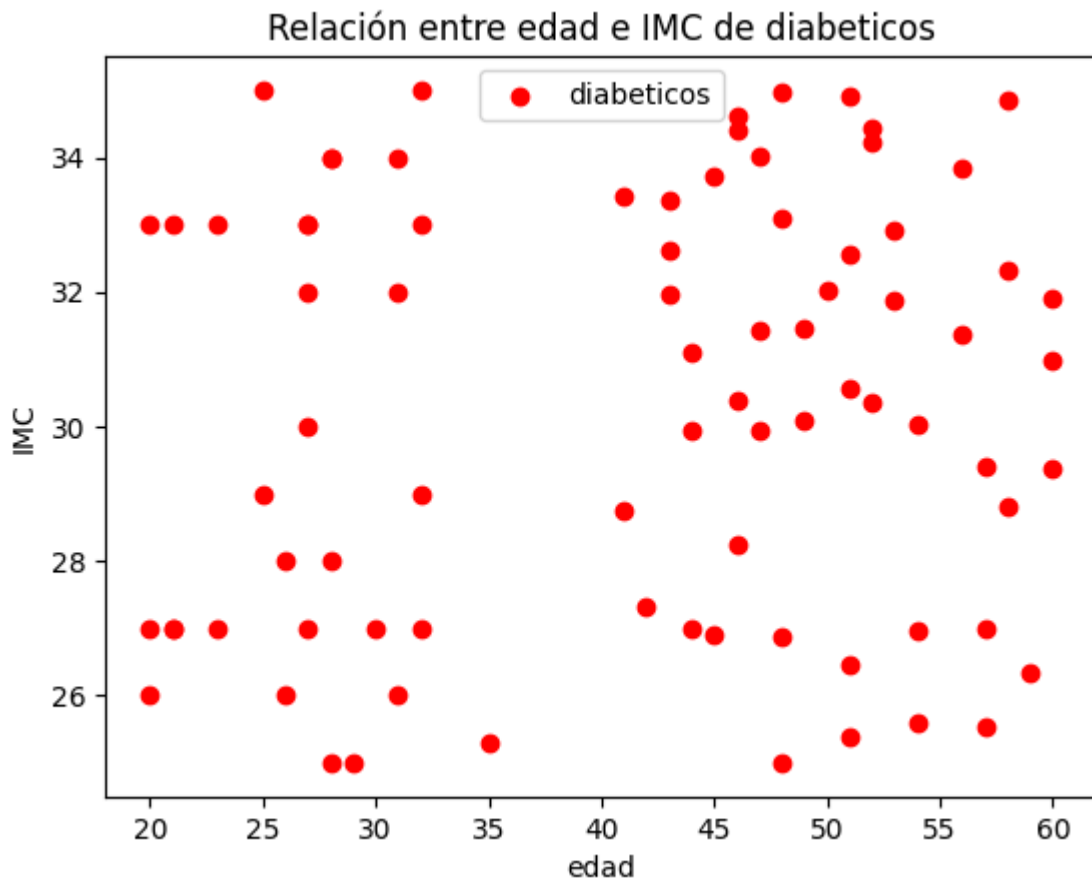
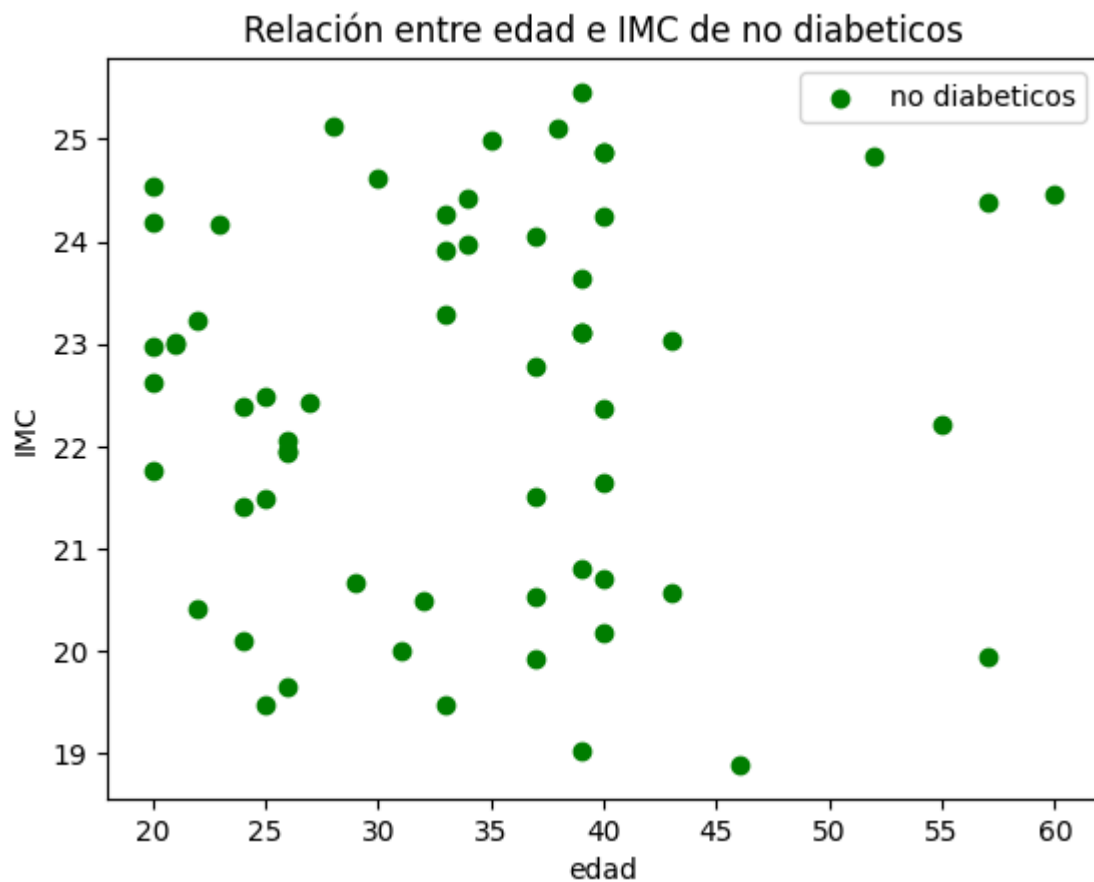


