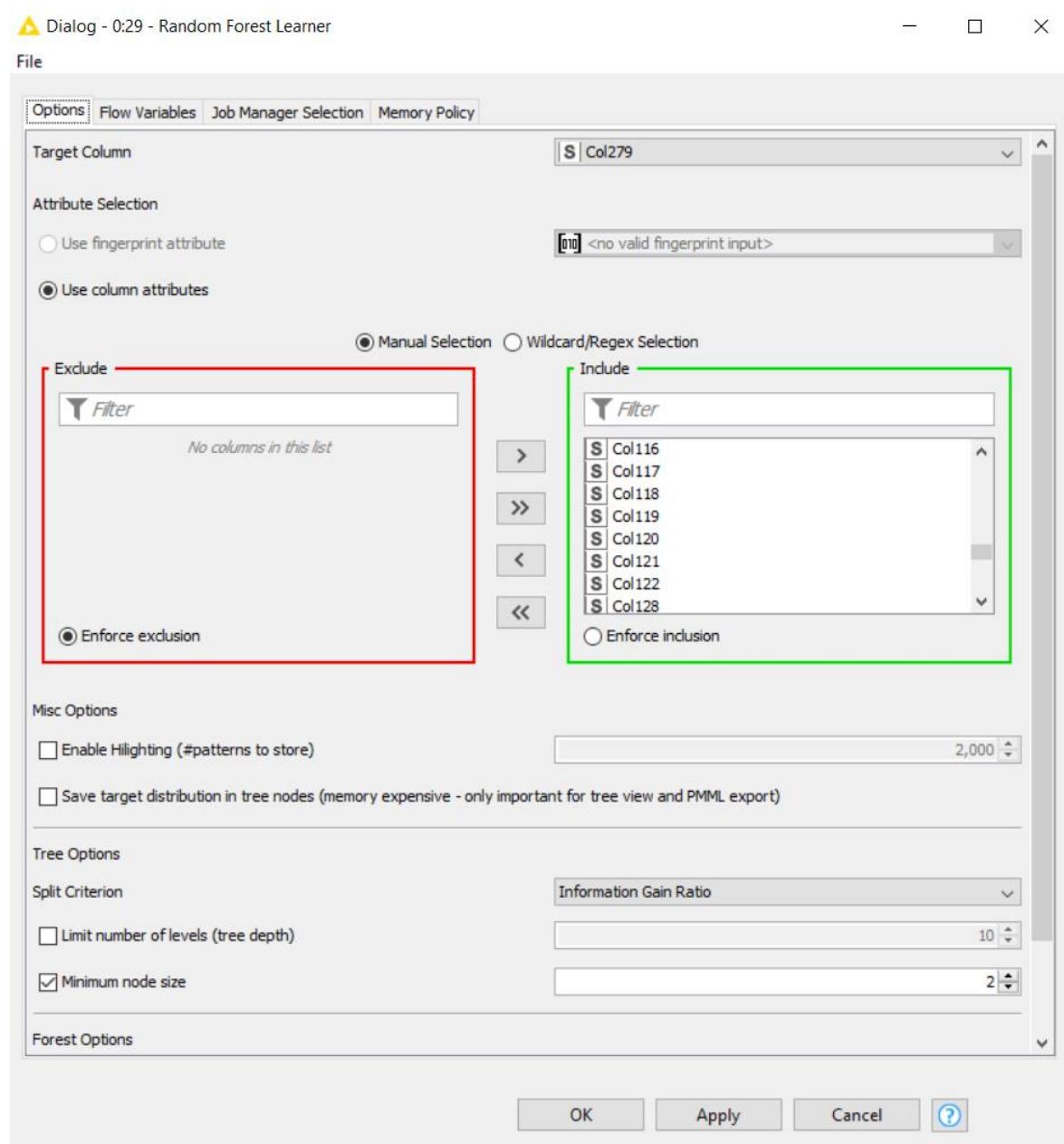


Fig 74. Configuración nodo Random Forest Learner

## Configuración del nodo Random Forest Predictor

Configuración dejada por default.

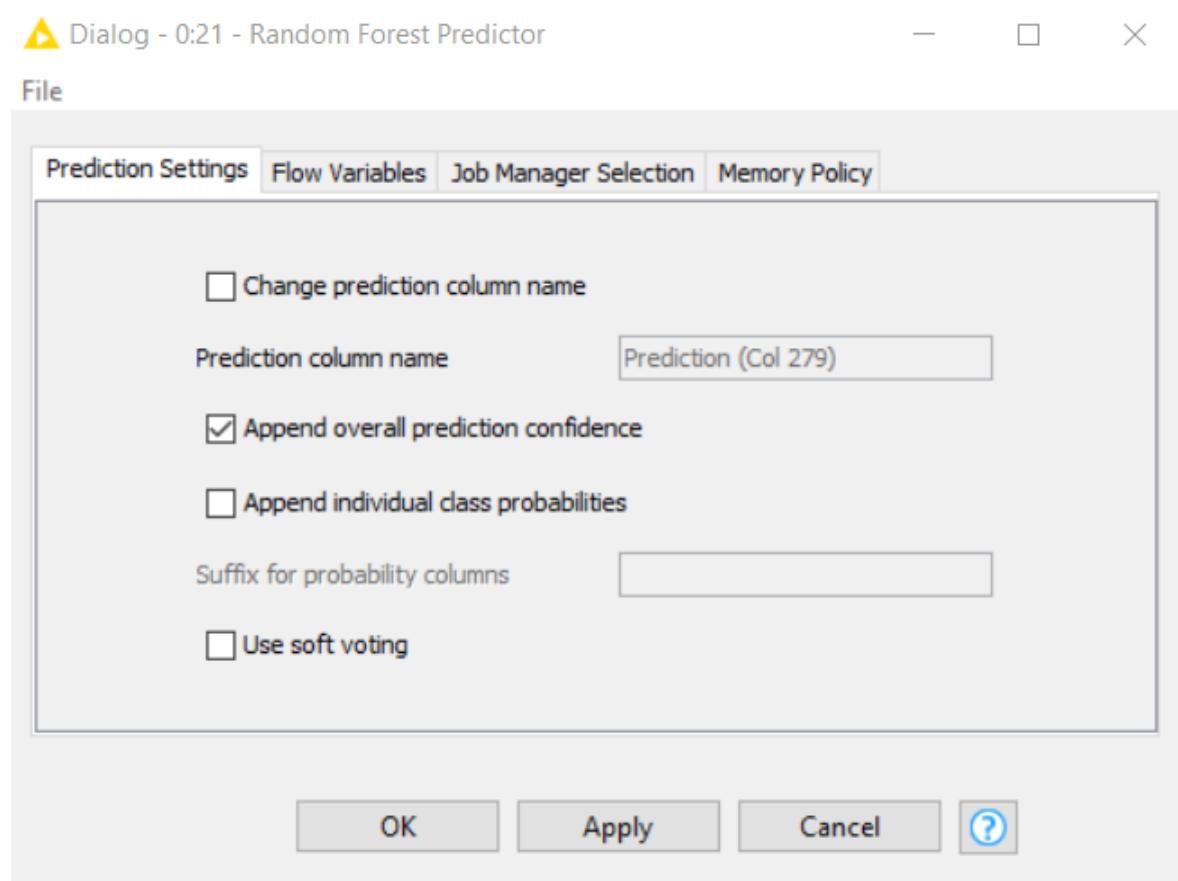
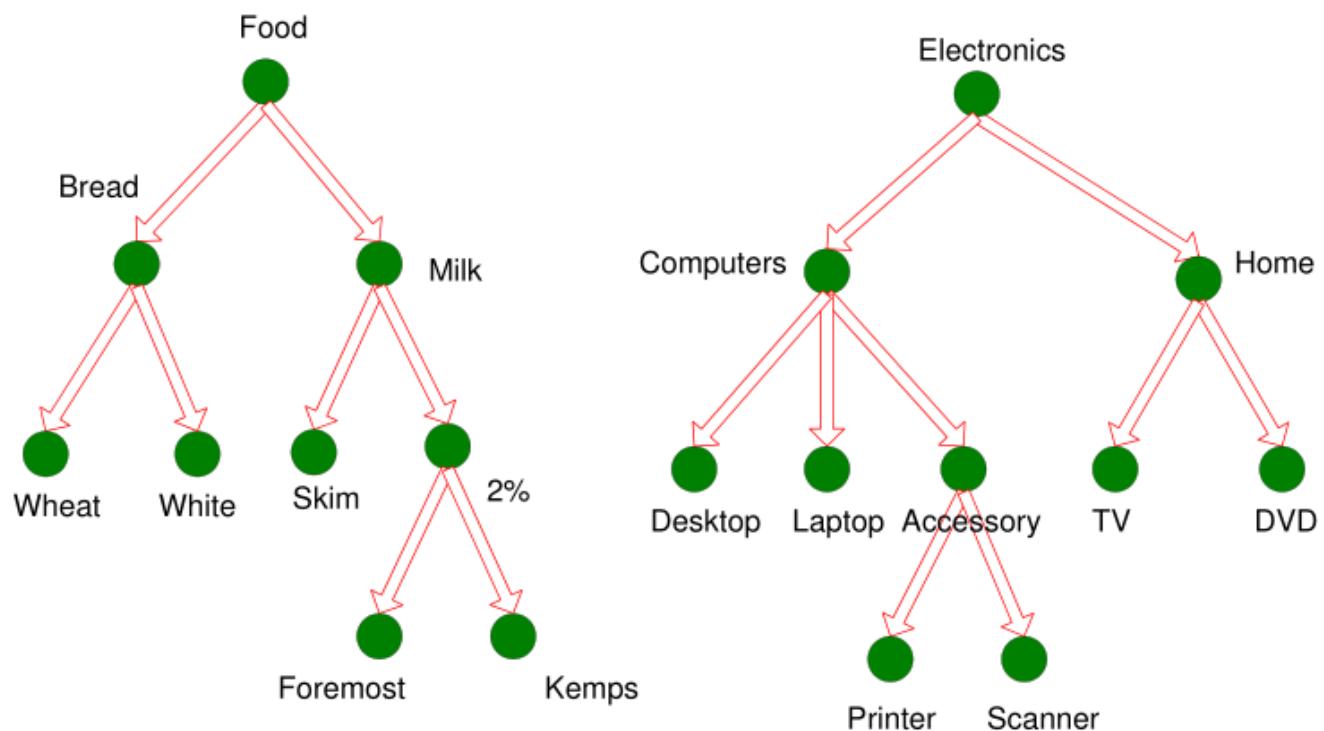


Fig 75. Configuración nodo Random Forest Predictor

## Reglas de asociación



## Reglas de Asociación

### Marco teórico

En esta sección abarcaremos el tema de las reglas de asociación. Recordemos que los algoritmos de reglas de asociación tienen como objetivo principal encontrar relaciones dentro un conjunto de transacciones. En este marco definimos transacción como un grupo de eventos que están asociados de alguna forma, como, por ejemplo:

- ✓ El carrito de compras en una tienda de autoservicio.
- ✓ El conjunto de libros adquiridos en una librería por una misma persona.

A cada uno de los eventos o elementos que forman parte de una transacción se le conoce como item y a un conjunto de ellos itemset. Una transacción puede estar formada por uno o varios items, en el caso de ser varios, cada posible subconjunto de ellos es un itemset distinto.

Una regla de asociación se define como una implicación del tipo “si X entonces Y” ( $X \Rightarrow Y$ ), donde X e Y son itemsets o items individuales. El lado izquierdo de la regla recibe el nombre de antecedente o left-hand-side (LHS) y el lado derecho el nombre de consecuente o right-hand-side (RHS). Por ejemplo, la regla  $\{A, B\} \Rightarrow \{C\}$  significa que, cuando ocurren A y B, también ocurre C.

Existen varios algoritmos diseñados para identificar itemsets frecuentes y reglas de asociación. A continuación, se describen algunos de los más utilizados.

### Apriori

Apriori fue uno de los primeros algoritmos desarrollados para la búsqueda de reglas de asociación y sigue siendo uno de los más empleados, tiene dos etapas:

1. Identificar todos los itemsets que ocurren con una frecuencia por encima de un determinado límite (itemsets frecuentes).
2. Convertir esos itemsets frecuentes en reglas de asociación.

Con la finalidad de ilustrar el funcionamiento del algoritmo, se emplea un ejemplo sencillo. Supóngase la siguiente base de datos de un centro comercial en la que cada fila es una transacción. En este caso, el término transacción hace referencia a todos los productos comprados bajo un mismo ticket (misma cesta de la compra). Las letras A, B, C y D hacen referencia a 4 productos (items) distintos.

#### Transacción

$\{A, B, C, D\}, \{A, B, D\}, \{A, B\}, \{B, C, D\}, \{B, C\}, \{C, D\}, \{B, D\}$

Antes de entrar en los detalles del algoritmo, conviene definir una serie de conceptos:

### Soporte

El soporte del ítem o ítemset X es el número de transacciones que contienen X dividido entre el total de transacciones.

## Confianza

La confianza de una regla “Si X entonces Y” se define acorde a la ecuación

$\text{confianza}(X \Rightarrow Y) = \text{soporte}(\text{unión}(X, Y)) / \text{soporte}(X)$ , donde  $\text{unión}(XY)$  es el ítemset que contienen todos los ítems de X y de Y. La confianza se interpreta como la probabilidad  $P(Y|X)$ , es decir, la probabilidad de que una transacción que contiene los ítems de X, también contenga los ítems de Y.

Volviendo al ejemplo del centro comercial, puede observarse que, el artículo A, aparece en 3 de las 7 transacciones, el artículo B en 6 y ambos artículos juntos en 3. El soporte del ítem {A} es por lo tanto del 43%, el del ítem {B} del 86% y del ítemset {A, B} del 43%. De las 3 transacciones que incluyen A, las 3 incluyen B, por lo tanto, la regla “clientes que compran el artículo A también compran B”, se cumple, acorde a los datos, un 100% de las veces. Esto significa que la confianza de la regla  $\{A \Rightarrow B\}$  es del 100%.

## Descripción del trabajo

En este caso la aplicación que buscamos con las reglas de asociación es identificar las características que tiene alguien con una arritmia de tipo 1, por ejemplo, y generar las reglas para poder realizar un primer diagnóstico rápido.

En cuestión de los atributos a usar hay que recordar que para generar reglas de asociación es necesario que todos sean del tipo discreto. Por esta razón en primera instancia se pensó en usar todos los atributos del tipo categórico que nos ofrecía desde un inicio nuestro conjunto de datos.

Tras un análisis más cuidadoso concluimos que elegir un solo canal sería un buen primer acercamiento, de la misma manera se recurrió a técnica de discretización de datos con el fin de poder usar los atributos continuos que hacían referencia a la frecuencia cardíaca, al peso y a la altura. Atributos que a nuestro parecer pueden ser factores importantes para el diagnóstico.

Los atributos seleccionados se presentarán en la Tabla 7. La representación en la tabla no implica la forma en la que se usaron para generar las reglas de asociación, en secciones futuras se profundizará en el pretratamiento que se le dio al conjunto de datos para cumplir con nuestro propósito.

## Fragmento del diccionario de datos utilizado

#	Nombre	Significado	Tipo	Dominio
2	Sex	Género	Nominal	0 =Masculino; 1 = Femenino
3	Height	Estatura en cm	Lineal	132-190
4	Weight	Peso en kg	Lineal	10-104
15	Heart rate	Número de latidos del corazón por minuto	Lineal	0-88
260 a 269	channel V5	...	Lineal	0-14.9
270 a 279	channel V6	...	Lineal	-4-0
280	class	Clasificación de la arritmia	Nominal	[1,16]

## Pretratamiento

Durante el desarrollo del flujo de trabajo es super importante siempre tener presente que, para las reglas de asociación es necesario contar con atributos discretos. Bajo esta premisa fue necesario recurrir a un proceso de discretización de los datos.

En herramientas como KNIME contamos con nodos que cumplen con esta tarea. En el caso del software Orange en su versión 3.28 aun no contamos con un nodo especializado. Debido a la necesidad se recurrió a hacerlo con la herramienta Excel.

La discretización básicamente se hizo en los tipos de arritmias, creando un nuevo atributo para cada uno. De la misma manera se genero una nueva clase para pasar de continuo a discreto los atributos referentes a la frecuencia cardiaca, la estatura y peso de los individuos.

Para el atributo de la frecuencia cardiaca se hizo una investigación con el fin de establecer lo que es considerado como una frecuencia cardiaca normal. Tras consultar varias fuentes se concluyó que una frecuencia cardiaca normal se encuentra en el intervalo de 60 a 100. Se hizo una formula en Excel donde en una nueva columna llamada Latido normal a los valores del atributo frecuencia los etiqueta con un 1 si se encuentra dentro del intervalo preestablecido. En caso de ser un valor mayor o menos entonces lo etiquetara con un 0.

En la Fig. 76 podremos observar el atributo original y el nuevo atributo creado con el fin de contener la etiqueta asignada a cada registro.

O	KK
Heart rate	Latido Norm F
63	1
53	0
3	1
75	1
71	1
?	0
84	1
70	1
67	1
4	1
64	1
63	1
70	1
72	1
73	1
56	0
72	1
76	1
67	1
70	1
66	1
66	1
76	1
66	1
77	1
⋮	⋮

Fig 76. Atributo Heart Rate, Atributo nuevo

Ese proceso fue por el lado del atributo Heart Rate. Tras el análisis se decidió que también se debería incluir atributos referentes a la estatura y el peso. La forma en la que estos se incorporaron fue calculando el IMC. Tras una investigación se encontró que en ocasiones la falta o el exceso de peso puede ser un factor para desarrollar una arritmia. Con esta información se generó un nuevo atributo que recibió el nombre de Peso Normal, donde tras calcular el índice de masa corporal se procede a etiquetar. En caso de que este índice se encuentre dentro del rango considerado como un peso saludable se recibe la etiqueta de 1, en caso contrario será asignado un cero.

En la Fig. 77 observamos los dos atributos empleados para este cálculo y el nuevo atributo generado.

	C	D	KL
	Height	Weight	Peso normal
0	190	80	1
1	165	64	1
0	172	95	0
0	175	94	0
0	190	80	1
0	169	51	0
1	160	52	1
1	162	54	1
0	168	56	1
1	167	67	1
0	170	72	0
1	165	86	0
1	172	58	1
0	170	73	0
1	160	88	0
1	150	48	1
0	171	59	1
1	158	58	1
0	165	63	1
1	166	72	0
1	160	58	1
0	169	67	1
1	153	75	0

Fig 77. Atributo Peso, Altura, Atributo nuevo

El último paso del pretratamiento que se le dio al conjunto de datos fue el de discretizar el atributo CLASS, aquel que hace referencia a todos los tipos de arritmia documentados. En este caso se generaron 16 atributos nuevos con el fin de que cada uno de estos atributos indique la presencia de cada tipo. Es decir, si por ejemplo tenemos el primer atributo que hace referencia al primer tipo, este tendrá valor de 1 en caso de que este presente en el individuo y un 0 en caso contrario. Esta misma lógica está en los demás atributos.

En la Fig. 78 podemos observar cada uno de los nuevos atributos creados.

Fig 78 Atributo CLASS, 16 atributos nuevos para cada tipo

## Reglas de asociación

La construcción de un flujo de datos para reglas de asociación es netamente sencilla en el software Orange. Es importante señalar que es necesario descargar una extensión. Para esto es necesario entrar al menú Opciones, en el iremos a la opción Add-ons. Una vez que ingresamos se abrirá una ventana como la que se muestra en la Fig. 79. En esta ventana ya solo tenemos que buscar aquella cuyo nombre es **Associate**. En ella encontraremos los dos nodos que emplearemos para el flujo de trabajo.

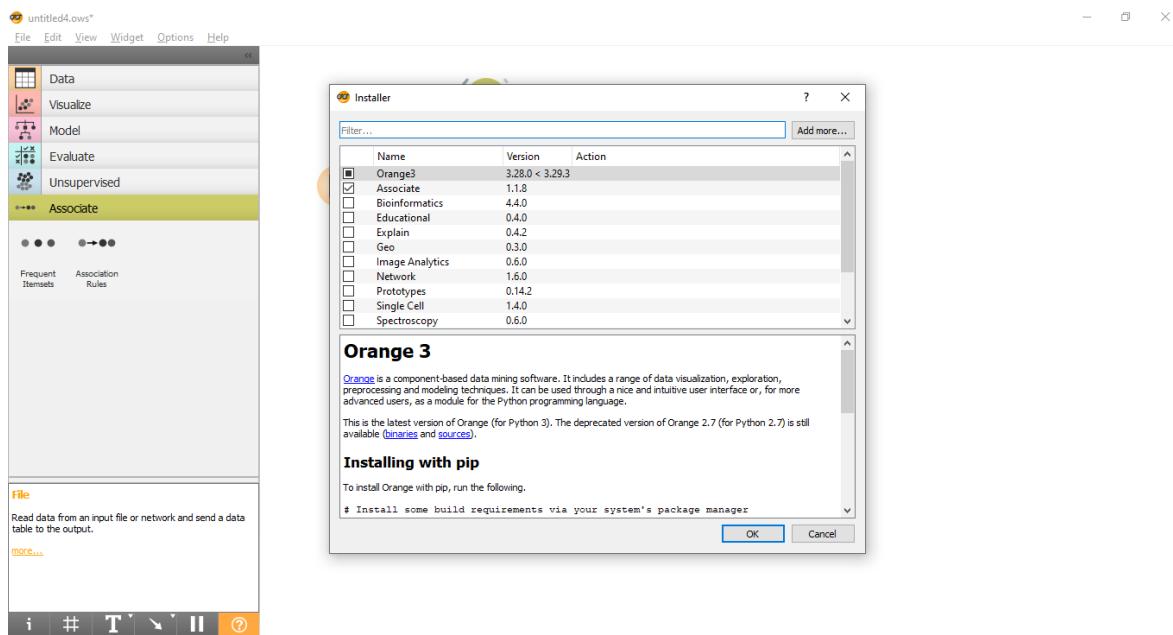


Fig 79 Ventana instalación extensión

Una vez que tenemos listo el software con las herramientas necesarias basta configurar el nodo File para poder generar las reglas de asociación. Según la documentación de los nodos Frequent Itemsets y Association Rules ambos tienen como flujo de entrada el conjunto de datos. La salida son las instancias que concuerden con los criterios ingresado. En las figuras siguientes veremos a detalle su salida y daremos las primeras impresiones.

En el caso del nodo Association Rules. Este nodo implementa el algoritmo de minería de patrones frecuentes FP-growth con optimización de bucketing para bases de datos condicionales de pocos elementos. Para inducir reglas de clasificación, genera reglas para todo el conjunto de ítems y omite las reglas cuando el consecuente no coincide con uno de los valores de la clase.

En la sección buscar reglas de asociación, del mismo nodo, se puede establecer criterios para la inducción de reglas:

- ✓ Soporte mínimo: porcentaje de todo el conjunto de datos cubierto por la regla completa (antecedente y consecuente).
- ✓ Confianza mínima: proporción del número de ejemplos que encajan en el lado derecho (consecuente) entre los que encajan en el lado izquierdo (antecedente).
- ✓ Max. número de reglas: limitar el número de reglas que genera el algoritmo. Demasiadas reglas pueden ralentizar el widget considerablemente.

