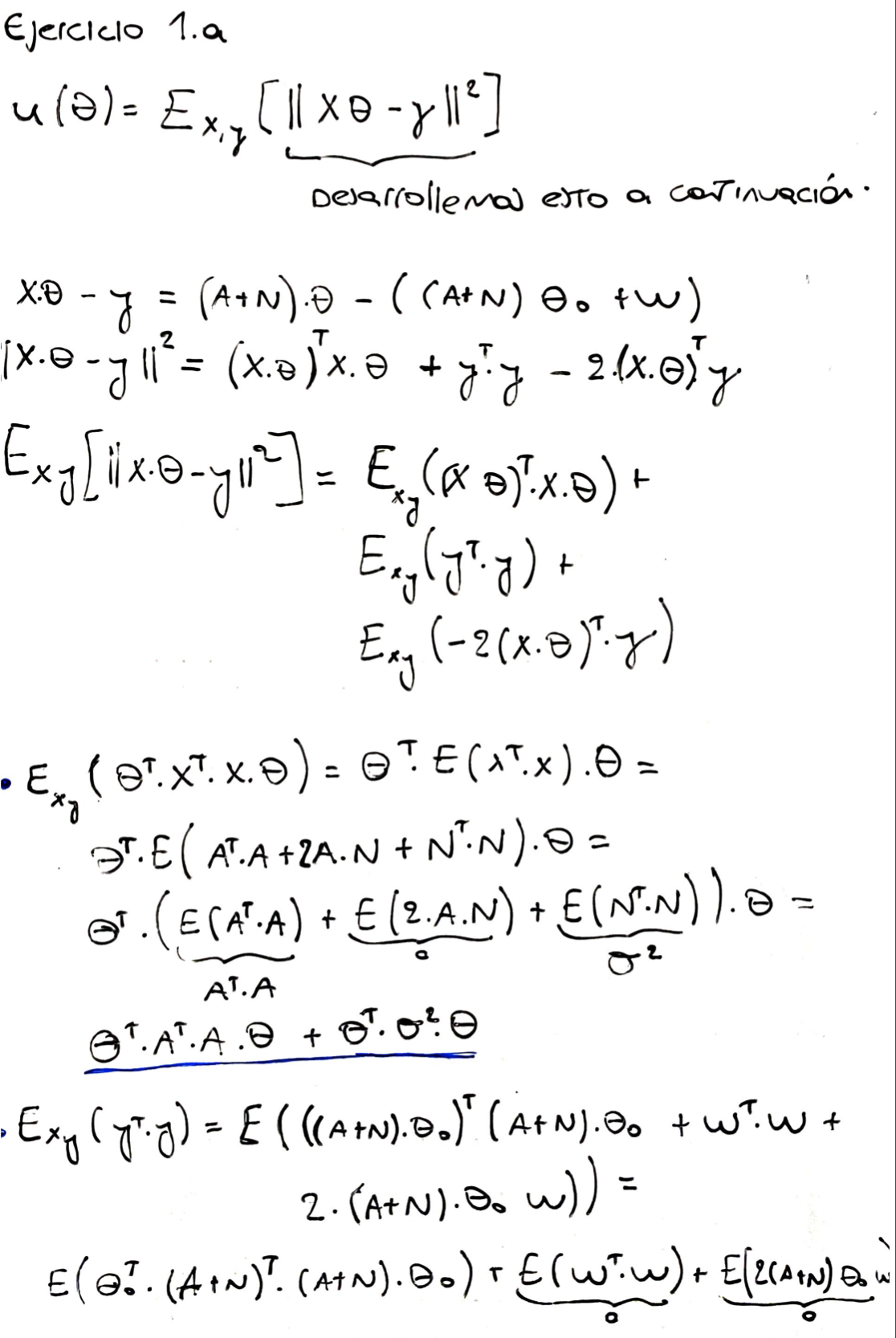
