1. EJERCICIO SOBRE LA BÚSQUEDA ITERATIVA DE ÓPTIMOS

1.2.a) Considerar la función E(u, v) = $(u^3e^{(v-2)}-2v^2e^{-u})^2$. Calcular analíticamente y mostrar la expresión del gradiente de la función E(u,v)

El método de obtención del vector gradiente de una función cualquiera f(x,y) es con el cálculo del vector cuyas coordenadas corresponden con las derivadas parciales de esta función con respecto a cada una de sus variables, en este caso x e y.

Por tanto, para mostrar la expresión del gradiente de la función E necesitaremos derivar la función E(u,v) = $(u^3e^{(v-2)}-2v^2e^{-u})^2$ con respecto a u y v.

Recordemos algunos conceptos básicos sobre las derivadas compuestas.

- *La derivada de $y = f(x)^n$ es $y' = n \cdot f(x)^{n-1} \cdot f'(x)$*
- *La derivada de un producto: $f(x) = u \cdot v \rightarrow f'(x) = u' \cdot v + u \cdot v'$*
- *La derivada de $e^{f(x)} = e^{f(x)} \cdot f'(x)$*

Derivada de E(u,v) con respecto a u (v cuenta como un número independiente más):

- $(u^3e^{(v-2)})' = (u^3)' \cdot (e^{(v-2)}) + (u^3) \cdot (e^{(v-2)})' = 3u^2 \cdot (e^{(v-2)}) + u^3 \cdot (e^{(v-2)}) \cdot 0 = 3u^2e^{(v-2)}$
- $(v^2e^{-u})' = (v^2)' \cdot (e^{-u}) + (v^2) \cdot (e^{-u})' = 0 \cdot (e^{-u}) + v^2 \cdot (e^{-u}) \cdot (-1) = - v^2e^{-u}$
- $(u^3e^{(v-2)}-2v^2e^{-u})' = (u^3e^{(v-2)})' - 2(v^2e^{-u})' = 3u^2e^{(v-2)} - 2(- v^2e^{-u}) = 3u^2e^{(v-2)} + 2\, v^2e^{-u}$
- ***dEu: $E'(u,v) = 2(u^3e^{(v-2)}-2v^2e^{-u}) \cdot (u^3e^{(v-2)}-2v^2e^{-u})' = 2(u^3e^{(v-2)}-2v^2e^{-u})\,(3u^2e^{(v-2)} + 2\, v^2e^{-u})$***

Derivada de E(u,v) con respecto a v (u cuenta como un número independiente más):

- $(u^3e^{(v-2)})' = (u^3)' \cdot (e^{(v-2)}) + (u^3) \cdot (e^{(v-2)})' = 0 \cdot (e^{(v-2)}) + u^3 \cdot (e^{(v-2)}) \cdot 1 = u^3e^{(v-2)}$
- $(v^2e^{-u})' = (v^2)' \cdot (e^{-u}) + (v^2) \cdot (e^{-u})' = 2v \cdot (e^{-u}) + v^2 \cdot (e^{-u}) \cdot 0 = 2ve^{-u}$
- $(u^3e^{(v-2)}-2v^2e^{-u})' = (u^3e^{(v-2)})' - 2(v^2e^{-u})' = u^3e^{(v-2)} - 2(2ve^{-u}) = u^3e^{(v-2)} - 4ve^{-u}$
- ***dEv: $E'(u,v) = 2(u^3e^{(v-2)}-2v^2e^{-u}) \cdot (u^3e^{(v-2)}-2v^2e^{-u})' = 2(u^3e^{(v-2)}-2v^2e^{-u})\,(u^3e^{(v-2)} - 4ve^{-u})$***

De esta forma ya tenemos la derivada parcial con respecto a cada variable de la función, las cuales forman el gradiente de la forma gradE = (dEu(u,v), dEv(u,v)). Veamos un ejemplo:

Para el punto (2, 3) de la función E obtendríamos el valor

$$E(2,3) = (2^3e^{(3-2)}-2\cdot3^2e^{-2})^2 = (8e-18e^{-2})^2 = (21.75 - 2.44)^2 = 373.03;$$

pero su gradiente sería el vector (dEu(u,v), dEv(u,v)):

$$dEu(2,3) = 2(2^3e^{(3-2)}-2\cdot3^2e^{-2})(3\cdot2^2e^{(3-2)}+2\cdot3^2e^{-2}) =$$

$$= 2(8e-18e^{-2})(12e+18e^{-2}) =$$

$$= 2 \cdot 19{,}31 \cdot 35.06 = 1353{,}84$$

$$dEv(2,3)= 2(2^3e^{(3-2)}-2\cdot3^2e^{-2})(2^3e^{(3-2)}-4\cdot3e^{-2}) =$$

$$= 2(8e-18e^{-2})(8e-12e^{-2}) =$$

$$= 2 \cdot 19{,}31 \cdot 20{,}12 = 777{,}12$$

Por tanto, el gradiente de E en el punto (2,3) sería el vector (1353,84; 777,12).

