Reconocimiento de Patrones
TP2 - Clasificación

Nicolás San Martín
Sebastián Sujarchuk
Eric Brandwein

Agosto 2020

1 Introducción

En este TP, implementaremos clasificadores de conjuntos de datos. En particular, clasificadores:

- por cuadrados mínimos;
- por discriminante de Fisher;
- y por regresión logística.

Además, realizaremos pruebas con distintos conjuntos de datos, tanto generados por nosotros como extraídos de librerías de análisis de datos. Por último, mostraremos los resultados de estas pruebas para comparar.

2 Clasificador por cuadrados mínimos

Implementamos el clasificador por cuadrados mínimos para K clases, y lo probamos para un conjunto de datos en $\mathbb{R}^2$ pertenecientes a tres clases diferentes. En primer lugar, del dataset "wines" de scikit-learn, extraímos un conjunto de datos manteniendo de cada vector solamente las primeras dos coordenadas para que los datos pertenezcan a $\mathbb{R}^2$, y manteniendo los datos de las tres clases. Es decir que queremos probar nuestro clasificador para clasificar los datos provenientes de tres clases. La dimensionalidad de los datos era originalmente 13, pero elegimos las dos primeras dimensiones de acuerdo al requisito de la consigna. Los datos así obtenidos se pueden ver en la Figura 1. A simple vista, puede verse que no son linealmente separables y por lo tanto podemos esperar que el *accuracy* no sea cercano a 1. Tanto en este caso como en el siguiente, el modelo se entrenó con un dataset de entrenamiento, y luego se lo evaluó con un set de testing.

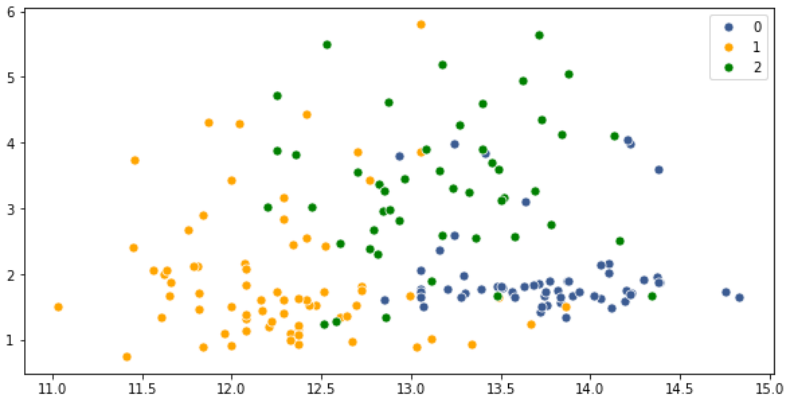

Figure 1: Dataset generado desde el dataset "wines" de scikit-learn. Cada color corresponde a una clase diferente.

Efectivamente, el accuracy alcanzado por la aplicación de este clasificador sobre este conjunto de datos fue del $0,76$. En la Figura 2 pueden observarse los resultados.

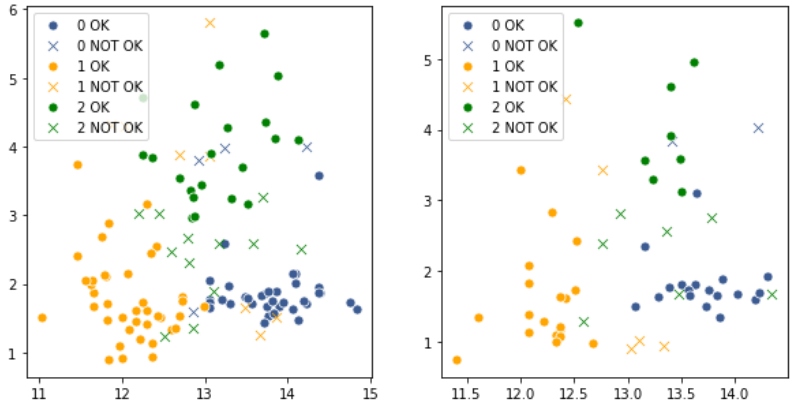

Figure 2: Clasificación por el clasificador de cuadrados mínimos sobre el dataset de vinos. El color de un punto corresponde con la clasificación verdadera, y la forma corresponde con el resultado de la clasificación predicha por el algoritmo.

Luego de esto, utilizamos un dataset generado sintéticamente, con puntos más separables. Se trata de puntos en $\mathbb{R}^2$ obtenidos de tres diferentes distribuciones Gaussianas con la misma varianza pero medias distintas, tal como puede verse en la Figura 3.