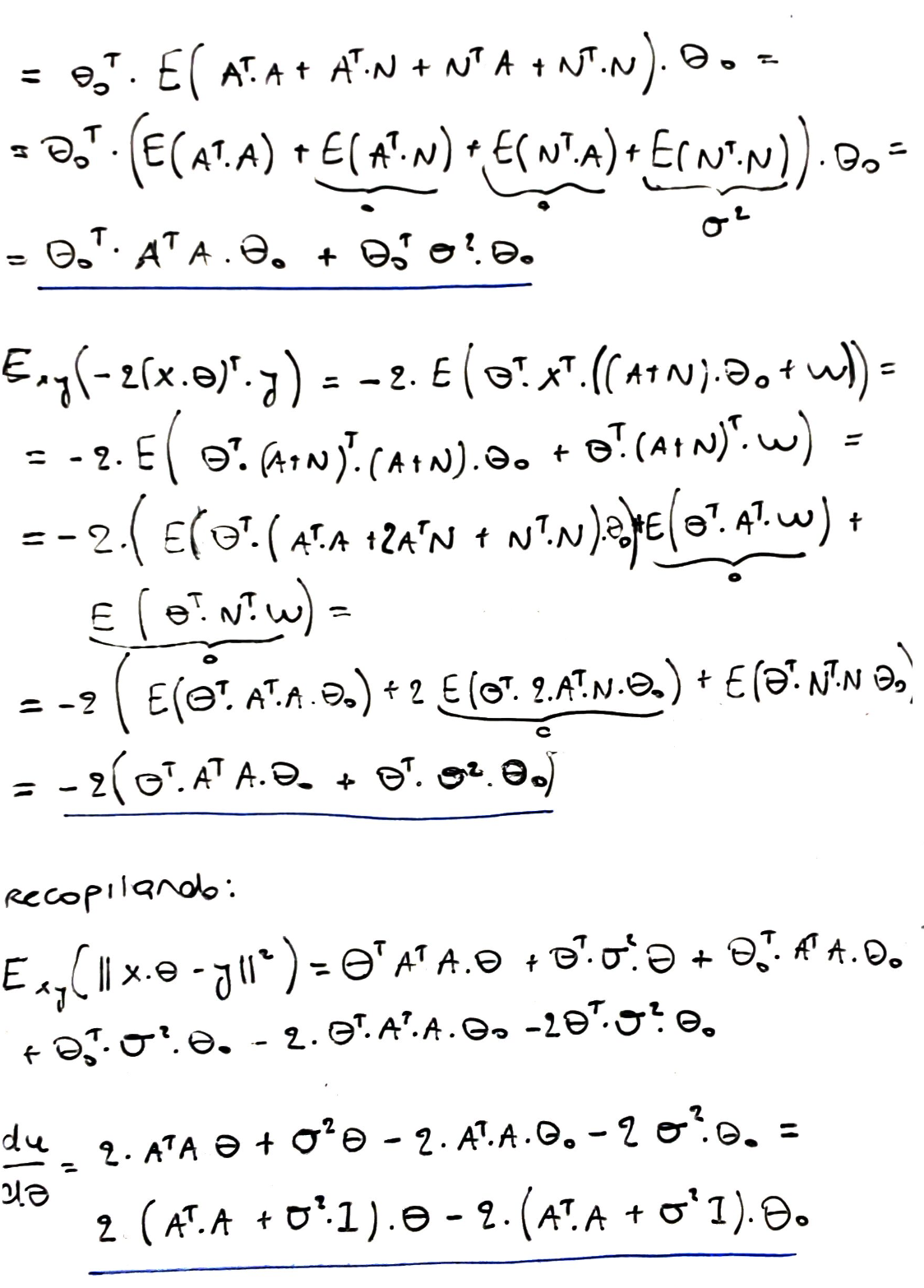
**Obligatorio 5: SGD**