1.2.b) y c) ¿Cuántas iteraciones tarda el algoritmo en obtener por primera vez un valor de E(u,v) inferior a 10^{-14} y en qué coordenadas ocurre?

Con un punto inicial ($\mathbf{w_0}$) en (1,1), una tasa de aprendizaje 0.1 y la condición de parada de obtener un valor menor a 10^{-14} llegamos al número deseado en 10 iteraciones, con las coordenadas (1.16, 0.91).

```
Ejercicio1: Apartado 2:

    b) Numero de iteraciones:  10
    c) Coordenadas obtenidas: ( 1.16 ,  0.91 )
```

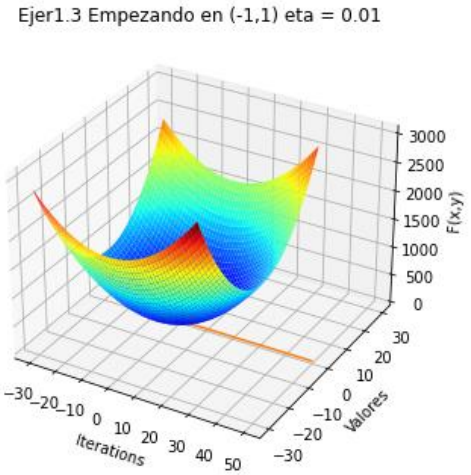
1.3.a) Considerar la función $f(x, y) = (x+2)^2 + 2(y-2)^2 + 2\sin(2\pi x)\sin(2\pi y)$. Usar gradiente descendente para minimizar la función y generar un gráfico de cómo desciende el valor con las iteraciones con tasa de aprendizaje 0.1 y 0.01 y partiendo del punto (-1,1). Comentar las diferencias y dependencias de eta (tasa de aprendizaje).

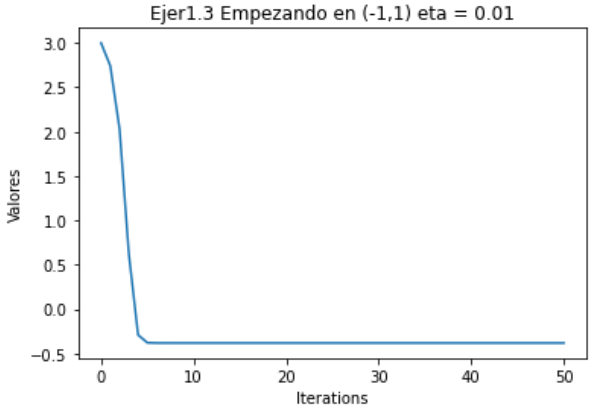
Para realizar el gradiente descendente volvemos a necesitar las derivadas de la función con respecto a x y a y, que en este caso no se explicará el procedimiento de obtención, pero son:

$$f'_x(x, y) = 2(x+2) + 4\pi\cos(2\pi x)\sin(2\pi y)$$

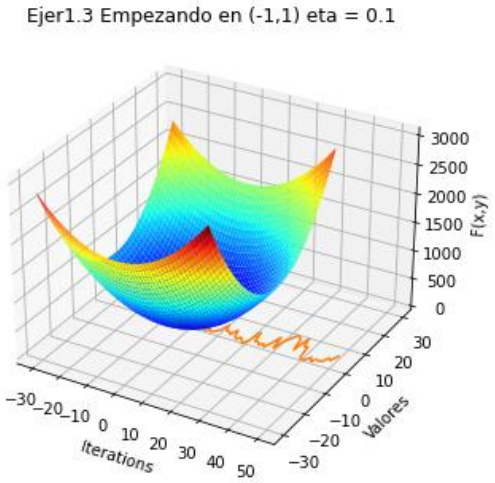
$$f'_y(x, y) = 4(y-2) + 4\pi\sin(2\pi x)\cos(2\pi y)$$

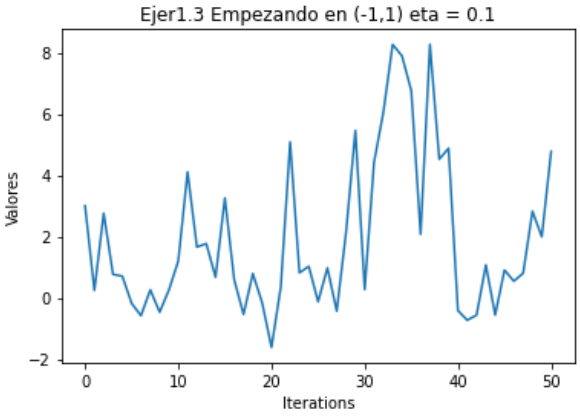
- Con una tasa de 0.01 y un punto de partida de (-1,1) obtenemos en 50 iteraciones las coordenadas del mínimo en (-1.27 , 1.29) con el valor -0.38.