Si se marca Inducir clasificación (itemset → clase) reglas, el nodo solo generará reglas que tengan un valor de clase en el lado derecho (consecuente) de la regla.

Si el campo de reglas de búsqueda automática está marcado, el nodo ejecutará la búsqueda en cada cambio de parámetros. Puede ser lento para conjuntos de datos con muchos atributos, por lo que es buena idea pulsar Find rules solo cuando los parámetros están establecidos.

También se pueden filtrar las reglas por:

- ✓ Antecedentes:
  - Contiene: filtrará reglas haciendo coincidir expresiones regulares separadas por espacios en elementos precedentes.
  - Elementos mínimos: número mínimo de elementos que deben figurar en un antecedente.
  - Max. artículos: número máximo de elementos que pueden aparecer en un antecedente.
- ✓ Consecuentes:
  - Contiene: filtrará reglas haciendo coincidir expresiones regulares separadas por espacios en elementos consecuentes.
  - Elementos mínimos: número mínimo de elementos que deben aparecer en una consecuente.
  - Max. artículos: número máximo de elementos que pueden aparecer en una consecuente.

Para terminar con este nodo, si se marca la opción de Aplicar estos filtros en búsqueda, el nodo limitará la generación de reglas sólo a reglas que coincidan con los filtros. Si no se activa, se generan todas las reglas, pero sólo se muestran las coincidencias.

En el caso del nodo Frequent Itemsets encuentra elementos frecuentes en un conjunto de datos basado en una medida de soporte para la regla. Contamos con la sección de información sobre el conjunto de datos. Donde la opción de "Expand all" expande el árbol de conjuntos de elementos frecuentes, mientras que "Collapse all" lo contrae.

En Buscar itemsets se puede establecer criterios para la búsqueda Itemset como, por ejemplo:

- ✓ Soporte mínimo: una proporción mínima de instancias de datos que deben soportar (contener) el conjunto de elementos para que se genere. Para conjuntos de datos grandes es normal establecer un soporte mínimo inferior (por ejemplo, entre 2% - 0,01%).
- ✓ Max. número de conjuntos de elementos: limita la cantidad ascendente de conjuntos de elementos generados. Los conjuntos de elementos se generan en ningún orden particular.

Si la búsqueda automática de elementos está activada, el widget ejecutará la búsqueda en cada cambio de parámetros. Podría ser lento para grandes conjuntos de datos, así que pulsando Buscar elementos sólo cuando los parámetros están establecidos Es una buena idea.

También contamos con la opción de filtrar conjuntos de elementos, donde si estás buscando un elemento o conjuntos de elementos específicos, se puede filtrar los resultados por expresiones regulares. Se debe separar las expresiones regulares por comas para filtrar con más de una palabra. Para esta búsqueda tenemos las siguientes opciones:

- ✓ Contiene: filtrará conjuntos de elementos por expresiones regulares.
- ✓ Elementos mínimos: número mínimo de elementos que deben aparecer en un conjunto de elementos. Si 1, se mostrarán todos los conjuntos de elementos. Aumentándolo a, digamos, 4, sólo mostrará conjuntos de elementos con cuatro o más elementos.
- ✓ Max. artículos: número máximo de elementos que deben aparecer en un conjunto de elementos. Si desea encontrar, por ejemplo, sólo conjuntos de elementos con menos de 5 elementos, debe establecer este parámetro a 5.

Si se marca Aplicar estos filtros en búsqueda, el widget filtrará los resultados en tiempo real. Preferiblemente no marcada para grandes conjuntos de datos. Si la selección de envío automático está activada, los cambios se comunican automáticamente.

## Medidas

Para tener más claro el cómo se llevará la evaluación de los resultados obtenidos es importante definir conceptos importantes que forman parte de las reglas, así como los parámetros importantes de las mismas.

En primera instancia vamos a definir lo que es el consecuente y el antecedente. Una forma muy sencilla de entender estos dos conceptos es mediante un condicional, es decir, Si <antecedente> entonces <consecuente>. ¿Qué quiere decir esto? Si sucede el antecedente entonces a continuación sucederá el consecuente. Al antecedente también se le conoce como “parte izquierda” y al consecuente como “parte derecha”.

Sabemos que en un conjunto de datos muy grande probablemente no estemos interesados en todos los elementos que pueden extraerse de los datos, sino en los conjuntos de elementos que son de algún tipo de interés, tal vez desde una perspectiva comercial o desde otra perspectiva.

Por esta razón tenemos las medidas de interés. En general son los parámetros de medición importantes en las reglas de asociación como lo son el soporte, la confianza y el lift o levantamiento.

## Soporte

El soporte nos dice qué tan frecuente es un elemento o un conjunto de elementos en todos los datos. Básicamente, nos dice qué tan popular es un conjunto de elementos en el conjunto de datos dado.

El soporte nos dice qué tan importante o interesante es un conjunto de elementos en función de su número de apariciones. Esta es una medida importante porque en datos reales, hay millones y miles de millones de registros, y trabajar en cada ítem no tiene sentido porque en millones de compras, si un sólo usuario compra dos libros no nos interesa.

Pero el soporte solo no es suficiente. Aunque es importante, por sí solo no nos dice las reglas que se necesitan para aprovechar esta gran cantidad de datos.

## Confianza

Una regla consta de dos partes: antecedente y consecuencia. La confianza nos dice qué tan probable es un consecuente cuando ha ocurrido el antecedente. La confianza se calcula utilizando valores de soporte, de esta manera:

$$\text{conf}(X \Rightarrow Y) = \frac{\text{sop}(X \cap Y)}{\text{sop}(X)} = \frac{|X \cap Y|}{|X|}$$

Fig 80 Fórmula Confianza

Si un elemento es frecuente en un conjunto de datos, entonces hay una alta probabilidad de que una transacción de un elemento menos frecuente contenga el elemento más frecuente, inflando así la confianza. Podemos superar esto al dividir el soporte del conjunto de elementos con el producto del soporte de todos los elementos presentes en el conjunto de elementos para evitar que haya sido sólo una coincidencia. Esto se conoce como levantamiento.

Levantamiento nos dice qué tan probable es el consecuente cuando el antecedente ya ha ocurrido, teniendo en cuenta el soporte de ambos antecedentes y consecuentes.

Usando estas medidas, se han implementado varios algoritmos para encontrar las reglas de asociación desde una base de datos, como apriori y FP-growth.

## Evaluación de resultados

Al ser 16 los posibles tipos de arritmia se seleccionaron los primeros tres, los cuales tienen la mayor frecuencia, para generar y analizar las reglas de asociación. Tomando esto en consideración en la Fig. 81 podemos observar la salida del nodo Association Rules.

Fig 81 Salida Association Rules

El conjunto de datos es pequeño, de cualquier manera, no todas las reglas de asociación son relevantes o representan algo para nuestro objetivo. Hay que recordar que lo que busca esta sección de la investigación es el identificar las características de alguien que tiene arritmia de alguno de los 16 tipos.

Analizaremos unas cuantas reglas de asociación para poder evaluarlas y comprender las implicaciones de cada una. Con el fin de tener un análisis más completo analizaremos un conjunto de reglas con un soporte alto, y un conjunto más con un soporte bajo.

En la Fig. 82 veremos el conjunto de reglas de asociación con un soporte alto.

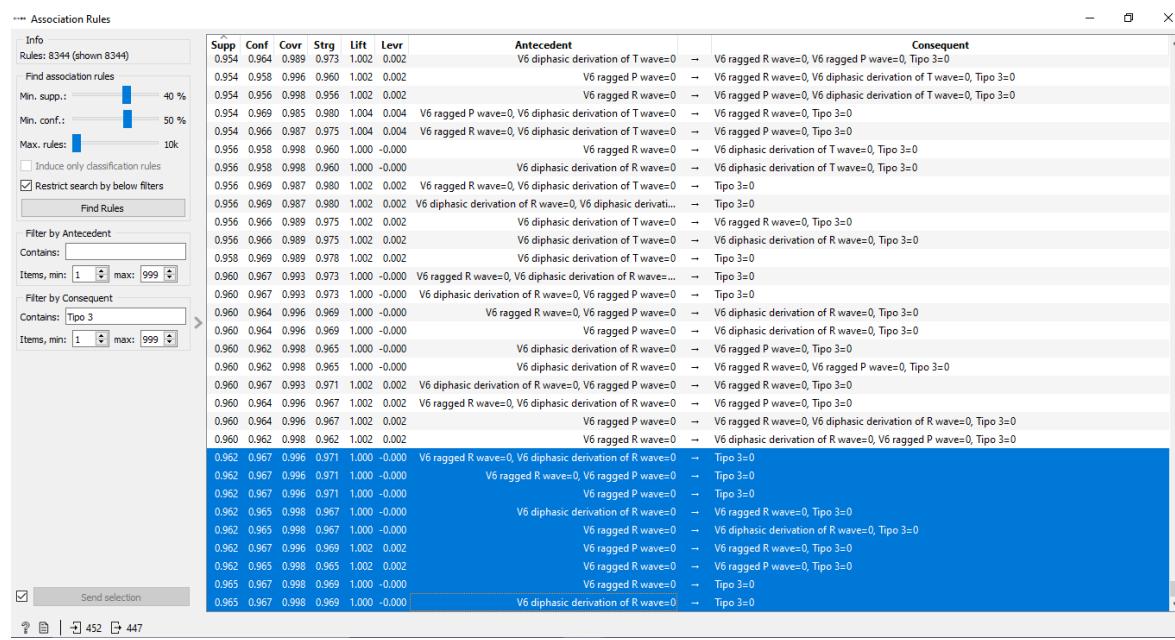


Fig 82 Reglas de Asociación conjunto 1

Nos enfocaremos en aquellas reglas de asociación donde el consecuente es únicamente el tipo. Estas reglas tienen un soporte de 0.962, tienen una confianza mayor a 0.960 y un Lift de 1. Podrían parecer solo números y mediciones, pero una vez que se interpretan arrojan cosas interesantes.

Notemos por ejemplo que una persona que tiene valores normales en su canal V6 es muy poco probable que presente una arritmia de tipo tres. En este caso las reglas que nos interesan son aquellas que tienen como consecuente un tipo de arritmia. Es importante enfocarnos en estos puesto que son las que cumplen con el propósito general propuesto.

En la Fig. 83 veremos el conjunto de reglas de asociación con un soporte bajo.

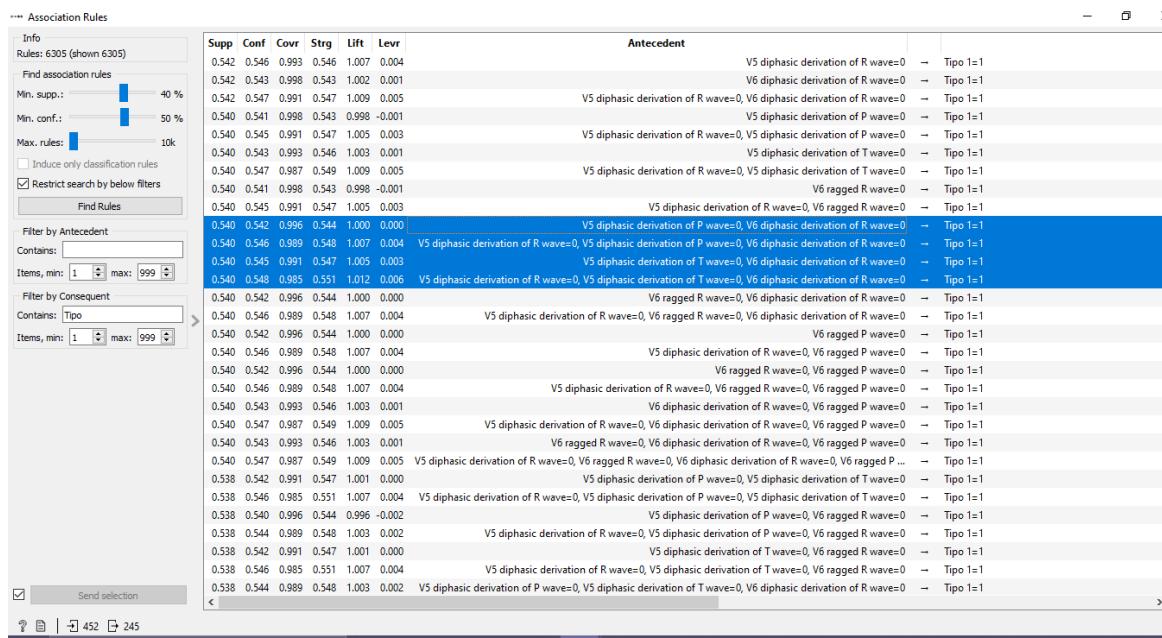


Fig 83 Reglas de Asociación conjunto 2

Que el atributo tipo 1 tenga un valor de 1 implica que hay presencia de ese tipo de arritmia. En el caso de estas reglas de asociación notamos que el valor del soporte y de la confianza no es tan elevado como en el caso anterior. Esto en ningún caso implica que sean malas, sino que se deben buscar forma de mejorarlas. Tal vez una selección diferente de atributos que estén más relacionado con nuestro consecuente aumentaría los valores de confianza y soporte.

Ver tantos números y tantas reglas de asociación puede llegar a ser muy confuso. Frente a esta situación es mejor el uso del nodo Frecuent Itemsets. La forma en la que está desplegada la información es mucho más fácil para la interpretación.

En las figuras siguientes encontraremos este nodo aplicado en 2 de los 16 tipos de arritmia. En la parte posterior a cada figura encontraremos la interpretación de los resultados.

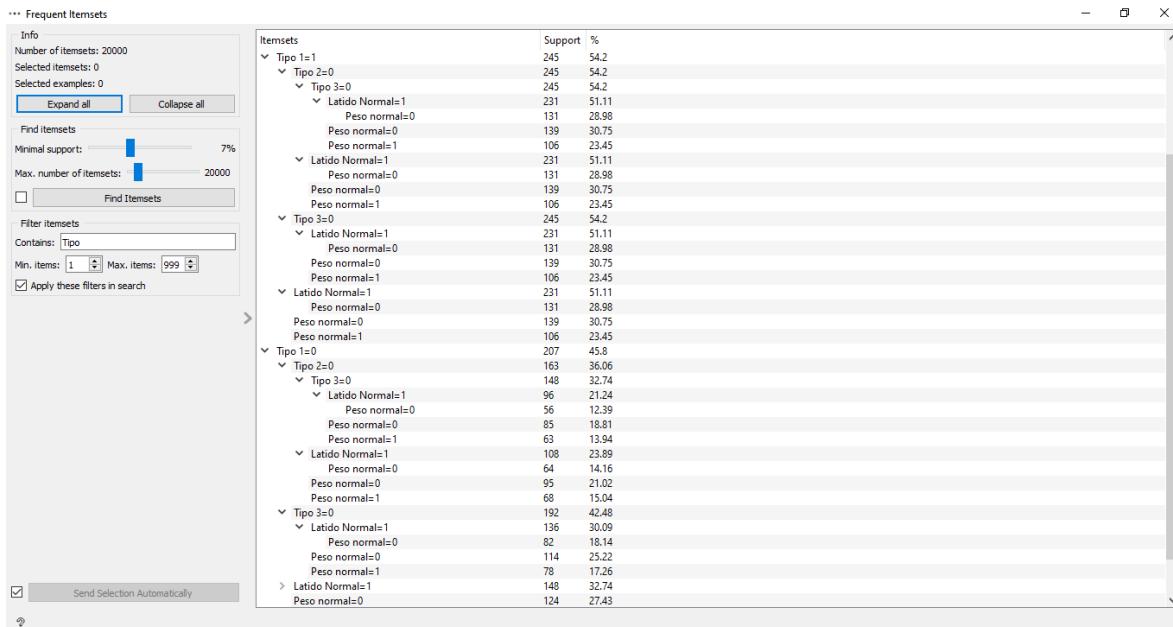


Fig 84 Frequent Itemsets para Tipo 1

Para la interpretación de los resultados hay que recordar que el atributo Tipo 1 se creó tras haber realizado el proceso de discretización de los datos. Siendo este el caso si Tipo 1 tiene valor de uno nos encontramos en el caso en el que este tipo de arritmia está presente. En caso contrario es el caso en el que no está presente, implicando entonces la presencia de otro tipo de arritmia.

Enfocado a los resultados primero analizaremos el caso en el que está presente. Esta información la podemos encontrar en el primer bloque, los atributos que se despliegan a partir de Tipo 1=1. La primera parte se puede interpretar como que Tipo 1 tendrá el valor de cero si Tipo 2 y Tipo 3 tienen valor de cero. Esto se evidente puesto que en la clasificación de nuestro conjunto de datos el tipo 1 se refiere a cuando una persona no tiene arritmia. Siendo este el caso sería ilógico que una persona que no tiene arritmia presente un positivo a cualquiera de los otros tipos.

Continuando sobre este mismo eje tenemos que una persona que no tiene arritmia debe tener latidos normales sin importar si tiene o no sobrepeso. Esto no es una regla absoluta porque como lo podemos ver en los diferentes indicadores a penas si se tiene un soporte de 231 con un porcentaje que apenas sobrepasa el 50%.

Esto es por parte de la sección en la que Tipo 1=1. El caso en el que Tipo 1=0, implica que el individuo en efecto presenta un tipo de arritmia. De la misma manera notamos que lo asocia con los valores cero de Tipo 2 y Tipo 3. Implicando entonces que se tiene otro tipo de arritmia. De la misma manera notamos que el peso y el Heart Rate es indistinto.

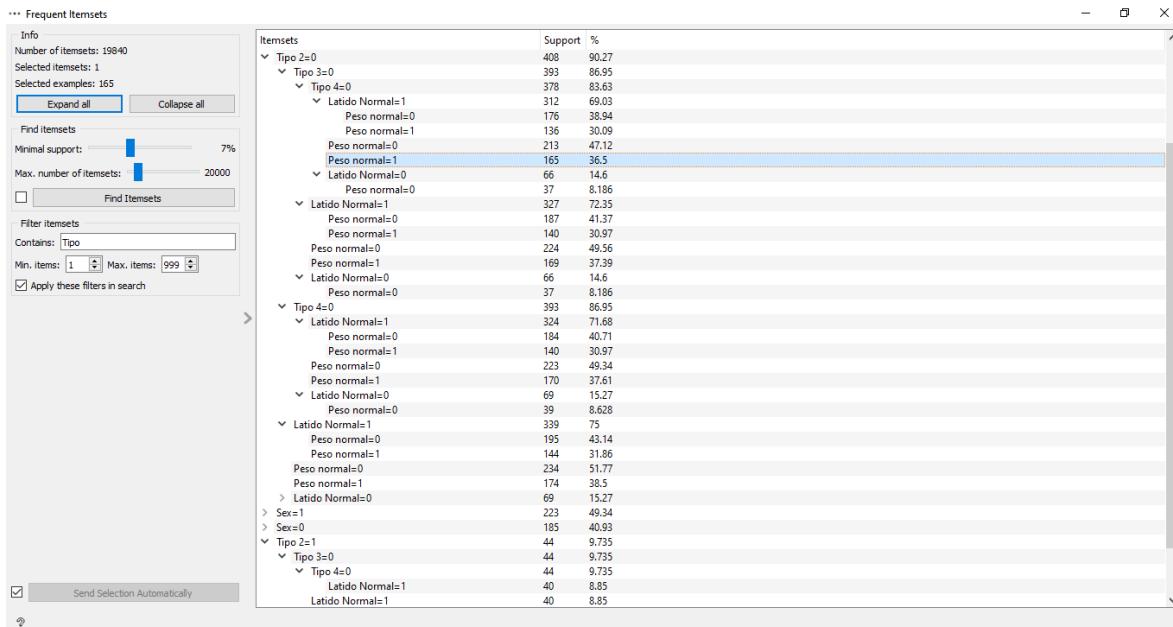


Fig 85 Frequent Itemsets para Tipo 2

Como en el caso anterior es importante tener en mente que el atributo Tipo 2 se creó tras el proceso de discretización. De la misma manera un 1 en su valor representa la presencia de este tipo de arritmia mientras que un 0 representa su ausencia implicando otro tipo o que no hay arritmia.

Yendo por el camino en el que el tipo dos adquiere el valor de 1 vemos que el tipo 3 y 4 están ausentes, como era de esperarse. Lo interesante es que el tipo 2 puede estar presente incluso si se tienen latidos considerados normales. Tal vez para reglas más precisas se deberías considerar atributos más ligados a esta.

Por otro lado, tenemos el caso en el que está ausente, estos caminos son más largos pues, al ser 16 tipos es muy más probable que no sea del tipo 2. Tras el análisis general de los resultados obtenidos el considerar otros atributos tal vez generarían mejores reglas de asociación. Igualmente hay que tener en cuenta que el proceso de discretización es muy importante para las reglas de asociación.

Esto se debe a la aplicación de estas reglas. Las transacciones reportan la presencia o ausencia de un ítem, por tanto, entre más atributos de este tipo es más probable encontrar asociación entre estos atributos discretizados y nuestro consecuente.