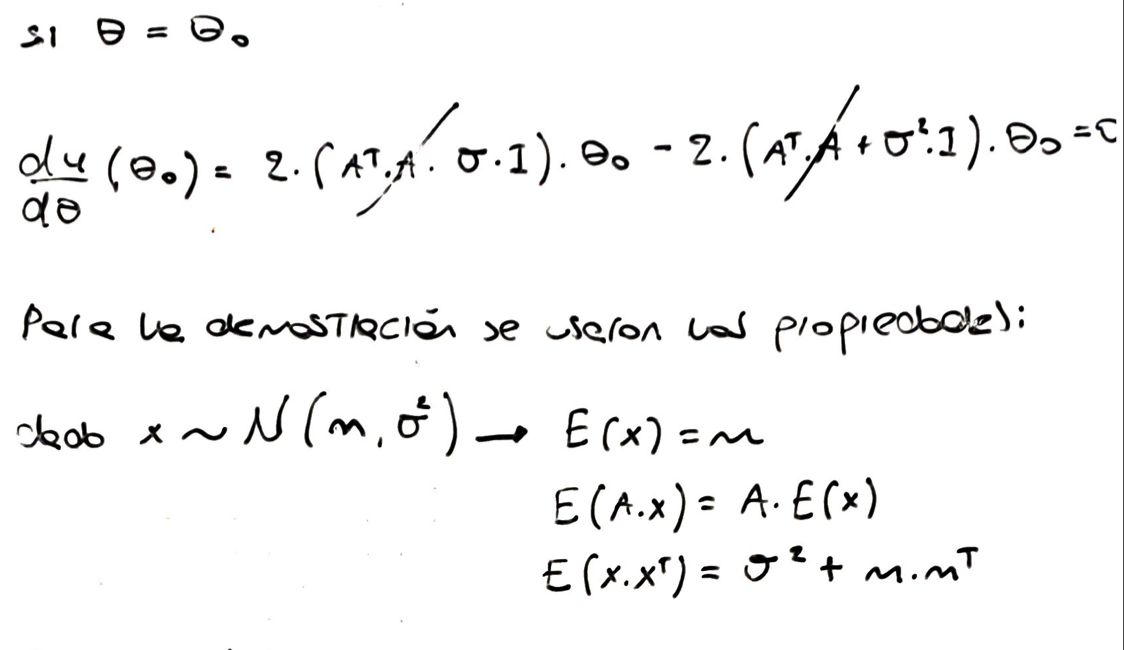
**Lucía Bouza - 4.289.797-0**

**Ejercicio 1**

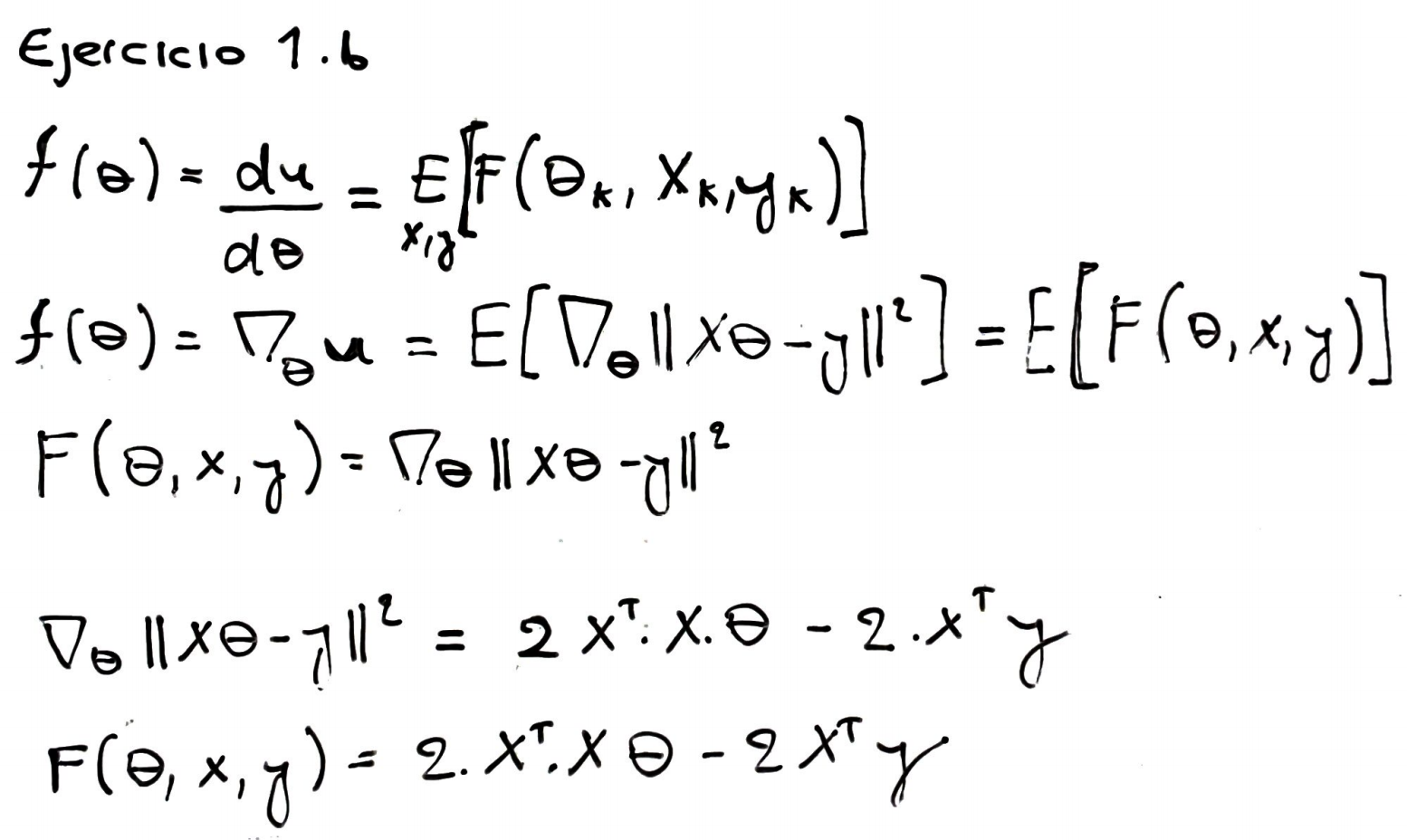
**Parte A**

****

****

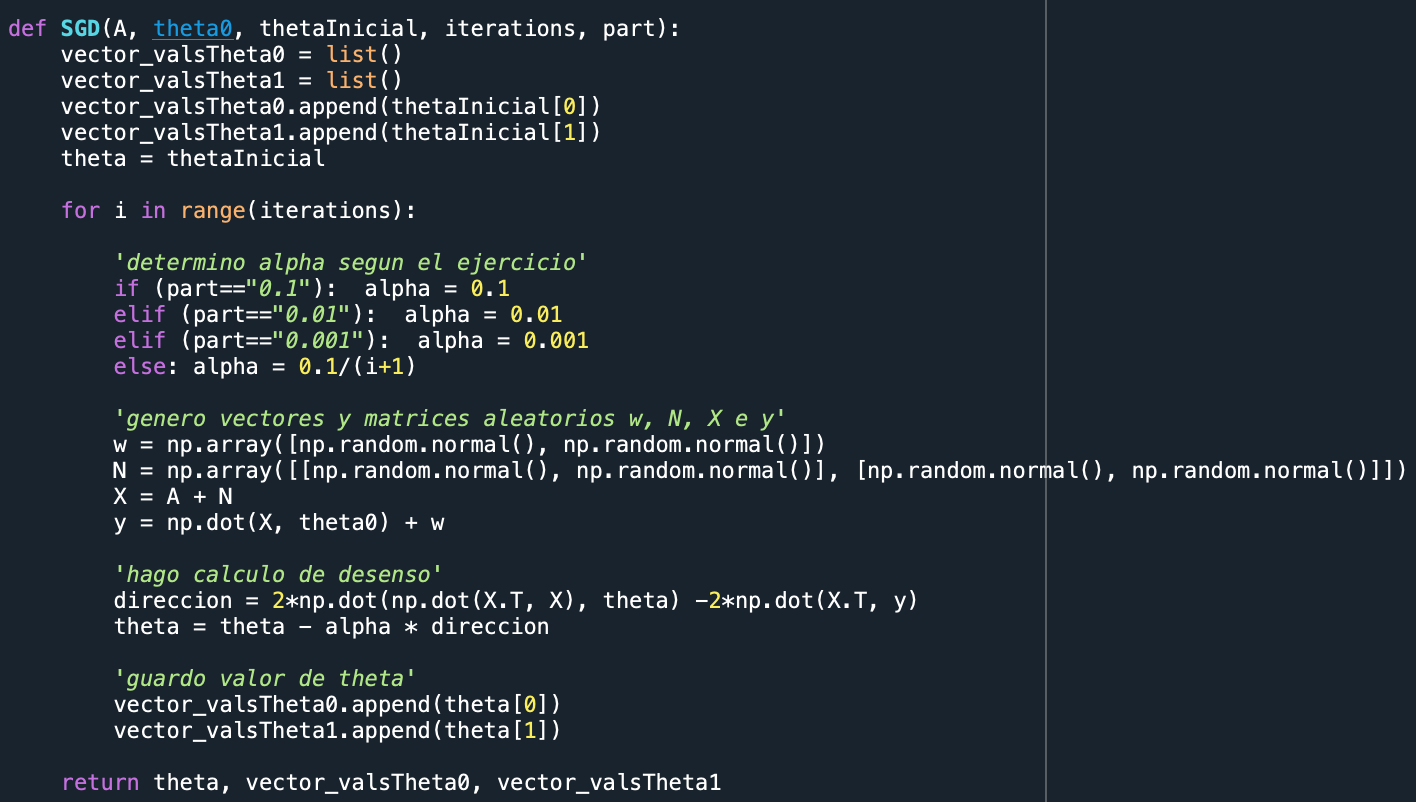
****

**Parte B**

****

**Parte C**

Para este ejercicio se realizó la implementación del algoritmo SGD, la cual puede verse en el archivo Ej1.py. A continuación se mostrará el código del algoritmo