- Con una tasa de 0.1 y un punto de partida de (-1,1) obtenemos en 50 iteraciones las coordenadas del mínimo en (-2.96, 0.57) con el valor 4.77.





<u>Conclusión sobre diferencias y dependencias:</u>

Para un mismo número de iteraciones y partiendo del mismo punto inicial, los resultados obtenidos son bastante diferentes en un caso y otro. Esto es debido a la diferencia de la tasa de aprendizaje. La importancia de una buena elección en la tasa de aprendizaje reside en que de esta depende lo drástico que es el cambio del valor de w, que son las coordenadas de la función por las que iteramos. Si la tasa es un valor muy pequeño, podemos avanzar por la w a un ritmo perjudicialmente lento, pero si es demasiado grande podemos saltarnos repetidas veces el mínimo que buscamos.

Es por esto, por lo que vemos en el ejemplo un cambio significativo en los resultados. El ejercicio limita el número de iteraciones y no prioriza que se alcance un mínimo o un valor menor que otro, por lo que el resultado final será aquel que se obtenga cuando las 50 iteraciones se hayan realizado. Con una tasa de 0.01 llegamos al mínimo de -0.38, que se encuentra en las coordenadas (-1.27 , 1.29). Este mínimo, en comparación con el obtenido para una tasa de 0.1 parece ser bastante mejor resultado, pues este segundo valor alcanzado es el que se encuentra en las coordenadas (-2.96 , 0.57) de 4.77.

La explicación más coherente es que con la tasa más baja de las dos, difícilmente cambiemos los valores de w tan bruscamente como para saltarnos el mínimo que buscamos. Un problema podría ser que avance tan lento que se haya quedado a medio camino de un resultado mejor al que habría llegado en más iteraciones. Por otra parte, la tasa mayor parece haber avanzado más rápido y haber esquivado inintencionadamente el valor más óptimo al que optaba en ese número de iteraciones.

Estas conclusiones pueden verse reflejadas en las gráficas, ya que en una de ellas (la que alcanza -0.38) vemos que por cada iteración obtiene un valor más pequeño que en el anterior siendo siempre descendente, mientras que la otra se encuentra valores más altos y más bajos debido a las oscilaciones que hace. La tasa perfecta sería aquella con el valor que no haga el avance demasiado lento ni demasiado brusco.

1.3.b) Obtener el mínimo y los valores de las variables en las que se alcanzan cuando se parte de (-0.5, -0.5), (1,1), (2.1,-2.1), (-3,3) y (-2,2) y obtener una tabla con los valores. Comentar la dependencia del punto inicial

ETA = 0.01		
PUNTO DE PARTIDA	**COORDENADAS DEL MINIMO**	**MINIMO**
(-0.5,-0.5)	(-0.79 , -0.13)	9.13
(1.0, 1.0)	(0.68 , 1.29)	6.44
(2.1,-2.1)	(0.15 , -0.10)	12.49
(-3.0, 3.0)	(-2.73 , 2.71)	-0.38
(-2.0, 2.0)	(-2.00 , 2.00)	0.00

ETA = 0.1		
PUNTO DE PARTIDA	**COORDENADAS DEL MINIMO**	**MINIMO**
(-0.5,-0.5)	(-3.03 , 1.67)	1.6
(1.0, 1.0)	(-1.78 , 2.06)	0.82
(2.1,-2.1)	(-2.31 , 1.87)	1.47
(-3.0, 3.0)	(-0.27 , 1.42)	2.68
(-2.0, 2.0)	(-1.80 , 2.36)	1.78

Como conclusión, vemos que existe una relación directa entre el punto del que se parte y la calidad de la solución que se obtiene. Tras una investigación sobre el método correcto en la elección del punto, he contrastado que en muchos de los casos en los que se utiliza este método, el punto de partida es aleatorio, lo cual me hace suponer que no debe haber un método formalizado para la elección de este primer punto más allá de la experimentación.

Al tener un tiempo de ejecución y un número de iteraciones limitados, los puntos de los que se nos pide partir parecen estar lo suficientemente cerca del mínimo como para que el experimento tenga sentido, pero poniéndome en la situación de que querer minimizar una función iterativamente con libertad en las coordenadas de comienzo, no tengo forma de intuir la distancia al mínimo de la función sin haber experimentado con ella antes.

Es verdad que, con una misma tasa de aprendizaje, el punto inicial puede ser determinante para encontrar el mínimo de la función, ya que cuanto más cerca esté, menos iteraciones necesitará para alcanzarlo siempre que la tasa de aprendizaje sea la correcta.

Evaluando los datos obtenidos en las tablas, los valores más bajos cuando la tasa de aprendizaje es 0.01 se obtienen partiendo de las coordenadas (-3,3) y (-2,2), pero para una tasa de 0.1, los valores mejores valores se obtienen cuando el punto inicial es alguno entre (-0.5,-0.5), (1,1) y (2.1,-2.1), lo cual impide sacar una conclusión clara.

