ALGORITMO ISOLATION FOREST

DETECCIÓN DE PACIENTES CON DIABETES ATÍPICOS

Grupo 6 - Grupo Colina

Alumnos

Valdiviezo Goicochea Wisner Ernan [22200217]

Príncipe Velasquez Ivan [22200037]

Morales Damasco Cristian [22200029]

Saavedra Nieto Rissel [22200041]

Sernaque Gutierrez Jose [22200063]

Profesor

GELBER CHRISTIAN USCUCHAGUA FLORES

Curso: Big Data

Resumen Ejecutivo

Justificacion del Proyecto

La **diabetes** es una enfermedad crónica con una prevalencia en constante aumento a nivel mundial. La detección temprana y el manejo adecuado de la diabetes son cruciales para prevenir complicaciones graves que impactan la calidad de vida de los pacientes y los sistemas de salud.

Sin embargo, **identificar tempranamente** a las personas **con mayor riesgo** de desarrollar diabetes o que requieren **atención médica** especializada es un desafío. Este proyecto busca desarrollar herramientas innovadoras para un análisis más eficiente de los datos de pacientes y así optimizar la prevención, el diagnóstico temprano y la personalización de los tratamientos.

Objetivo del proyecto

El objetivo principal del proyecto es aplicar el algoritmo **Isolation Forest** para analizar un conjunto de datos de pacientes con diabetes y detectar aquellos con perfiles atípicos. El análisis se centrará en identificar patrones inusuales en las variables clínicas, como **la glucosa en sangre**, **la presión arterial**, el índice **de masa corporal** y **otros factores de riesgo**, para **detectar** individuos que se desvían significativamente de la población general.

Resumen Ejecutivo

Impacto del proyecto en la sociedad

Este proyecto busca generar un impacto positivo en la sociedad al mejorar la salud pública a través de:

- **Prevención:** Detectar tempranamente personas con alto riesgo de desarrollar diabetes.
- **Tratamiento:** Identificar pacientes que requieren atención médica especializada o un manejo más personalizado de su condición, lo que puede incluir intervenciones tempranas para prevenir complicaciones.
- **Eficiencia:** Optimizar recursos para que los profesionales de la salud puedan focalizar sus esfuerzos en los pacientes que más lo necesitan.

Beneficio social y/o comercial del proyecto

Los beneficios sociales del proyecto incluyen:

- Reducción de la carga económica que la diabetes representa para los sistemas de salud y las familias.
- Mayor conocimiento sobre los factores de riesgo y las características de la diabetes.

Los beneficios comerciales del proyecto incluyen:

- Potencial desarrollo de herramientas de análisis de datos para la industria de la salud.
- Identificación de pacientes de alto riesgo para una mejor gestión de la enfermedad.
- Prevención de complicaciones y mejora en la atención al paciente.

Resumen Ejecutivo

• Vinculación con los ODS:

El proyecto se alinea con el **ODS 3: Salud y bienestar**, específicamente con las metas:

- **3.4:** Reducir en un tercio la mortalidad prematura por enfermedades no transmisibles mediante la prevención y el tratamiento y promover la salud mental y el bienestar.
- **3.8:** Lograr la cobertura sanitaria universal, en particular la protección contra los riesgos financieros, el acceso a servicios de salud esenciales de calidad y el acceso a medicamentos y vacunas seguros, eficaces, asequibles y de calidad para todos.

• Carácter innovador:

El proyecto se caracteriza por su innovación al aplicar el **algoritmo Isolation Forest** para detectar **anomalías** en datos de pacientes con diabetes. Esta técnica, basada en aprendizaje automático no supervisado, permite identificar perfiles atípicos que podrían pasar desapercibidos con **métodos tradicionales.**

El uso de herramientas de **Big Data en R** para el análisis y visualización de los datos también contribuye al carácter innovador del proyecto.

Esto incluye:

- Explorar relaciones entre variables.
- Identificar patrones complejos.
- Generar gráficos informativos.

Introduccion

• Objetivo Principal

Desarrollar e implementar un modelo de Machine Learning utilizando el algoritmo **Isolation Forest** en **R** para analizar un conjunto de datos de pacientes con diabetes, con la finalidad de identificar perfiles atípicos que posiblemente puedan requerir atención médica especializada o un manejo personalizado de su condición, contribuyendo así a la mejora de la salud pública y el avance hacia el **ODS 3: Salud y bienestar.**

• Objetivos Específicos:

Implementación del modelo Isolation Forest:

- Entrenar el modelo Isolation Forest utilizando la librería isolationForest en R.
- Ajustar los hiperparámetros del modelo, como el número de árboles y la profundidad máxima, para optimizar su rendimiento en la detección de anomalías.

Visualización e interpretación de resultados:

- Generar visualizaciones como box plot, histogramas y pair plots para una interpretación gráfica.
- Interpretar los resultados del análisis en el contexto de la diabetes, considerando las implicaciones para la atención médica, la prevención y el tratamiento.

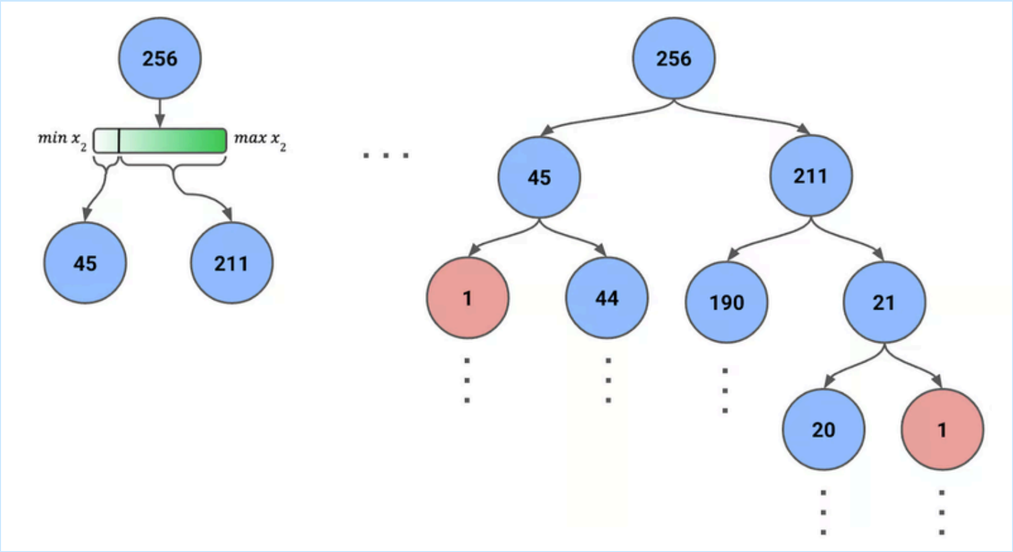
Identificación y análisis de perfiles atípicos:

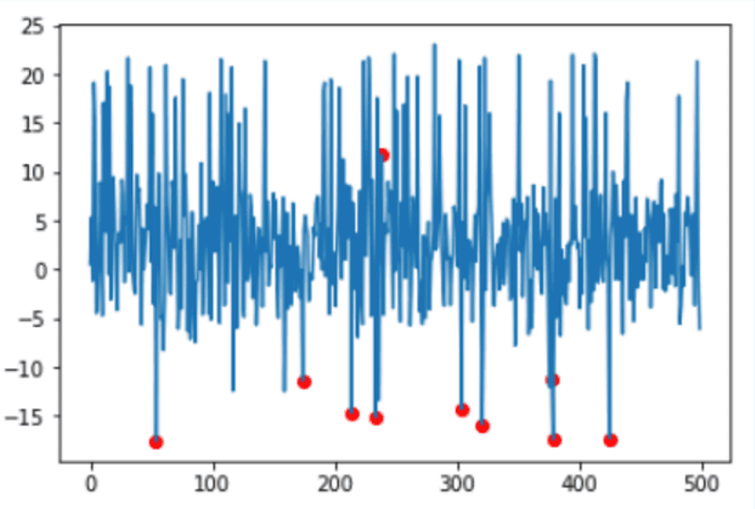
- Aplicar el modelo entrenado para obtener las puntuaciones de anomalía para cada paciente en el conjunto de datos.
- Analizar las características de los pacientes identificados como atípicos, observando las variables que más contribuyen a su puntuación de anomalía.

