Trabajo practico Nro. 1

INTELIGENCIA ARTIFICIAL

Gonzalo Bengoechea | 38254089 | 14-10-2023

A continuación, se adjuntan las imágenes del desarrollo de los distintos puntos solicitados en el trabajo practico, los cuales pueden ser buscados de forma más específica en el siguiente índice:

Índice

1.

A. La principal distinción al ejecutar el código y llevar a cabo el entrenamiento del modelo reside en que, al emplear el conjunto de datos A, la convergencia se alcanza de manera rápida. No obstante, al entrenar el modelo con el conjunto B, el proceso de convergencia se prolonga significativamente.

B. Para tratar de entender esta discrepancia, procedimos a representar visualmente ambos conjuntos de datos mediante gráficos de dispersión. Al analizar los gráficos, se destaca principalmente que, en el caso del conjunto B, es factible llevar a cabo una clasificación prácticamente perfecta de los datos utilizando un límite de decisión lineal adecuado. A continuación, se presentan los gráficos de dispersión:

Imagen del dataset A:

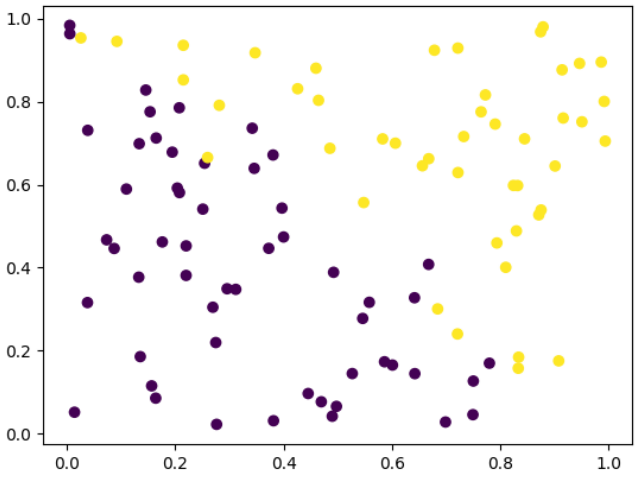
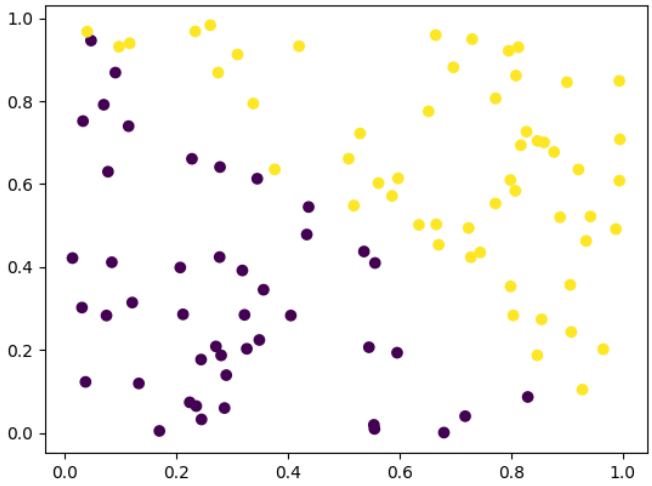


Imagen del dataset B:



Para abordar esta cuestión, podemos respaldarnos en la lógica presentada en el inciso c, donde se realizaron diversas modificaciones y se constató que la única que produjo

mejoras en la convergencia fue la inclusión de un término de regularización en la función de costo. Con esta información como base, podemos plantear la hipótesis de que el desafío en la convergencia del conjunto B puede estar relacionado con la carencia de regularización. Debido a esta posible falta, es plausible que el conjunto B sea más complejo y más susceptible al sobreajuste en comparación con el conjunto A.

C. I. El learning rate es un parámetro crítico en algoritmos de aprendizaje automático, como en modelos de regresión logística. Juega un papel fundamental en la velocidad y estabilidad del proceso de entrenamiento de estos modelos.