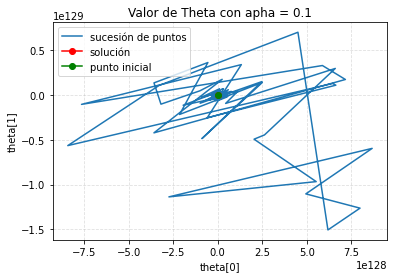


Los valores de N y w fueron creados con distribución normal, independientes entre si.

**Alpha = 0.1**

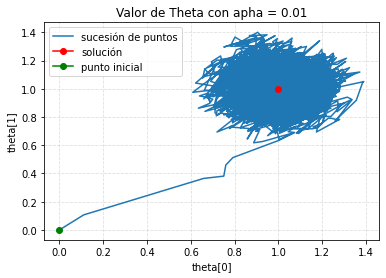
el resultado con alpha = 0.1 es: [1.89027940e+121 1.24073824e+121]

Para todas las pruebas realizadas se constató que la mayoría de las veces la sucesión no converge al valor deseado. El paso parece ser demasiado grande.

****

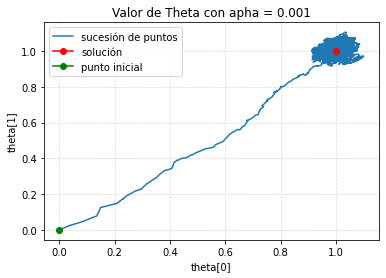
**Alpha = 0.01**

El resultado con alpha = 0.01 es: [0.93652755 1.08602043]. Observamos que tiende a la solución Theta0=[1,1]

****

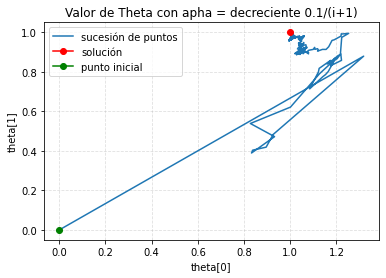
**Alpha = 0.001**

el resultado con alpha = 0.001 es: [1.07473511 0.98233012]. Observamos que converge a los valores de Theta0=[1,1], de forma muy similar que para alpha= 0.01, pero convergiendo a una bola más pequeña.



