

# Trabajo Práctico N°4: Clasificador K-Neighbours

Introducción al Aprendizaje  
Automatizado



Alumno: Navall, Nicolás Uriel. N-1159/2.

b)

### Espirales sin ruido: K-neighbours

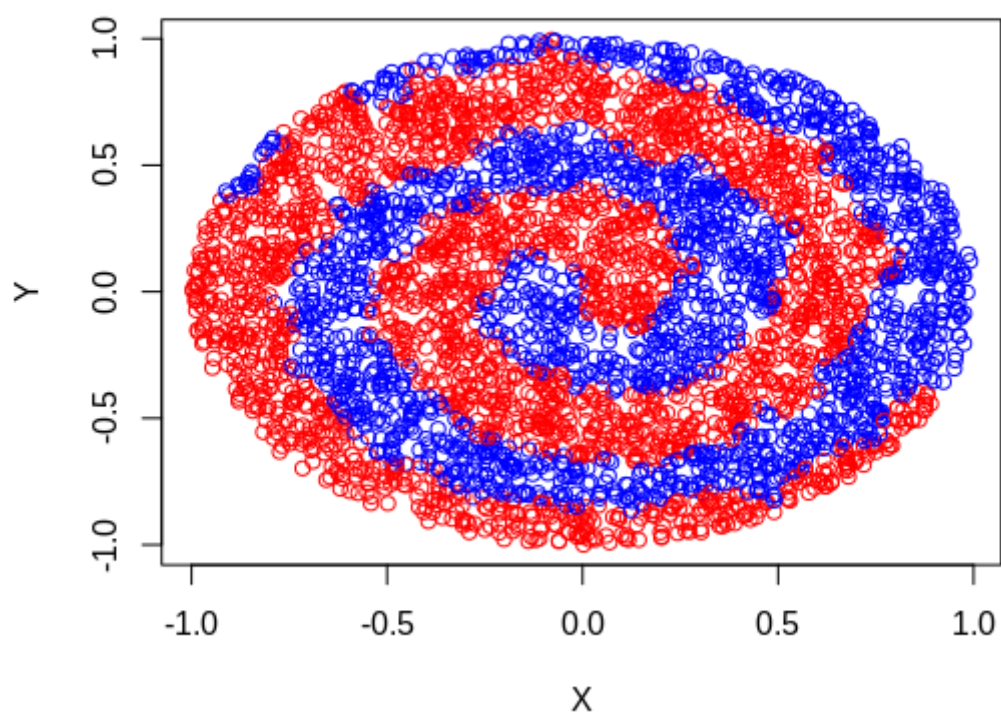


Figura 1.1

### Espirales sin ruido: Arboles

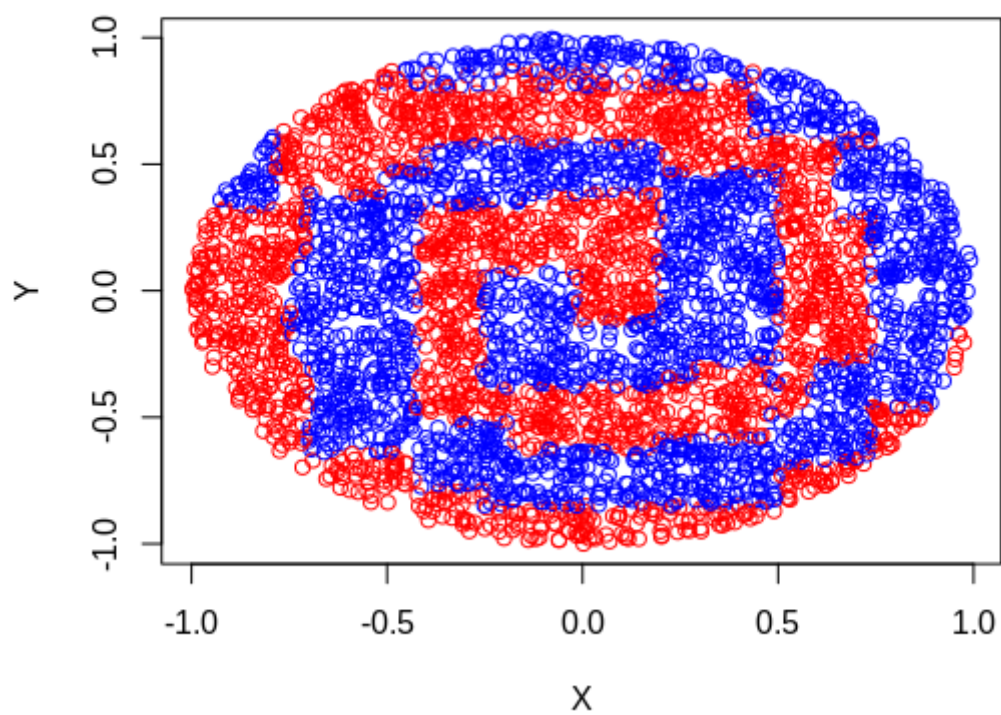


Figura 1.2

### Espirales con ruido: K-neighbours

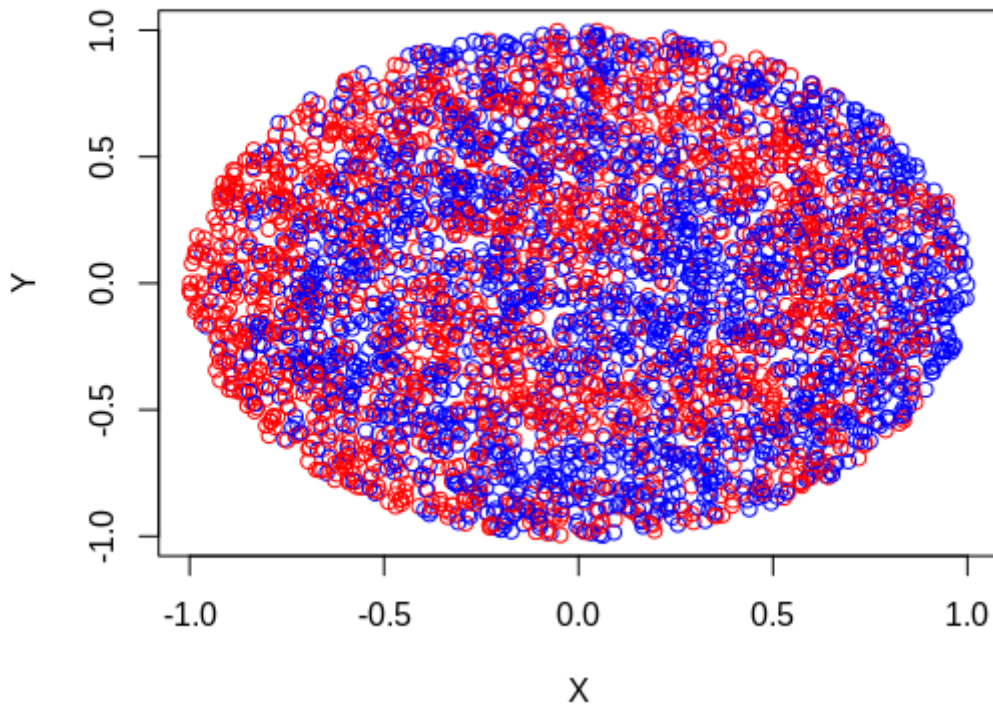


Figura 1.3

### Espirales con ruido: Árboles

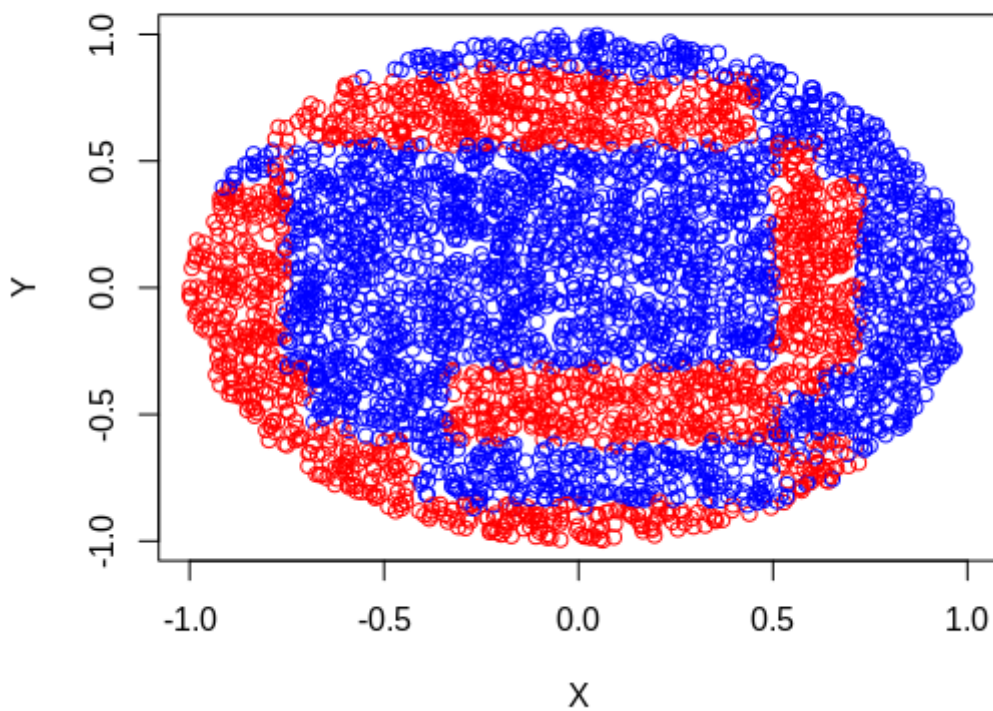


Figura 1.4

Es entendible que el método de k-vecinos obtenga un mejor resultado clasificando el espiral cuando no existe ruido con otras componentes, ya que no tiene las limitaciones de las “reglas” que tienen los árboles, permitiéndole realizar clasificaciones más complejas. También es de esperar como al

agregar dos componentes que no ayudan a clasificar los valores el clasificador de k-vecinos pierda tanta precisión, ya que hace que los valores de distancia entre los puntos cambien mucho (por que le da igual relevancia a todas las componentes cuando calcula la distancia entre valores), llevando así a que se utilicen k valores para la votación que no necesariamente son los más cercanos al punto a evaluar.

Si creo que es importante señalar que más dimensiones no necesariamente significa un peor resultado, pero si hay un número de componentes que no afectan a la catalogación relativamente grande comparado con el número de componentes que sí ayudan a definir la clase (como es este caso), entonces esta implementación del método de k-vecinos debería evitarse.