El valor del learning rate determina la rapidez con la que el algoritmo de optimización convergerá hacia los valores óptimos de los parámetros del modelo. Si el learning rate es muy pequeño, el modelo puede converger lentamente y requerir más iteraciones para alcanzar una solución óptima. Por otro lado, si el learning rate es muy grande, el algoritmo podría no converger o incluso divergir, lo que significa que los parámetros nunca alcanzarán una solución estable.

Además, el learning rate adecuado es esencial para la estabilidad del entrenamiento. Un learning rate demasiado alto puede hacer que el modelo oscile alrededor del mínimo óptimo o incluso diverja, lo que se conoce como "exploding gradients". Por otro lado, un learning rate muy pequeño puede hacer que el modelo se quede atascado en mínimos locales o tome mucho tiempo en converger.

En este contexto, se realizaron pruebas con varios valores de learning rate, tanto altos como bajos, pero no se logró mejorar el rendimiento del entrenamiento del algoritmo. Por lo tanto, se ha llegado a la conclusión de que la solución a este problema no radica en la modificación del learning rate.

II. Se intentó abordar el problema disminuyendo el learning rate en cada iteración mediante la fórmula learning_rate = 1 / i^2, lo que permitió que ambos conjuntos de datos convergieran. Sin embargo, el proceso de convergencia requirió un número sustancialmente alto de iteraciones, oscilando entre 9 y 14 millones de iteraciones. A pesar de este esfuerzo, los resultados obtenidos no parecen generar predicciones precisas.

III. El escalado de datos es un paso crucial en el preprocesamiento de datos al trabajar con modelos de regresión logística y otros algoritmos de aprendizaje automático. Ayuda a mejorar la velocidad y estabilidad de algoritmos de optimización como el descenso de gradiente.

El escalado de datos es importante porque, si las características no están en la misma escala, algunas pueden dominar sobre otras en términos de contribución a la función de costo, lo que ralentiza la optimización. Además, el escalado ayuda a igualar la importancia relativa de las diferentes características, ya que, si algunas tienen rangos de valores mucho mayores que otras, el modelo podría darles un peso desproporcionado en la predicción.

Sin embargo, es relevante notar que, a pesar de la importancia del escalado, en su caso, realizar el escalado de las características (X) antes del entrenamiento no parece haber tenido ningún efecto en la convergencia de los algoritmos. El dataset B aún no logra converger. Esto sugiere que el problema de convergencia en el conjunto B no se

debe a la falta de escalado de las características. Es posible que el problema esté relacionado con la naturaleza de los datos o con otros factores que requieran una estrategia diferente para abordarlo.

IV. La aplicación de un término de regularización al entrenamiento de un modelo de regresión logística puede tener un impacto significativo en la capacidad del modelo para generalizar y evitar el sobreajuste. La regresión logística es propensa al sobreajuste cuando se entrena con conjuntos de datos ruidosos o cuando se ajusta de manera demasiado ajustada a los datos de entrenamiento. Aquí se explican cómo afecta la regularización al entrenamiento de un modelo de regresión logística:

La regularización, ya sea L1 (Lasso) o L2 (Ridge), introduce un término adicional en la función de costo que penaliza los valores grandes de los coeficientes del modelo. Esto evita que los coeficientes se vuelvan muy grandes en magnitud, lo que puede llevar al sobreajuste. Un coeficiente grande significa que el modelo está dando demasiada importancia a una característica específica y, por lo tanto, está más sujeto a adaptarse a ruido en los datos de entrenamiento. La regularización ayuda a suavizar los coeficientes y, por lo tanto, a evitar el sobreajuste.

Agregando un término de regularización a la función de costo, el gradiente nos queda con un término de un valor lambda sumando únicamente. Haciendo esto, y probando con varios valores de lambda, se logró que el dataset B llegue a converger (utilizando lambda =0.0225), sin modificar el comportamiento del entrenamiento del dataset A.