ISOLATION FOREST

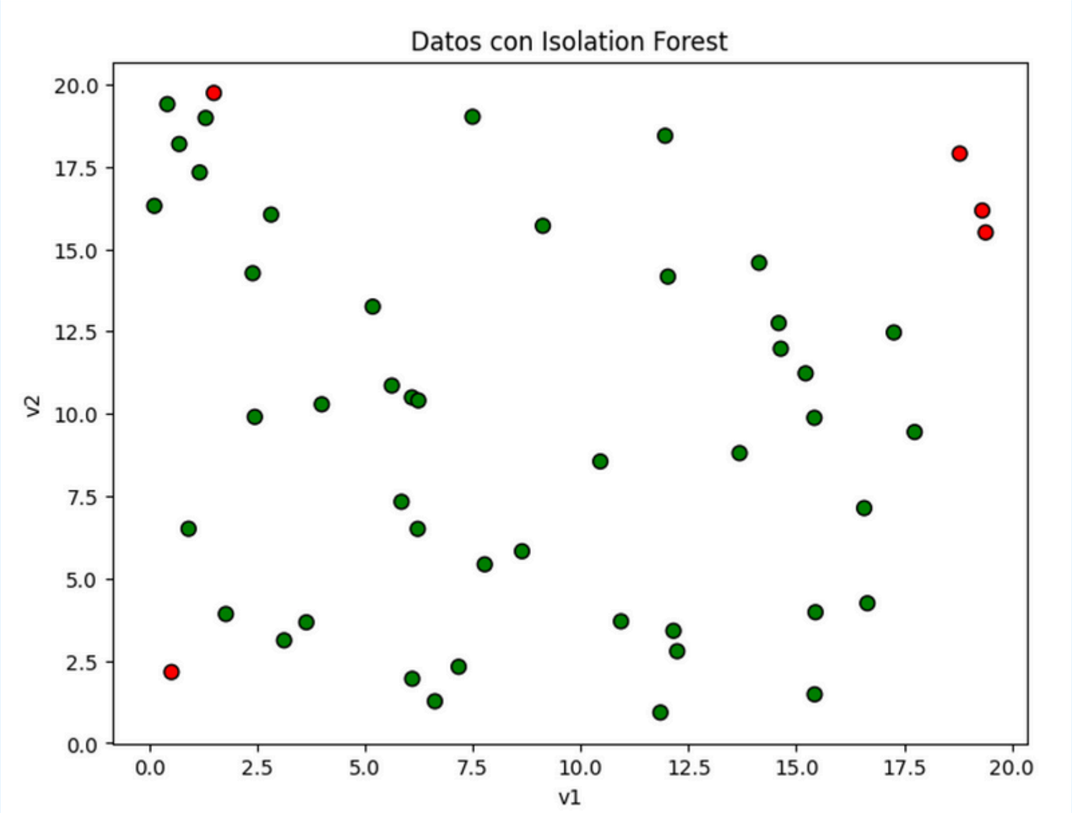
Isolation Forest (iForest) es un algoritmo no supervisado diseñado para la detección de anomalías en grandes volúmenes de datos. A diferencia de métodos tradicionales que modelan las instancias normales, iForest identifica las anomalías al aislarlas explícitamente, aprovechando que estas son pocas y diferentes, lo que las hace más fáciles de separar del resto.

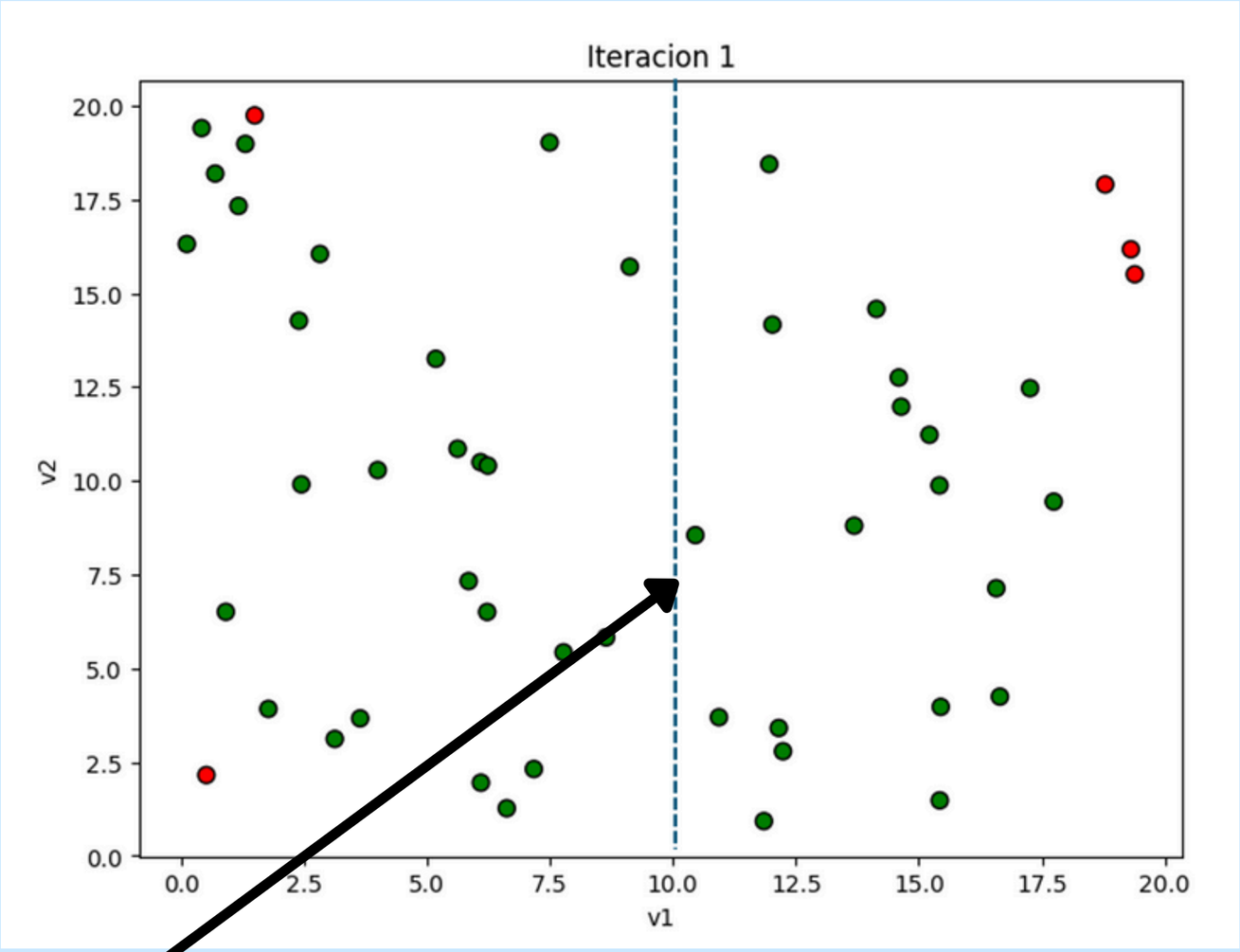
El algoritmo se basa en la construcción de un Bosque de Árboles de Aislamiento (iTrees), donde los nodos internos dividen el espacio de datos seleccionando aleatoriamente un atributo y un punto de división. Las anomalías, al ser distintas, requieren menos divisiones (menor profundidad) para ser aisladas. Posteriormente, se calcula una puntuación de anomalía normalizando la longitud promedio del camino en el bosque completo.