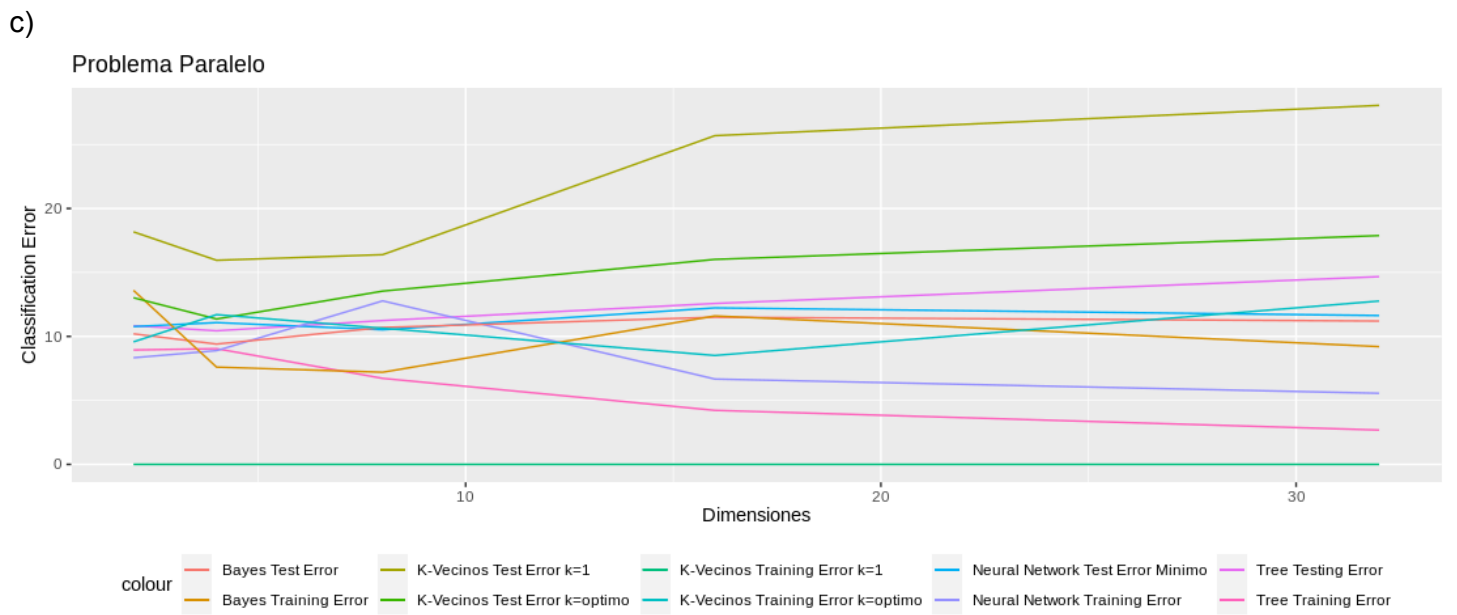


Figura 2.1

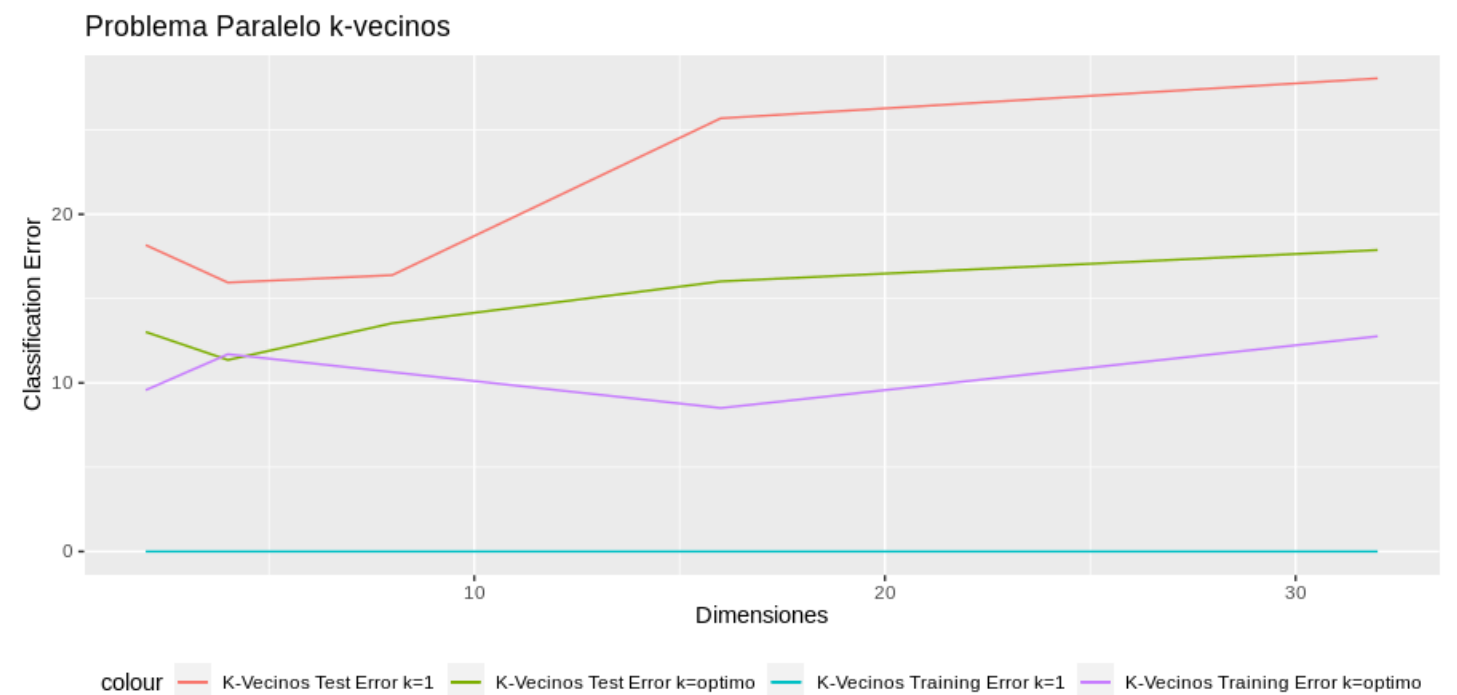


Figura 2.2

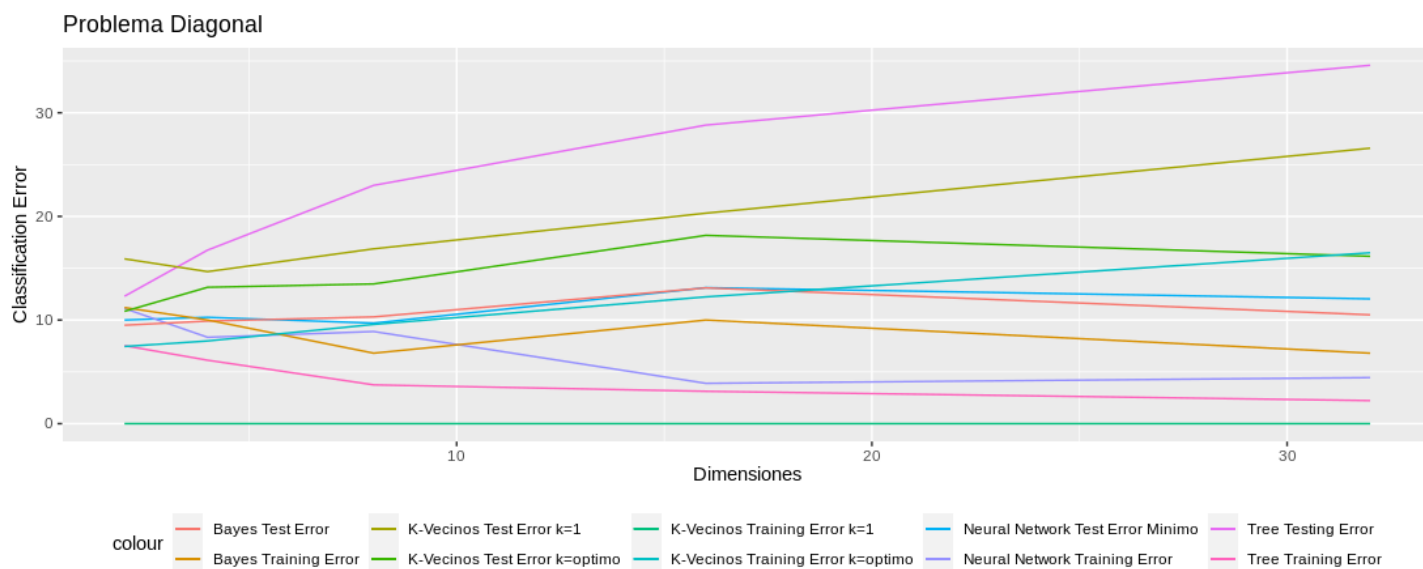


Figura 2.3

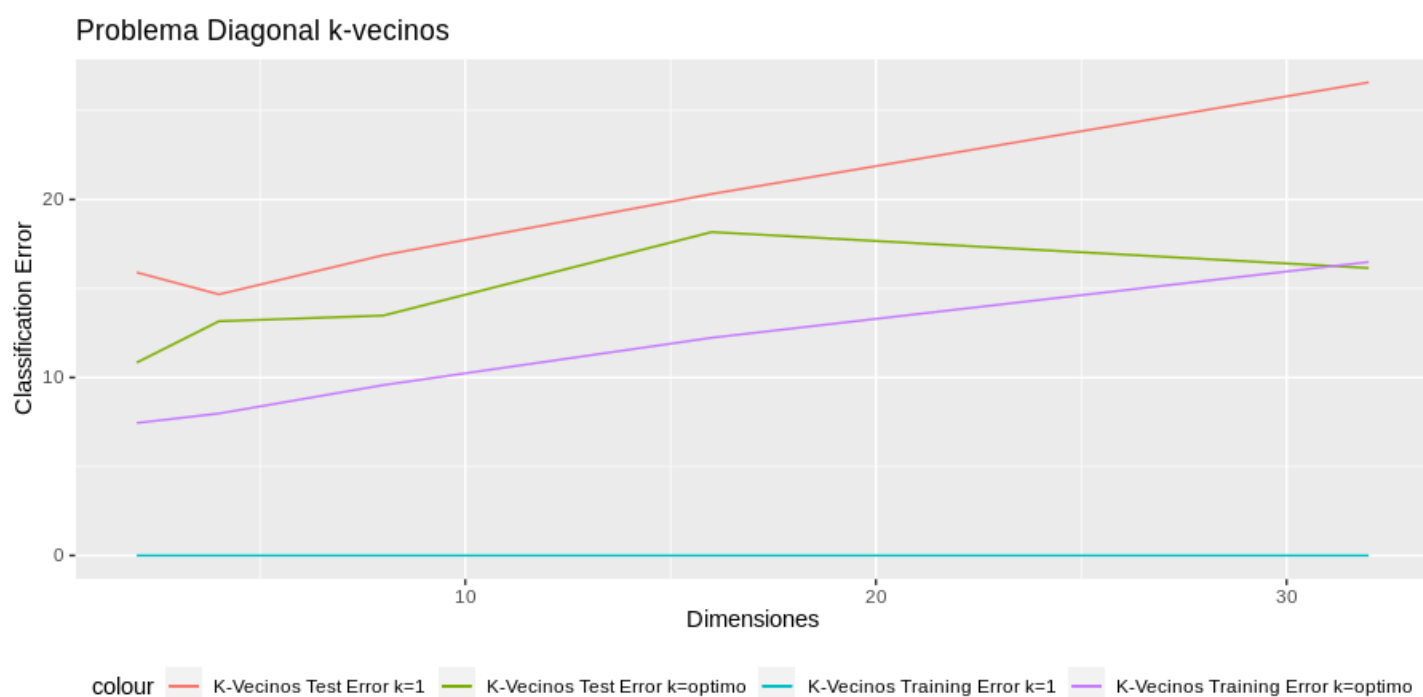


Figura 2.4

Los errores de training de los clasificadores k-vecinos cuando  $k=1$  en ambos problemas es igual a 0 ya que, como el clasificador solo revisa la clase del punto más cercano al punto a evaluar, va a asignarle a todos los puntos del dataset de training la misma clasificación que estos mismos tienen. Esto podría cambiarse para que en el caso que  $k=1$  se busque el segundo punto más cercano al calcular el error de training, ya que el obtenido no nos da mucha información sobre nuestro clasificador.

Es de esperarse que los errores de test en ambos problemas sean menores cuando el  $k$  es óptimo en comparación con cuando este es 1, ya que el  $k$  se optimiza con el set de datos de validación, que busca representar al “universo” de valores del problema, y como el set de “test” forma parte de este universo,

El aumentar las dimensiones en los problemas diagonal y paralelo no aumenta en gran manera el error como lo hacía en el problema de los espirales, ya que cada componente de los valores es relevante para clasificarlos.