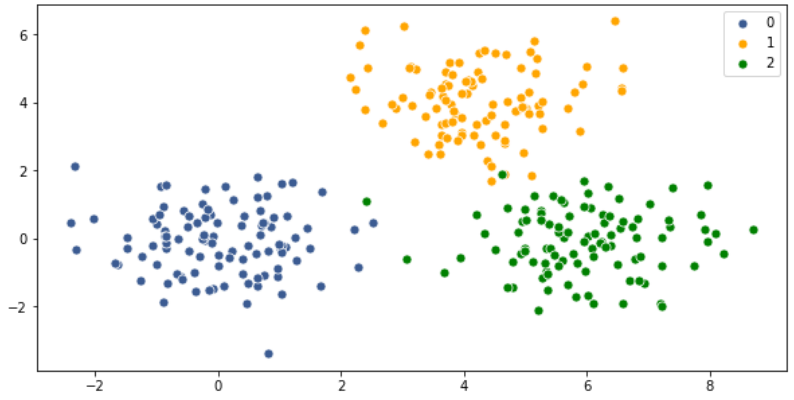

Figure 3: Datos sintéticos de tres clases con distribución Gaussiana en $\mathbb{R}^2$.

Utilizando el clasificador con este dataset obtenemos un accuracy de 1 y el resultado de dicha clasificación está graficado en la Figura 4.

3 Discriminante de Fisher

Implementamos un discriminante de Fisher para 2 clases, y otro para K clases en general. Los conjuntos de datos utilizados para la prueba de estos discriminantes también provinieron del dataset "wines" de scikit-learn; para el primero, mantuvimos solamente las primeras dos coordenadas de cada vector de dato, como en el punto anterior, y mantuvimos datos de solo dos clases, como se puede ver en la Figura 5. Para el segundo, mantuvimos las primeras tres coordenadas de cada vector de dato, y mantuvimos datos de las tres clases, como se puede ver en la Figura 7.

El discriminante de Fisher nos da un hiperplano sobre el cual proyectar los puntos, de forma tal que minimiza la varianza intraclases y maximiza la interclases. Dados lo puntos proyectados, asumimos que cada clase tiene una distribución gaussiana. Esto significa que debemos estimar dos parámetros μ y σ para conocer a cada una. De esta forma, utilizamos los puntos proyectados para calcular la media y varianza empíricas como estimadores de los

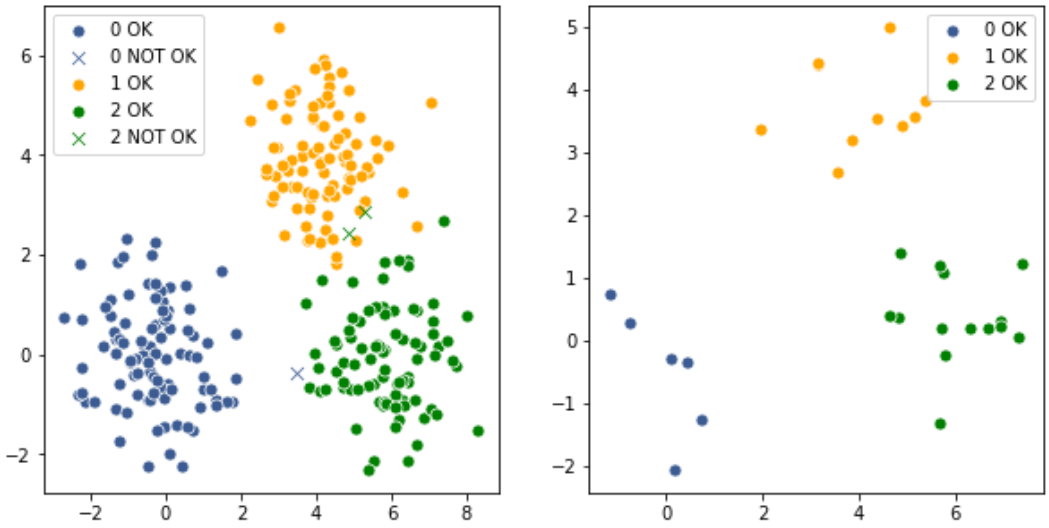

Figure 4: Resultado de clasificar los datos sintéticos de tres clases usando el clasificador de cuadrados mínimos.

parámetros buscados. Así, obtenemos las distribuciones de las proyecciones de cada una de las clases en el hiperplano en cuestión. Luego, para clasificar un punto nuevo, calculamos la probabilidad de dicho punto proyectado, para la distribución de cada una de las clases y esto nos da un vector de probabilidades. Lo que hacemos entonces es asignar al punto a la clase de mayor probabilidad.