## Anexos

En la Fig. 86 observamos el flujo de trabajo completo. Como se mencionó este es muy sencillo pues el nodo por sí mismo se conecta directamente al nodo de la carga del flujo de datos. Como se ha venido comentando el proceso de discretización se generó antes de la carga del flujo de datos a Orange. Por lo tanto, solamente se emplearon los nodos que nos ayudan a cumplir con el objetivo planeado

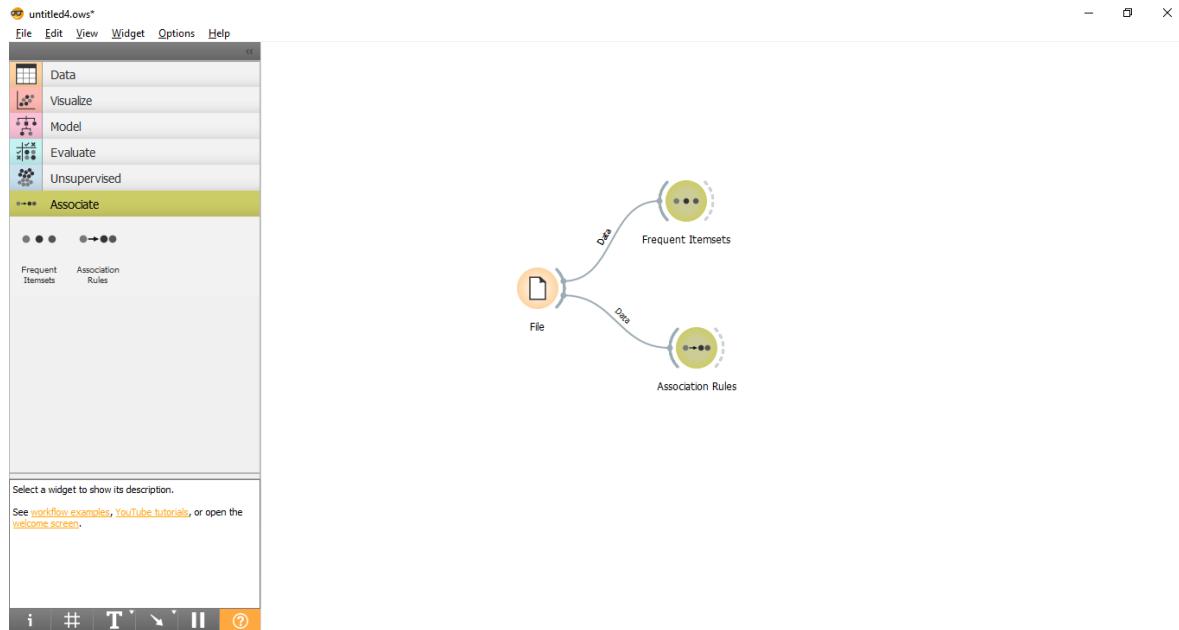
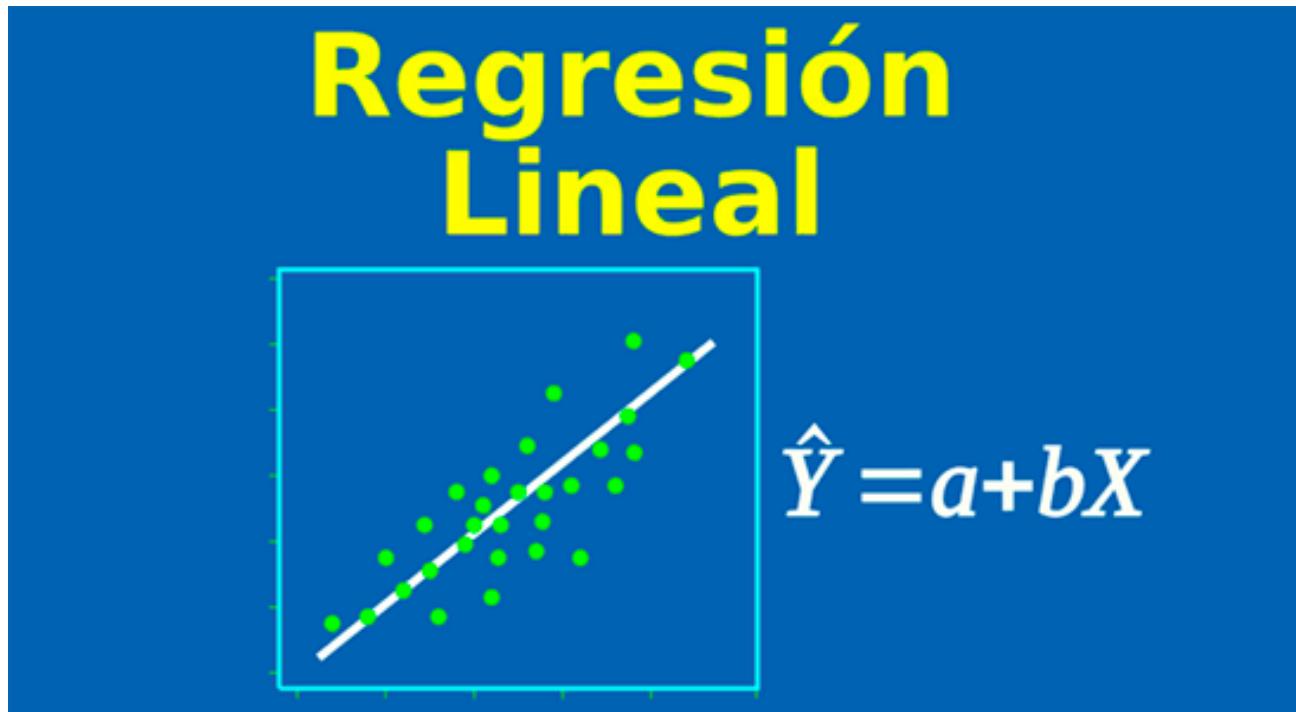


Fig 86 Workflow Reglas de asociación

## Regresión Lineal



## Regresión lineal

### Enunciado del Problema

El conjunto de datos con el que hemos estado trabajando a lo largo de este reporte, contiene parámetros que se obtuvieron de personas que tienen arritmias, hemos ya trabajado con ellos en varios métodos anteriores, veremos en este apartado como nos servirán estos parámetros para la regresión lineal.

El objetivo de este apartado es predecir el número de latidos del corazón de los pacientes, en base a la edad de este, utilizando modelos predictivos de regresión lineal.

### Diccionario de Datos

#	Nombre	Significado	Tipo	Dominio
1	Age	Edad en años	Lineal	0-89
15	Heart rate	Número de latidos del corazón por minuto	Lineal	0-88

### Desarrollo: Proceso KDD

#### Carga del archivo:

El conjunto de datos a usar se llama arrhythmia.data

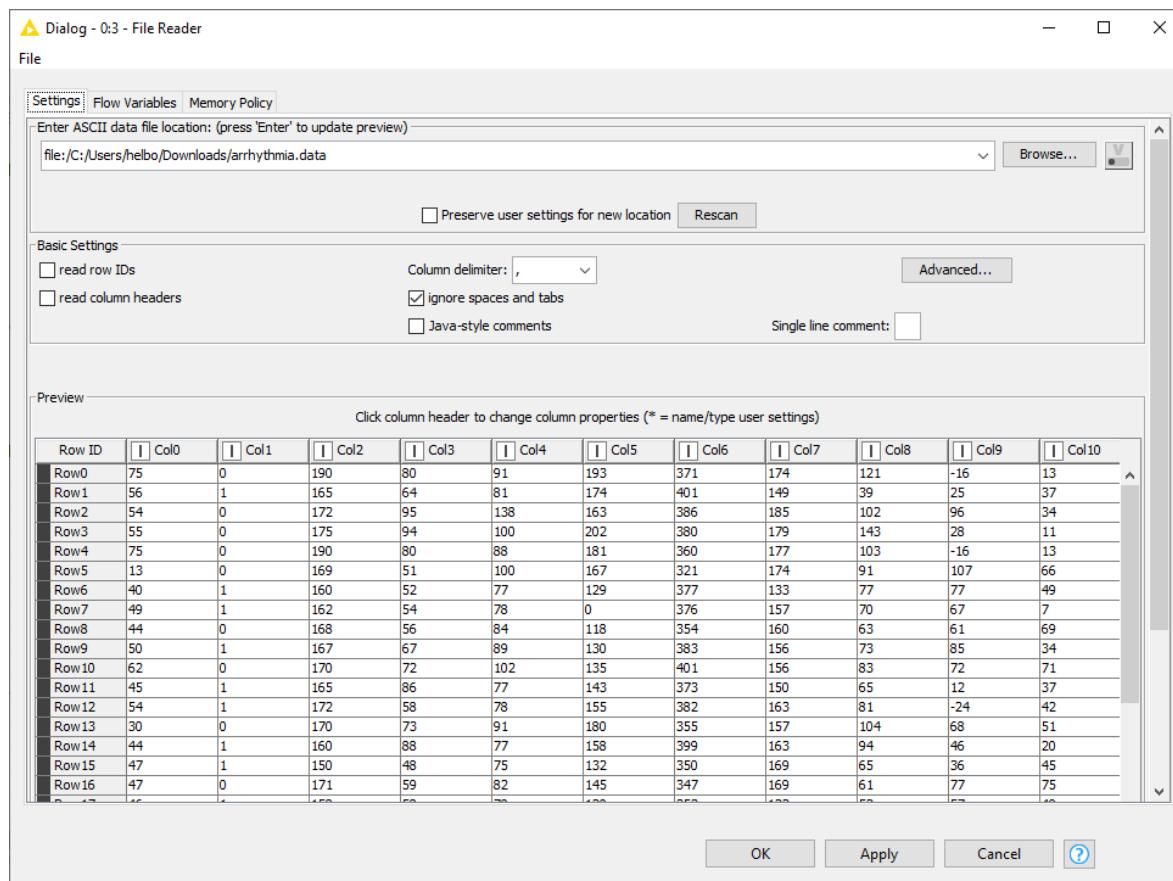
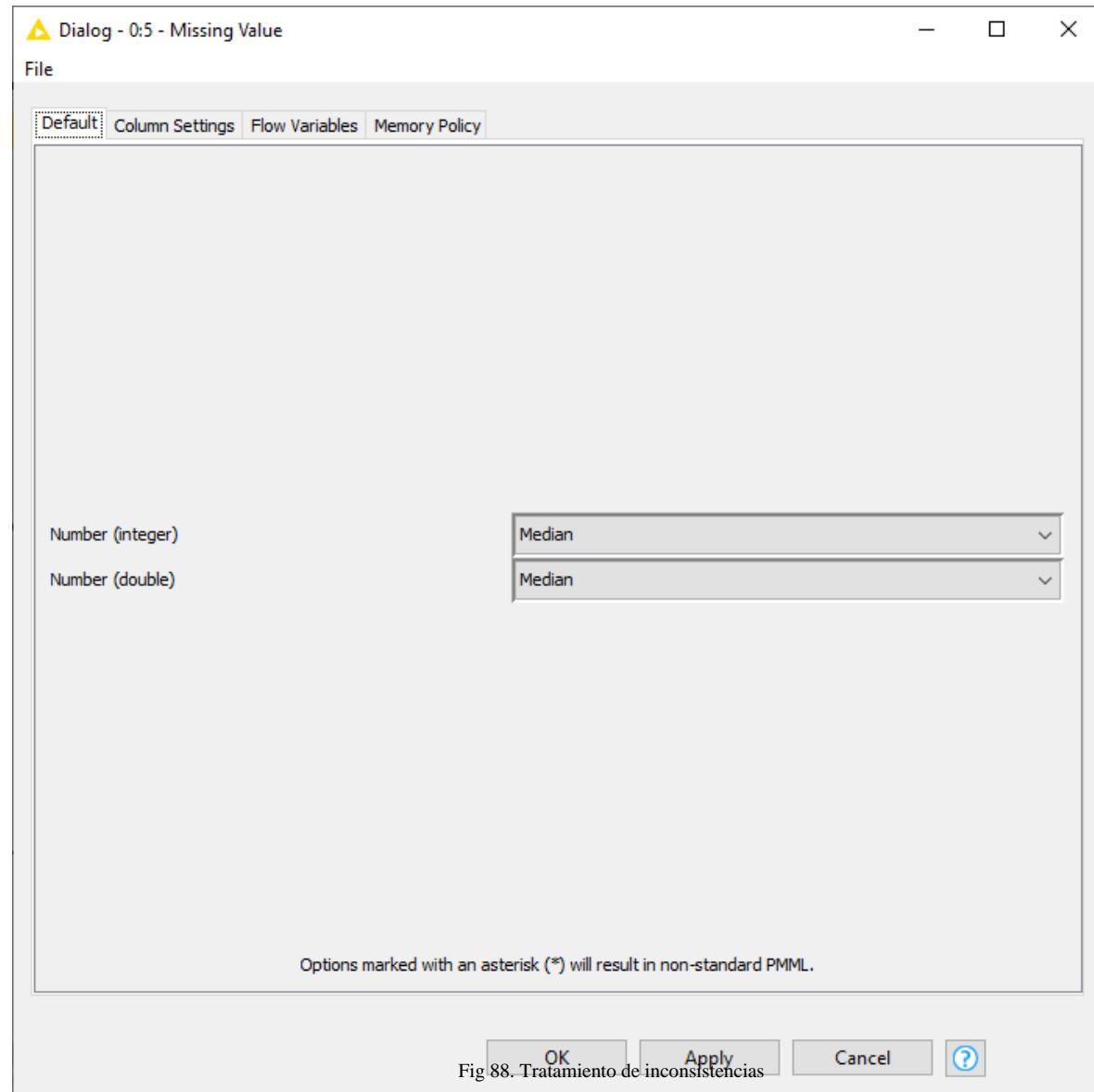


Fig 87. Configuración Carga de Archivos

## Limpieza de los Datos

Al revisar el conjunto de datos en busca de datos nulos, nos encontramos que tiene inconsistencias por lo que procedemos a hacer una limpieza de estos datos para no afectar las predicciones de nuestro modelo.



## Aplicación

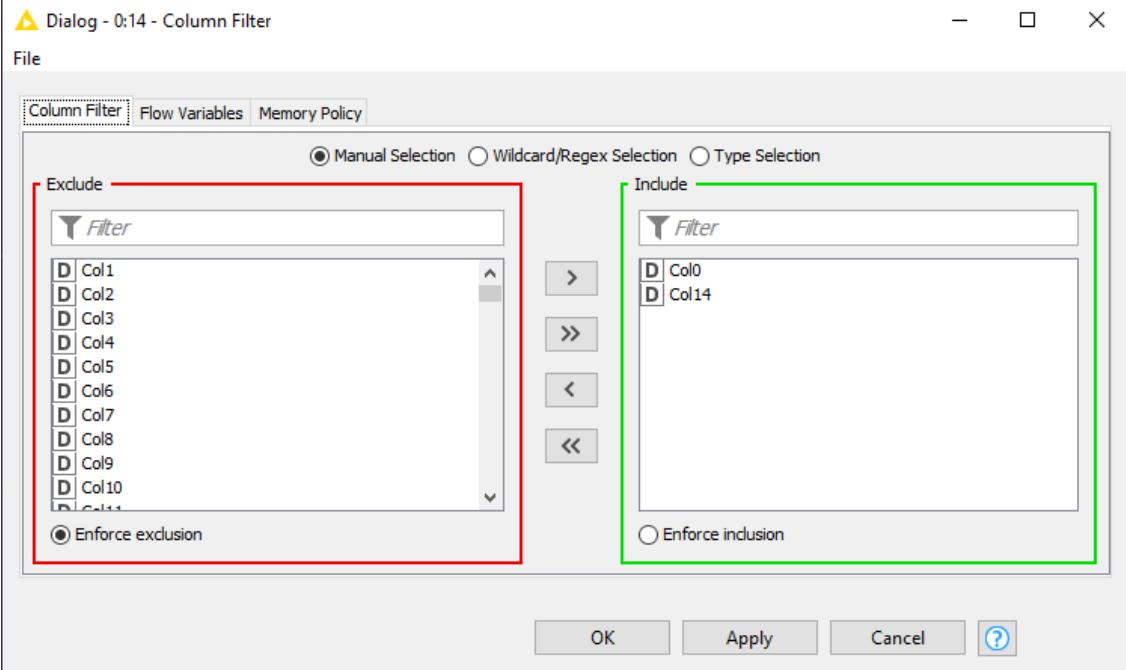
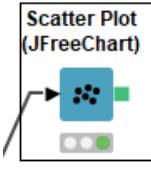
Herramienta	KNIME
Parámetros de Configuración	<p>Ya que vemos que nuestro conjunto de datos es bastante amplio lo filtraremos antes para solo enfocarnos en los atributos que nos servirán para realizar nuestras predicciones.</p> 

Fig 89. Filtro de nuestro conjunto de datos

Filtered table - 0:14 - Column Filter		
<a href="#">File</a> <a href="#">Edit</a> <a href="#">Hilite</a> <a href="#">Navigation</a> <a href="#">View</a>		
Table "default" - Rows: 452 <a href="#">Spec</a> - Columns: 2 <a href="#">Properties</a> <a href="#">Flow Variables</a>		
Row ID	[D] Col0	[D] Col14
Row0	75	63
Row1	56	53
Row2	54	75
Row3	55	71
Row4	75	72
Row5	13	84
Row6	40	70
Row7	49	67
Row8	44	64
Row9	50	63
Row10	62	70
Row11	45	72
Row12	54	73
Row13	30	56
Row14	44	72
Row15	47	76
Row16	47	67
Row17	46	70
Row18	73	66
Row19	57	66
Row20	28	76
Row21	45	66
Row22	36	77
Row23	57	69
Row24	40	68

Fig 90. Datos usados para el modelo

<b>Parámetros de Configuración</b>	Ocupamos en nodo Scatter Plot para visualizar nuestros datos en una representación grafica.
	 Fig 91. Nodo empleado Scatter Plot
Seleccionamos nuestras variables dependientes e independientes, vemos que nuestra variable independiente es la edad y la dependiente por su lado es el numero de latidos.	

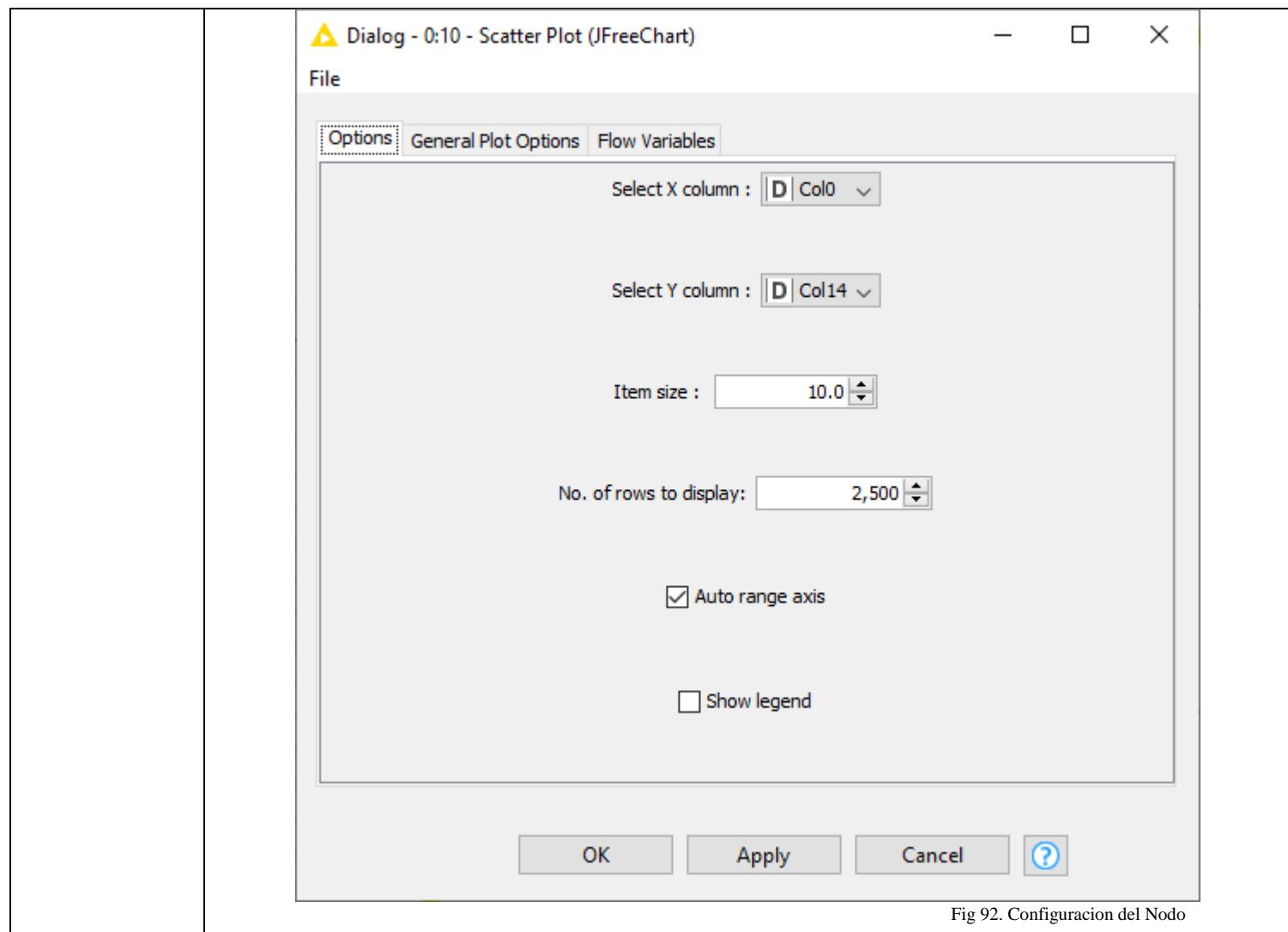
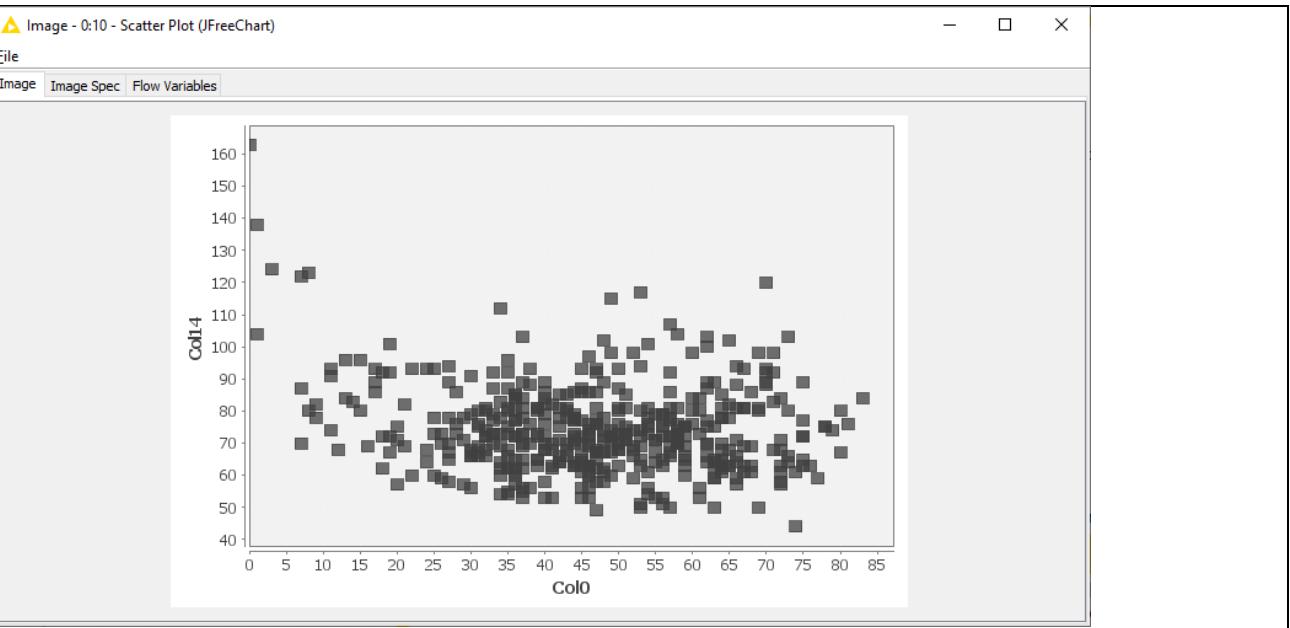
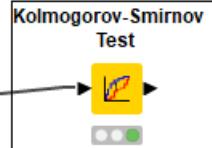
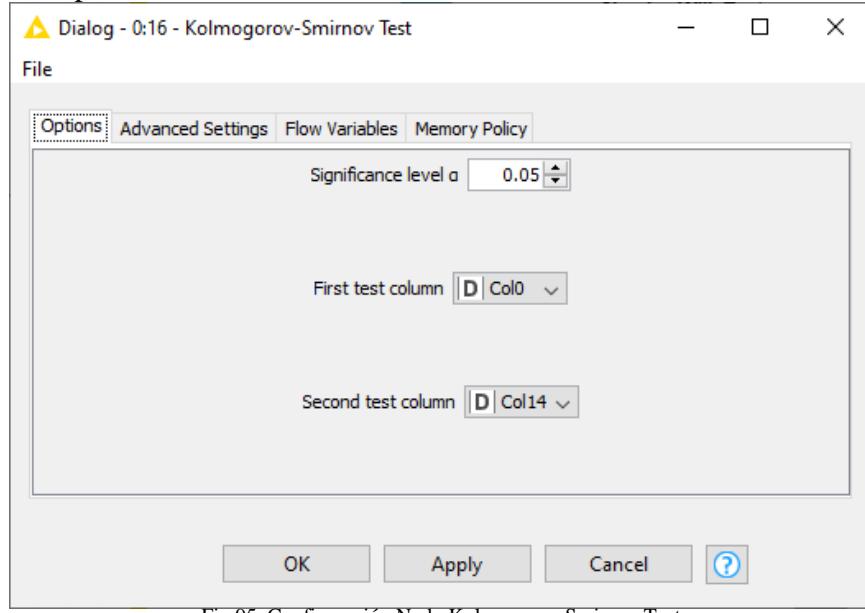