Finalmente, pienso que la dependencia del punto inicial es tan relevante como la de la tasa o el número de iteraciones máximo y, cambiando únicamente esto no aseguramos obtener mejoras en los resultados.

1.4 ¿Cuál sería su conclusión sobre la verdadera dificultad de encontrar el mínimo global de una función arbitraria?

Encontrar el mínimo global de una función arbitraria puede llegar a tener muchísimas dificultades. Para empezar, utilizar el gradiente descendente necesita de parámetros definidos por nosotros mismos sin un criterio exacto de cuál es el correcto, como por ejemplo el valor de la tasa de aprendizaje. Este debe ser el correcto para que no avance lento ni esquive los mínimos. Otros puntos influyentes son el número de iteraciones máximo permitidos (que cuantos más sean, más cercano al mínimo quedara el resultado, pero un número excesivo puede hacerlo ineficiente) y el punto de partida (que puede ser aleatorio).

Otro error frecuente puede ser quedarse atrapado en un mínimo local, ya que el gradiente cuenta con derivada de la función, que es su pendiente, y de tener una tasa de aprendizaje demasiado grande puede estar continuamente saltando de un lado a otro del mínimo local sin salir. También, encontrar un máximo entre dos valles puede plantear el dilema de cuál de los dos es el más profundo e inducirnos a un error.

Como conclusión, los principales errores, o al menos aquellos que dependen de nosotros, residen en la elección de una tasa de aprendizaje correcta, un punto de partida no muy lejano y limitar la cantidad de iteraciones permitida para que no deje de ser eficiente la búsqueda.

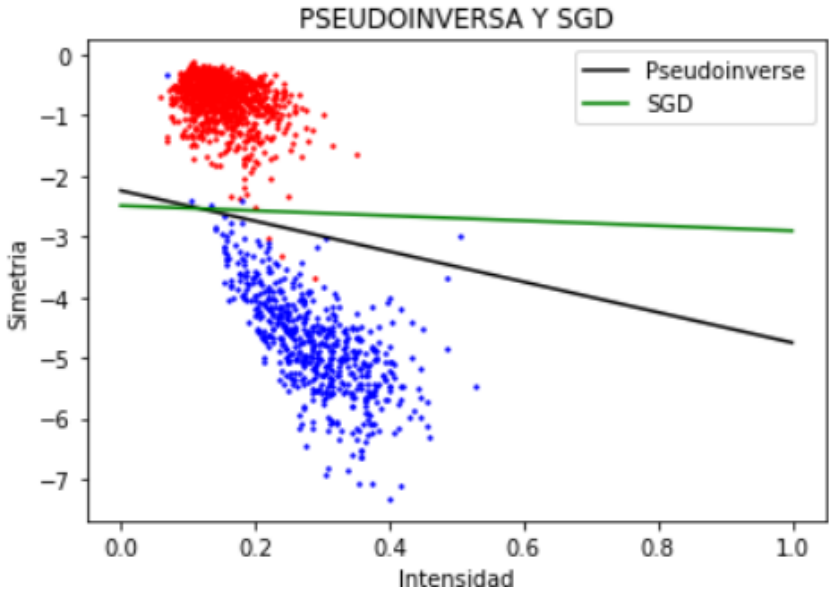
2. EJERCICIO SOBRE REGRESION LINEAL

2.1 Utilizar el algoritmo de la pseudo-inversa y el Gradiente descendente estocástico (SGD), además de las etiquetas {-1,1} para cada número. Pintar las soluciones obtenidas junto con los datos usados en el ajuste. Valorar la bondad del resultado usando Ein y Eout.

Para el uso de la función de pseudoinversa será necesario contar con unos datos de entrenamiento y de etiquetas pasados como parámetros para su posterior uso con la función np.linalg.pinv(datosEntrenamiento).

Por otra parte, para la función de gradiente descendente estocástico necesitaremos también dos conjuntos de datos, un umbral y una tasa de aprendizaje. Para el tamaño del batch he seguido las recomendaciones del profesor fijando un valor de 32, aunque cualquiera superior a este e inferior a 128 habría sido razonable.

Con esta nueva función únicamente se utiliza una muestra aleatoria del total por cada iteración, lo cual es la principal diferencia con el algoritmo de gradiente descendente anterior.