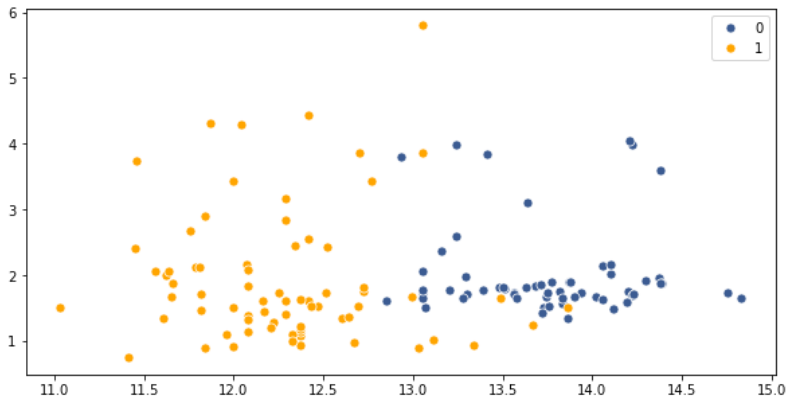

Figure 5: Dataset de dos clases generado desde el dataset "wines" de scikit-learn.

Usando el clasificador de Fisher para dos clases, y el dataset mencionado anteriormente, alcanzamos una accuracy de $0,97$. En la Figura 6 se observa un gráfico con los puntos clasificados, indicando sus clases de pertenencia y la clase asignada por el clasificador. También se grafica la recta sobre la cual el clasificador proyecta los datos para clasificarlos.

Usando el clasificador de Fisher para tres clases, y el dataset de tres clases mencionado anteriormente, alcanzamos una accuracy de $0,81$. Los resultados de la clasificación pueden verse en la Figura 8.

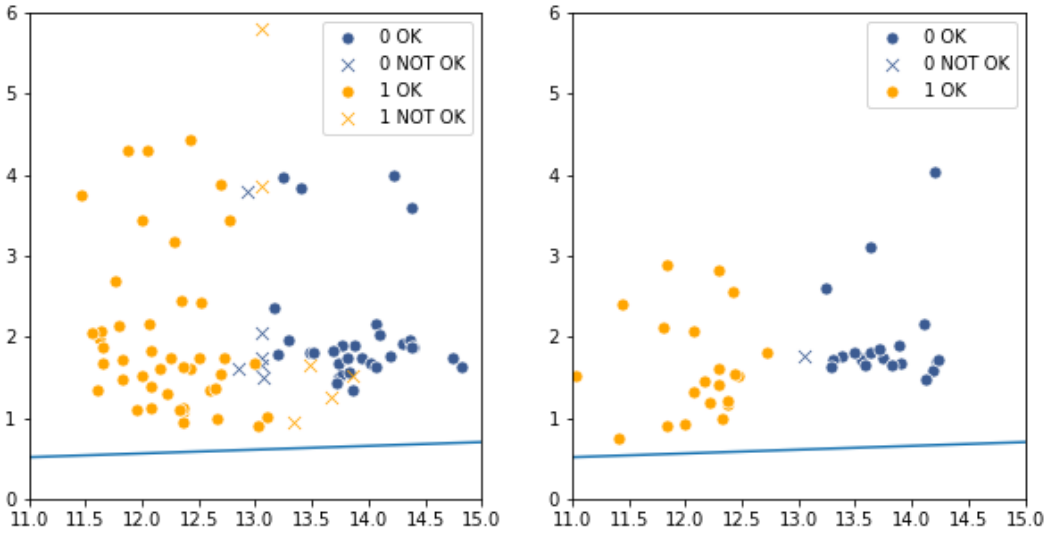

Figure 6: Resultados de la clasificación de Fisher para dos clases, junto a la recta en la que el algoritmo proyecta los puntos.