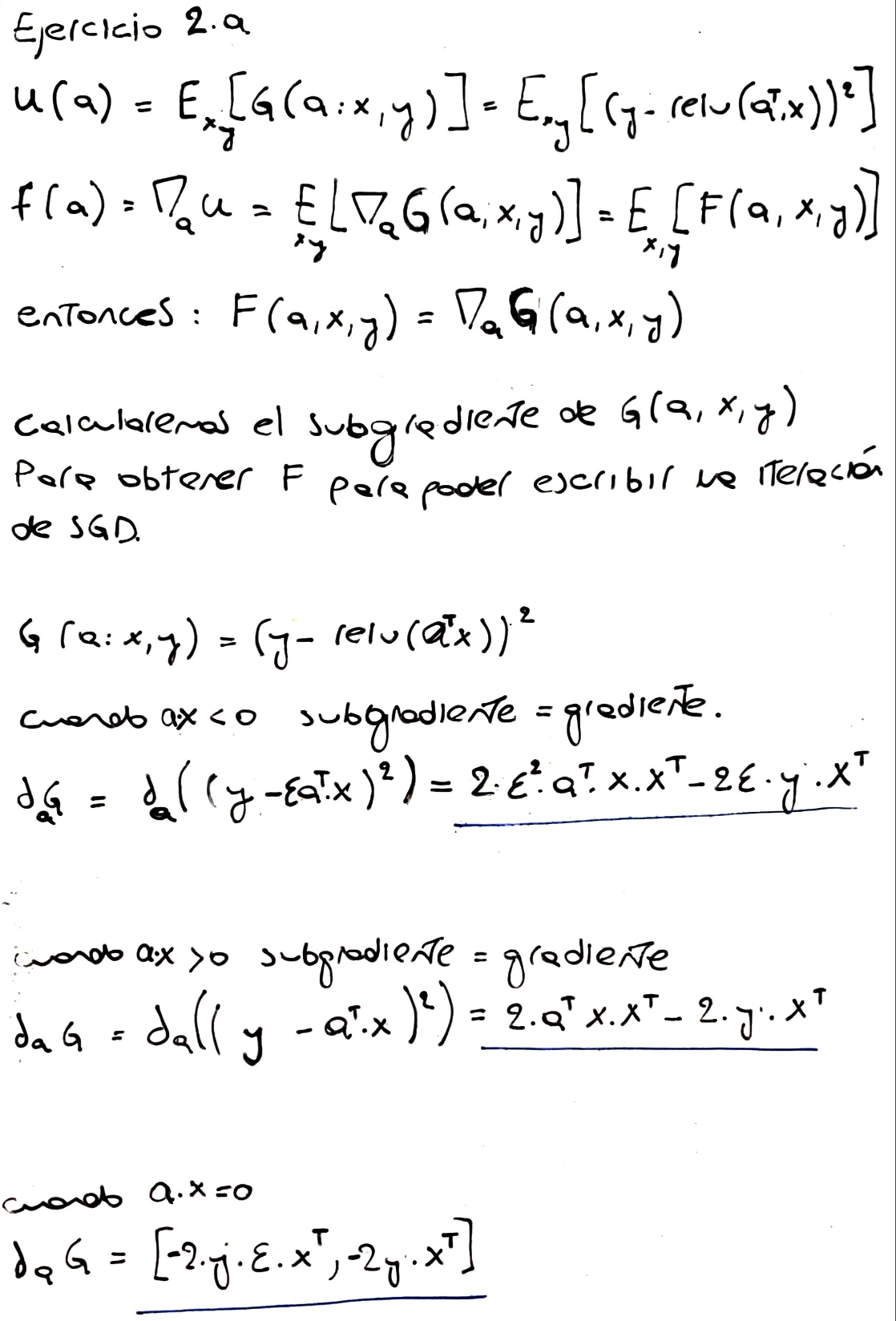
**Alpha decreciente: 0.1/(iteración)**

el resultado con alpha = decreciente 0.1/(i+1) es: [1.00098318 0.99888913]. Aquí podemos observar que el resultado tiende a acercarse a la solución, de una forma más lenta pero certera a medida que en alpha desciende.