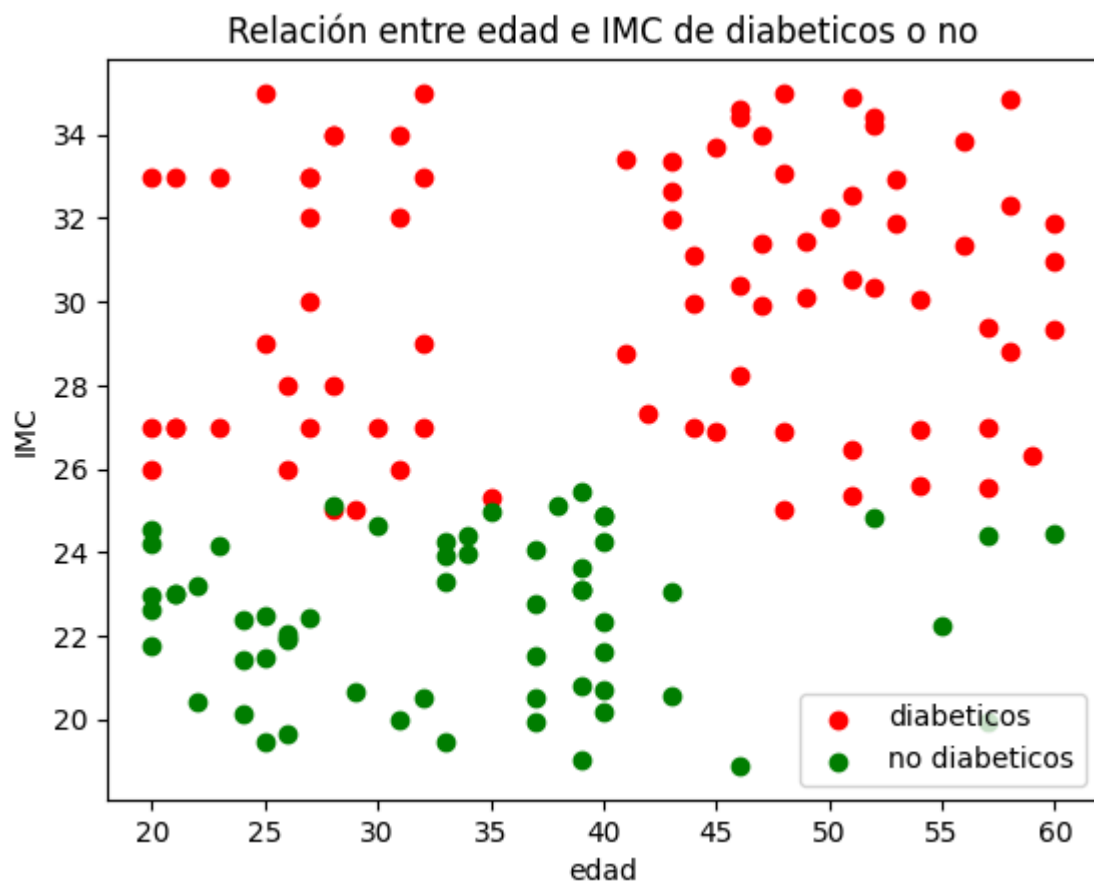
Empecemos con los datos separados:





Tal como podemos ver, en nuestra muestra mientras más edad más personas con diabetes hay, sin embargo también pueden haber personas con no mucha y, al tener un gran IMC, tener diabetes, o puede también no cumplirse ninguna de estas cosas.