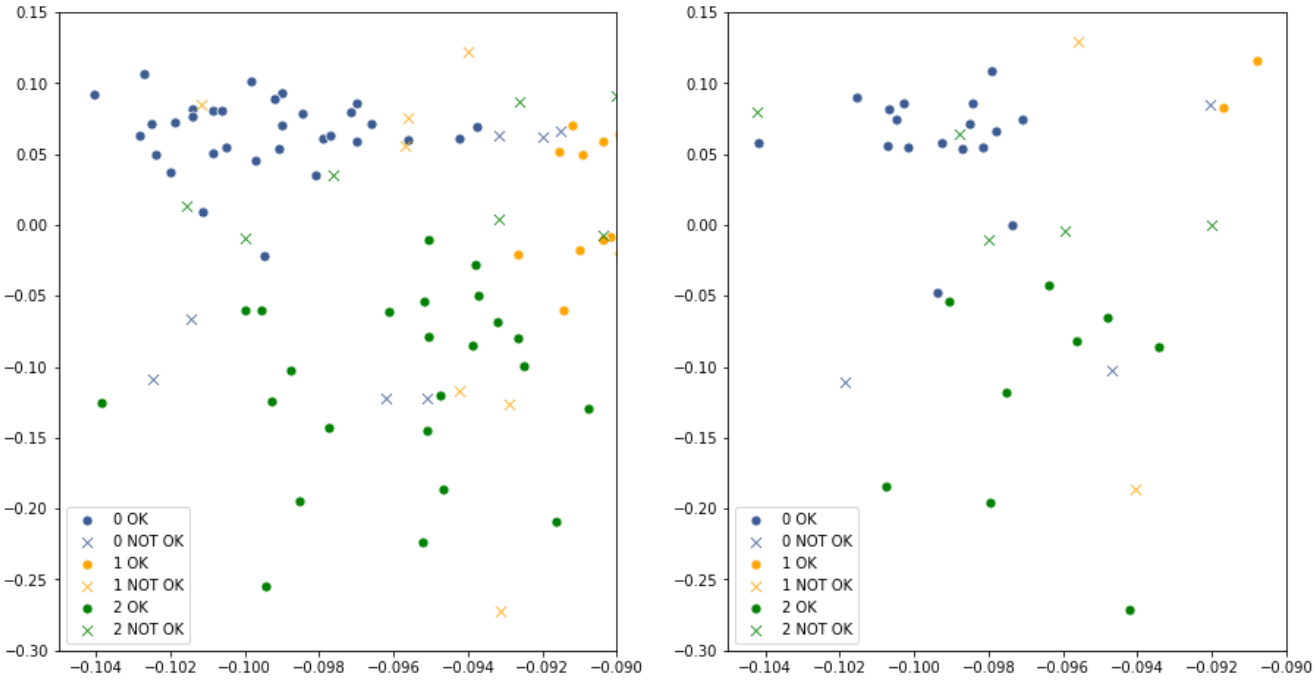

Figure 7: Dataset de tres clases subconjunto de $\mathbb{R}^3$ generado desde el dataset "wines" de scikit-learn.