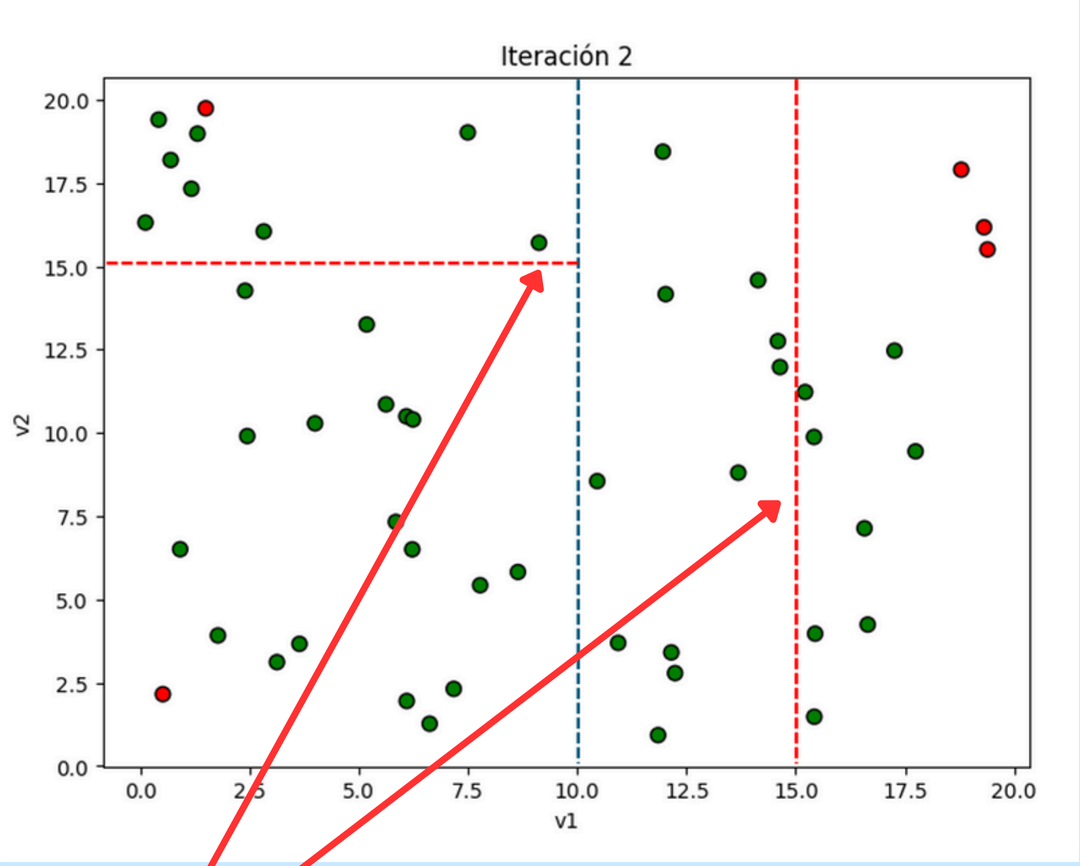
EJEMPLO



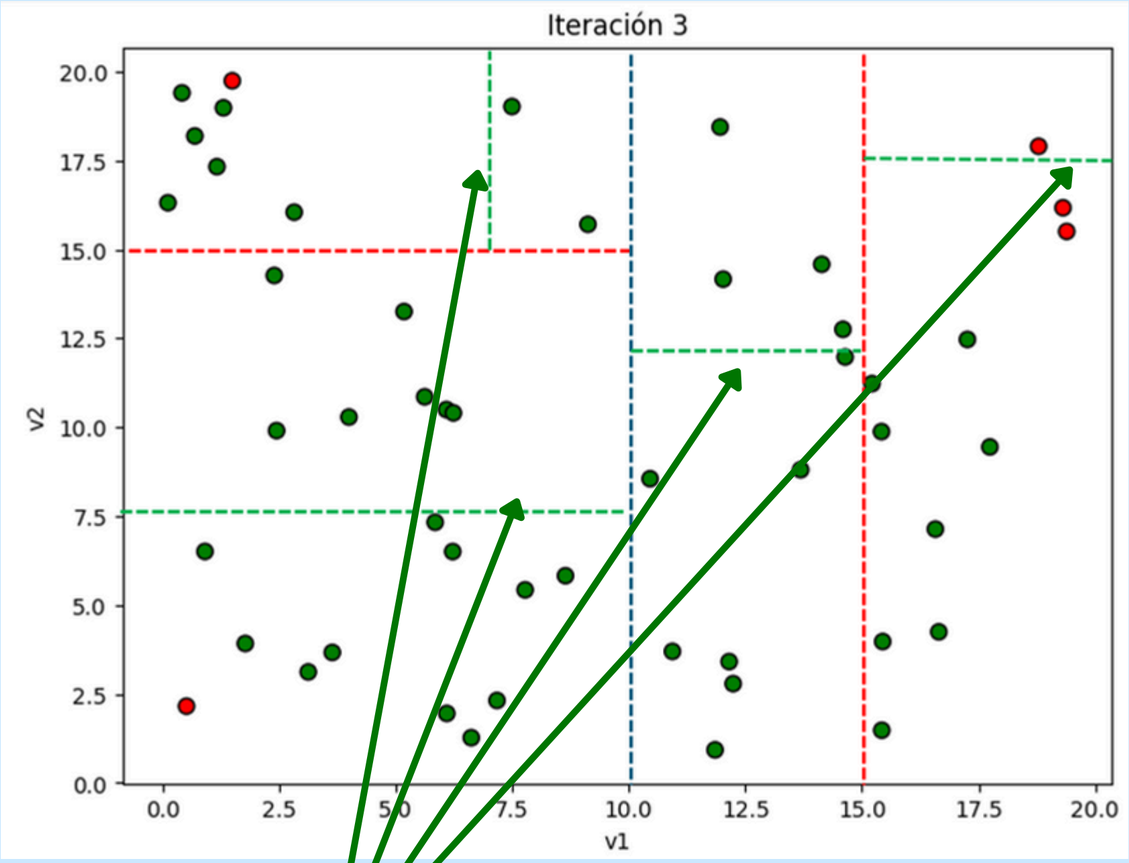


PASO 1

EJEMPLO



PASO 2



PASO 3

FUNCIONAMIENTO

Etapa de Entrenamiento

En la etapa de entrenamiento, se construyen múltiples árboles de aislamiento a partir de submuestras de los datos. Cada árbol divide los datos recursivamente seleccionando un atributo y un punto de división aleatorios, lo que permite identificar las instancias que requieren menos particiones para ser aisladas.

Algorithm 1 : $iForest(X, t, \psi)$

Inputs: X - input data, t - number of trees, ψ - sub-sampling size

Output: a set of t *iTrees*

Initialize $Forest$
set height limit $l = ceiling(\log_2 \psi)$
for $i = 1$ to t **do**
  $X' \leftarrow sample(X, \psi)$
  $Forest \leftarrow Forest \cup iTree(X', 0, l)$
end for
return $Forest$

Algorithm 2 : $iTree(X, e, l)$

Inputs: X - input data, e - current tree height, l - height limit

Output: an iTree

if $e \geq l$ or $|X| \leq 1$ **then**
  return *exNode*$\{Size \leftarrow |X|\}$
else
  let Q be a list of attributes in X
  randomly select an attribute $q \in Q$
  randomly select a split point p from max and min values of attribute q in X
  $X_l \leftarrow filter(X, q < p)$
  $X_r \leftarrow filter(X, q \geq p)$
  return *inNode*$\{Left \leftarrow iTree(X_l, e+1, l),$
    $Right \leftarrow iTree(X_r, e+1, l),$
    $SplitAtt \leftarrow q,$
    $SplitValue \leftarrow p\}$
end if

FUNCIONAMIENTO

Etapa de Evaluación

En la etapa de evaluación, se calcula la longitud promedio del camino que recorre cada instancia a través de los árboles. Las anomalías, al ser más fáciles de aislar, tienen caminos más cortos, y esto se utiliza para determinar la puntuación de anomalía.

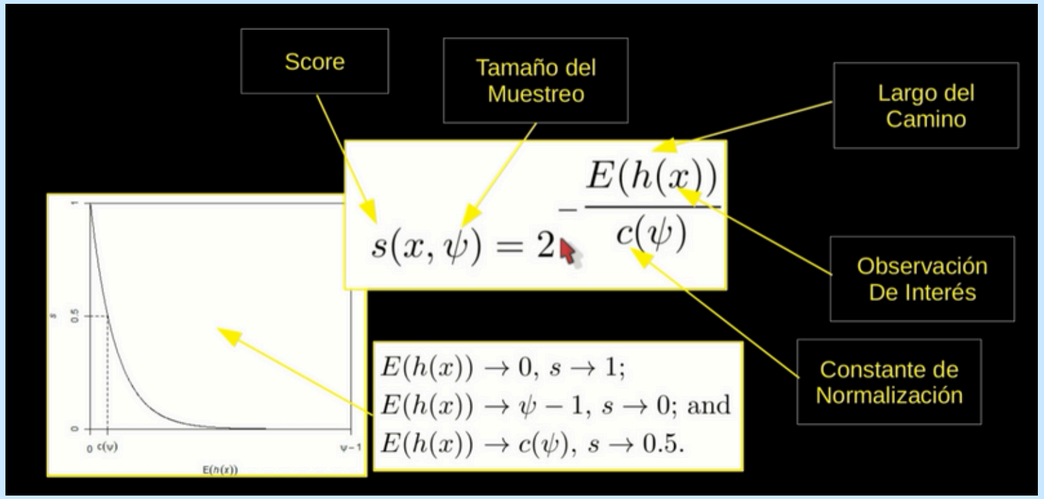
Algorithm 3 : $PathLength(x, T, hlim, e)$

Inputs : x - an instance, T - an iTree, $hlim$ - height limit, e - current path length; to be initialized to zero when first called

Output: path length of x

```
if T is an external node or e ≥ hlim then
    return e + c(T.size) {c(.) is defined in Equation 1}
end if
a ← T.splitAtt
if x_a < T.splitValue then
    return PathLength(x, T.left, hlim, e + 1)
else {x_a ≥ T.splitValue}
    return PathLength(x, T.right, hlim, e + 1)
end if
```

Score de Anomalía



La puntuación de anomalía s(x) se interpreta de la siguiente manera:

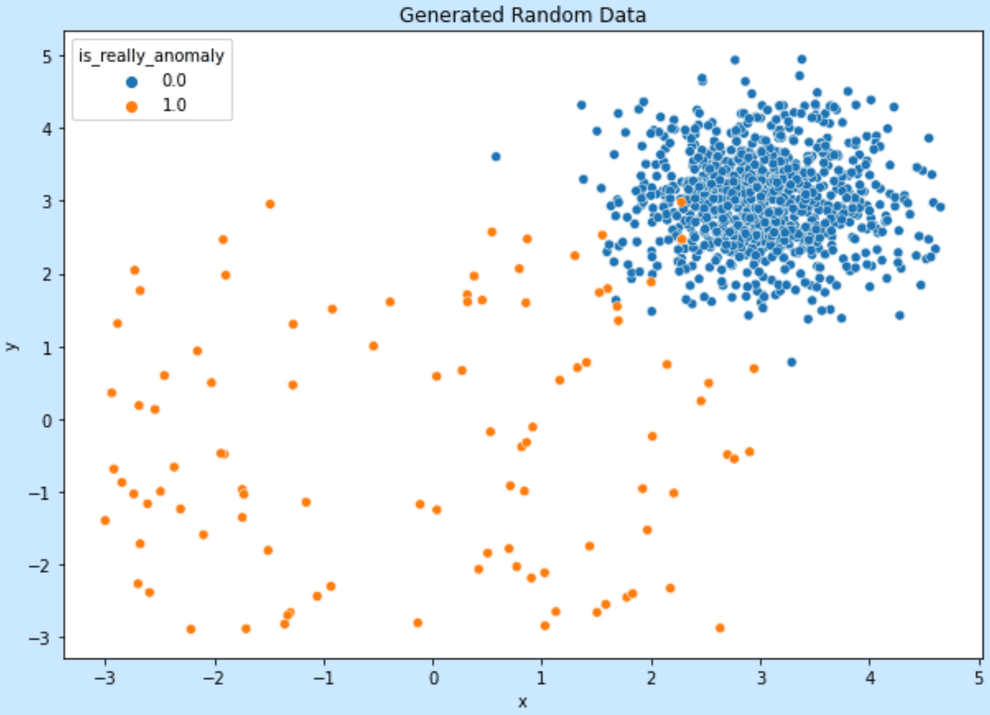
Si s(x) =1: La instancia es anómala.