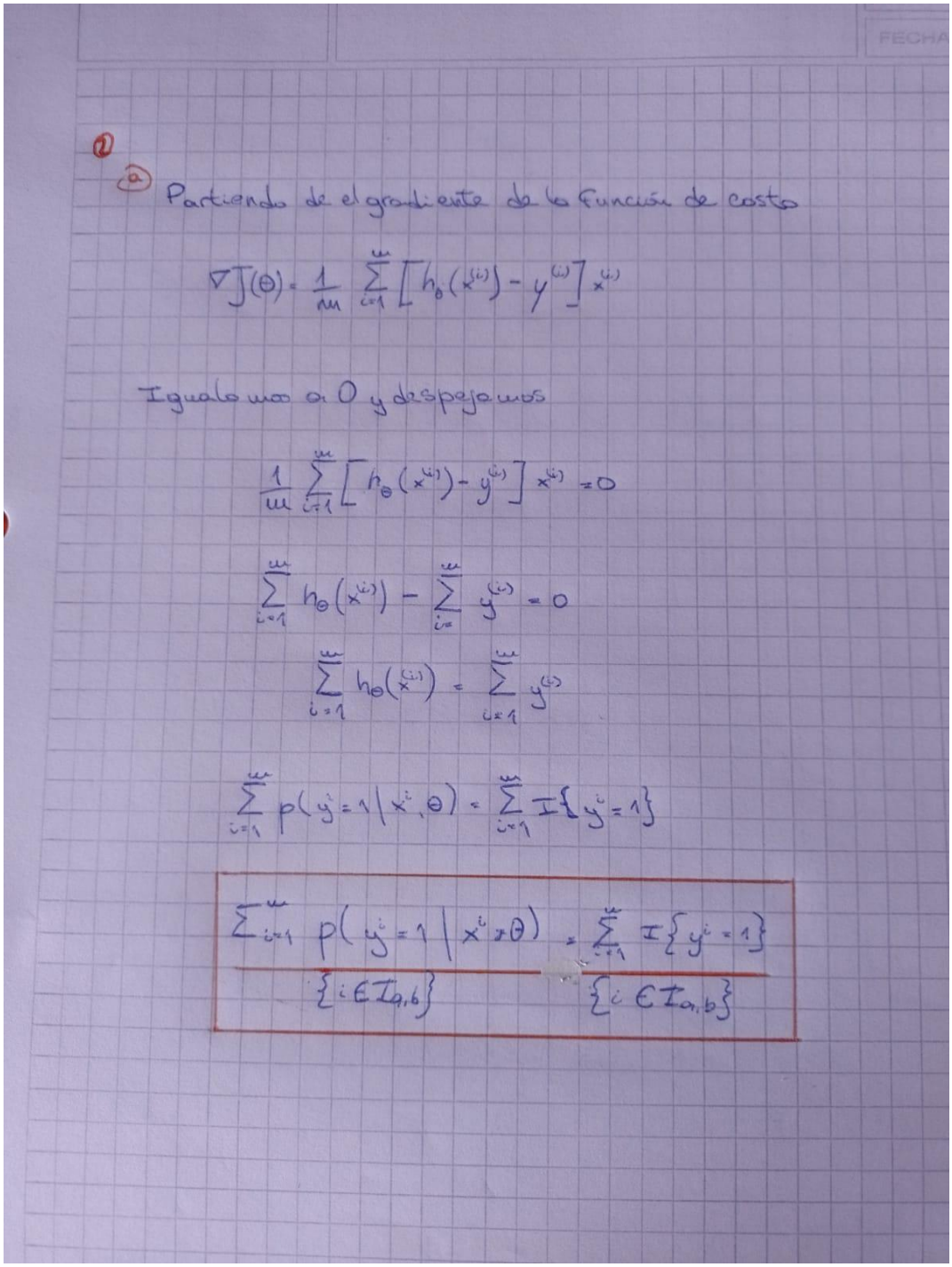
V. Se realizaron pruebas añadiendo ruido gaussiano a los datos, tanto a las características (features) como a las etiquetas (labels). Cabe mencionar que cada iteración se considera como un caso independiente debido a que los valores generados son pseudoaleatorios, lo que significa que los resultados pueden variar en cada ejecución.

En el caso de la adición de ruido a las características, los modelos mostraron un rendimiento notable durante el entrenamiento, logrando converger en menos de 10 mil iteraciones en la mayoría de los intentos. Además, se observó que se generó un límite de decisión adecuado para los datos.