Los errores obtenidos para cada una de las funciones son los siguientes:

```
Ejercicio 2 Apartado 1


Bondad del resultado para pseudoinversa:
Ein:  0.07918658628900384
Eout:  0.13095383720052572

Bondad del resultado para grad. descendente estocastico:
Ein:  0.09283220246648519
Eout:  0.15827970239606062
```

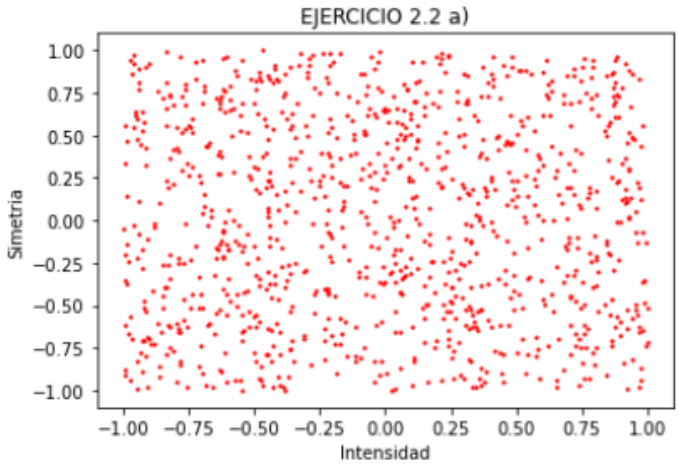
Observando los resultados podemos ver que el error es menor para la pseudoinversa. A esta conclusión se llega sabiendo que el error se calcula como el cuadrado de la diferencia entre el valor obtenido y el esperado. Por tanto, cuanto mayor sea el error, más distancia habrá desde el valor real al alanzado. Esta conclusión se puede ver reflejada en la gráfica también, pues la recta de la pseudoinversa hace una mejor división de los datos.

El valor del error con el gradiente descendente estocástico podría ser modificado si alteramos el valor de atributos como la tasa de aprendizaje, número de iteraciones máxima o umbral que cruzar. Por lo general, es recomendable el uso de este algoritmo a pesar de obtener peores errores debido a que la aleatoriedad de las muestras permite escapar de mínimos locales y es más eficiente.

2.2 En este caso aumentamos la complejidad del modelo lineal usado para observar como se transforman los errores Ein y Eout. Utilizamos la función simula_unif(N,2,size).

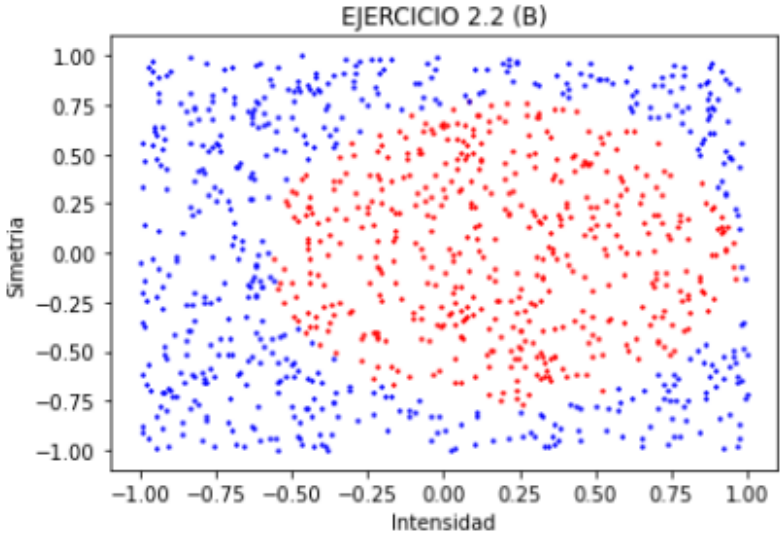
a) Pintar el mapa de punto 2D

Mapa sacado de la función simula_unif(N, d, size) con una N de 1000, una d con valor 2 y un size de 1. Esto significa obtener 1000 coordenadas 2D dentro del cuadrado [-1,1] x [-1,1].



b) Pintar el mapa de etiquetas obtenido

Corresponde al mapa anterior, pero diferenciando los valores en dos subconjuntos según su etiqueta.



c) Estimar el error de ajuste Ein usando Gradiente Descendente Estocástico.

Este es el error obtenido para la recta que separaría los datos en dos subconjuntos. Por algún error de cálculo se obtiene una cifra superior a 1, el cual es el máximo, pero nos deja intuir que el valor será realmente alto y por tanto, el resultado, poco satisfactorio.

```
Ejercicio 2 Apartado 2 c)

Ein:  1.012363233570044
```

d) Valor medio de los errores in y out

Tras 1000 ejecuciones diferentes, vemos que la media de los errores es cercana a 1 también, por lo que cojamos los datos que cojamos, vemos que este ajuste no es el mejor que podría darse.

```
Ejercicio 2 Apartado 2 d)

Ein medio:  0.9511694810904 79
Eout medio:  0.9554529813366135
```

e) Valore qué tan bueno considera que es el ajuste con este modelo lineal a la vista de los valores medios obtenidos en Ein y Eout

El error cuadrático es una medida de ajuste que informa sobre cómo de cerca están los valores obtenidos de los que se predicen. A pesar de estar al cuadrado para evitar trabajar con valores negativos, el error no deja de ser una distancia, y cuanto mayor sea, peor será lo que representa. Sabiendo que los valores reales están entre -1