Si s(x) =0.5: La instancia es normal.

Si s(x) =es cercano a 0.5 para todas las instancias, el conjunto de datos no contiene anomalías claras.

Relación con el Proyecto

El algoritmo Isolation Forest será implementado en R utilizando la librería isolationForest. Se analizarán variables clínicas como glucosa en sangre, presión arterial e índice de masa corporal (IMC) para detectar anomalías significativas en los datos de pacientes con diabetes.



Aplicaciones en Otros Contextos

- Ciberseguridad
- Industria Financiera
- Gestión Industrial

IMPLEMENTACION DE ISOLATION FOREST

Librerias usadas

```r
if (!require(solitude)) install.packages("solitude", dependencies = TRUE)
if (!require(dplyr)) install.packages("dplyr")
if (!require(gridExtra)) install.packages("gridExtra")
if (!require(ggplot2)) install.packages("ggplot2")
if (!require(GGally)) install.packages("GGally")
if (!require(VIM)) install.packages("VIM")

library(solitude)
library(dplyr)
library(gridExtra)
library(ggplot2)
library(GGally)
library(VIM)
```

IMPLEMENTACION DE ISOLATION FOREST

Lectura

```r
df <- read.csv(file.choose(), sep = ",")
head(df)
summary(df)
print("Estructura del dataset:")
str(df)
print("Resumen estadístico del dataset:")
summary(df)
print("Valores nulos por columna:")
colSums(is.na(df))
```

Grafico de cajas

```r
columns_to_plot <- colnames(df)[1:(ncol(df))]
plots <- list()

# Diagramas de cajas
for (col in columns_to_plot) {
  p <- ggplot(df, aes_string(y = col)) +
    geom_boxplot(fill = "steelblue", color = "black") +
    labs(title = col, y = "") +
    theme_minimal()
  plots[[col]] <- p
}

do.call(grid.arrange, c(plots, ncol = 3))
```

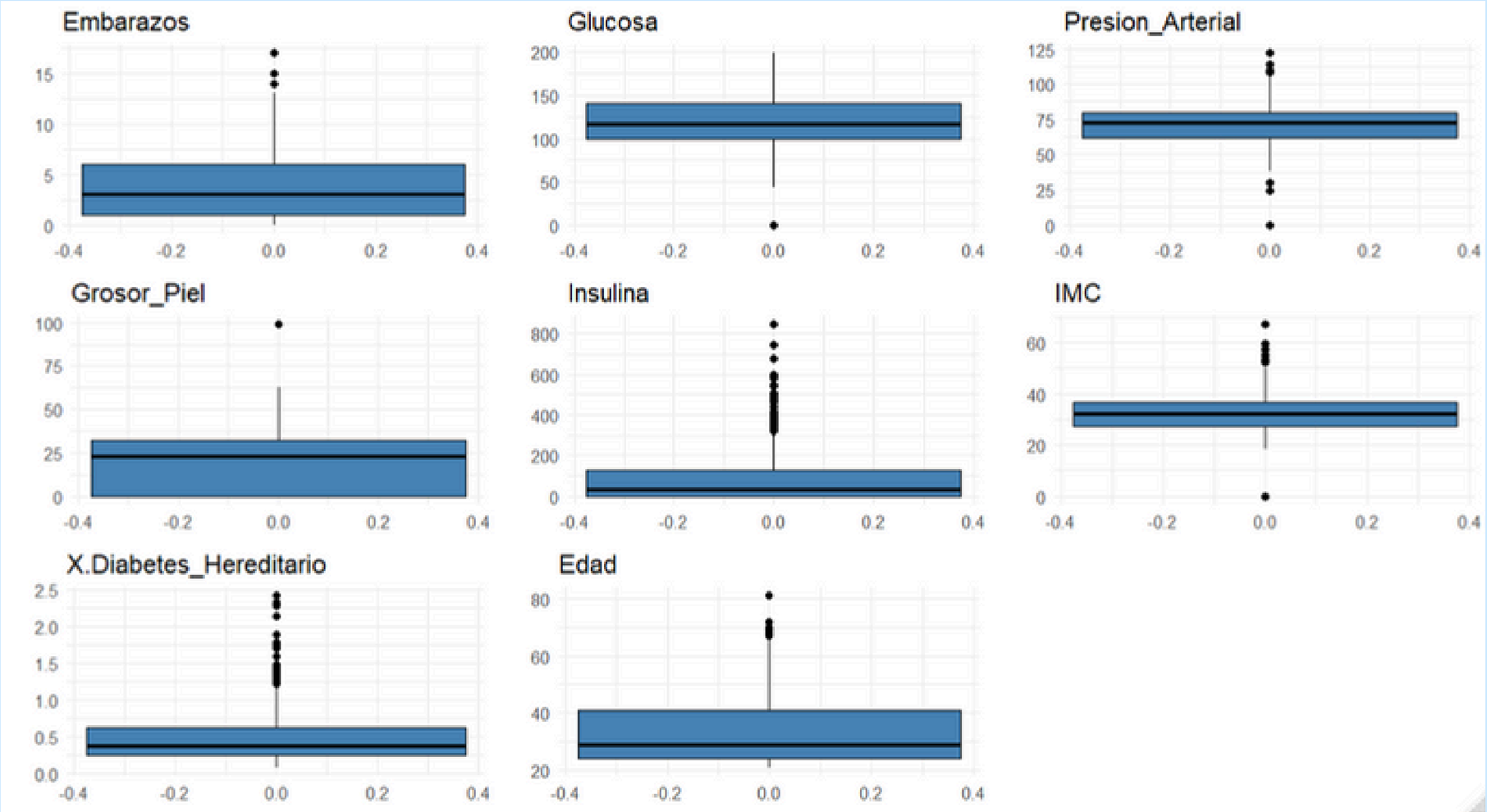
IMPLEMENTACION DE ISOLATION FOREST

```
> summary(df)
   Embarazos        Glucosa      Presion_Arterial  Grosor_Piel       Insulina          IMC         X.Diabetes_Hereditario
 Min.   : 0.000   Min.   :  0.0   Min.   :  0.00   Min.   : 0.00   Min.   :  0.0   Min.   : 0.00   Min.   :0.0780
 1st Qu.: 1.000   1st Qu.: 99.0   1st Qu.: 62.00   1st Qu.: 0.00   1st Qu.:  0.0   1st Qu.:27.30   1st Qu.:0.2437
 Median : 3.000   Median :117.0   Median : 72.00   Median :23.00   Median : 30.5   Median :32.00   Median :0.3725
 Mean   : 3.845   Mean   :120.9   Mean   : 69.11   Mean   :20.54   Mean   : 79.8   Mean   :31.99   Mean   :0.4719
 3rd Qu.: 6.000   3rd Qu.:140.2   3rd Qu.: 80.00   3rd Qu.:32.00   3rd Qu.:127.2   3rd Qu.:36.60   3rd Qu.:0.6262
 Max.   :17.000   Max.   :199.0   Max.   :122.00   Max.   :99.00   Max.   :846.0   Max.   :67.10   Max.   :2.4200
      Edad
 Min.   :21.00
 1st Qu.:24.00
 Median :29.00
 Mean   :33.24
 3rd Qu.:41.00
 Max.   :81.00
```

```
> str(df)
'data.frame':   768 obs. of  8 variables:
 $ Embarazos             : int  6 1 8 1 0 5 3 10 2 8 ...
 $ Glucosa               : int  148 85 183 89 137 116 78 115 197 125 ...
 $ Presion_Arterial      : int  72 66 64 66 40 74 50 0 70 96 ...
 $ Grosor_Piel           : int  35 29 0 23 35 0 32 0 45 0 ...
 $ Insulina              : int  0 0 0 94 168 0 88 0 543 0 ...
 $ IMC                   : num  33.6 26.6 23.3 28.1 43.1 25.6 31 35.3 30.5 0 ...
 $ X.Diabetes_Hereditario: num  0.627 0.351 0.672 0.167 2.288 ...
 $ Edad                  : int  50 31 32 21 33 30 26 29 53 54 ...
```

```
> colSums(is.na(df))
             Embarazos                Glucosa       Presion_Arterial           Grosor_Piel               Insulina
                     0                      0                      0                     0                      0
                   IMC X.Diabetes_Hereditario                   Edad
                     0                      0                      0
```