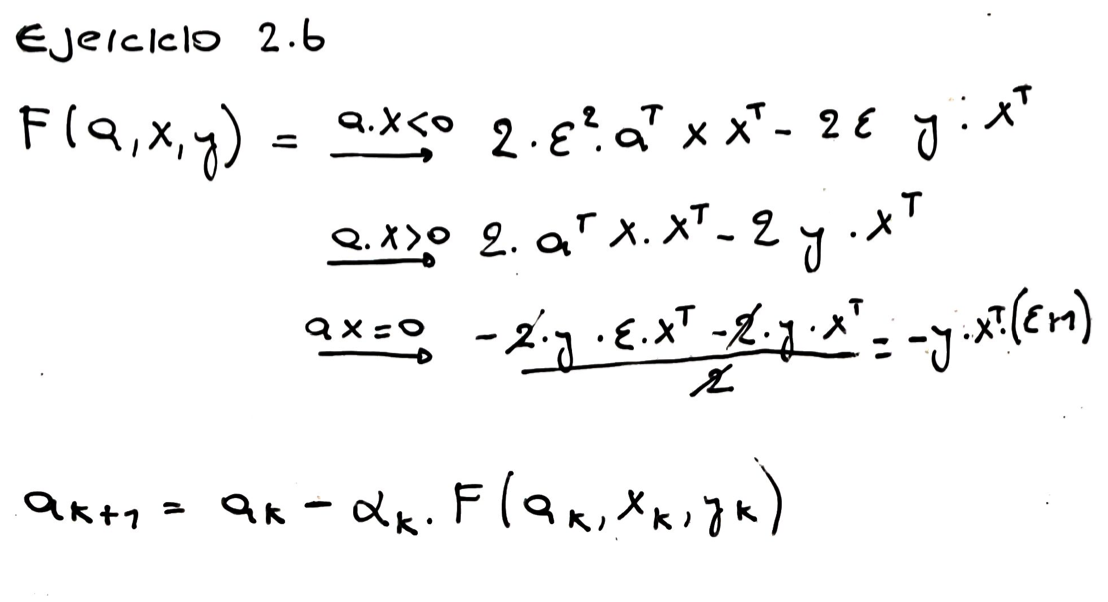
**Ejercicio 2**

**Parte A**

****

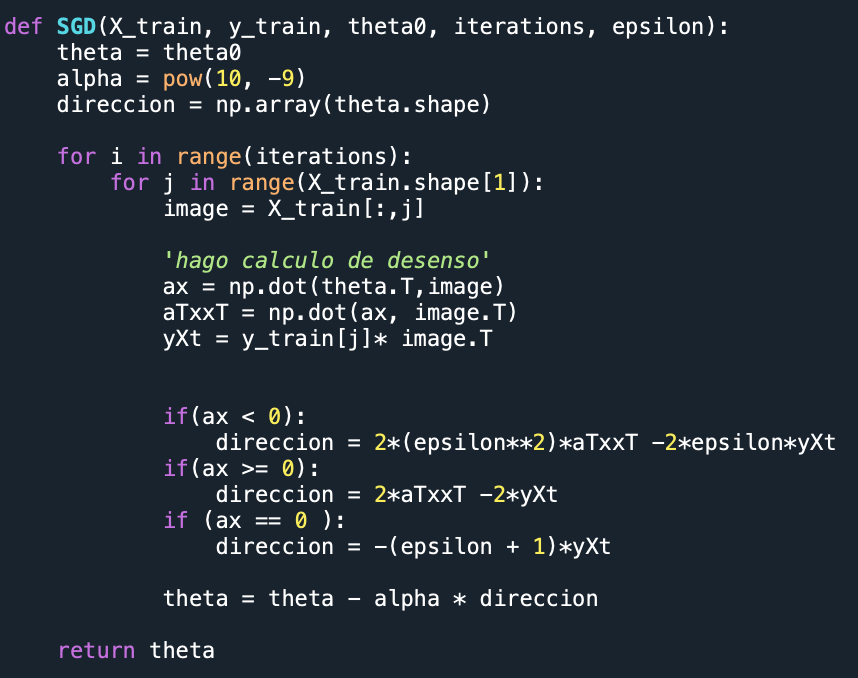
**Parte B**

Para el caso en que la función no es diferenciable se definió utilizar la dirección media de todas las posibles.



**Parte C**

Para este ejercicio se realizó la implementación del algoritmo SGD, la cual puede verse en el archivo Ej2.py. A continuación se mostrará el código del algoritmo.