Por otro lado, al agregar ruido a las etiquetas, no se experimentó ninguna mejora en el proceso de entrenamiento. El conjunto de datos B continuó sin alcanzar la convergencia. Esto sugiere que la adición de ruido a las etiquetas no ha tenido un impacto positivo en la capacidad de entrenamiento de los modelos y que el problema de convergencia persiste en el caso del conjunto B.

D. En realidad, las SVM (Support Vector Machines) son menos susceptibles a problemas de convergencia, ya que muestran una menor sensibilidad a la elección de hiperparámetros y a la calidad de los datos, en comparación con la regresión logística. Esto se debe en parte al uso de la función de pérdida de Hinge por parte de las SVM, que está diseñada para maximizar el margen entre las clases. En consecuencia, las SVM se centran en los datos que se encuentran cerca del borde de decisión y que están clasificados incorrectamente.

2. A.



B. Tener un modelo perfectamente calibrado no implica necesariamente alcanzar una precisión perfecta, ya que incluso con una calibración ideal, el modelo podría cometer

errores de predicción debido a un umbral de decisión incorrecto. Esto podría llevar a que algunos casos se clasifiquen incorrectamente como falsos negativos o falsos positivos. Por lo tanto, aunque el promedio de clasificaciones sea impecable, el modelo podría estar cometiendo errores en cuanto a cuándo clasifica los ejemplos como positivos o negativos.

Del mismo modo, tener una precisión perfecta no garantiza una calibración perfecta. La precisión se centra en la exactitud de las predicciones de clase, en términos de verdaderos positivos y verdaderos negativos, lo que no necesariamente garantiza que las probabilidades estimadas sean exactas en términos de reflejar la probabilidad real.

Un modelo puede lograr una alta precisión simplemente ajustando un umbral de probabilidad muy alto para predecir la clase positiva, lo que podría resultar en una precisión alta, pero las probabilidades estimadas podrían no estar bien calibradas, especialmente en el rango de probabilidades más bajas.

En resumen, la calibración y la precisión son dos aspectos distintos en la evaluación de un modelo de clasificación binaria. Un modelo puede estar perfectamente calibrado sin lograr una precisión perfecta, y viceversa. La calibración se refiere a la precisión de las probabilidades estimadas, mientras que la precisión se enfoca en la exactitud de la clasificación de ejemplos en términos de verdaderos positivos y verdaderos negativos.

C. La regularización L2, también conocida como Ridge, es una técnica comúnmente empleada en la regresión logística y otros modelos de aprendizaje automático para prevenir el sobreajuste y mejorar el rendimiento. Su influencia en la calibración del modelo puede ser tanto positiva como negativa, dependiendo de diversos factores, en especial del valor del hiperparámetro de regularización Lambda.

Efectos positivos en la calibración:

Reducción del sobreajuste: La regularización L2 contribuye a disminuir el riesgo de sobreajuste del modelo. El sobreajuste puede resultar en problemas de calibración en datos de prueba, y la regularización ayuda al modelo a generalizar de manera más efectiva, mejorando, por lo tanto, la calibración en datos no observados.
Efectos negativos en la calibración:

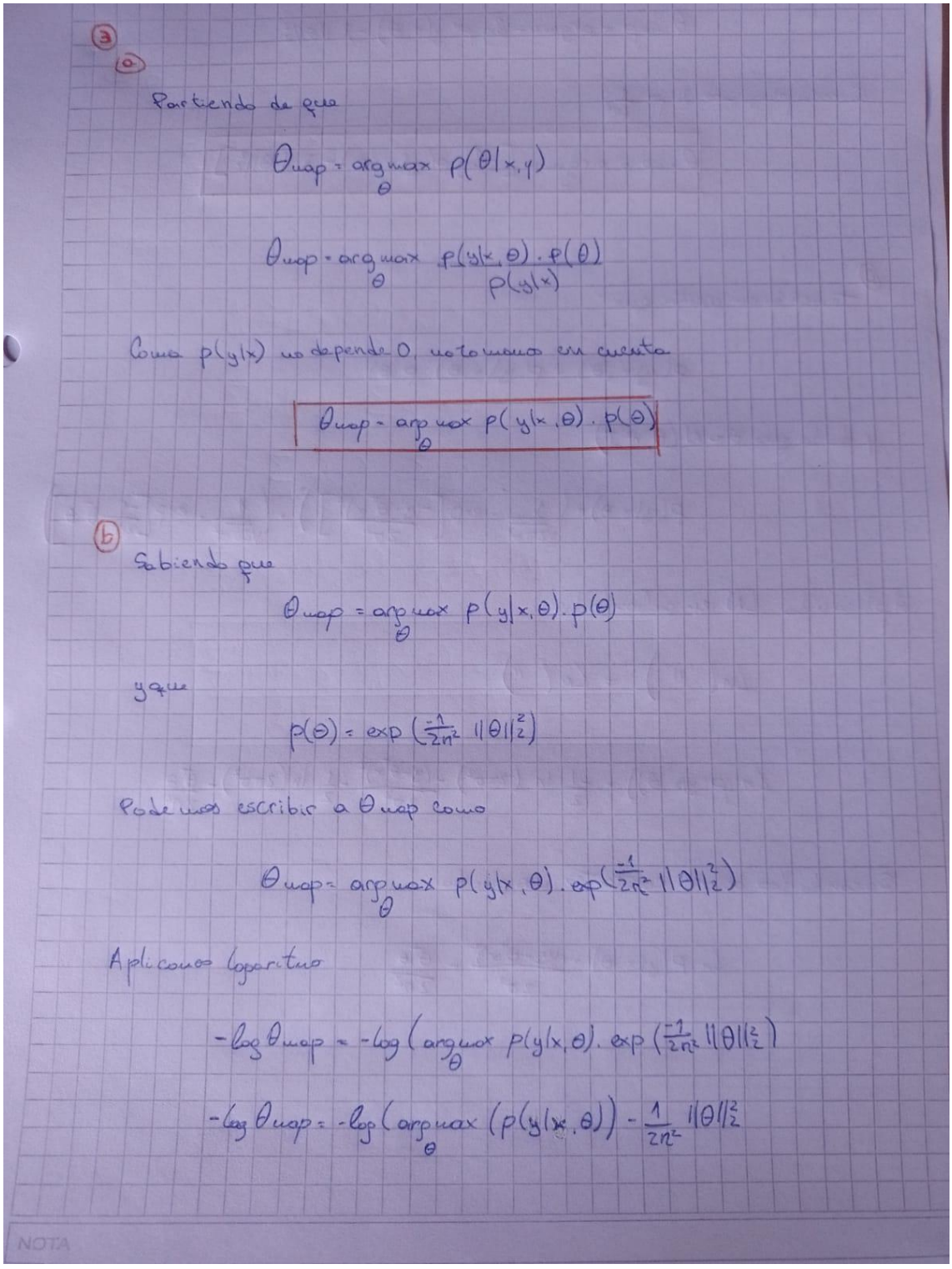
Sesgo en los coeficientes: La regularización L2 introduce un término de penalización que favorece la simplicidad del modelo, limitando los valores de los coeficientes. Esto puede reducir la sensibilidad del modelo a las características de los datos, afectando negativamente su capacidad para modelar con precisión las relaciones subyacentes.

Ajuste del hiperparámetro de regularización (λ): El impacto de la regularización L2 en la calibración depende en gran medida del valor de λ. Un valor pequeño de λ permite que los coeficientes se ajusten más cerca de los datos de entrenamiento, lo que puede resultar en una mejor calibración en el conjunto de entrenamiento, pero aumenta el riesgo de sobreajuste en datos de prueba. Un valor grande de λ incrementa la regularización y simplifica el modelo, lo que puede perjudicar la calibración si el modelo se vuelve demasiado simple para capturar la complejidad de los datos.