IMPLEMENTACION DE ISOLATION FOREST



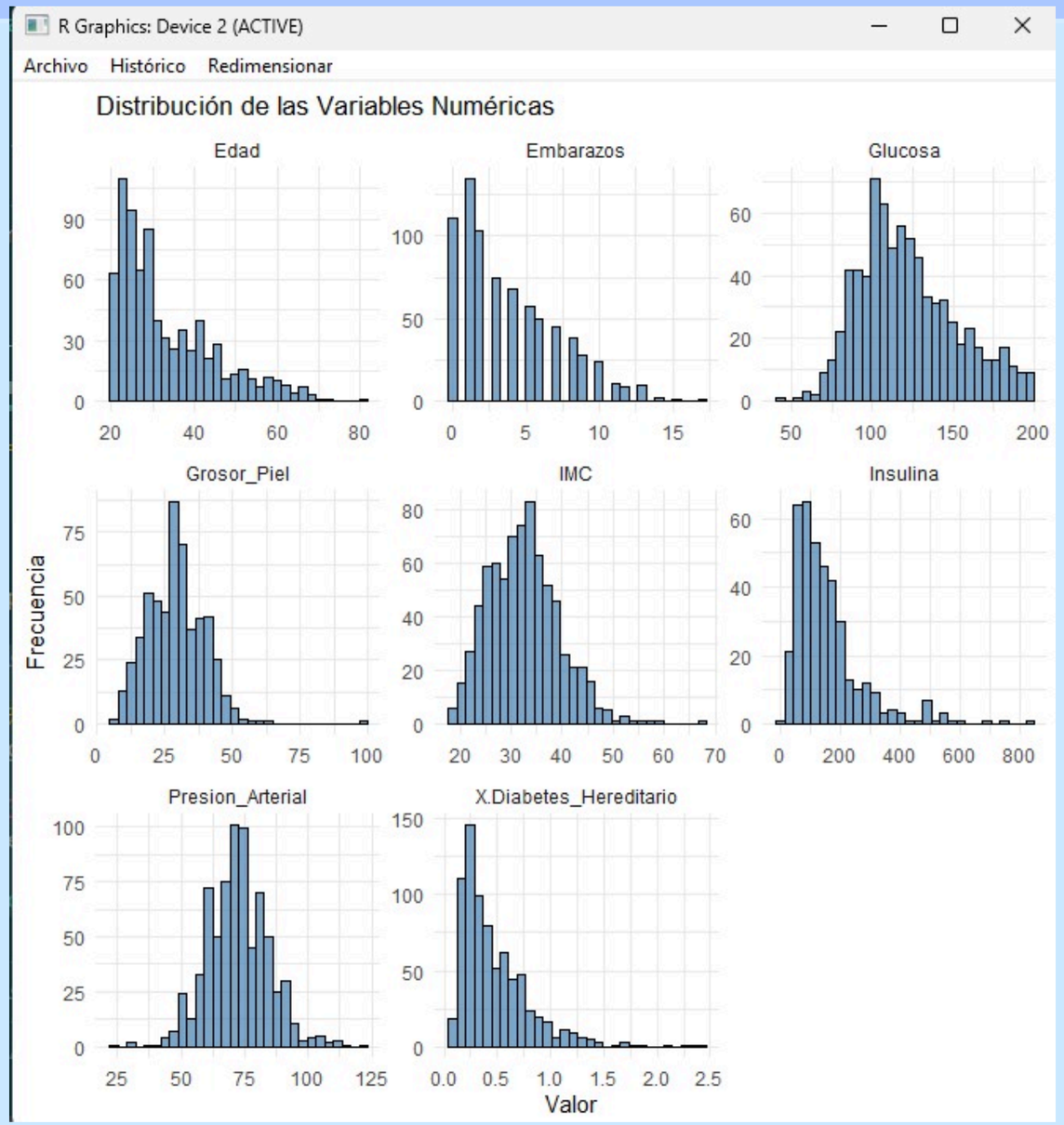
IMPLEMENTACION DE ISOLATION FOREST

Histograma

```r
# Histograma de variables numéricas
df %>%
  select_if(is.numeric) %>%
  pivot_longer(everything(), names_to = "key", values_to = "value") %>%
  ggplot(aes(value)) +
  geom_histogram(bins = 30, fill = "steelblue", color = "black", alpha = 0.7) +
  facet_wrap(~ key, scales = 'free') +
  theme_minimal() +
  labs(title = "Distribución de las Variables Numéricas", x = "Valor", y = "Frecuencia")
```

IMPLEMENTACION DE ISOLATION FOREST

Histograma



IMPLEMENTACION DE ISOLATION FOREST

Aplicacion de Isolation Forest

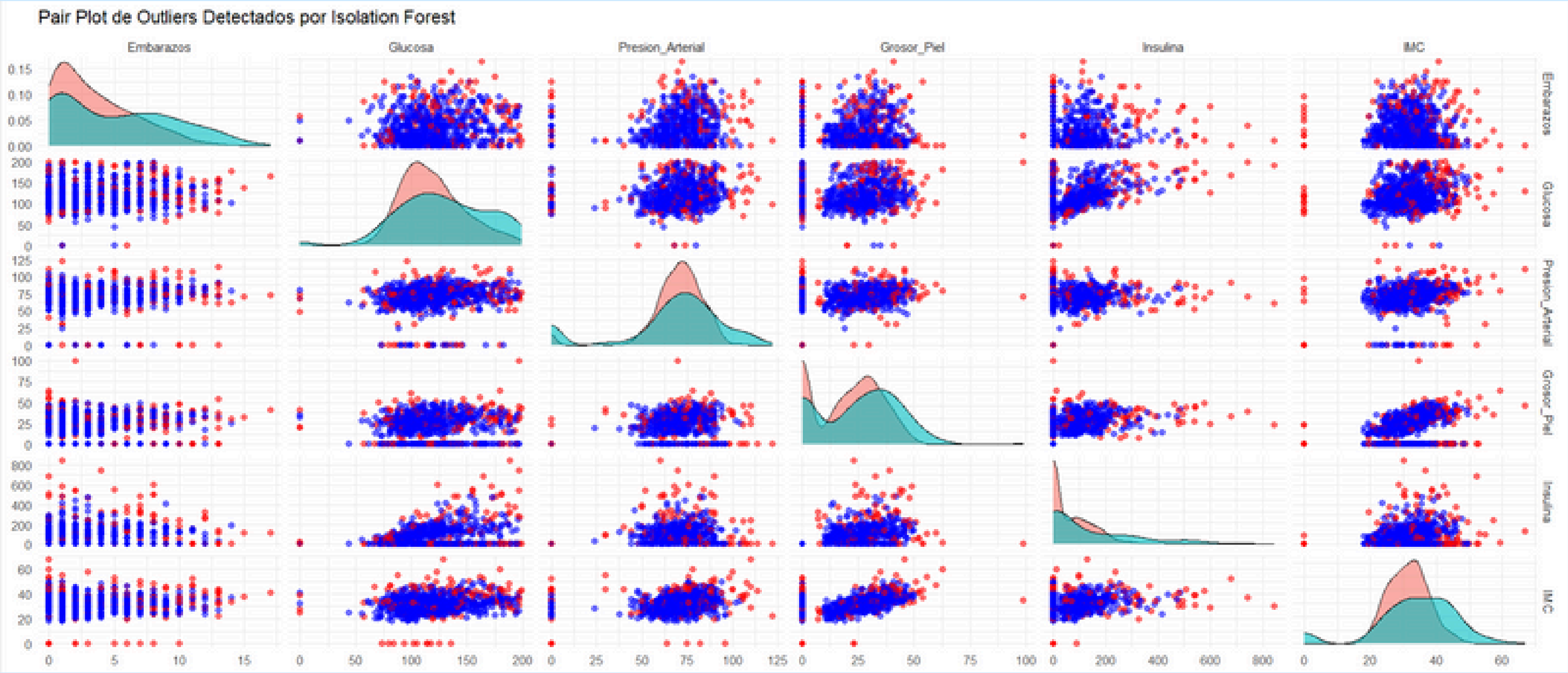
```
X_scaled <- scale(df %>% select_if(is.numeric))

iso_model <- isolationForest$new(sample_size = nrow(X_scaled), num_trees = 100, seed = 42)
iso_model$fit(X_scaled)

predictions <- iso_model$predict(X_scaled)
df$outlier <- ifelse(predictions$anomaly_score > quantile(predictions$anomaly_score, 0.8), "Outlier",
"Normal")
```

IMPLEMENTACION DE ISOLATION FOREST

Aplicacion de Isolation Forest

```r
# Visualización: Pair Plot inicial
ggpairs(df,
        columns = 1:6,
        aes(color = outlier, alpha = 0.6),
        upper = list(continuous = wrap("points", size = 2)),
        lower = list(continuous = wrap("points", size = 2)),
        diag = list(continuous = wrap("densityDiag", alpha = 0.6))) +
  scale_color_manual(values = c("Normal" = "blue", "Outlier" = "red")) +
  labs(title = "Pair Plot de Outliers Detectados por Isolation Forest") +
  theme_minimal()
```