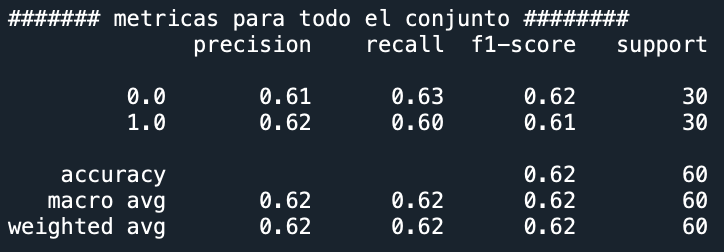


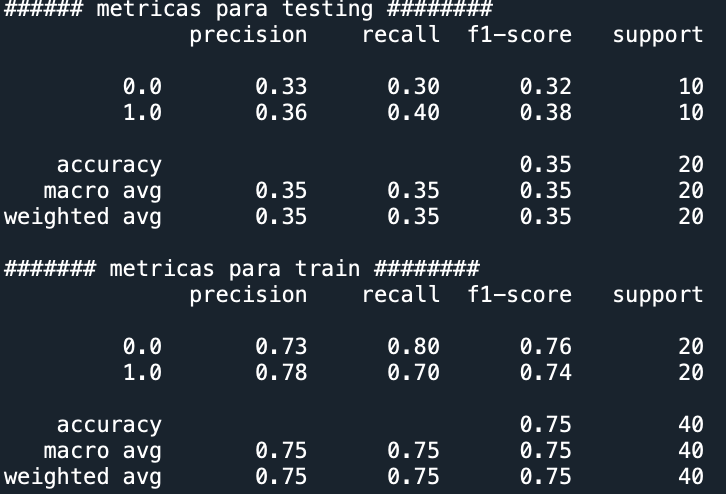
El procedimiento para aplicar el método de clasificación consistió en cargar los datos, separarlos en conjunto de entrenamiento y test, entrenar el clasificador SGD con el conjunto de entrenamiento y luego evaluar el modelo obtenido en el conjunto de test. En el entrenamiento, dado que eran solo 40 muestras, se definió iterar en ellas, 69 veces.

En la evaluación también se visualizó el resultado del clasificador sobre el conjunto de muestras sobre el que fue entrenado.

Para la evaluación se utilizaron los reportes de sklearn, donde se obtienen métricas de accurancy, precision, recall y F para los conjuntos. También, para poder visualizar mejor, se realizarán las gráficas solicitadas en la parte D.

Resultados obtenidos



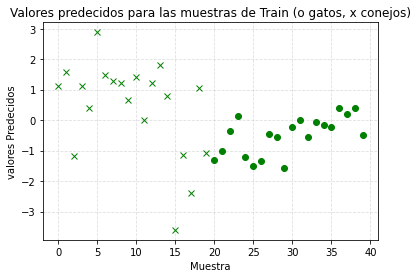


Se puede observar que en el conjunto de entrenamiento los resultados son mejores que en el de test. Esto es habitual, ya que el conjunto de entrenamiento es muy pequeño y sufre sobreajuste.

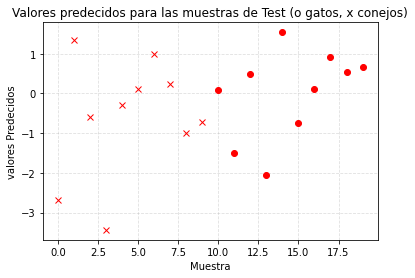
**Parte D**

Se pide graficar z = a.T \* x para los conjuntos de entrenamiento y test.

Se representan a continuación la grafica de los valores z en función del número de muestra. Se espera que las imágenes de conejos (cuyo valor de clasificación es 1) tengan valores de z mayores a 0, mientras que las imágenes de gatos (cuyo valor de clasificación es 0) tengan valores de z menores a 0.



Podemos ver que en el conjunto de test se cumple para la mayoría de los puntos (los valores de métricas se pueden encontrar en la parte C).



Para el conjunto de test los valores no son tan buenos. Para los conejos solamente clasifica correctamente 4 de 10, mientras que para los gatos clasifica correctamente 3 de 10.