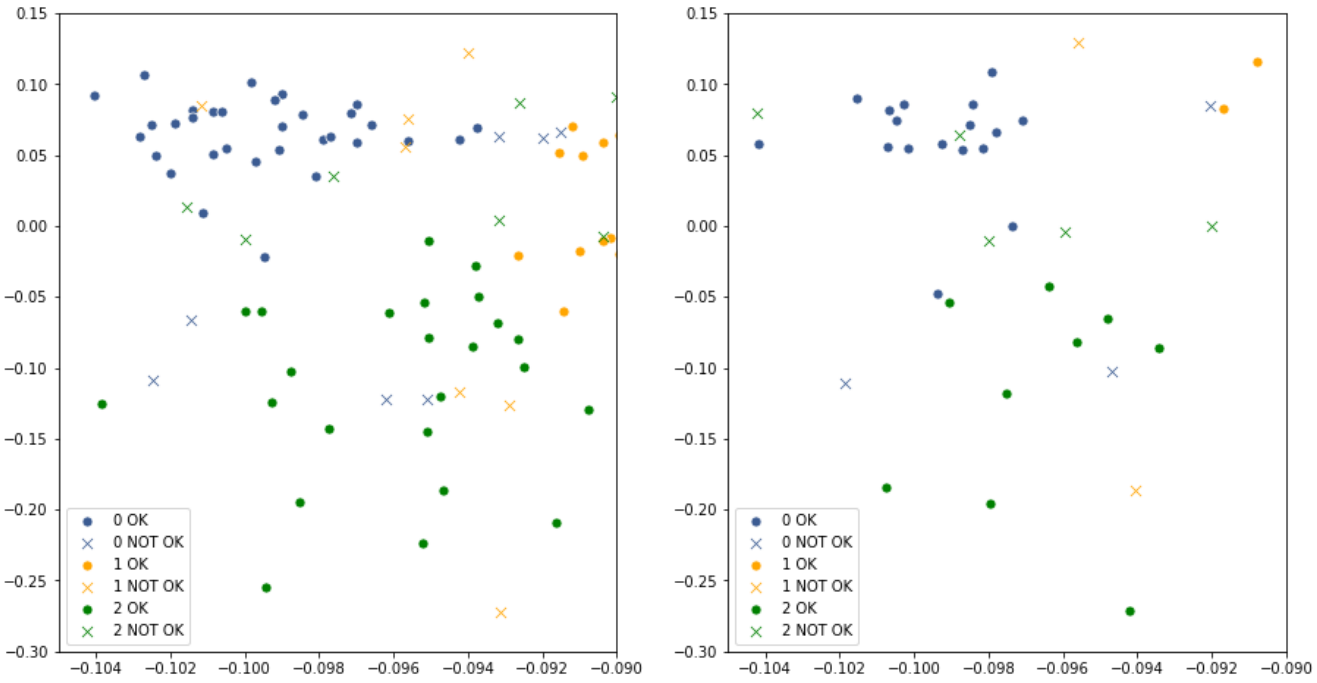

Figure 8: Resultados de la clasificación de Fisher para tres clases, proyectadas en un plano.

4 Regresión logística

En este punto, implementamos el clasificador de regresión logística para dos clases. Primeramente, lo utilizamos para clasificar puntos esparcidos en el plano linealmente separables, obtenidos, como previamente, del dataset "wines" de scikit-learn, visibles en la Figura 5. Los resultados de la clasificación pueden verse en la Figura 9.

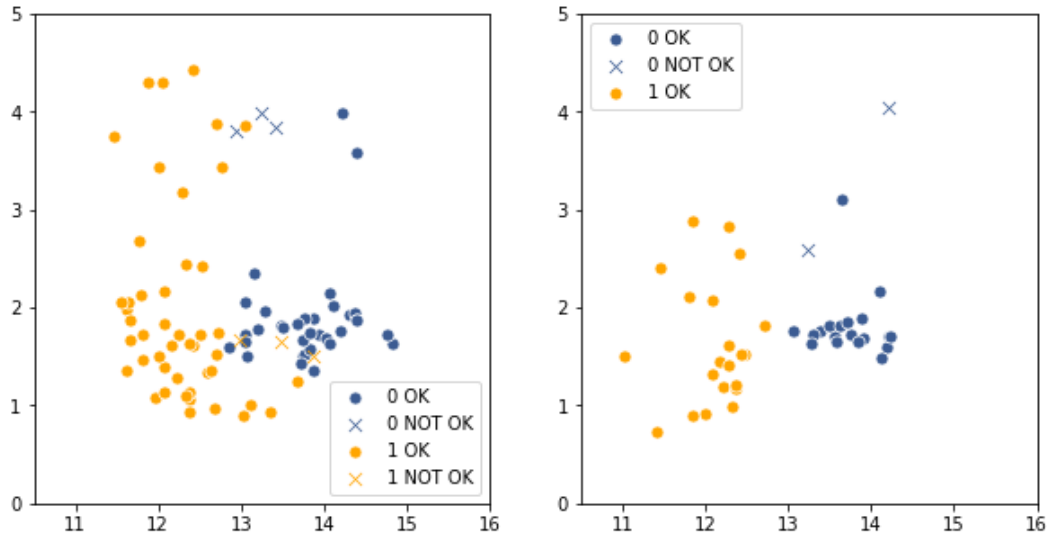

Figure 9: Resultado de clasificar un subconjunto del dataset de dos clases descripto anteriormente.

Con este clasificador obtuvimos un accuracy de $1,0$, o sea, todos los puntos fueron bien clasificados. Esto es debido a que el dataset original, como se ve en el gráfico, es muy bien separable.

Para lograr clasificar datos no separables linealmente, implementamos el mismo clasificador pero con una *basis function*, que convierte las coordenadas de los puntos de tal forma que sí sean separables, idealmente. En este caso, utilizamos una basis function elíptica. El conjunto de datos que utilizamos para probarlo tenía la forma de una nube de puntos centrada en el $(0,0)$ perteneciente a la primera clase, rodeada de un "anillo" de puntos perteneciente a la segunda. Para generar la primera clase de puntos, utilizamos una distribución Gaussiana multivariada de media $\mathbf{0}$. Para la segunda clase de puntos, generamos cada uno con un ángulo con distribución uniforme entre 0 y 2π, y un radio con distribución Gaussiana. Esto representa cada punto en coordenadas polares, que luego convertimos en coordenadas cartesianas.