A continuación el gráfico completo:



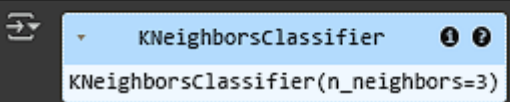
Para realizar el modelo KNN debemos instanciar el objeto

KNeighborsClassifier(*n_neighbors=vecinos*) y luego entrenarlo con los datos correspondientes.

```
[7] datos = data[["edad", "imc"]]
    clase = data["diabetes"]

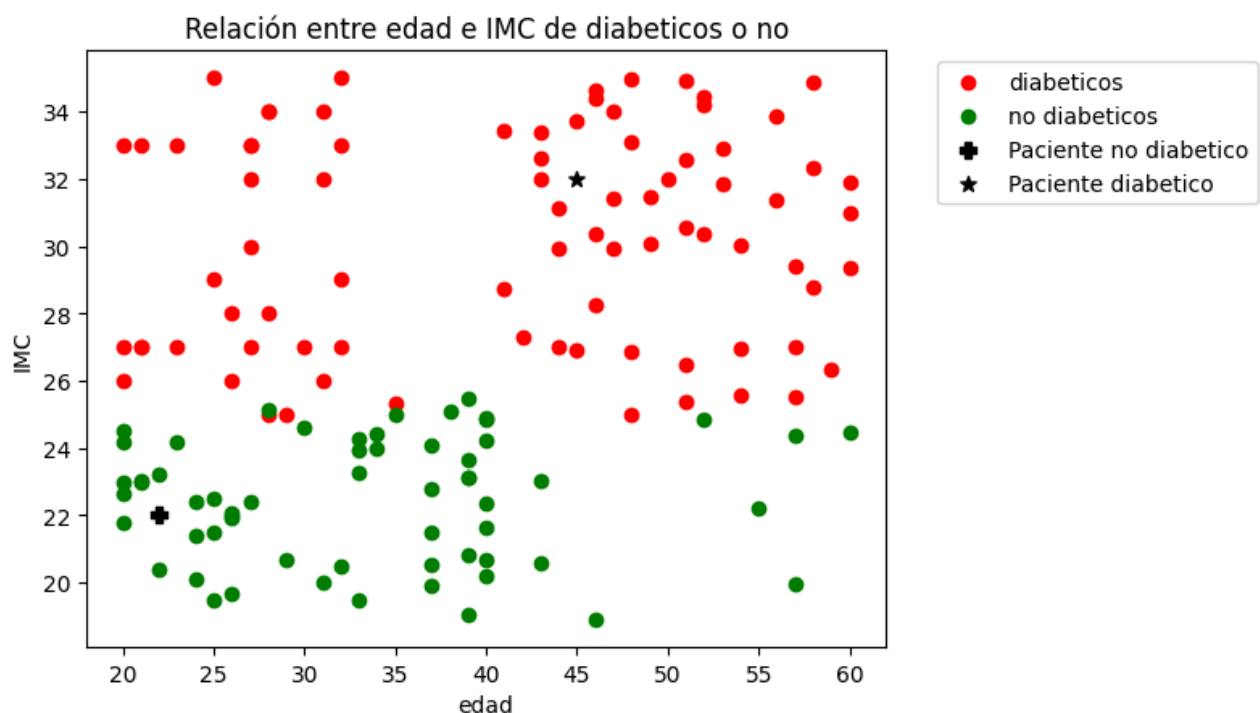
[8] clasificador = KNeighborsClassifier(n_neighbors=3)

    clasificador.fit(datos, clase)
```



Ya teniendo nuestro modelo creado podemos crear pacientes e intentar ver que probabilidad hay de que sean o no diabéticos

Por ejemplo:



El paciente diabetico tiene 45 años y un IMC es 32, nuestro modelo lo clasifica como [1]
El paciente no diabetico tiene 22 años y un IMC es 22, nuestro modelo lo clasifica como [0]

Para calcular la distancia entre los pacientes se establecen distancias de 3 tipos posibles:

Distancia Euclidiana

$$d(x, y) = \sqrt{\sum_{i=1}^k (x_i - y_i)^2}$$

Distancia Manhattan

$$d(x, y) = \sum_{i=1}^k |x_i - y_i|$$

Distancia Minkowski

$$d(x, y) = \left(\sum_{i=1}^k |x_i - y_i|^p \right)^{\frac{1}{p}}$$

Con esta distancia se establece una circunferencia y los k valores más cercanos son tomados como los k vecinos candidatos a establecer, en este caso, si un paciente tiene o no diabetes.

La principal ventaja de aplicar KNN es su simplicidad: con acomodar ciertos datos ya tendremos nuestro modelo.

Las desventajas son:

- Son muy costosos computacionalmente.
- Son muy sensibles a la normalización.