Fig 92. Configuracion del Nodo

	<p>Fig 93. Representación gráfica de nuestro conjunto de datos</p>
<b>Parámetros de Configuración</b>	<p>Haremos uso del nodo Kolmogorov para poder determinar el p-Value que nos servira mas adelante para mas calculos, la configuración la veremos mas adelante.</p>  <p>Fig 94. Nodo Kolmogorov-Smirnov Test</p>
	<p>Colocaremos un valor de significancia del 0.05 y seleccionaremos nuestras variables dependientes e independientes.</p>  <p>Fig 95. Configuración Nodo Kolmogorov-Smirnov Test</p>

<b>Parámetros de Configuración</b>	<p>Ocupamos el nodo Linear Regression Learner para crear el modelo de decisiones que usara toda la muestra de nuestro conjunto de datos.</p> <p>Fig 96. Nodo empleado Linear Regression Learner</p> <p>Seleccionamos nuestra variable objetivo y la variable que nos ayudara en nuestro modelo</p> <p>Fig 97. Configuracion del Nodo</p>
------------------------------------	--

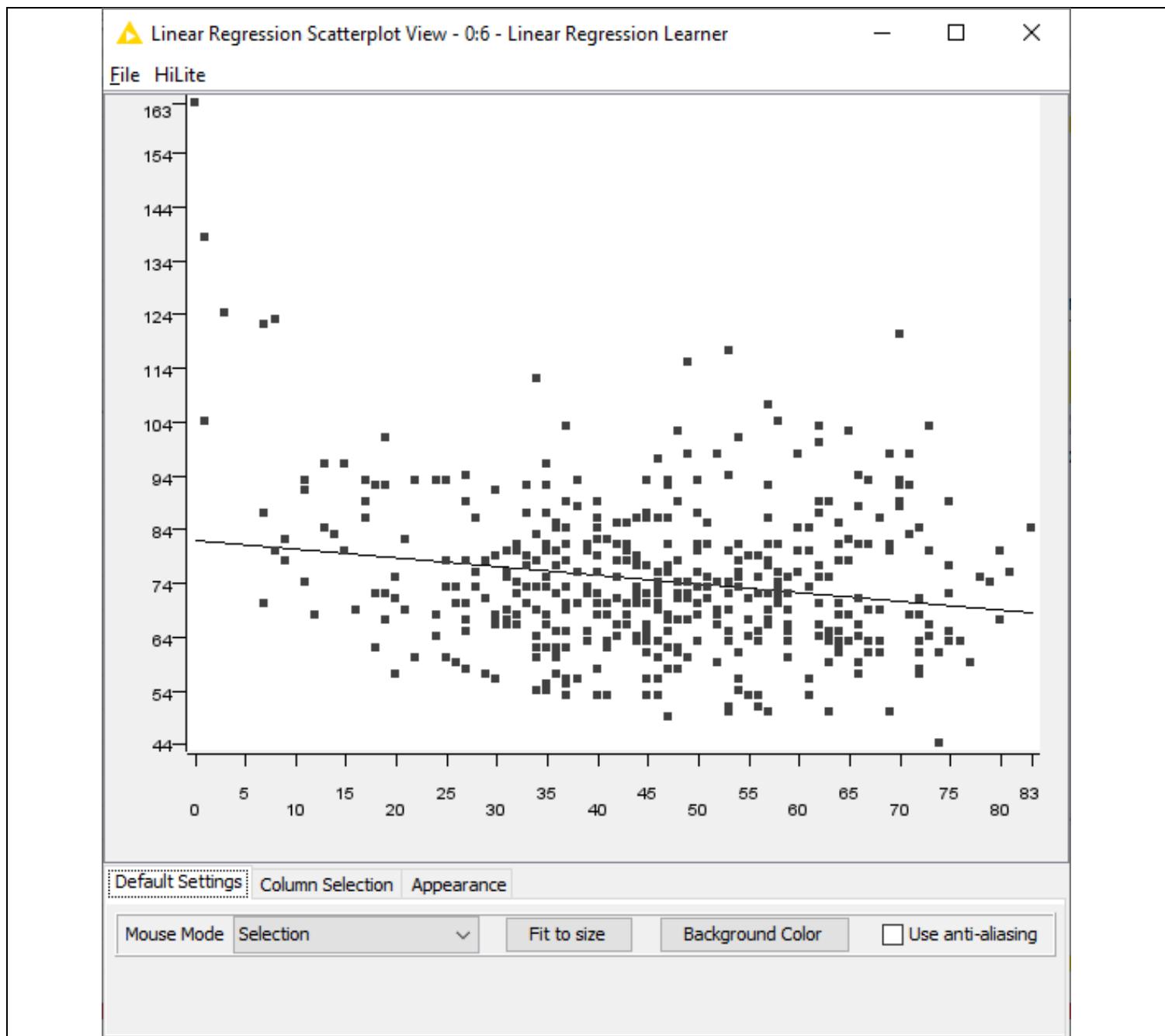


Fig 98. Salida grafica de nuestro modelo

### Parámetros de Configuración

En este nodo necesitaremos el modelo que entrenamos en el nodo anterior y el conjunto de muestra inicial.

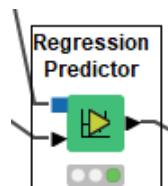


Fig 99. Nodo empleado Regression Predictor

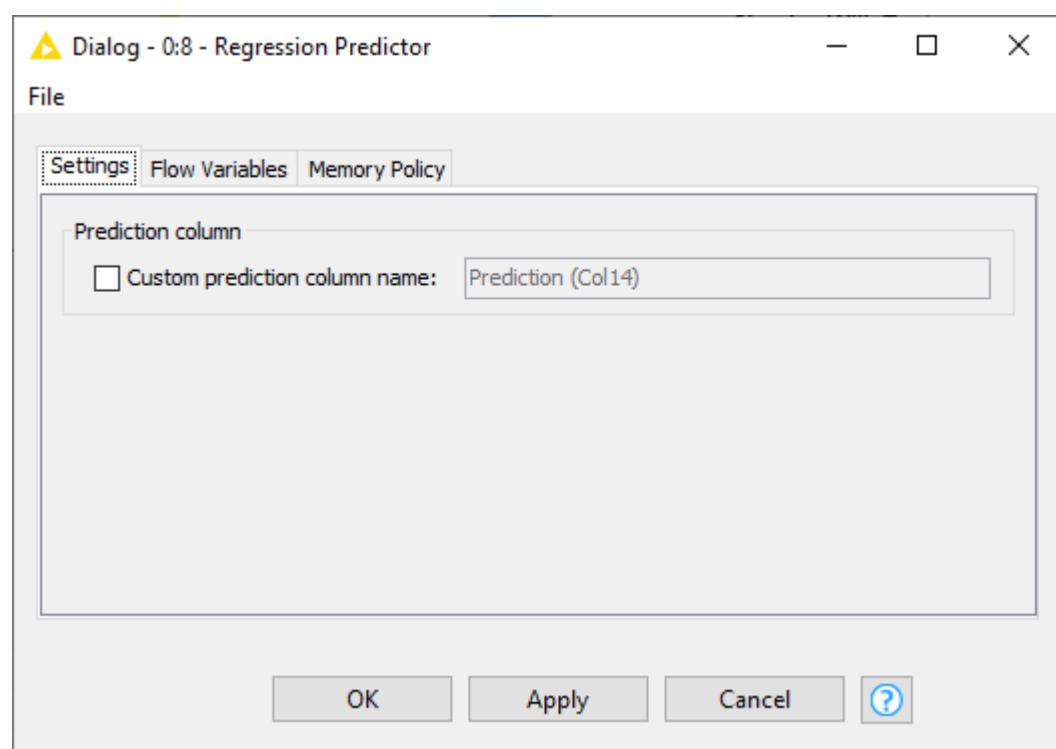


Fig 100. Configuracion del nodo

**▲ Predicted data - 0:8 - Regression Predictor**

File Edit Hilite Navigation View

Table "default" - Rows: 452 Spec - Columns: 3 Properties Flow Variables

Row ID	Col0	Col14	Predicti...
Row0	75	63	69.846
Row1	56	53	72.918
Row2	54	75	73.241
Row3	55	71	73.079
Row4	75	72	69.846
Row5	13	84	79.869
Row6	40	70	75.504
Row7	49	67	74.049
Row8	44	64	74.857
Row9	50	63	73.887
Row10	62	70	71.948
Row11	45	72	74.696
Row12	54	73	73.241
Row13	30	56	77.121
Row14	44	72	74.857
Row15	47	76	74.372
Row16	47	67	74.372
Row17	46	70	74.534
Row18	73	66	70.169
Row19	57	66	72.756
Row20	28	76	77.444
Row21	45	66	74.696
Row22	36	77	76.151
Row23	57	69	72.756
Row24	40	68	75.504

Fig 101. Tabla de salida de nuestro nodo Predictor

**Parámetros de Configuración**

Este nodo nos ayudara para calcular los residuales de nuestra predicción y los datos originales veremos mas adelante como deberemos configurarlo.

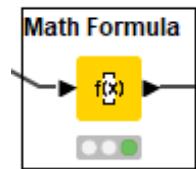


Fig 102. Nodo empleado Math Formula

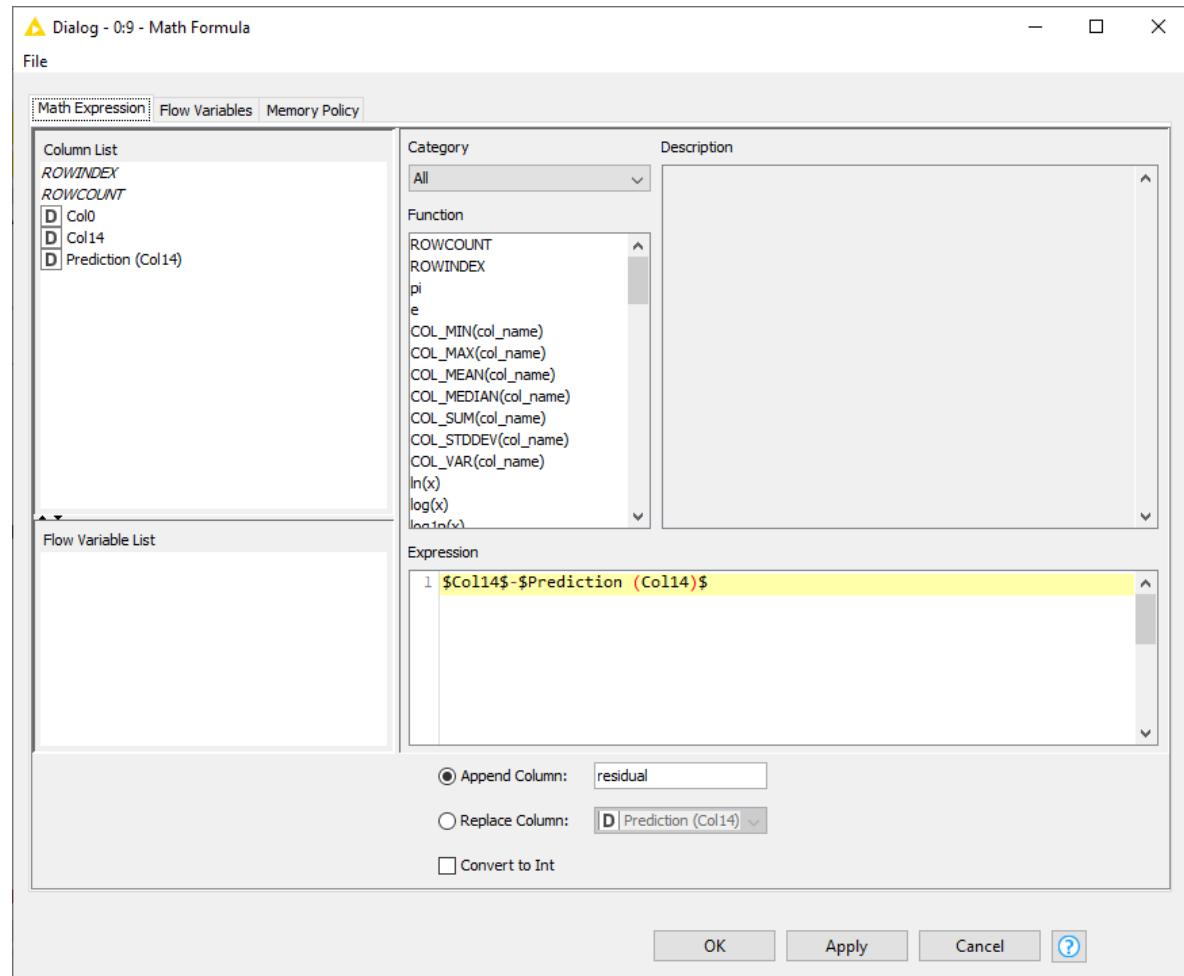


Fig 103. Configuracion del nodo

⚠ Output data - 0:9 - Math Formula

File Edit Hilita Navigation View

Table "default" - Rows: 452 Spec - Columns: 4 Properties Flow Variables

Row ID	Col0	Col14	Predict...	residual
Row0	75	63	69.846	-6.846
Row1	56	53	72.918	-19.918
Row2	54	75	73.241	1.759
Row3	55	71	73.079	-2.079
Row4	75	72	69.846	2.154
Row5	13	84	79.869	4.131
Row6	40	70	75.504	-5.504
Row7	49	67	74.049	-7.049
Row8	44	64	74.857	-10.857
Row9	50	63	73.887	-10.887
Row10	62	70	71.948	-1.948
Row11	45	72	74.696	-2.696
Row12	54	73	73.241	-0.241
Row13	30	56	77.121	-21.121
Row14	44	72	74.857	-2.857
Row15	47	76	74.372	1.628
Row16	47	67	74.372	-7.372
Row17	46	70	74.534	-4.534
Row18	73	66	70.169	-4.169
Row19	57	66	72.756	-6.756
Row20	28	76	77.444	-1.444
Row21	45	66	74.696	-8.696
Row22	36	77	76.151	0.849
Row23	57	69	72.756	-3.756
Row24	40	68	75.504	-7.504
Row25	44	63	74.857	-11.857
Row26	34	83	76.474	6.526
Row27	31	67	76.959	-9.959
Row28	56	79	72.918	6.082

Fig 104. Tabla de salida de nuestro nodo Predictor

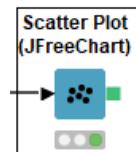


Fig 105. Nodo empleado Regression Predictor

### Parámetros de Configuración

Al igual que al inicio usaremos este nodo para ver la representación gráfica de los residuales con nuestra variable dependiente, veremos mas adelante la configuración.

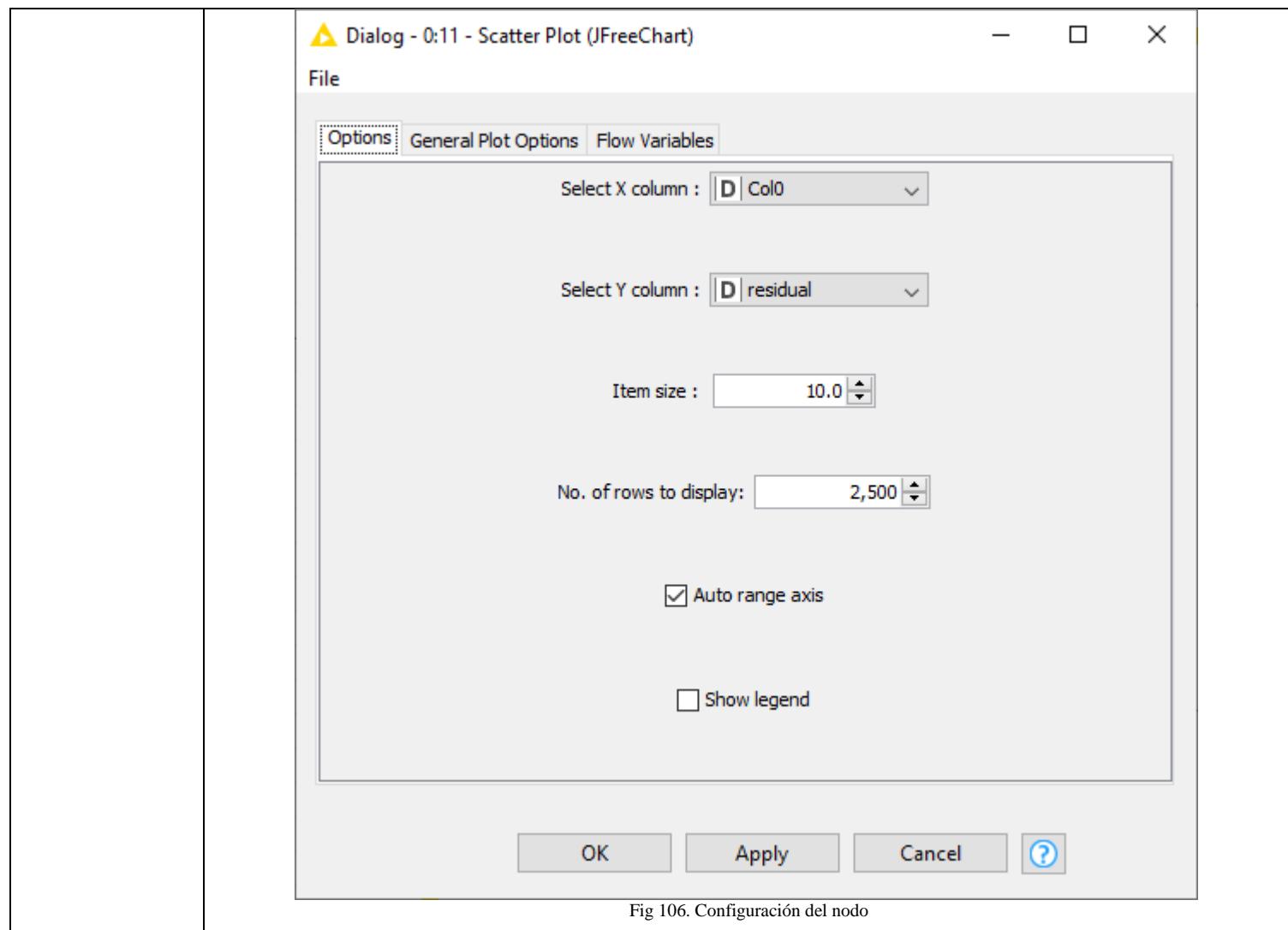


Fig 106. Configuración del nodo

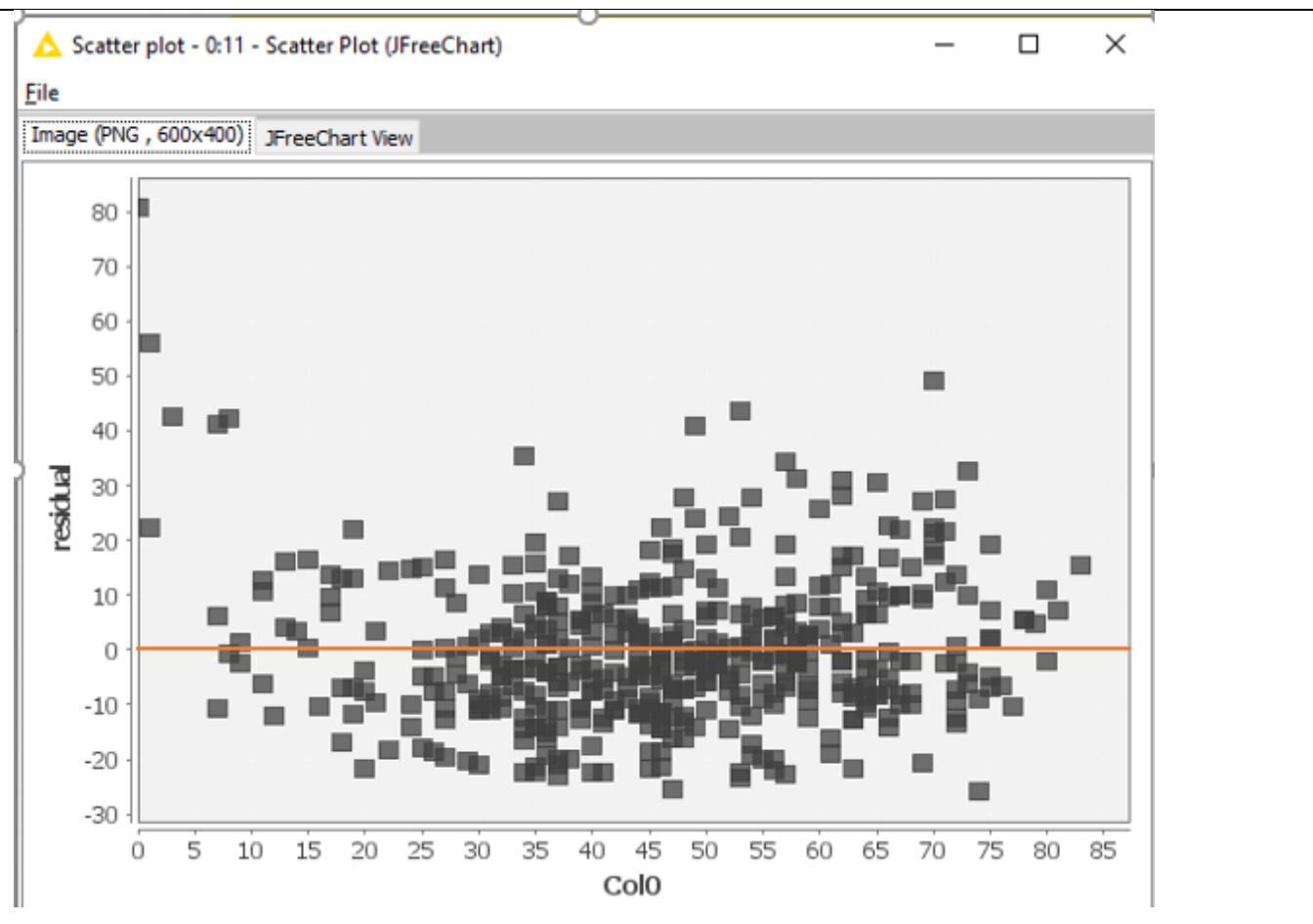


Fig 107. Grafico Representativo

## Significancia Estadística

Ahora bien este término en pocas palabras nos dice si el resultado que obtuvimos tiene probabilidades de que haya sido generado al azar, para determinarlo tendremos en cuenta el resultado de p-Value que obtuvimos del nodo de Kolmogorov-Test que vemos que nos indica que el p-Value es 0.

Por lo tanto si  $\alpha=0.05$  como nivel de significancia,  $p-Value < \alpha$  se rechaza  $H_0$  y se concluye que hay una relación estadísticamente significativa entre la edad del paciente y el número de latidos por minuto de cada paciente.