En resumen, la incorporación de la regularización L2 en la función objetivo de la regresión logística puede afectar tanto positiva como negativamente la calibración del

modelo. La elección adecuada de λ es esencial para encontrar un equilibrio entre prevenir el sobreajuste y mantener una calibración adecuada en los datos de prueba. La elección de λ debe basarse en las características de los datos y la relación entre sesgo y varianza del modelo.

3.



(a)

Partiendo de que

$$\theta_{map} = \arg\max_{\theta} p(\theta|x,y)$$

$$\theta_{map} = \arg\max_{\theta} \frac{p(y|x,\theta)\cdot p(\theta)}{p(y|x)}$$

Como $p(y|x)$ no depende de θ, no lo tomamos en cuenta

$$\boxed{\theta_{map} = \arg\max_{\theta} p(y|x,\theta)\cdot p(\theta)}$$

(b)

Sabiendo que

$$\theta_{map} = \arg\max_{\theta} p(y|x,\theta)\cdot p(\theta)$$

y que

$$p(\theta) = \exp\left(\frac{-1}{2n^2}\|\theta\|_2^2\right)$$

Podemos escribir a θ_{map} como

$$\theta_{map} = \arg\max_{\theta} p(y|x,\theta)\cdot \exp\left(\frac{-1}{2n^2}\|\theta\|_2^2\right)$$

Aplicamos logaritmo

$$-\log\theta_{map} = -\log\left(\arg\max_{\theta} p(y|x,\theta)\cdot \exp\left(\frac{-1}{2n^2}\|\theta\|_2^2\right)\right)$$

$$-\log\theta_{map} = -\log\left(\arg\max_{\theta}\left(p(y|x,\theta)\right) - \frac{1}{2n^2}\|\theta\|_2^2\right)$$

NOTA

Aplicando exponencial de ambos lados

$$\theta_{map} = \arg\min_{\theta} -\log p(y|x,\theta) - \lambda \|\theta\|_2^2$$

Si reemplazamos λ por $\frac{1}{2\sigma^2}$

$$\boxed{\theta_{map} = \arg\min_{\theta} \left[-\log p(y|x,\theta) - \lambda \|\theta\|_2^2 \right]}$$

(2) Teniendo en cuenta que

$$p(y|x,\theta) = p(y|x,\theta) \cdot p(\theta)$$

y dado que $\theta \sim N(\theta, \sigma^2 I)$

$$p(y|x\theta) = \left(\frac{1}{\sqrt{2\pi\sigma^2}} \cdot \exp\left[\frac{-(y-\theta^T x)^2}{2\sigma^2} \right] \right) \cdot \left(\frac{1}{\sqrt{2\pi\sigma^2}} \cdot \exp\left[\frac{-1}{2\sigma^2} \theta^T \theta \right] \right)$$