IMPLEMENTACION DE ISOLATION FOREST



IMPUTACIÓN DE VALORES

```r
# Imputacion de valores faltantes
cols_imputar <- c("Glucosa", "Presion_Arterial", "Grosor_Piel", "Insulina", "IMC")
df[cols_imputar ] <- lapply(df[cols_imputar ], function(x) ifelse(x == 0, NA, x))
df_imputado <- kNN(df, variable = cols_imputar )
```

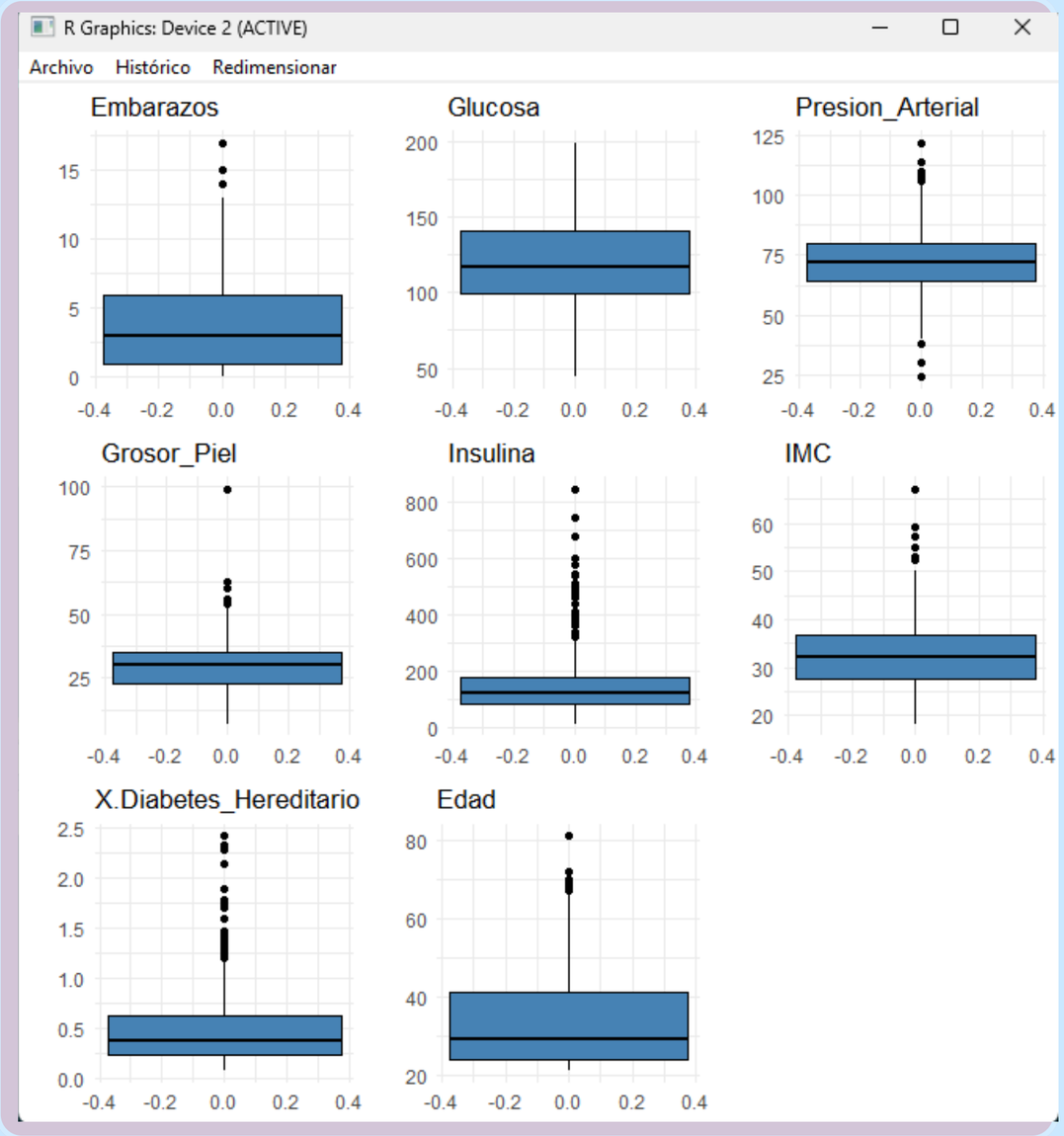
ANTES

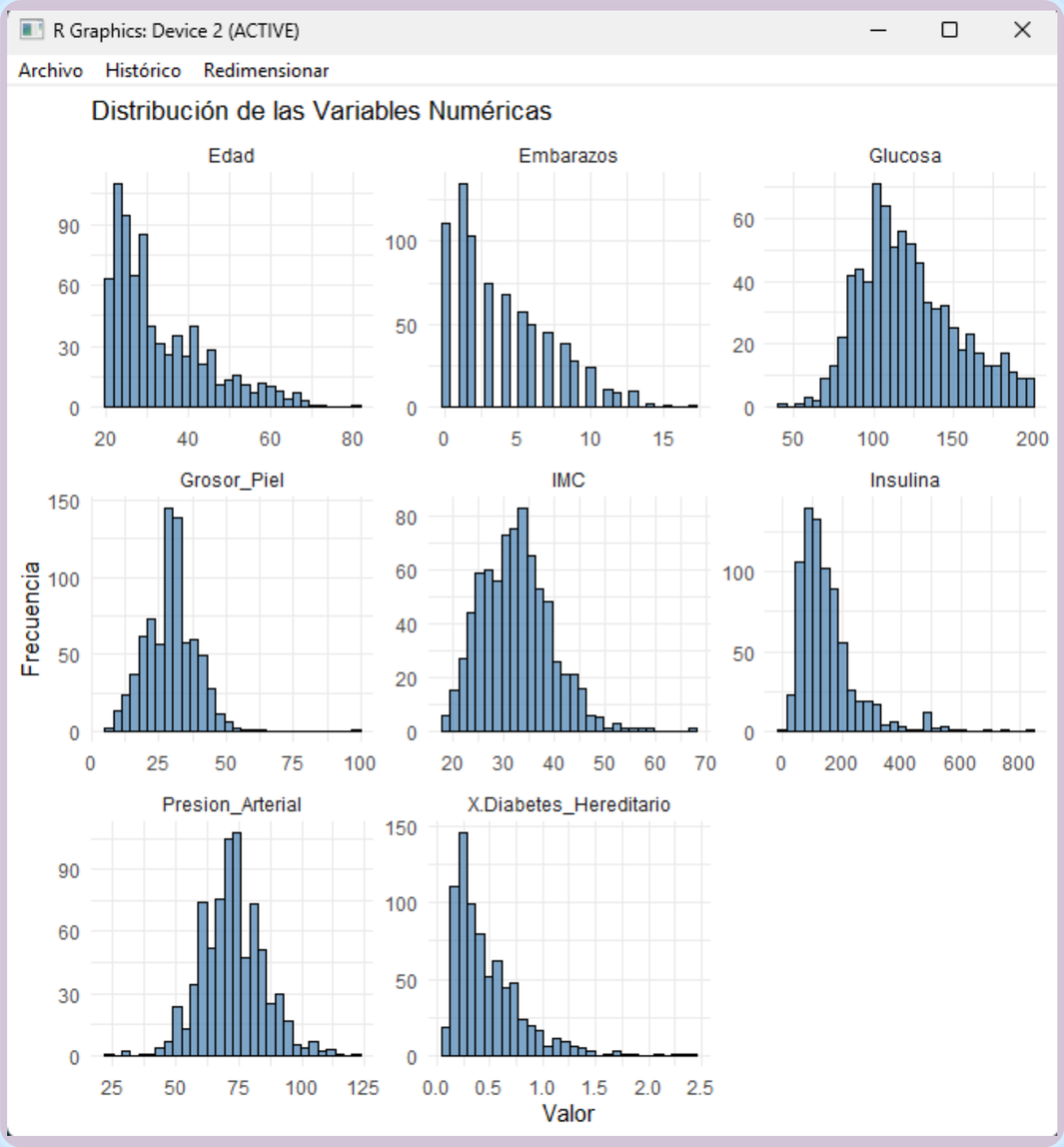
	Embarazos	Glucosa	Presion_Arterial	Grosor_Piel	Insulina	IMC	X.Diabetes_Hereditario	Edad	outlier
1	6	148	72	35	0	33.6	0.627	50	Normal
2	1	85	66	29	0	26.6	0.351	31	Normal
3	8	183	64	0	0	23.3	0.672	32	Normal
4	1	89	66	23	94	28.1	0.167	21	Normal
5	0	137	40	35	168	43.1	2.288	33	Outlier
6	5	116	74	0	0	25.6	0.201	30	Normal
7	3	78	50	32	88	31.0	0.248	26	Normal
8	10	115	0	0	0	35.3	0.134	29	Outlier
9	2	197	70	45	543	30.5	0.158	53	Outlier
10	8	125	96	0	0	0.0	0.232	54	Outlier
11	4	110	92	0	0	37.6	0.191	30	Normal
12	10	168	74	0	0	38.0	0.537	34	Normal
13	10	139	80	0	0	27.1	1.441	57	Outlier
14	1	189	60	23	846	30.1	0.398	59	Outlier
15	5	166	72	19	175	25.8	0.587	51	Normal