La probabilidad de que los resultados mostrados se deban al azar es de 0.

## Regresión Lineal Múltiple



## Regresión Múltiple

### Enunciado del Problema

El conjunto de datos con el que hemos estado trabajando a lo largo de este reporte, contiene parámetros que se obtuvieron de personas que tienen arritmias, hemos ya trabajado con ellos en varios métodos anteriores, veremos en este apartado como nos servirán estos parámetros para la regresión lineal múltiple.

El objetivo de este apartado es predecir el número de latidos del corazón de los pacientes, en base a la edad, el peso, la duración media entre el inicio de las ondas P y Q en mseg, la duración media entre el inicio de Q y el desplazamiento de ondas T en mseg, la duración media de la onda T en mseg, la duración media de la onda P en mseg y el ángulo vectorial en grados en el plano frontal de QRS, utilizando modelos predictivos de regresión lineal multiple.

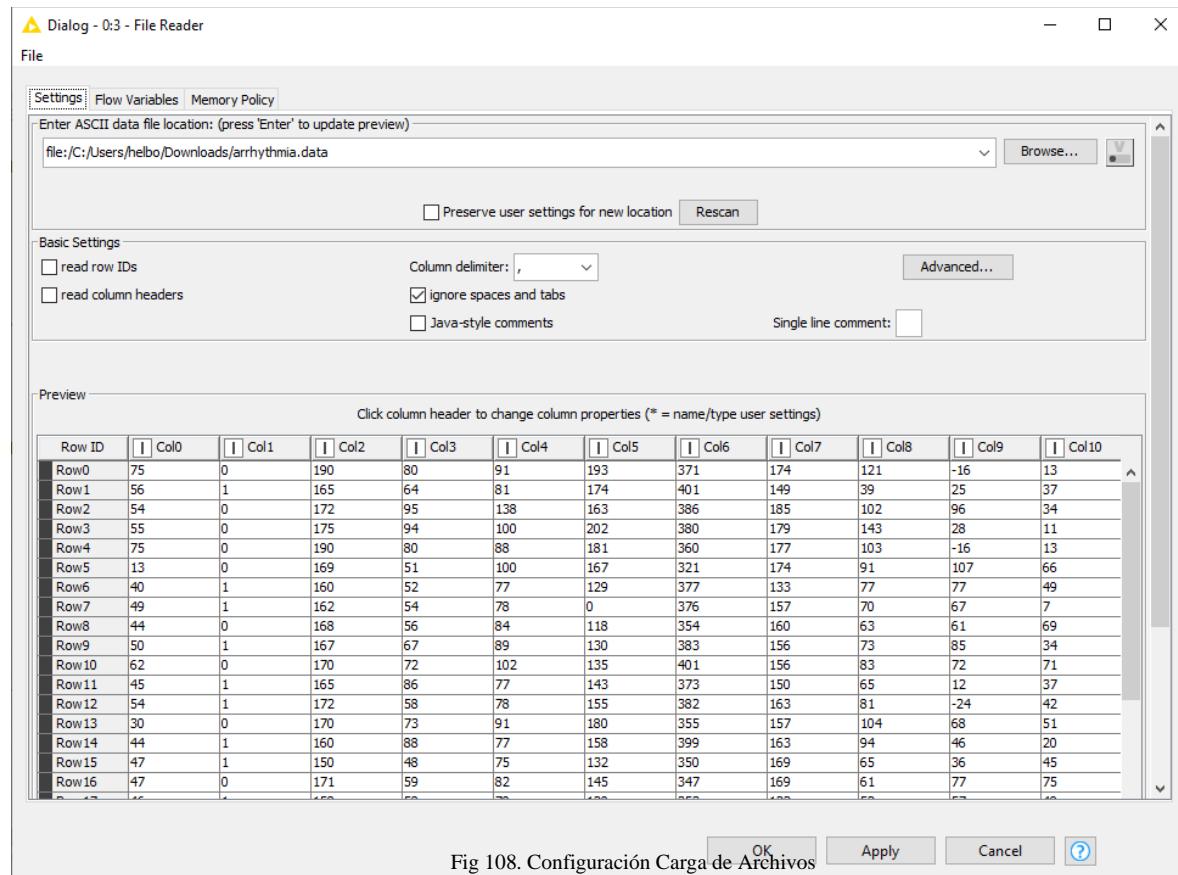
### Diccionario de Datos

#	Nombre	Significado	Tipo	Dominio
<b>1</b>	Age	Edad en años	Lineal	0-89
<b>4</b>	Weight	Peso en kg	Lineal	10-104
<b>6</b>	P-R interval	Duración media entre el inicio de las ondas P y Q en mseg	Lineal	0-524
<b>7</b>	Q-T interval	Duración media entre el inicio de Q y el desplazamiento de ondas T en mseg	Lineal	241-509
<b>8</b>	T interval	Duración media de la onda T en mseg	Lineal	0-205
<b>9</b>	P interval	Duración media de la onda P en mseg	Lineal	-172-169
<b>10</b>	QRS	Ángulo vectorial en grados en el plano frontal de QRS	Lineal	-144-177
<b>15</b>	Heart rate	Número de latidos del corazón por minuto	Lineal	0-88

## Desarrollo: Proceso KDD

### Carga del archivo:

El conjunto de datos a usar se llama arrhythmia.data



## Limpieza de los Datos

Al revisar el conjunto de datos en busca de datos nulos, nos encontramos que tiene inconsistencias por lo que procedemos a hacer una limpieza de estos datos para no afectar las predicciones de nuestro modelo.

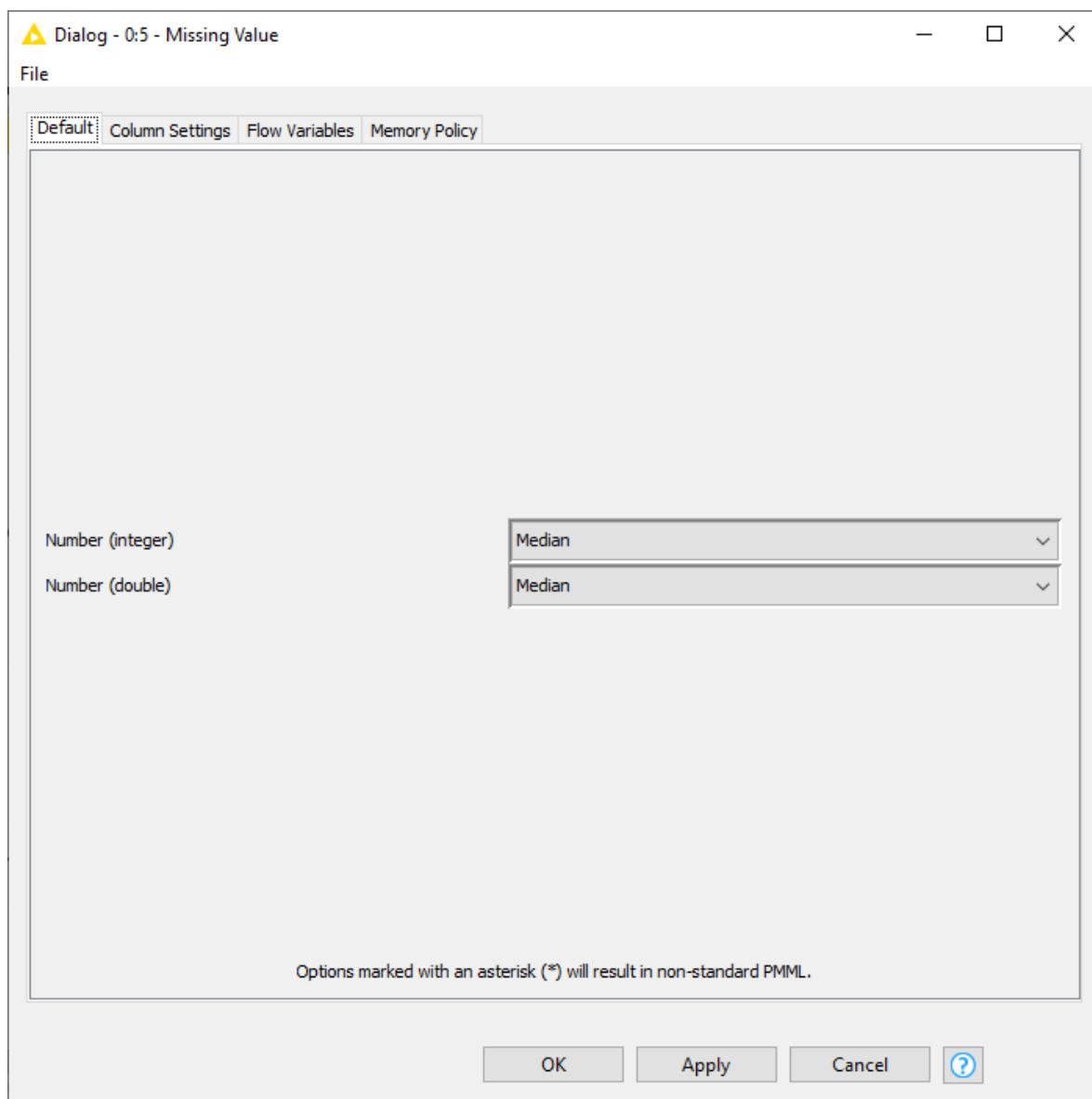
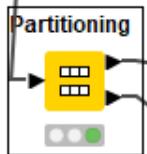
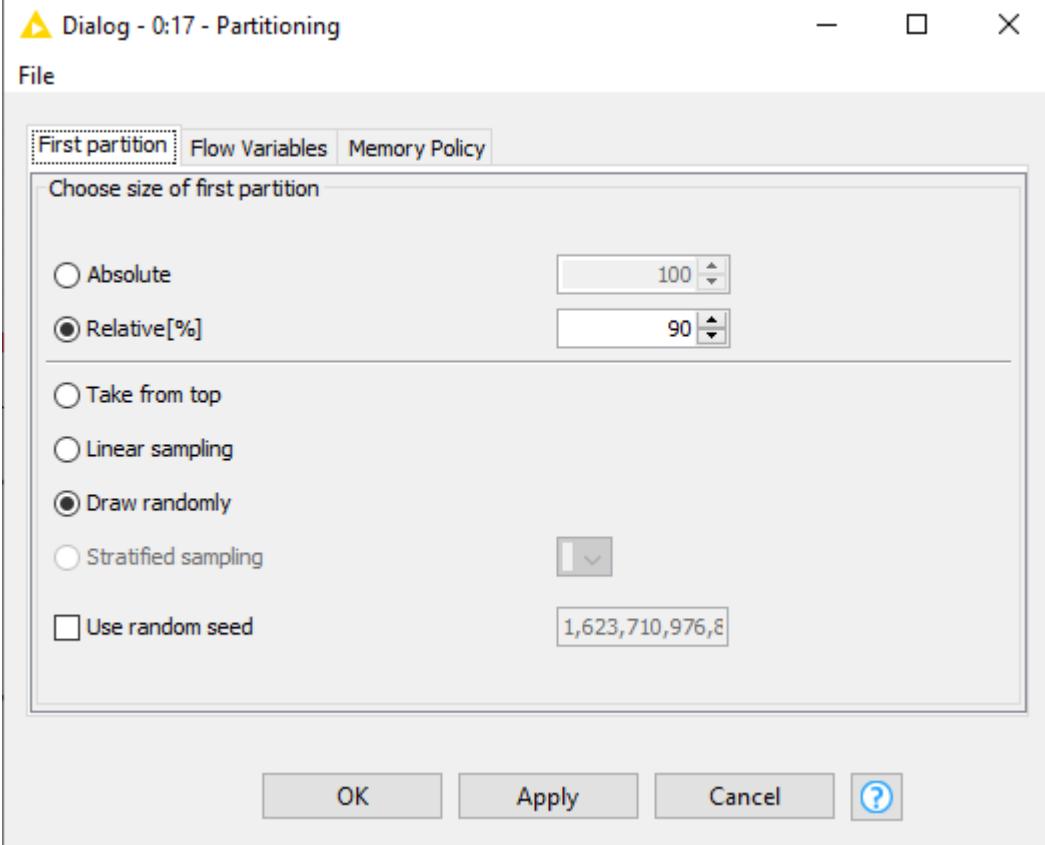


Fig 109. Tratamiento de inconsistencias

## Aplicación

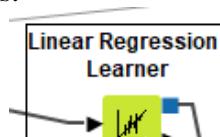
<b>Herramienta</b>	KNIME
<b>Parámetros de Configuración</b>	<p>Necesitaremos dos particiones para este modelo, lo partiremos para usar una de ellas colmo entrenamiento y la otra para la etapa de la predicción, tenemos la configuración mas adelante.</p>  <p>Fig 110. Nodo Partitioning</p>
	 <p>Fig 111. Configuración Nodo Partitioning</p>

First partition (as defined in dialog) - 0:17 - Partitioning											
File Edit Hilita Navigation View											
Table "default" - Rows: 406 Spec - Columns: 280 Properties Flow Variables											
Row ID	[D] Col0	[D] Col1	[D] Col2	[D] Col3	[D] Col4	[D] Col5	[D] Col6	[D] Col7	[D] Col8	[D] Col9	[D]
Row0	75	0	190	80	91	193	371	174	121	-16	13
Row1	54	0	172	95	138	163	386	185	102	96	34
Row2	55	0	175	94	100	202	380	179	143	28	11
Row3	40	1	160	52	77	129	377	133	77	77	49
Row4	49	1	162	54	78	0	376	157	70	67	7
Row5	44	0	168	56	84	118	354	160	63	61	69
Row6	50	1	167	67	89	130	383	156	73	85	34
Row7	62	0	170	72	102	135	401	156	83	72	71
Row8	45	1	165	86	77	143	373	150	65	12	37
Row9	54	1	172	58	78	155	382	163	81	-24	42
Row10	30	0	170	73	91	180	355	157	104	68	51
Row11	44	1	160	88	77	158	399	163	94	46	20
Row12	47	1	150	48	75	132	350	169	65	36	45
Row13	46	1	158	58	70	120	353	122	52	57	49
Row14	73	0	165	63	91	154	392	175	83	73	-24
Row15	57	1	166	72	82	181	399	158	79	-12	28
Row16	28	1	160	58	83	251	383	189	183	50	39
Row17	36	1	153	75	71	132	364	169	82	62	56
Row18	57	1	165	59	75	157	406	143	92	4	10
Row19	40	1	153	55	82	140	388	149	82	52	17
Row20	44	0	169	80	109	128	382	195	60	-34	112
Row21	34	0	170	73	94	186	373	224	125	90	52
Row22	31	1	160	54	95	161	407	168	83	10	48
Row23	56	1	164	65	90	164	420	381	99	-8	153

Fig 112. Primera Partición

Second partition (remaining rows) - 0:17 - Partitioning											
File Edit Hilita Navigation View											
Table "default" - Rows: 46 Spec - Columns: 280 Properties Flow Variables											
Row ID	[D] Col0	[D] Col1	[D] Col2	[D] Col3	[D] Col4	[D] Col5	[D] Col6	[D] Col7	[D] Col8	[D] Col9	[D]
Row1	56	1	165	64	81	174	401	149	39	25	37
Row4	75	0	190	80	88	181	360	177	103	-16	13
Row5	13	0	169	51	100	167	321	174	91	107	66
Row6	47	0	171	59	82	145	347	169	61	77	75
Row7	45	0	169	67	90	122	336	177	78	81	78
Row8	24	1	163	53	92	157	370	142	68	64	45
Row9	46	0	165	66	91	176	372	161	79	42	-24
Row10	54	1	160	63	82	158	410	141	87	25	41
Row11	37	1	171	85	94	148	377	142	81	30	-56
Row12	19	0	165	50	96	151	373	147	102	68	175
Row13	64	1	155	88	82	194	342	138	126	-4	41
Row14	50	0	184	96	94	160	360	203	89	26	143
Row15	28	1	159	56	96	153	340	152	90	-16	19
Row16	58	0	186	18	87	166	372	150	96	-1	63
Row17	41	1	155	56	80	134	362	156	58	59	46
Row18	51	1	160	80	85	167	355	151	127	36	36
Row19	34	1	155	54	86	127	369	157	78	23	10
Row20	39	1	160	62	80	123	375	153	67	60	42
Row21	34	1	167	60	63	164	396	139	84	49	10
Row22	45	1	162	61	75	122	357	143	72	50	8
Row23	46	1	153	70	84	153	383	181	86	-14	54
Row24	54	0	170	78	113	216	414	193	170	50	64
Row25	44	0	178	89	106	183	380	147	94	-2	77
Row26	50	0	168	80	95	159	358	166	96	-50	50

Fig 113. Segunda Partición

<b>Parámetros de Configuración</b>	Ocupamos el nodo Linear Regression Learner para crear el modelo de decisiones que usara toda la muestra de nuestro conjunto de datos.
	
Seleccionamos nuestra variable objetivo y las variables que nos ayudaran en nuestro modelo	

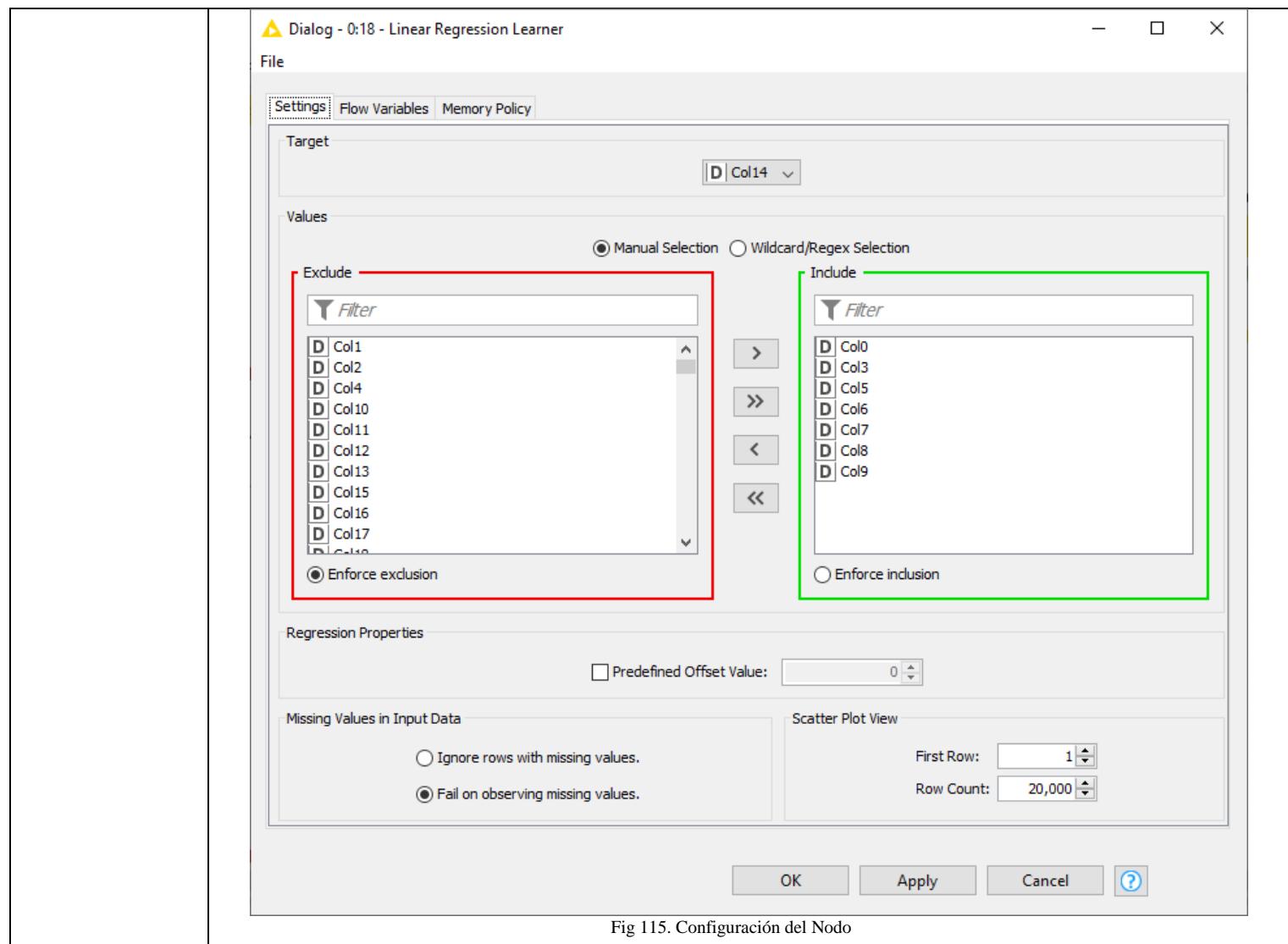


Fig 115. Configuración del Nodo

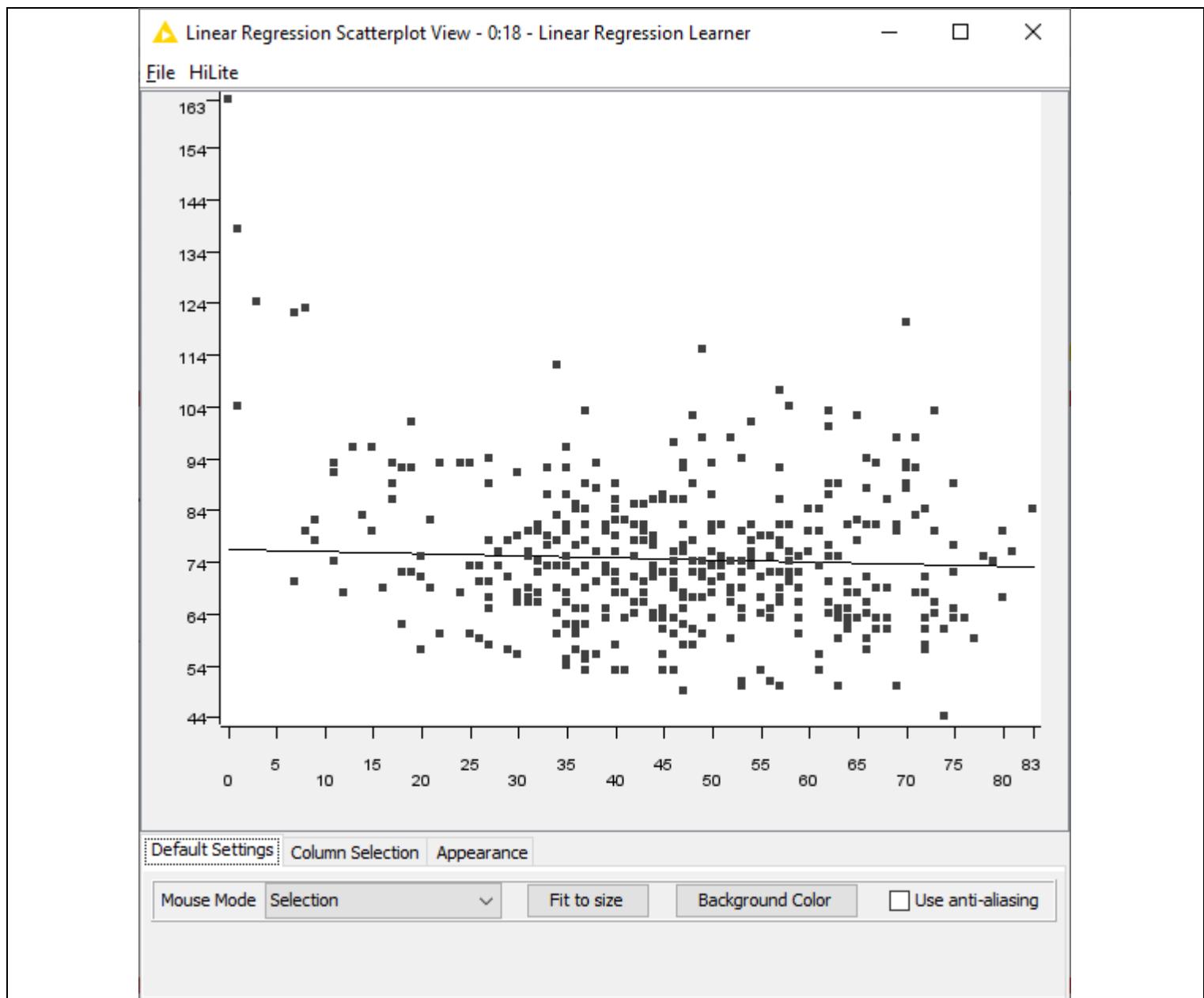


Fig 116. Salida gráfica de nuestro modelo

### Parámetros de Configuración

En este nodo haremos uso de las particiones que realizamos en el nodo de partitioning una que es la que entrenamos y la de prueba.