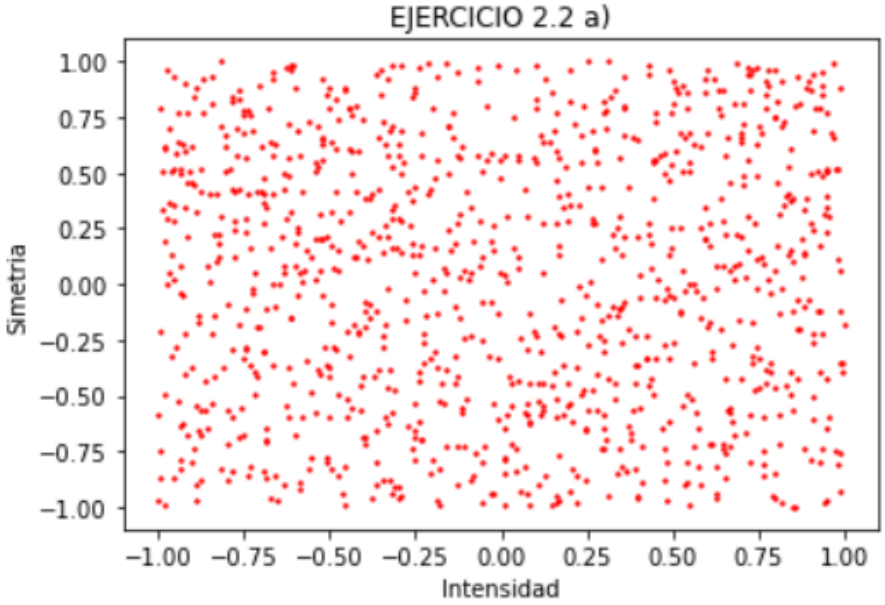
y 1 en ambos ejes, que la distancia media de los alcanzados sea de casi 1 indica que los puntos están bastante más alejados de lo que se pretendía.

Además, a partir de la gráfica vemos la dificultad de trazar una línea que delimite un conjunto de datos del otro, pues uno de los dos está totalmente rodeado por el opuesto. De esta forma es realmente difícil definir una función lineal con un error cercano a 0 y se puede calificar el modelo lineal como poco acertado.

2.2 REPETIR EL EXPERIMENTO PARA UN MODELO NO LINEAL

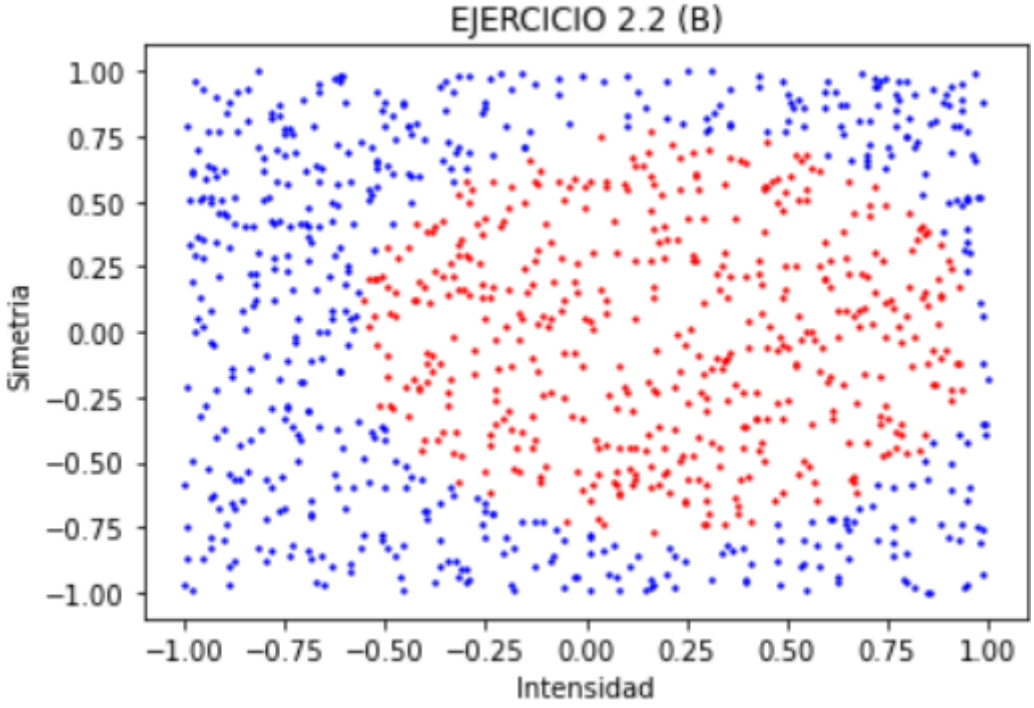
a) Pintar el mapa de punto 2D

Aquí podemos ver el nuevo mapa para la repetición del experimento.



b) Pintar el mapa de etiquetas obtenido

El mapa anterior diferenciando por etiquetas.



c) Estimar el error de ajuste Ein usando Gradiente Descendente Estocástico.