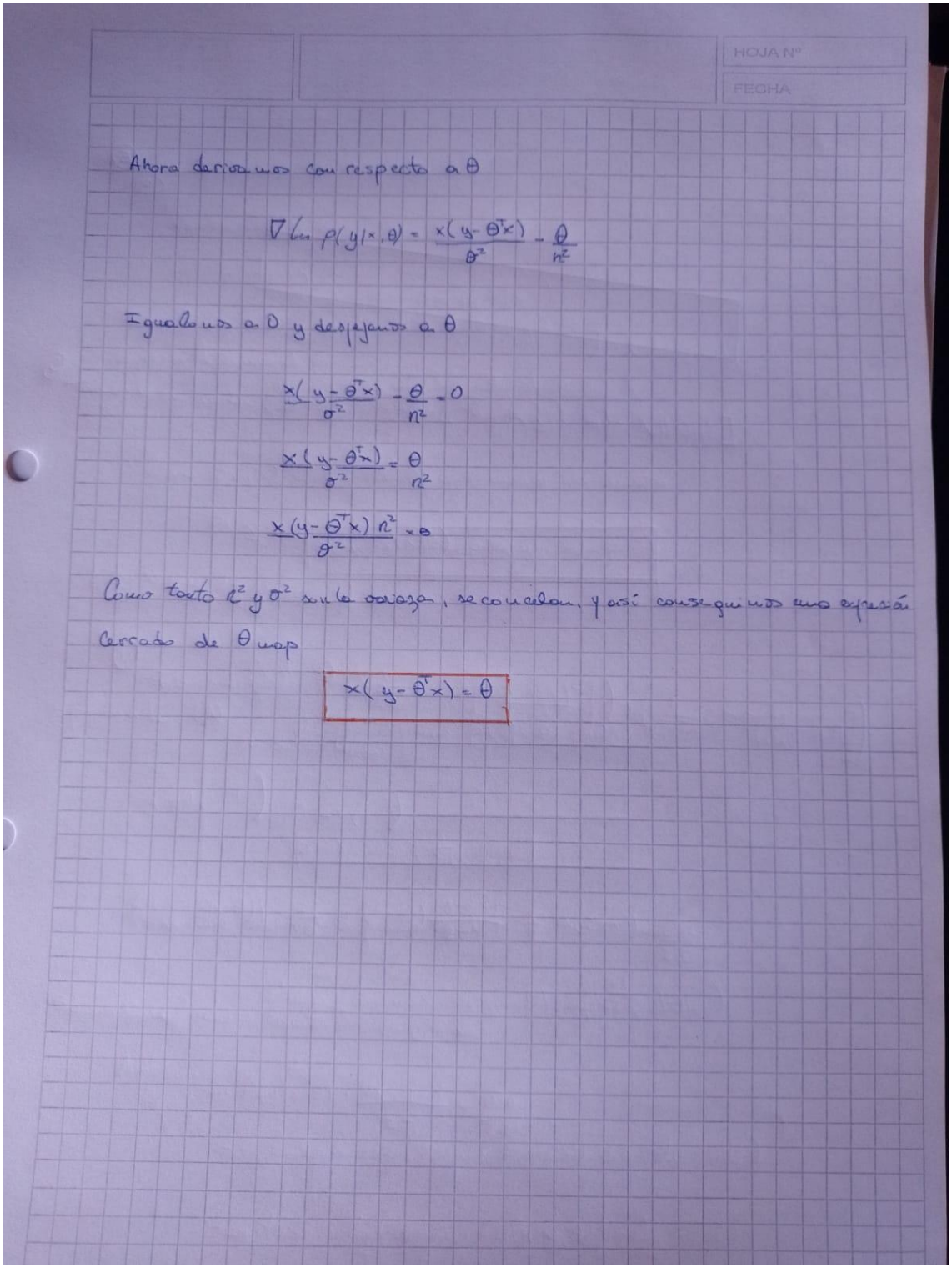
Aplicando LN en ambos lados

$$Ln(\ \ ) = Ln(\ \ )$$

$$Ln(p(y|x,\theta)) = \frac{-1}{2} \ln(2\pi\sigma^2) - \frac{(y-\theta^T x)^2}{2\sigma^2} - \frac{1}{2} \ln(2\pi\sigma^2) - \frac{\theta^T \theta}{2\sigma^2}$$

Sacando los valores constantes, nos queda:

$$Ln\ p(y|x,\theta) = \frac{(y-\theta^T x)^2}{2\sigma^2} - \frac{\theta^T \theta}{2\sigma^2}$$



4. A. Se adjunta el cogido realizado

   B. El cálculo del factor de compresión se basa en la cantidad de bits utilizados en una imagen original y en una imagen comprimida. Para calcular los bits de la imagen original, se considera la resolución de la imagen multiplicada por la cantidad de bits por píxel, que en este caso es 24 (8 bits para cada uno de los 3 canales de color: rojo,

verde y azul).

Luego, para los bits de la imagen comprimida, se tiene en cuenta la resolución de la imagen y se divide por el logaritmo en base 2 de 16, que representa la cantidad de colores utilizados en la compresión. Esto establece que cada píxel se representa con 4 bits.

El factor de compresión se calcula dividiendo la cantidad de bits de la imagen original por la cantidad de bits de la imagen comprimida. En este caso, al reducir la cantidad de colores a un total de 16, el factor de compresión es de 6, lo que significa que la imagen comprimida utiliza solo el 16.67% de la cantidad de bits que requeriría una imagen sin comprimir para representar la misma información.