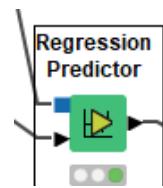


Fig 117. Nodo empleado Regression Predictor

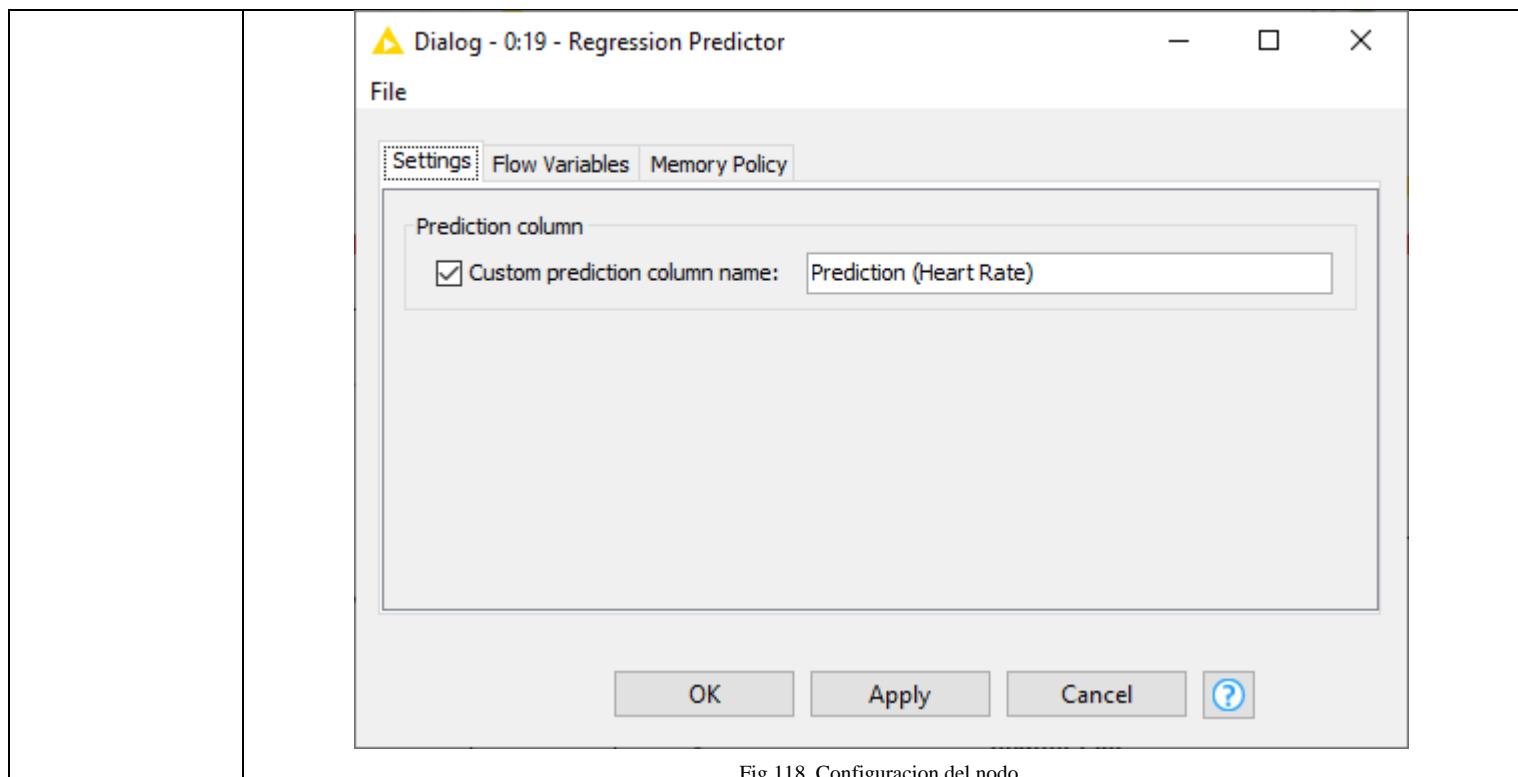


Fig 118. Configuracion del nodo

**▲ Predicted data - 0:19 - Regression Predictor**

File Edit Hilite Navigation View

Table "default" - Rows: 46 Spec - Columns: 281 Properties Flow Variables

Row ID	Col271	Col272	Col273	Col274	Col275	Col276	Col277	Col278	Col279	Col280	Predict...
Row1	8.5	0	0	0	0.2	2.1	20.4	38.8	6	60.269	
Row4	13.1	-3.6	0	0	-0.1	3.9	25.4	62.8	7	75.106	
Row5	12.2	-2.8	0	0	0.9	2.2	13.5	31.1	14	89.826	
Row16	9.4	-1.7	0	0	0.6	2.3	19.5	41.1	10	78.809	
Row21	8.3	-1.8	0	0	0.8	1.1	11.7	19.6	1	83.292	
Row53	16.5	0	0	0	0.4	2.1	44	61.6	16	73.089	
Row58	12.8	-2.1	0	0	0.7	-0.8	25.7	15.2	6	71.781	
Row66	7.3	-2.1	0	0	0.6	-0.3	11	8.3	2	61.705	
Row77	7.5	-2.2	0	0	0.4	-0.6	14.7	10.7	2	68.832	
Row86	15.4	-2.1	0	0	0.3	-2.2	32.4	3.4	2	75.777	
Row91	4.9	-0.9	0.5	0	0.3	0.4	9.6	12.8	4	78.983	
Row109	8.8	0	0	0	0.4	0.3	26.4	28.8	1	75.18	
Row117	3.1	-1.8	0	0	0.6	1.6	1.8	13	1	83.124	
Row126	6.3	-2.9	0	0	0.7	1.4	5.7	15.7	1	77.154	
Row134	7.1	-1.4	0	0	0.6	2.9	12.8	38.9	10	74.772	
Row139	6.4	-1.3	0	0	0.6	1.5	12	24	1	78.128	
Row147	8.6	-0.8	0	0	0.7	1.8	17.5	31.9	1	75.225	
Row148	5.3	-0.7	0	0	0.4	1.9	9.2	23.2	1	71.593	
Row153	8.7	0	0	0	0.4	1.5	22.6	33.4	6	65.997	
Row156	11.6	0	0	0	0.7	1.4	27.8	35.9	1	76.178	
Row159	6.6	-2	0	0	0.3	1.5	9.7	22.9	1	70.708	
Row162	8.9	0	0	0	1	3.2	33	61.8	1	66.646	
Row170	8.2	-1.3	0	0	0.5	0.8	17.9	24.1	1	67.822	
Row171	8.1	-6.5	0	0	0.6	0.8	-5.9	0.1	1	76.172	

Fig 119. Tabla de salida de nuestro nodo Predictor

**Parámetros de Configuración**

Este nodo nos ayudara para calcular los residuales de nuestra predicción y los datos originales veremos mas adelante como deberemos configurarlo

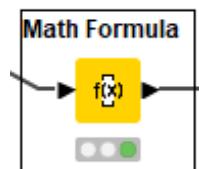


Fig 120. Nodo empleado Math Formula

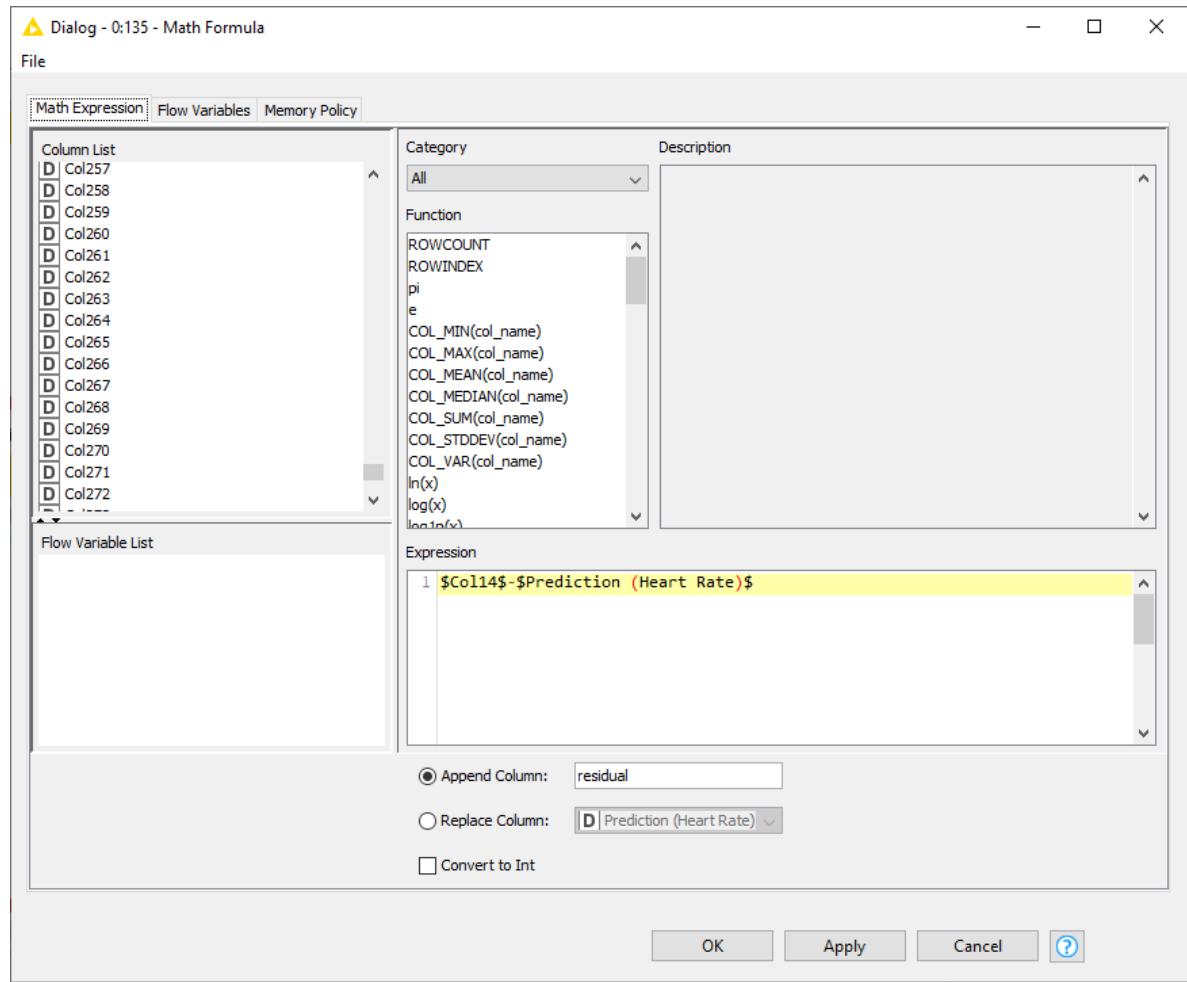


Fig 121. Configuracion del nodo

Table "default" - Rows: 46 Spec - Columns: 282 Properties Flow Variables												
Row ID	Col272	Col273	Col274	Col275	Col276	Col277	Col278	Col279	Predicti...	residual		
Row1	0	0	0	0.2	2.1	20.4	38.8	6	60.269	-7.269		
Row4	-3.6	0	0	-0.1	3.9	25.4	62.8	7	75.106	-3.106		
Row5	-2.8	0	0	0.9	2.2	13.5	31.1	14	89.826	-5.826		
Row16	-1.7	0	0	0.6	2.3	19.5	41.1	10	78.809	-11.809		
Row21	-1.8	0	0	0.8	1.1	11.7	19.6	1	83.292	-17.292		
Row53	0	0	0	0.4	2.1	44	61.6	16	73.089	-9.089		
Row58	-2.1	0	0	0.7	-0.8	25.7	15.2	6	71.781	-15.781		
Row66	-2.1	0	0	0.6	-0.3	11	8.3	2	61.705	-7.705		
Row77	-2.2	0	0	0.4	-0.6	14.7	10.7	2	68.832	1.168		
Row86	-2.1	0	0	0.3	-2.2	32.4	3.4	2	75.777	-8.777		
Row91	-0.9	0.5	0	0.3	0.4	9.6	12.8	4	78.983	6.017		
Row109	0	0	0	0.4	0.3	26.4	28.8	1	75.18	-7.18		
Row117	-1.8	0	0	0.6	1.6	1.8	13	1	83.124	2.876		
Row126	-2.9	0	0	0.7	1.4	5.7	15.7	1	77.154	-7.154		
Row134	-1.4	0	0	0.6	2.9	12.8	38.9	10	74.772	-4.772		
Row139	-1.3	0	0	0.6	1.5	12	24	1	78.128	6.872		
Row147	-0.8	0	0	0.7	1.8	17.5	31.9	1	75.225	-13.225		
Row148	-0.7	0	0	0.4	1.9	9.2	23.2	1	71.593	-6.593		
Row153	0	0	0	0.4	1.5	22.6	33.4	6	65.997	-11.997		
Row156	0	0	0	0.7	1.4	27.8	35.9	1	76.178	0.822		
Row159	-2	0	0	0.3	1.5	9.7	22.9	1	70.708	2.292		
Row162	0	0	0	1	3.2	33	61.8	1	66.646	-5.646		
Row170	-1.3	0	0	0.5	0.8	17.9	24.1	1	67.822	2.178		
Row171	-6.5	0	0	0.6	0.8	-5.9	0.1	1	76.172	-3.172		

Fig 122. Tabla de salida de nuestro nodo Predictor

### Parámetros de Configuración

Al igual que al inicio usaremos este nodo para ver la representación gráfica de los residuales con nuestra variable dependiente, veremos más adelante la configuración.

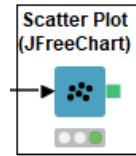


Fig 123. Nodo empleado Regression Predictor

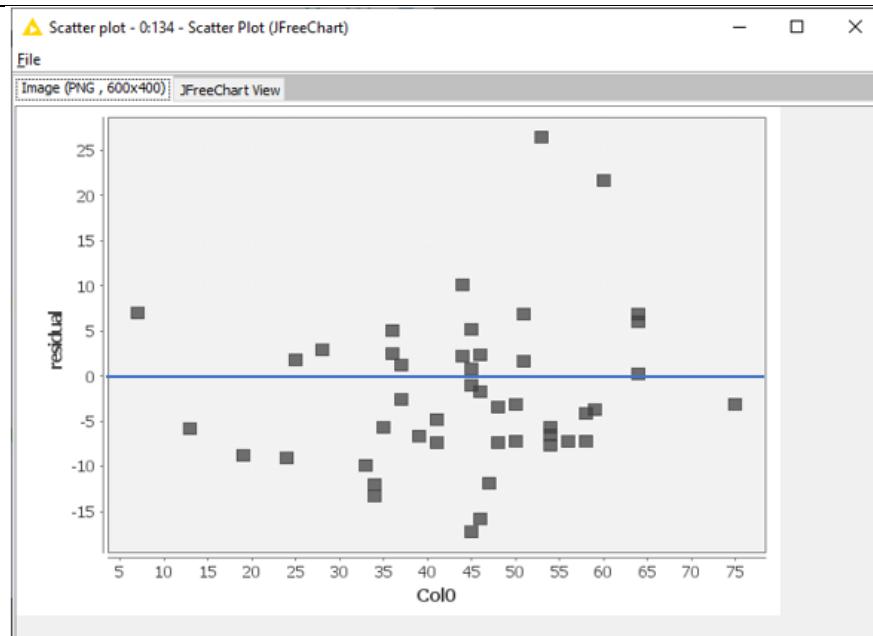


Fig 125. Configuración del nodo

<b>Parámetros de Configuración</b>	Este nodo nos ayudara para calcular el p-Value que usaremos para determinar la significancia estadistica de nuestras variables.

Fig 126. Nodo empleado Kolmogorov-Smirnov Test

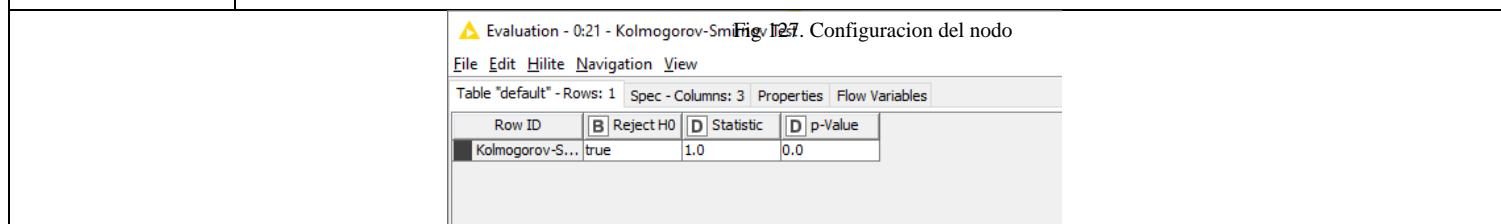
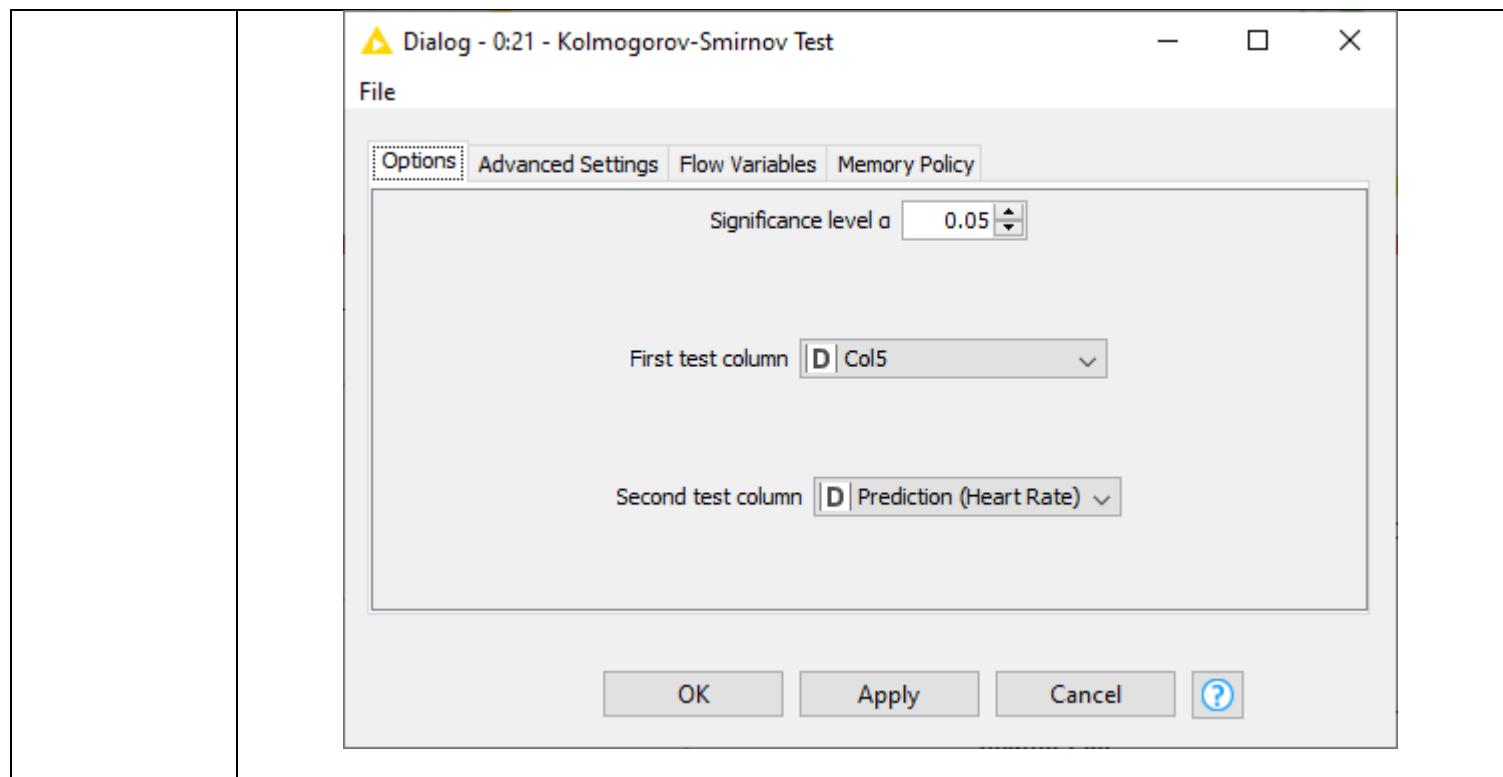


Fig 127. Configuracion del nodo

**Parámetros de Configuración** Por ultimo el nodo scorer nos brindara mas informacion acerca de nuestro modelo, veremos la configuracion mas adelante.