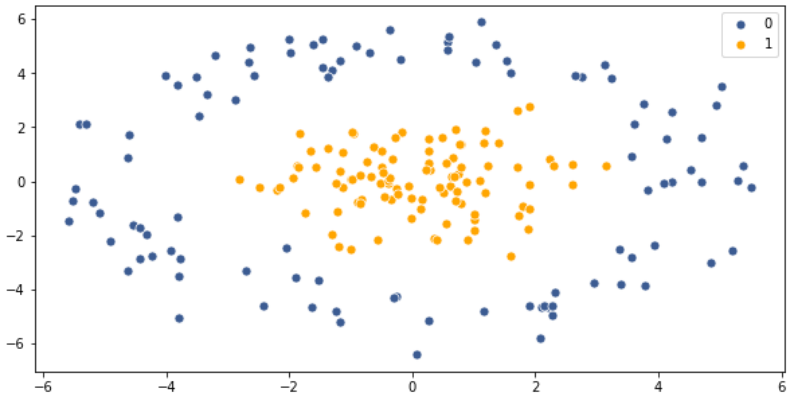

Figure 10: Dataset generado para el clasificador de fisher con una basis function elíptica.

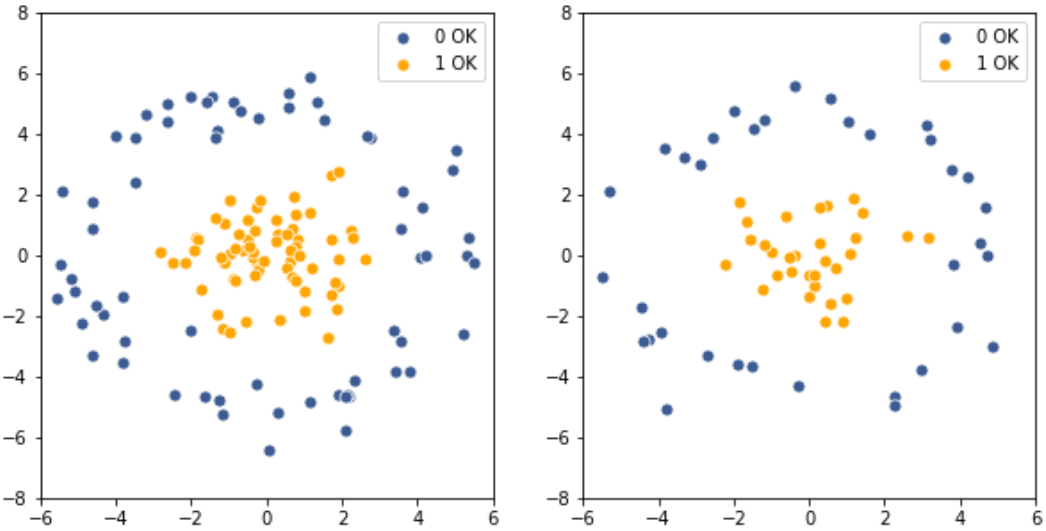

Figure 11: Resultado de clasificar un subconjunto del dataset de dos clases descripto anteriormente.

El accuracy alcanzado al clasificar los datos aplicando la basis function elíptica antes de aplicar el clasificador de regresión logística fue del $0,96$, con lo cual podemos decir que los datos pudieron ser separados, dado que es una accuracy razonablemente alta. Uno podría esperar que el mismo clasificador sin basis function tendría resultados abismales, ya que los datos no son para nada separables por un plano.

A Apéndice de notación

El color de los puntos en cada gráfico define la clase verdadera del punto correspondiente. En donde corresponde, la forma define si fue bien clasificado o no, siendo un círculo para bien clasificado o una cruz para mal clasificado. Mal clasificado es NOT OK, bien clasificado es OK. Por ejemplo, un punto 1 NOT OK es un punto de la clase 1 que fue mal clasificado.

Por cada figura de clasificación de puntos hay dos gráficos. El de la izquierda es la clasificación sobre el conjunto de training, y el de la derecha es la clasificación sobre un conjunto de testing.