De nuevo, el valor puede ser elevado, pero bastante menor que el obtenido con un modelo lineal.

```
Ejercicio 2 Apartado 2 c)

Ein:  0.8301294414879897
```

d) Valor medio de los errores in y out

En este caso, tras 1000 ejecuciones también obtenemos una media, lo cual significa que este modelo no lineal es el más correcto.

```
Ejercicio 2 Apartado 2 d)

Ein medio:  0.8105278406069459
Eout medio:  0.8136150049021186
```

e) Valore qué tan bueno considera que es el ajuste con este modelo lineal a la vista de los valores medios obtenidos en Ein y Eout

El caso en este experimento es bastante similar al anterior. A pesar de ser una gráfica nueva y diferente a la anterior, el conjunto de datos representado con el color rojo se encuentra totalmente rodeado por el azul, haciendo que el modelo lineal no sea el más adecuado para este caso. Esto repercute en el error haciendo que tenga unos valores más elevados de los que se deberían considerar y en que sea altamente probable que al añadir un nuevo elemento, este sea asignado con etiquetas que no le corresponden.

2.2.3 A la vista de los resultados de los errores promedios obtenidos en los dos experimentos. ¿Qué modelo considera que es el más adecuado? Justifique la decisión

Aunque ambos modelos obtienen errores elevados y, por tanto, ambos son poco deseables, tras la experimentación se puede observar que el error medio tanto en E_{in} como en E_{out}, es menor en el segundo experimento en el que se usa un vector de características con 6 elementos en lugar de 3.

Esto es debido a la ineficiencia de las funciones lineales para resolver este problema tan complejo que hace que el uso de características no lineales aporte resultados mejores. El modelo no lineal usado es cuadrático y es por lo que estima mejor que el otro.

BONUS.

Considerar la función $f(x, y) = (x+2)^2 + 2(y-2)^2 + 2\sin(2\pi x)\sin(2\pi y)$. Usar el método de Newton para minimizar la función y generar un gráfico de cómo desciende el valor con las iteraciones con tasa de aprendizaje 0.1 y 0.01 y partiendo del punto (-1,1). Comentar las diferencias y dependencias de eta (tasa de aprendizaje).

Para el método de Newton es necesario contar también con la segunda derivada con respecto a x, la segunda derivada con respecto a y, la derivada sobre y de la derivada sobre x y la derivada sobre x de la derivada sobre y. Para verlo más claro:

$$f(x, y) = (x+2)^2 + 2(y-2)^2 + 2\sin(2\pi x)\sin(2\pi y)$$

$$f'_x(x, y) = 2(x+2) + 4\pi\cos(2\pi x)\sin(2\pi y)$$

$$f'_y(x, y) = 4(y-2) + 4\pi\sin(2\pi x)\cos(2\pi y)$$

$$f''_x(x, y) = 2 - 8\pi^2\sin(2\pi x)\sin(2\pi y)$$

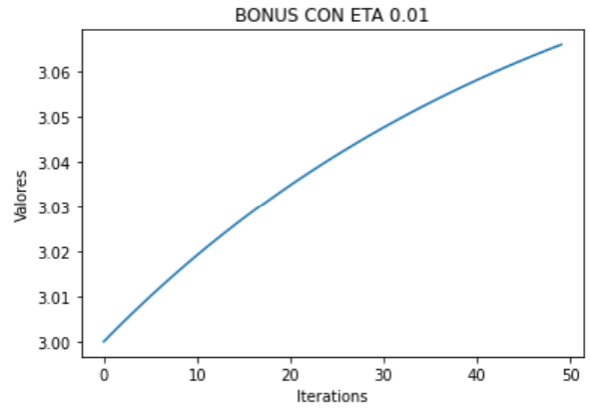
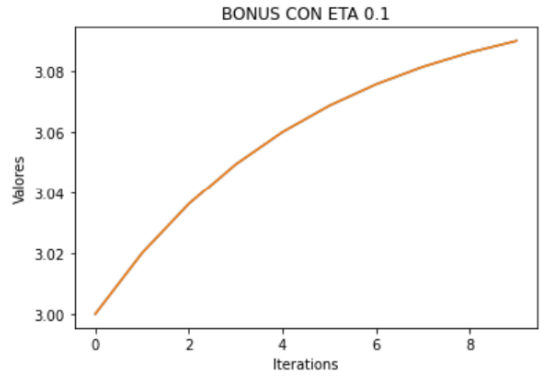
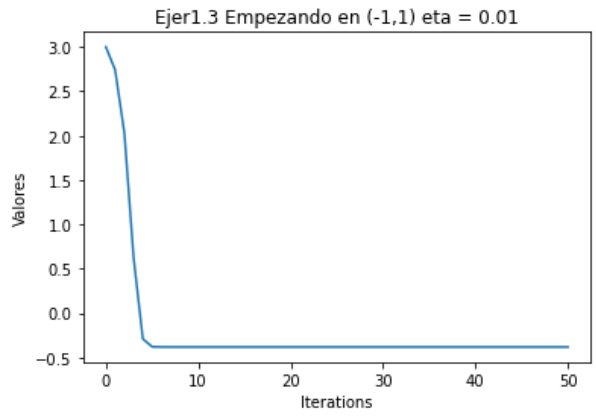
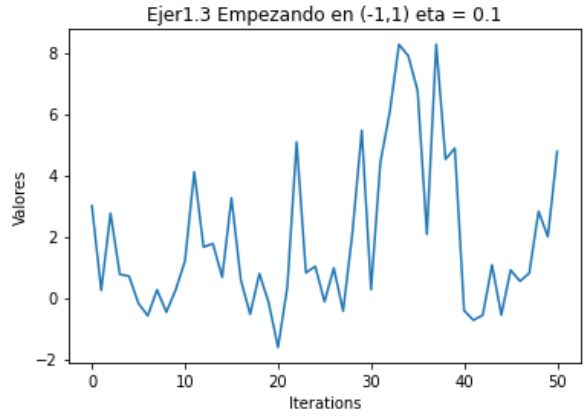
$$f''_y(x, y) = 4 - 8\pi^2\sin(2\pi x)\sin(2\pi y)$$