DESPUÉS

	Embarazos	Glucosa	Presion_Arterial	Grosor_Piel	Insulina	IMC	X.Diabetes_Hereditario	Edad	outlier
1	6	148	72	35	168	33.6	0.627	50	Normal
2	1	85	66	29	55	26.6	0.351	31	Normal
3	8	183	64	28	175	23.3	0.672	32	Normal
4	1	89	66	23	94	28.1	0.167	21	Normal
5	0	137	40	35	168	43.1	2.288	33	Outlier
6	5	116	74	27	112	25.6	0.201	30	Normal
7	3	78	50	32	88	31.0	0.248	26	Normal
8	10	115	74	37	285	35.3	0.134	29	Outlier
9	2	197	70	45	543	30.5	0.158	53	Outlier
10	8	125	96	36	130	36.6	0.232	54	Outlier
11	4	110	92	32	105	37.6	0.191	30	Normal
12	10	168	74	32	171	38.0	0.537	34	Normal
13	10	139	80	24	150	27.1	1.441	57	Outlier
14	1	189	60	23	846	30.1	0.398	59	Outlier
15	5	166	72	19	175	25.8	0.587	51	Normal

NUEVO ANÁLISIS EXPLORATORIO

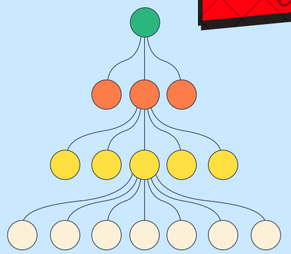




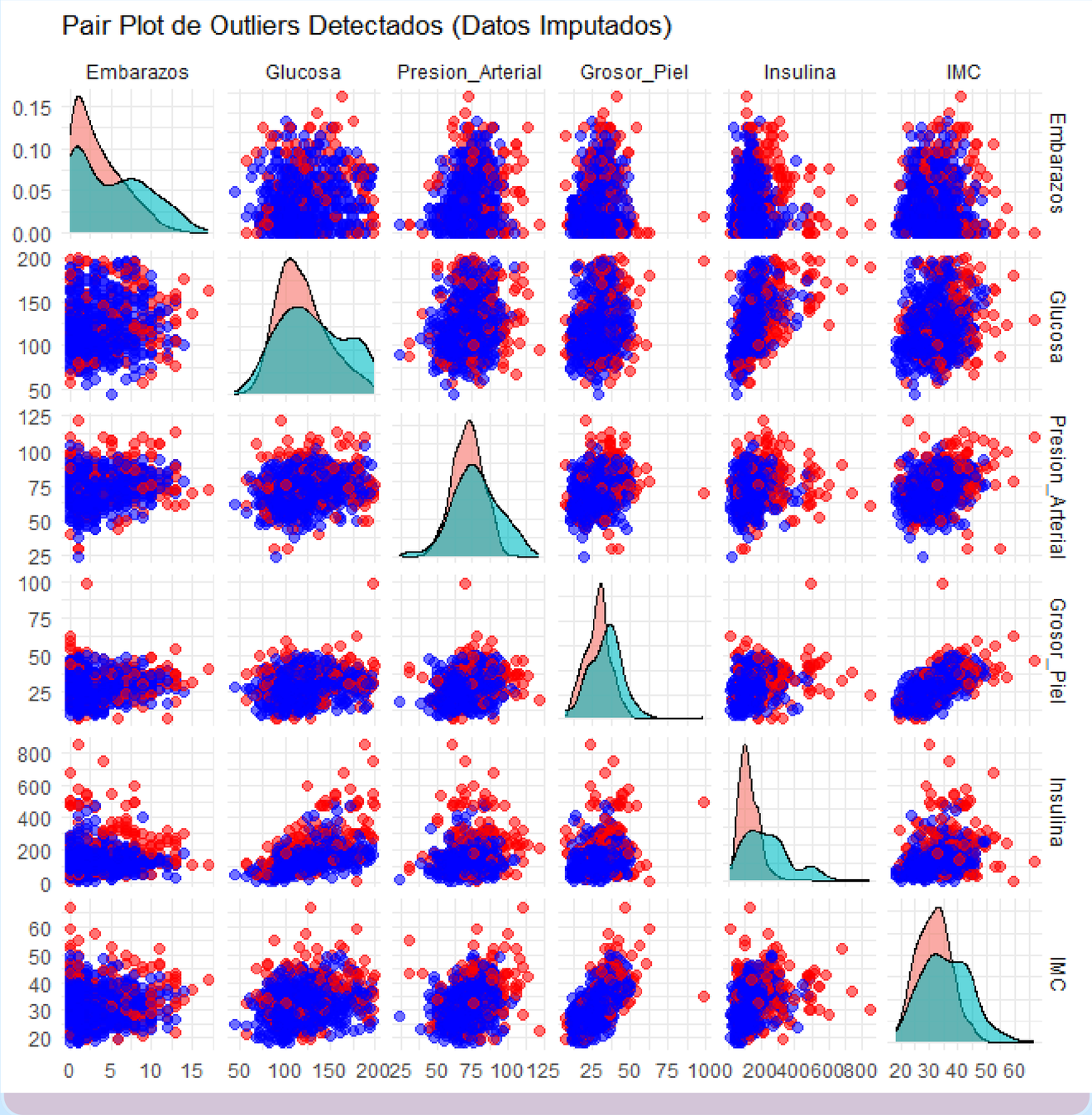
APLICACIÓN DE ISOLATION FOREST

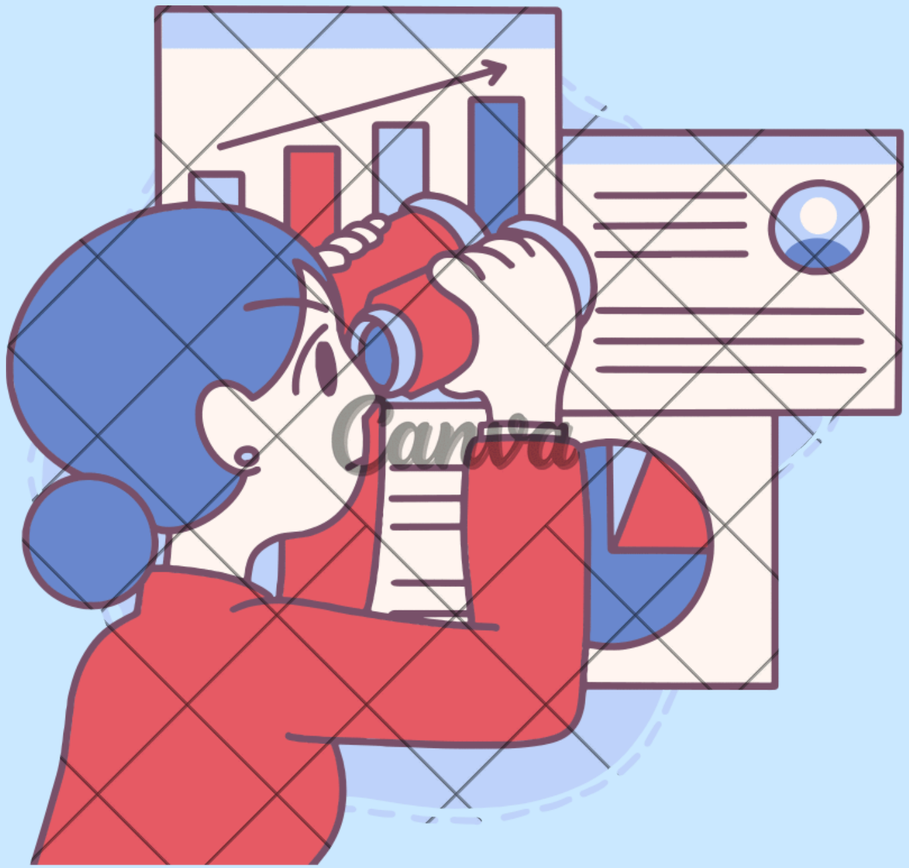
```r
# Nueva aplicacion de Isolation Forest
X_nueva_scaled <- scale(df_imputado %>% select_if(is.numeric))

nuevo_modelo_isolation <- isolationForest$new(sample_size = nrow(X_nueva_scaled), num_trees = 100, seed = 42)
nuevo_modelo_isolation$fit(X_nueva_scaled)

nueva_predicion <- nuevo_modelo_isolation$predict(X_nueva_scaled)
df_imputado$outlier <- ifelse(nueva_predicion$anomaly_score > quantile(nueva_predicion$anomaly_score, 0.8), "Outlier", "Normal")
```