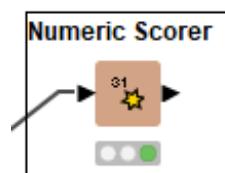
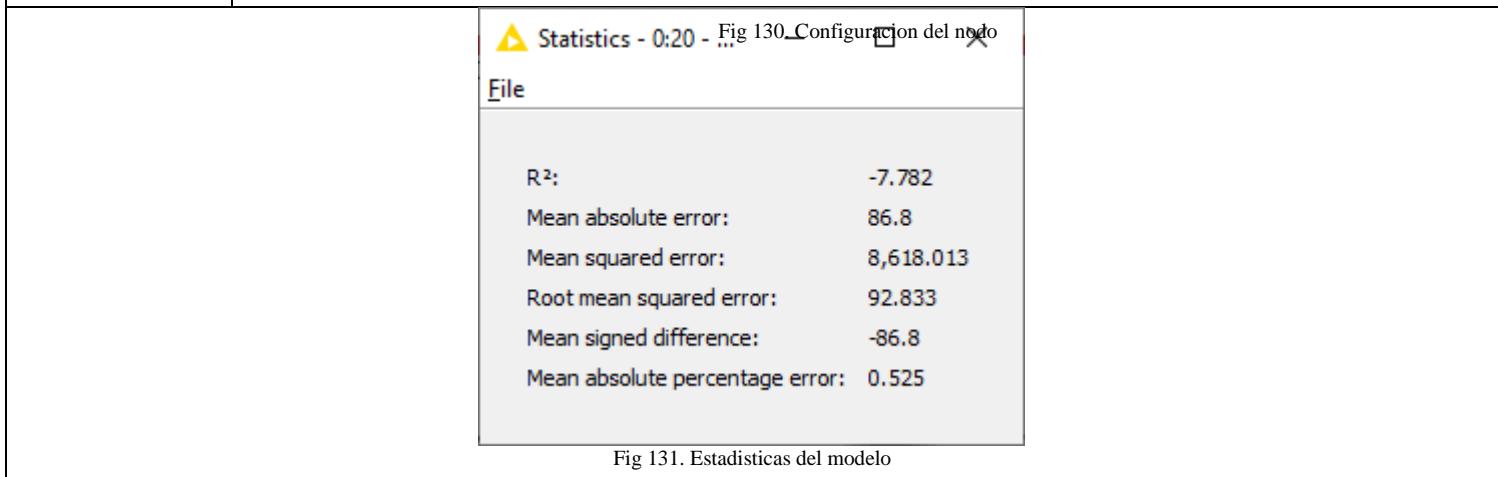
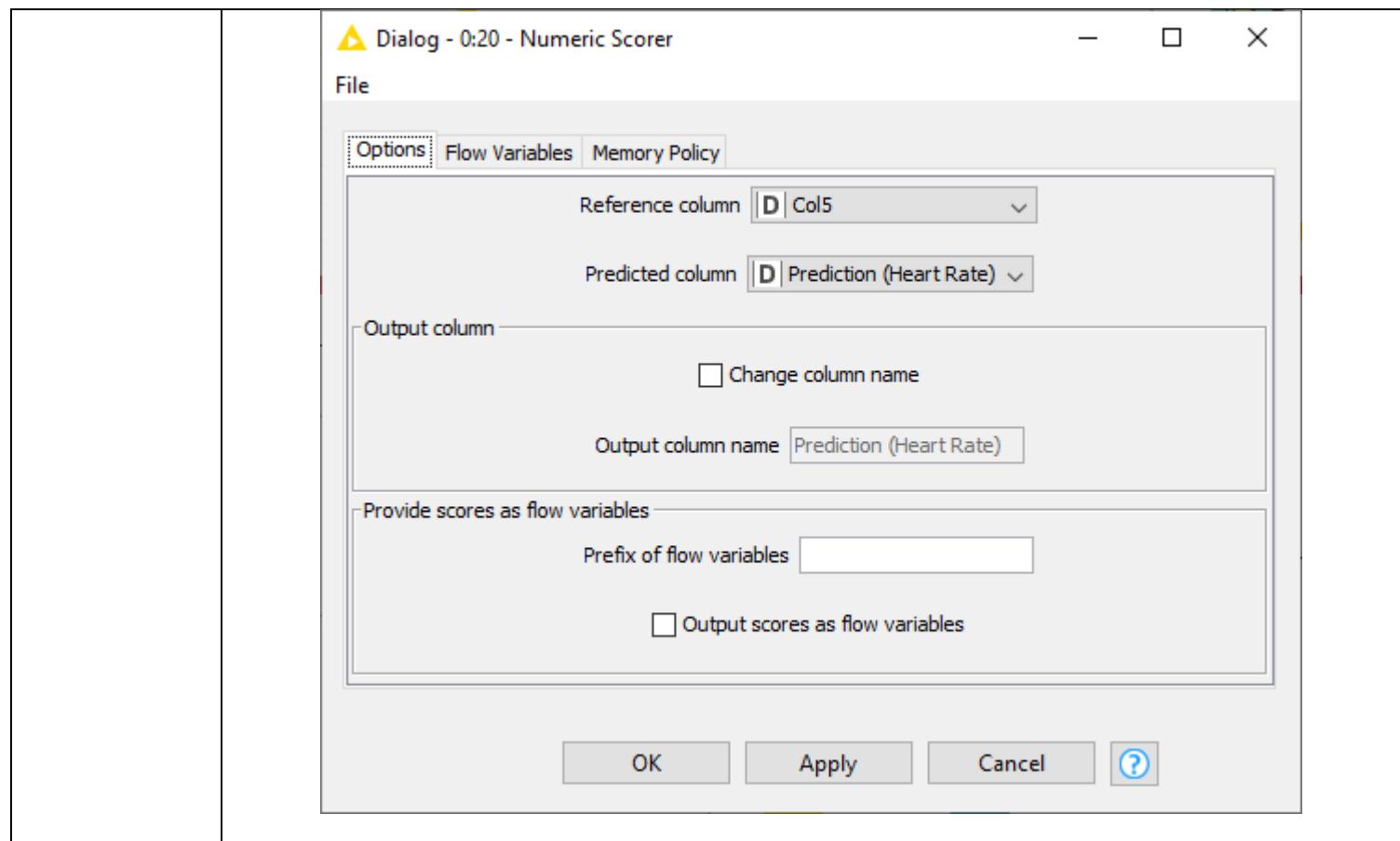


Fig 128. Resultados



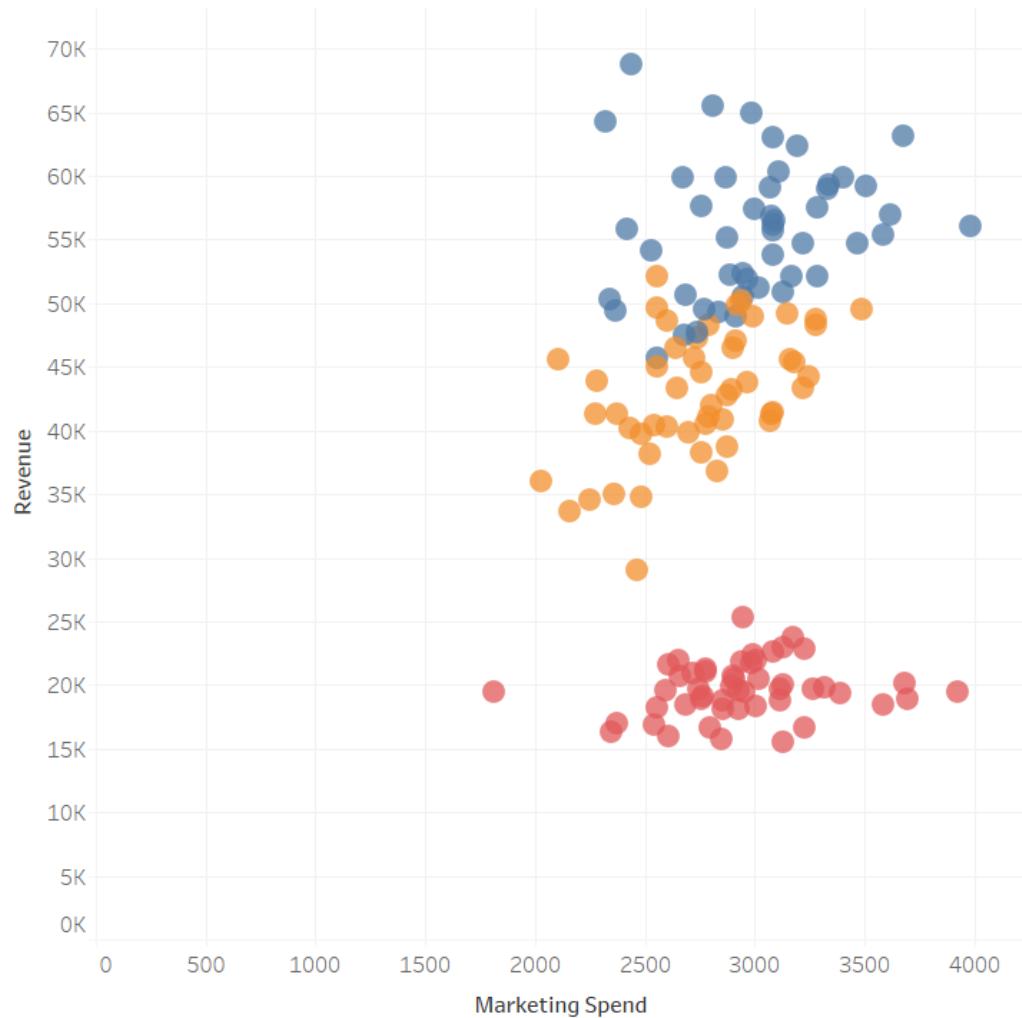
## Significancia Estadística

Ahora bien este término en pocas palabras nos dice si el resultado que obtuvimos tiene probabilidades de que haya sido generado al azar, para determinarlo tendremos en cuenta el resultado de p-Value que obtuvimos del nodo de Kolmogorov-Test que vemos que nos indica que el p-Value es 0.

Por lo tanto si  $\alpha=0.05$  como nivel de significancia,  $p\text{-Value} < \alpha$  se rechaza  $H_0$  y se concluye que hay una relación estadísticamente significativa entre la edad del paciente, el peso, la duración media entre el inicio de las ondas P y Q en mseg, la duración media entre el inicio de Q y el desplazamiento de ondas T en mseg, la duración media de la onda T en mseg, la duración media de la onda P en mseg y el ángulo vectorial en grados en el plano frontal de QRS y el número de latidos por minuto de cada paciente. La probabilidad de que los resultados mostrados se deban al azar es de 0.

## Clúster

### Clustering



## Clúster

**Tabla comparativa**

Algoritmo	Tipo	Características
Aglomerativo	Jerárquico	<ul style="list-style-type: none"> <li>• Inicia con cada elemento en un clúster, luego fusiona todos en uno de forma iterativa.</li> <li>• Utiliza una matriz de adyacencia con distancias entre puntos: single link, average link, complete link.</li> <li>• Genera un dendograma.</li> <li>• Alta complejidad: <math>O(n^2)</math></li> <li>• No es incremental (dinámico).</li> </ul>
MST Single Link	Jerárquico aglomerativo	<ul style="list-style-type: none"> <li>• Encontrar el número máximo de elementos conectados en un grafo.</li> <li>• Dos clústeres se fusionan si la distancia mínima entre ellos es menor o igual que una distancia umbral dada.</li> <li>• Complejidad en cada iteración: <math>O(n^2)</math></li> <li>• Los clúster se fusionan en orden creciente de la distancia encontrada en un Minimum Spanning Tree.</li> <li>• Efecto de cadena: clusters con elementos sin relación, sólo eran cercanos.</li> </ul>
MST Average Link	Jerárquico aglomerativo	<ul style="list-style-type: none"> <li>• Dos clúster se fusionan si la distancia promedio entre ellos es menor o igual que una distancia umbral dada.</li> <li>• Se debe examinar todo el grafo completo en cada iteración, y no solo el de umbral.</li> </ul>
MST Complete Link	Jerárquico aglomerativo	<ul style="list-style-type: none"> <li>• Dos clúster se fusionan si la distancia máxima entre ellos es menor o igual que una distancia umbral dada.</li> <li>• Encuentra cliques en vez de elementos conectados.</li> <li>• Clique: es un grafo máximo en el que hay un borde entre dos vértices.</li> <li>• Complejidad: <math>O(n^2)</math></li> <li>• Clúster más compactos.</li> <li>• Farthest Neighbor Algorithm.</li> </ul>
MST Divisivo	Jerárquico divisivo	<ul style="list-style-type: none"> <li>• Todos los elementos se colocan inicialmente en un cluster y se dividen repetidamente en dos hasta que todos los elementos estén en su propio cluster.</li> <li>• La distancia entre elementos de los clusters es el criterio de separación.</li> <li>• Se quitan los bordes del más grande a los más pequeños.</li> </ul>
MST Particional	Particional	<ul style="list-style-type: none"> <li>• Inicia con un clúster, y a partir de ese va generando el número de clúster ingresado; es inverso al aglomerativo.</li> <li>• Distancia promedio entre clúster como criterio de calidad en los resultados.</li> <li>• Alto número de combinaciones de posibles soluciones.</li> <li>• Identifica bordes inconsistentes utilizando el peso (distancia) de un borde en comparación con aquellos cercanos a él.</li> </ul>

		<ul style="list-style-type: none"> <li>• Complejidad: <math>O(n)</math></li> </ul>
Error al cuadrado	Particional	<ul style="list-style-type: none"> <li>• En este algoritmo se usa la minimización del error cuadrático para determinar a qué clúster pertenece el punto. Esta técnica la usa el algoritmo K-Medias.</li> <li>• Error cuadrático = <math>(\text{real} - \text{estimado})^2</math></li> </ul>
K-Means	Particional	<ul style="list-style-type: none"> <li>• K-medias es un método de agrupamiento, que tiene como objetivo la partición de un conjunto de <math>n</math> observaciones en <math>k</math> grupos en el que cada observación pertenece al grupo cuyo valor medio es más cercano.</li> <li>• <math>O(n dk + 1 \log n)</math>, donde <math>n</math> es el número de entidades a particionar</li> <li>• Utiliza la medida entre el K centro y sus <math>n</math> puntos.</li> <li>• Utiliza la medida Euclídea o Mattahan</li> </ul>
Nearest Neighbor	Particional	<ul style="list-style-type: none"> <li>• Supervisado: esto -brevemente- quiere decir que tenemos etiquetado nuestro conjunto de datos de entrenamiento, con la clase o resultado esperado dada «una fila» de datos.</li> <li>• Basado en Instancia: Esto quiere decir que nuestro algoritmo no aprende explícitamente un modelo (como por ejemplo en Regresión Logística o árboles de decisión). En cambio, memoriza las instancias de entrenamiento que son usadas como «base de conocimiento» para la fase de predicción.</li> <li>• Se utiliza en la resolución de multitud de problemas, como en sistemas de recomendación, búsqueda semántica y detección de anomalías.</li> <li>• Complejidad <math>O(kN^{(1-1/k)})</math></li> </ul>
PAM	Particional	<ul style="list-style-type: none"> <li>• La técnica de clustering de partición entorno a centroides (PAM) realiza una distribución de los elementos entre un número prefijado de clústeres o grupos.</li> <li>• Esta técnica recibe como dato de entrada el número de clústeres a formar además de los elementos a clasificar y la matriz de similitudes.</li> <li>• Explorar todas las posibles particiones es computacionalmente intratable.</li> <li>• Complejidad <math>O(k(n-k)2)</math>.</li> </ul>
CLARANS	Particional	<ul style="list-style-type: none"> <li>• Agrupación de aplicaciones grandes basadas en la búsqueda Aleatoria (Clustering Large Applications based upon RANdomized Search)</li> <li>• Selecciona aleatoriamente <math>k</math> objetos en el conjunto de datos como los medoides actuales.</li> <li>• Luego selecciona aleatoriamente un mediodes actual <math>xy</math> un objeto <math>y</math> que no es uno de los medoides actuales.</li> <li>• Complejidad de <math>k(n-k)</math>.</li> </ul>
Bond Energy	Particional	<ul style="list-style-type: none"> <li>• Usado para determinar cómo agrupar datos y cómo almacenarlos físicamente en disco en función de su uso.</li> <li>• La afinidad entre los atributos se basa en el uso de ellos en conjunto.</li> </ul>

		<ul style="list-style-type: none"> <li>• Usa medida de similitud</li> <li>• Los atributos que son usados juntos crean un clúster y son almacenados juntos.</li> </ul>
BIRCH	Particional	<ul style="list-style-type: none"> <li>• Reducción iterativa equilibrada y agrupamiento mediante jerarquía.</li> <li>• Hace uso del algoritmo CF: Función de Clúster(cluster feature)</li> <li>• Se define por la terna <math>CF=(N, LS, SS)</math>.</li> <li>• No funciona bien para datos de naturaleza no esférica, es decir, que los grupos no son agrupados de forma circular.</li> <li>• Complejidad de <math>O(n)</math>.</li> <li>• Clasificar una gran cantidad de datos.</li> <li>• Requiere escanear solo una vez la BD.</li> <li>• Asume que hay un espacio limitado de memoria.</li> <li>• Es incremental y jerárquico.</li> </ul>
DBSCAN	Mezclado	<ul style="list-style-type: none"> <li>• Clústeres con un tamaño y densidad mínimos.</li> <li>• Densidad: Número mínimo de puntos dentro de una cierta distancia el uno del otro.</li> <li>• Se ingresa el número mínimo de puntos en cualquier clúster.</li> <li>• La distancia Eps como distancia máxima para la medida de densidad.</li> <li>• El vecindario Eps es el conjunto de puntos dentro de la distancia Eps.</li> <li>• El número deseado de clústeres lo determina el algoritmo mismo.</li> </ul>
CURE	Mezclado	<ul style="list-style-type: none"> <li>• Buen manejo de valores atípicos</li> <li>• Es un algoritmo híbrido entre los dos enfoques jerárquico y particional</li> <li>• Se toma un número <math>c</math> de puntos representativos del grupo</li> <li>• Selecciona los <math>c</math> puntos más dispersos del clúster y los comprimen hacia el centroide por un factor de contracción <math>\alpha</math></li> </ul>
ROCK	Aglomerativo categórico	<ul style="list-style-type: none"> <li>• Usa alguna técnica aglomerativa</li> <li>• Una medida de similitud determina el par de puntos que serán unidos.</li> <li>• El enlace entre <math>(ki, kj)</math> es el número de enlaces entre dos clústeres.</li> <li>• Un par de puntos se consideran vecinos si su similitud está por encima de un umbral.</li> </ul>

## Diccionario de Datos

#	Nombre	Significado	Tipo	Dominio
1	Age	Edad en años	Lineal	0-89
2	Sex	Género	Nominal	0 = Masculino; 1 = Femenino
3	Height	Estatura en cm	Lineal	132-190
4	Weight	Peso en kg	Lineal	10-104
5	QRS duration	Promedio de la duración del QRS en mseg	Lineal	61-138
6	P-R interval	Duración media entre el inicio de las ondas P y Q en mseg	Lineal	0-524
7	Q-T interval	Duración media entre el inicio de Q y el desplazamiento de ondas T en mseg	Lineal	241-509
8	T interval	Duración media de la onda T en mseg	Lineal	0-205
9	P interval	Duración media de la onda P en mseg	Lineal	-172-169
10	QRS	Ángulo vectorial en grados en el plano frontal de QRS	Lineal	-144-177
11	T	Ángulo vectorial en grados en el plano frontal de T	Lineal	-93-170
12	P	Ángulo vectorial en grados en el plano frontal de P	Lineal	-170-180
13	QRST	Ángulo vectorial en grados en el plano frontal de QRST	Lineal	-170-180
14	J	Ángulo vectorial en grados en el plano frontal de J	Lineal	0-112
15	Heart rate	Número de latidos del corazón por minuto	Lineal	0-88
16	Q	Anchura media, en mseg. de la onda Q	Lineal	0-156
17	R	Anchura media, en mseg. de la onda R	Lineal	0-88
18	S	Anchura media, en mseg. de la onda S	Lineal	0-24
19	R'	Anchura media, en mseg. de la onda R'	Lineal	0

<b>20</b>	S'	Anchura media, en mseg. de la onda S'	Lineal	0-100
<b>21</b>	Number of intrinsic deflections	Número de deflexiones intrínsecas	Lineal	0-1
<b>22</b>	Existence of ragged R wave	Existencia de onda R irregular	Nominal	0-1
<b>23</b>	Existence of diphasic derivation of R wave	Existencia de derivación difásica de la onda R	Nominal	0-1
<b>24</b>	Existence of ragged P wave	Existencia de onda P irregular	Nominal	0-1
<b>25</b>	Existence of diphasic derivation of P wave	Existencia de derivación difásica de la onda P	Nominal	0-1
<b>26</b>	Existence of ragged T wave	Existencia de onda T irregular	Nominal	0-1
<b>27</b>	Existence of diphasic derivation of T wave, nominal	Existencia de derivación difásica de la onda T	Nominal	0-1
<b>28 a 39</b>	channel DII similar (16 a 27)	...	Nominal y Lineal	0-76
<b>40 a 51</b>	channel DIII	...	Nominal y Lineal	0-116
<b>52 a 63</b>	channel AVR	...	Nominal y Lineal	0-80
<b>64 a 75</b>	channel AVL	...	Nominal y Lineal	0-148
<b>76 a 87</b>	channel AVF	...	Nominal y Lineal	0128
<b>88 a 99</b>	channel V1	...	Nominal y Lineal	0-216
<b>100 a 111</b>	channel V2	...	Nominal y Lineal	0-108
<b>112 a 123</b>	channel V3	...	Nominal y Lineal	0-132
<b>124 a 135</b>	channel V4	...	Nominal y Lineal	0-92
<b>136 a 147</b>	channel V5	...	Nominal y Lineal	0-136
<b>148 a 159</b>	channel V6	...	Nominal y Lineal	0-148
<b>160</b>	JJ wave (channel DI)	Amplitud, * 0,1 milivoltios, de onda JJ	Lineal	-2.7-0
<b>161</b>	Q wave	Amplitud, * 0,1 milivoltios, de onda Q	Lineal	0-2.80

<b>162</b>	R wave	Amplitud, * 0,1 milivoltios, de onda R	Lineal	0-1.9
<b>163</b>	S wave	Amplitud, * 0,1 milivoltios, de onda SS	Lineal	0-0.095
<b>164</b>	R' wave	Amplitud, * 0,1 milivoltios, de onda R'	Lineal	0
<b>165</b>	S' wave	Amplitud, * 0,1 milivoltios, de onda S'	Lineal	-1.5-1.7
<b>166</b>	P wave	Amplitud, * 0,1 milivoltios, de onda P	Lineal	-8.7-3.7
<b>167</b>	T wave	Amplitud, * 0,1 milivoltios, de onda T	Lineal	-33.3-155.2
<b>168</b>	QRSA	Suma de áreas de todos los segmentos dividida por 10, (Área = ancho * alto / 2)	Lineal	-38.8-74.3
<b>169</b>	QRSTA	QRSA + 0.5 * ancho de onda T * 0.1 * altura de T onda. (Si T es difásico, entonces el segmento más grande es considerado)	Lineal	-3.9-1.9
<b>170 a 179</b>	channel DII	...	Lineal	-3.4-0
<b>180 a 189</b>	channel DIII	...	Lineal	0-19.2
<b>190 a 199</b>	channel AVR	...	Lineal	-16.5-0
<b>200 a 209</b>	channel AVL	...	Lineal	0-3.2
<b>210 a 219</b>	channel AVF	...	Lineal	-1.5-0
<b>220 a 229</b>	channel V1	...	Lineal	-1.5-3.4
<b>230 a 239</b>	channel V2	...	Lineal	-30.3-0
<b>240 a 249</b>	channel V3	...	Lineal	0-28.5
<b>250 a 259</b>	channel V4	...	Lineal	-43.3-0
<b>260 a 269</b>	channel V5	...	Lineal	0-14.9
<b>270 a 279</b>	channel V6	...	Lineal	-4-0
<b>280</b>	class	Clasificación de la arritmia	Nominal	[1,16]

## Desarrollo: Proceso KDD

### Tratamiento de los Datos

Statistics Table - 0:2 - Statistics

File Edit Hilito Navigation View

Table "default" - Rows: 280 Spec - Columns: 16 Properties Flow Variables

Row ID	Column	Min	Max	Mean	Std. de...	Variance	Skewness	Kurtosis	Overall ...	No. mis...	No. NaNs	N
Col0	Col0	0	83	46.471	16.467	271.15	-0.287	-0.203	21,005	0	0	0
Col1	Col1	0	1	0.551	0.498	0.248	-0.205	-1.967	249	0	0	0
Col2	Col2	105	780	166.188	37.17	1,381.634	13.724	208.988	75,117	0	0	0
Col3	Col3	6	176	68.17	16.591	275.255	0.167	4.957	30,813	0	0	0
Col4	Col4	55	188	88.92	15.364	236.065	2.566	11.075	40,192	0	0	0

Fig 132. Dominio de los datos antes de la estandarización

En algunos de los algoritmos de segmentación el hecho de no eliminar las unidades puede afectar directamente en su funcionamiento. Como se puede visualizar a través de la Figura 132, el dominio de los datos entre cada uno de los atributos varía mucho, generando conflictos para algoritmos de clustering que se basen en la distancia debido a que habrá atributos que dominen el estudio.