$$f''_{xy}(x, y) = 8\pi^2\cos(2\pi x)\cos(2\pi y)$$

$$f''_{yx}(x, y) = 8\pi^2\cos(2\pi x)\cos(2\pi y)$$

Experimento del 1.3.a realizado con el método de Newton.

NEWTON	SGD
Punto de partida: (-1,1) Valor de eta = 0.01 Iteraciones realizadas: 50 Coordenadas del valor minimo (ultimas): (-0.98 , 0.99) Valor mínimo (ultimo valor): (3.07) Punto de partida: (-1,1) Valor de eta = 0.01 Iteraciones realizadas: 50 Coordenadas del valor minimo (ultimas): (-0.97 , 0.98) Valor mínimo (ultimo valor): (3.09)	a) Punto de partida: (-1,1) Valor de eta = 0.01 Iteraciones realizadas: 50 Coordenadas del valor minimo (ultimas): (-1.27 , 1.29) Valor mínimo (ultimo valor): (-0.38) Punto de partida: (-1,1) Valor de eta = 0.1 Iteraciones realizadas: 50 Coordenadas del valor minimo (ultimas): (-2.96 , 0.57) Valor mínimo (ultimo valor): (4.77)

NEWTON	SGD
 	 

Experimento del 1.3.a realizado con el método de Newton.

SGD ETA = 0.01		
PUNTO DE PARTIDA	**COORDENADAS DEL MINIMO**	**MINIMO**
(-0.5,-0.5)	(-0.79 , -0.13)	9.13
(1.0, 1.0)	(0.68 , 1.29)	6.44
(2.1,-2.1)	(0.15 , -0.10)	12.49
(-3.0, 3.0)	(-2.73 , 2.71)	-0.38
(-2.0, 2.0)	(-2.00 , 2.00)	0.00

SGD ETA = 0.1		
PUNTO DE PARTIDA	**COORDENADAS DEL MINIMO**	**MINIMO**
(-0.5,-0.5)	(-3.03 , 1.67)	1.6
(1.0, 1.0)	(-1.78 , 2.06)	0.82
(2.1,-2.1)	(-2.31 , 1.87)	1.47
(-3.0, 3.0)	(-0.27 , 1.42)	2.68
(-2.0, 2.0)	(-1.80 , 2.36)	1.78

NEWTON ETA = 0.01		
PUNTO DE PARTIDA	**COORDENADAS DEL MINIMO**	**MINIMO**
(-0.5,-0.5)	(-0.45 , -0.52)	15.00
(1.0, 1.0)	(1.02 , 0.97)	11.20
(2.1,-2.1)	(2.17 , -2.07)	49.81
(-3.0, 3.0)	(-3.02 , 3.01)	3.07
(-2.0, 2.0)	(-2.00 , 2.00)	0.00

NEWTON ETA = 0.1		
PUNTO DE PARTIDA	**COORDENADAS DEL MINIMO**	**MINIMO**
(-0.5,-0.5)	(-0.33 , -0.57)	15.25
(1.0, 1.0)	(1.07 , 0.91)	11.34
(2.1,-2.1)	(0.77 , 6.89)	56.70
(-3.0, 3.0)	(-3.05 , 3.03)	3.11
(-2.0, 2.0)	(-2.00 , 2.00)	0.00