RESULTADO



	Embarazos	Glucosa	Presion_Arterial	Grosor_Piel	Insulina	IMC	X.Diabetes_Hereditario	Edad	outlier
1	6	148	72	35	168	33.6	0.627	50	Normal
2	1	85	66	29	55	26.6	0.351	31	Normal
3	8	183	64	28	175	23.3	0.672	32	Outlier
4	1	89	66	23	94	28.1	0.167	21	Normal
5	0	137	40	35	168	43.1	2.288	33	Outlier
6	5	116	74	27	112	25.6	0.201	30	Normal
7	3	78	50	32	88	31.0	0.248	26	Normal
8	10	115	74	37	285	35.3	0.134	29	Normal
9	2	197	70	45	543	30.5	0.158	53	Outlier
10	8	125	96	36	130	36.6	0.232	54	Normal
11	4	110	92	32	105	37.6	0.191	30	Normal
12	10	168	74	32	171	38.0	0.537	34	Normal
13	10	139	80	24	150	27.1	1.441	57	Outlier
14	1	189	60	23	846	30.1	0.398	59	Outlier
15	5	166	72	19	175	25.8	0.587	51	Normal



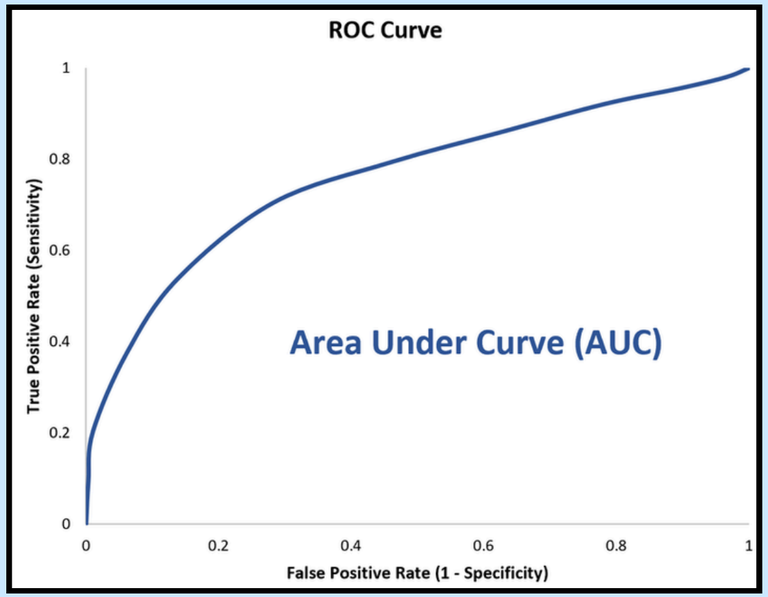


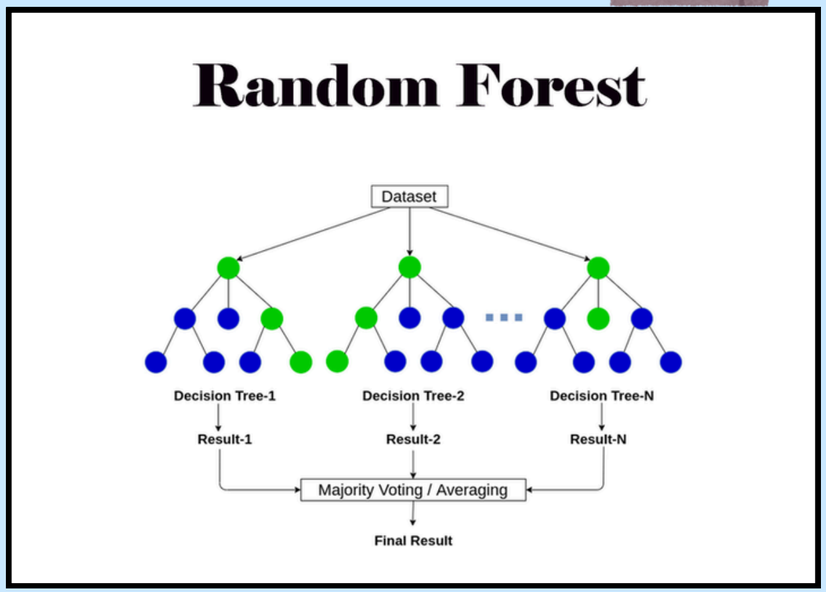
VISUALIZACIÓN FINAL

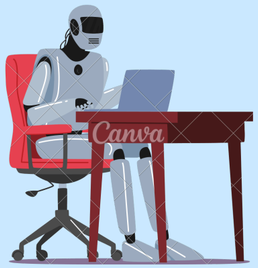
TRABAJOS FUTUROS

Clasificación Supervisada para Diagnóstico Médico

- Etiquetar los datos con categorías como "Diabetes", "Pre-Diabetes", "Sano".
- Aplicar algoritmos como Random Forest para clasificar pacientes.
- Evaluar el desempeño usando métricas como precisión, sensibilidad y área bajo la curva (AUC).



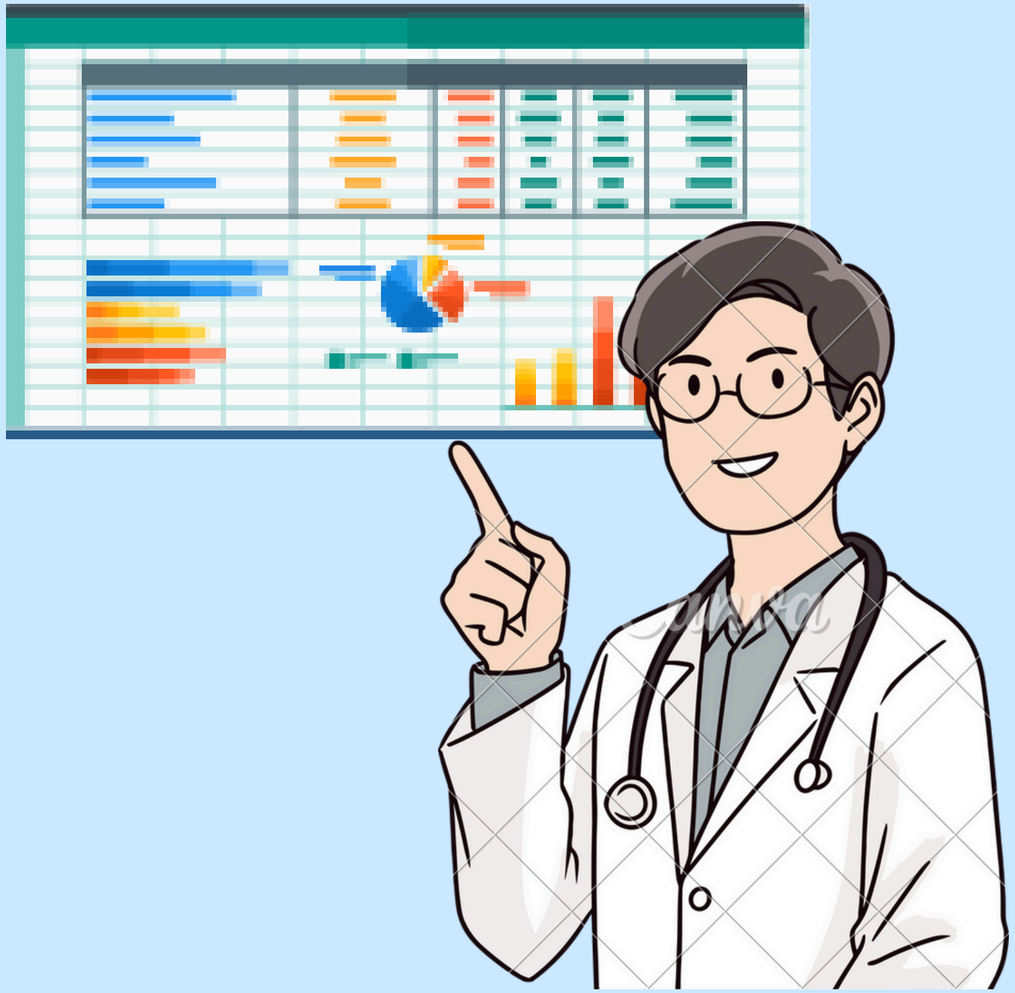




TRABAJOS FUTUROS

Integración con Datos de Salud Pública

- Realizar análisis comparativos con otros grupos poblacionales para evaluar cómo varían los factores de riesgo en diferentes regiones, edades o géneros.
- Estudiar si los outliers identificados efectivamente representan pacientes en riesgo o si requieren un ajuste adicional del modelo.



CONCLUSIONES

La implementación del modelo Isolation Forest en R permitió detectar outliers de manera más precisa y confiable al aplicar previamente un proceso de imputación de valores mediante el método KNN.

El algoritmo mostró su eficacia al identificar perfiles atípicos en variables clave como glucosa, IMC e insulina, destacando pacientes que se desvían significativamente del resto. Esto proporciona un punto de partida sólido para que un especialista médico evalúe estos casos y corrobore posibles diagnósticos tempranos o riesgos asociados, permitiendo una intervención más oportuna.

La detección confiable de anomalías mediante Isolation Forest facilita una mejor interpretación de los datos y sienta las bases para futuros análisis. Además de afinar los métodos gráficos de visualización, como es el caso del pair plot.