Normalized table - 0:129 - Normalizer

File Edit Hilito Navigation View

Table "default" - Rows: 452 Spec - Columns: 280 Properties Flow Variables

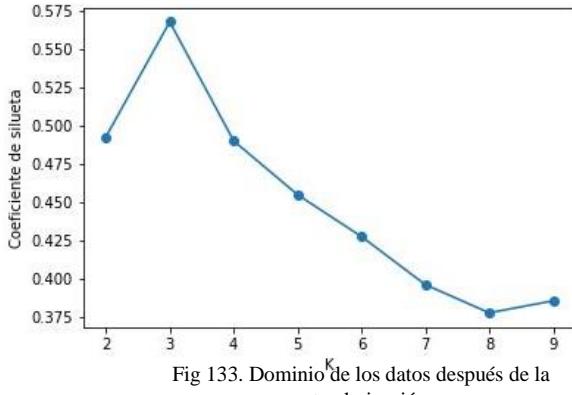
Row ID	[D] Col0	[D] Col1	[D] Col2	[D] Col3	[D] Col4	[D] Col5	[D] Col6	[D] Col7	[D] Col8	[D] Col9	[D] Col10	[D] Col11	[D] Col12	[D] Col13	[D] Col14	[D] Col15	[D] Col16
Row0	0.904	0	0.126	0.435	0.271	0.368	0.502	0.242	0.59	0.457	0.534	0.676	0.442	0.36	0.16	0	0.333
Row1	0.675	1	0.089	0.341	0.195	0.332	0.61	0.15	0.19	0.578	0.601	0.442	0.551	0.36	0.076	0	0.308
Row2	0.651	0	0.099	0.524	0.624	0.311	0.556	0.282	0.498	0.786	0.593	0.694	0.668	0.566	0.261	0	0.256
Row3	0.663	0	0.104	0.518	0.338	0.385	0.534	0.26	0.698	0.587	0.528	0.477	0.515	0.36	0.227	0	0.462
Row4	0.904	0	0.126	0.435	0.248	0.345	0.462	0.253	0.502	0.457	0.534	0.668	0.458	0.36	0.235	0	0.308
Row5	0.157	0	0.095	0.265	0.338	0.319	0.321	0.242	0.444	0.818	0.683	0.642	0.741	0.36	0.336	0	0.231
Row6	0.482	1	0.081	0.271	0.165	0.246	0.523	0.092	0.376	0.73	0.635	0.708	0.664	0.36	0.218	0	0.282
Row7	0.59	1	0.084	0.282	0.173	0	0.52	0.179	0.341	0.701	0.517	0.514	0.618	0.36	0.193	0	0.282
Row8	0.53	0	0.093	0.294	0.218	0.225	0.44	0.19	0.307	0.683	0.691	0.717	0.666	0.737	0.168	0	0.256
Row9	0.602	1	0.092	0.359	0.256	0.248	0.545	0.176	0.356	0.754	0.593	0.694	0.684	0.36	0.16	0	0.282
Row10	0.747	0	0.096	0.388	0.353	0.258	0.61	0.176	0.405	0.716	0.697	0.688	0.684	0.36	0.218	0.227	0.231
Row11	0.542	1	0.088	0.471	0.165	0.273	0.509	0.154	0.317	0.54	0.601	0.633	0.535	0.36	0.235	0	0.256
Row12	0.651	1	0.099	0.306	0.173	0.296	0.542	0.201	0.395	0.494	0.615	0.61	0.405	0.36	0.244	0	0.462
Row13	0.361	0	0.096	0.394	0.271	0.344	0.444	0.179	0.507	0.704	0.64	0.665	0.658	0.36	0.101	0	0.59
Row14	0.53	1	0.081	0.482	0.165	0.302	0.603	0.201	0.459	0.639	0.553	0.621	0.581	0.36	0.235	0	0.513
Row15	0.566	1	0.067	0.247	0.15	0.252	0.426	0.223	0.317	0.61	0.624	0.688	0.581	0.36	0.269	0	0.308
Row16	0.566	0	0.098	0.312	0.203	0.277	0.415	0.223	0.298	0.73	0.708	0.714	0.698	0.36	0.193	0	0.308
Row17	0.554	1	0.079	0.306	0.113	0.229	0.437	0.051	0.254	0.672	0.635	0.466	0.628	0.36	0.218	0	0.308
Row18	0.40	0	0.089	0.335	0.271	0.294	0.578	0.245	0.405	0.718	0.43	0.668	0.588	0.36	0.185	0	0.282
Row19	0.637	1	0.09	0.388	0.203	0.345	0.603	0.183	0.385	0.69	0.576	0.636	0.452	0.36	0.185	0	0.359
Row20	0.337	1	0.081	0.306	0.211	0.478	0.545	0.297	0.893	0.651	0.607	0.624	0.591	0.36	0.269	0.182	0.231
Row21	0.548	0	0.095	0.359	0.265	0.233	0.375	0.253	0.38	0.742	0.626	0.654	0.714	0.36	0.185	0	0.241
Row22	0.494	1	0.111	0.406	0.161	0.252	0.477	0.223	0.4	0.686	0.554	0.621	0.648	0.36	0.277	0	0.282
Row23	0.687	1	0.089	0.312	0.15	0	0.628	0.168	0.449	0.516	0.525	0.659	0.455	0.36	0.21	0	0.362
Row24	0.482	1	0.071	0.286	0.203	0.267	0.563	0.159	0.4	0.657	0.545	0.795	0.588	0.36	0.202	0	0.313
Row25	0.53	0	0.095	0.435	0.406	0.244	0.542	0.319	0.293	0.405	0.812	0.536	0.472	0.36	0.16	0.227	0.213
Row26	0.41	0	0.098	0.394	0.293	0.355	0.509	0.425	0.61	0.768	0.643	0.665	0.704	0.36	0.328	0	0.282
Row27	0.373	1	0.081	0.282	0.301	0.307	0.632	0.22	0.405	0.534	0.632	0.604	0.548	0.36	0.193	0	0.333
Row28	0.676	1	0.087	0.347	0.263	0.313	0.679	1	0.483	0.481	0.927	0.61	0.449	0.36	0.294	0	0.462
Row29	0.614	1	0.081	0.453	0.208	0.281	0.606	0.707	0.4	0.396	0.98	0.477	0.326	0.36	0.227	0	0.385
Row30	0.639	0	0.104	0.465	0.226	0.3	0.635	0.234	0.444	0.352	0.542	0.647	0.342	0.36	0.059	0	0.59
Row31	0.699	1	0.086	0.365	0.12	0.26	0.386	0.161	0.371	0.543	0.553	0.668	0.495	0.36	0.311	0	0.513
Row32	0.502	1	0.081	0.394	0.15	0.239	0.437	0.275	0.307	0.616	0.587	0.702	0.565	0.36	0.412	0.136	0.256
Row33	0.627	1	0.074	0.376	0.173	0.261	0.491	0.147	0.405	0.531	0.654	0.645	0.518	0.36	0.21	0	0.41
Row34	0.831	0	0.105	0.406	0.203	0.277	0.451	0.077	0.493	0.648	0.626	0.697	0.605	0.36	0.303	0	0.282
Row35	0.53	1	0.081	0.229	0.105	0.34	0.451	0.106	0.39	0.686	0.562	0.65	0.625	0.36	0.16	0	0.282
Row36	0.602	0	0.099	0.435	0.361	0.271	0.484	0.194	0.459	0.663	0.534	0.595	0.548	0.36	0.218	0.182	0.231
Row37	0.422	1	0.087	0.518	0.226	0.382	0.552	0.242	0.361	0.645	0.596	0.535	0.578	0.36	0.151	0	0.333

Fig 134. Coeficiente de silueta con K entre dos y nueve

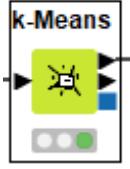
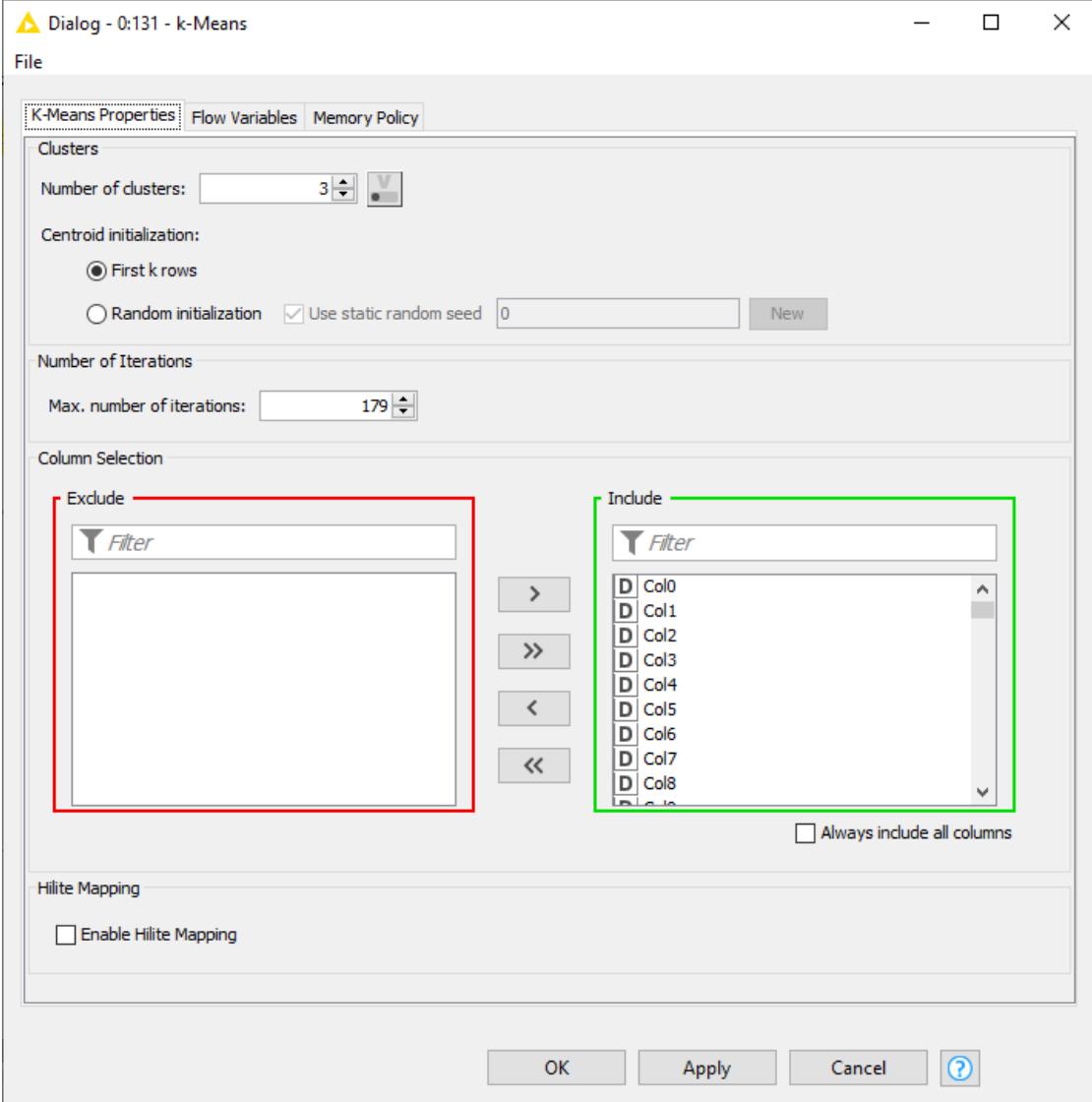
## Aplicación

El siguiente paso para realizar consiste en la aplicación de los algoritmos de clustering para determinar cuál es la mejor opción. Para ello primero determinaremos el valor de K idóneo con el algoritmo de K-Means. Esto solo para tener un valor de referencia.

Para ello se hará uso del coeficiente de silueta que permite medir valores como la cohesión entre elementos de un clúster y separación entre los clústeres. Es importante recordar que el coeficiente de silueta sólo funciona para algoritmos que funcionan con medida de distancia.

Fig 133. Dominio  $K$  de los datos después de la estandarización

El coeficiente de silueta es un valor entre menos uno y uno [-1,1]. El mejor valor de K es el que se aproxime más a uno. Auxiliándose de la figura 134 es fácil observar que el mejor valor de K para el algoritmo de K-Means es el tres. A partir de este punto, para los elementos que requieran un número de grupos esperados, se le asignará este valor(tres).

<b>Herramienta</b>	KNIME
<b>Parámetros de Configuración</b>	<p>Ocupamos en nodo K-Means para utilizar el algoritmo.</p>  <p>Fig 135. Nodo empleado K- means</p>
	 <p>Fig 136. Configuración del Nodo</p>

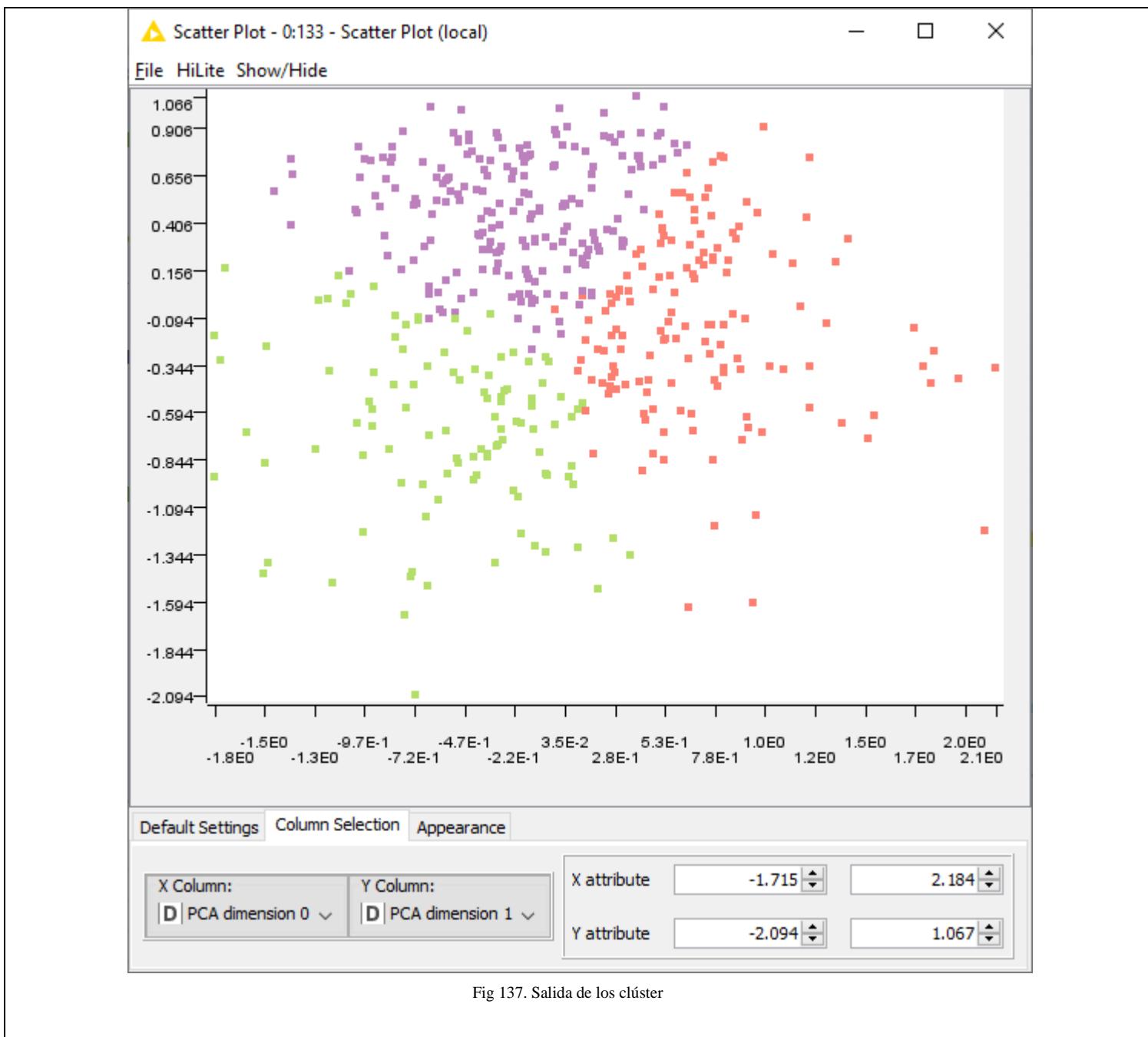


Fig 137. Salida de los clúster

## Anexos

### Flujos de trabajo de KNIME

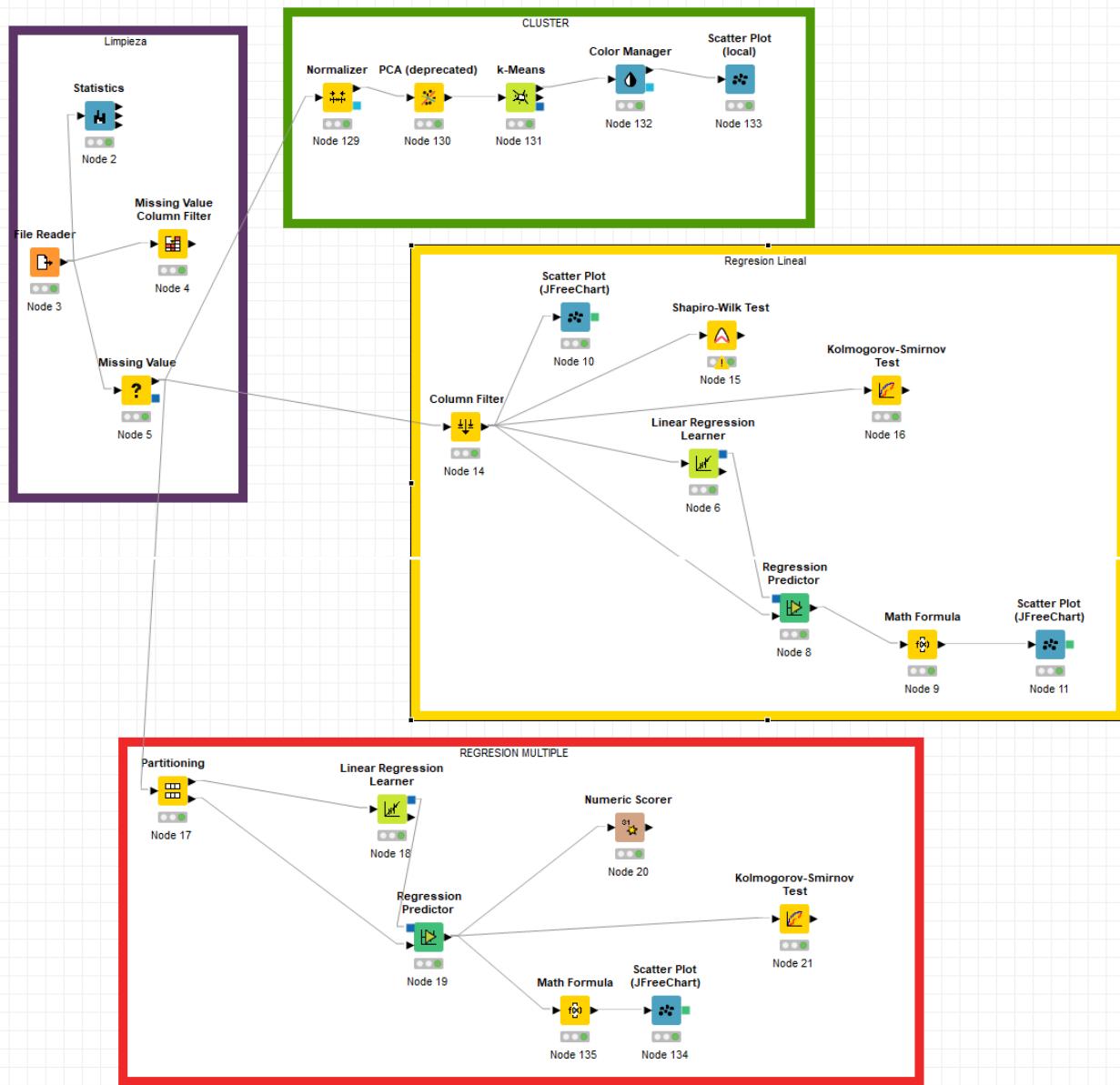


Fig 138. Flujo de Trabajo

## Referencias

- [1] Tom M. Mitchell, (1997), Machine Learning, Singapore, McGraw-Hill.
- [2] Paul E. Utgoff and Carla E. Brodley, (1990), 'An Incremental Method for Finding Multivariate Splits for Decision Trees', Machine Learning: Proceedings of the Seventh International Conference, (pp.58), Palo Alto, CA: Morgan Kaufmann.
- [3] Wei Peng, Juhua Chen and Haiping Zhou, of ID3, ' An Implementation Decision Tree Learning Algorithm', University of New South Wales, School of Computer Science & Engineering, Sydney, NSW 2032, Australia.
- [4]. Luz EJS, Schwartz WR, et al. ECG-based heartbeat classification for arrhythmia detection: a survey. *Comput. Methods Program. Biomed.* 2016;127:144–164. doi: 10.1016/j.cmpb.2015.12.008. [PubMed] [CrossRef] [Google Scholar]
- [5]. Wilkoff BL, Fauchier L, et al. 2015 HRS/EHRA/APHRS/SOLAECE expert consensus statement on optimal implantable cardioverter-defibrillator programming and testing. *EP Eur.* 2015;18(2):159–183. doi: 10.1093/europace/euv411. [PubMed] [CrossRef] [Google Scholar]
- [6]. Al-Khatib SM, Stevenson WG, et al. 2017 AHA/ACC/HRS guideline for management of patients with ventricular arrhythmias and the prevention of sudden cardiac death: executive summary. *Circulation.* 2018;138(13):e210–e271. doi: 10.1161/CIR.000000000000548. [PubMed] [CrossRef] [Google Scholar]
- [7] Bhardwaj, R., & Vatta, S. (2013). Implementation of ID3 algorithm. *International Journal of Advanced Research in Computer Science and Software Engineering*, 3(6).
- [8] Anand Bahety, ' Extension and Evaluation of ID3 – Decision Tree Algorithm'. University of Maryland, College Park
- [9]: J. Han, J. Pei, Y. Yin, R. Mao. (2004) Mining Frequent Patterns without Candidate Generation: A Frequent-Pattern Tree Approach.
- [10]: R. Agrawal, C. Aggarwal, V. Prasad. (2000) Depth first generation of long patterns.