Apuntes de Inteligencia Artificial – Semana 6

Ashley Vasquez
Apuntes del 09 de septiembre

***Abstract*—Este documento reúne y reformula los apuntes de la semana 6 del curso de Inteligencia Artificial. Incluye preguntas del quiz, instrucciones de la Tarea I, una breve nota sobre una actividad de IEEE y los contenidos principales de clase sobre regresión logística. Asimismo, se profundizó en la función de verosimilitud, el uso de logaritmos para simplificar derivadas, la regla de la cadena y la actualización de parámetros. Se añaden ejemplos prácticos (como el caso de la calabaza naranja / no naranja) para reforzar la comprensión del modelo.**

I. Preguntas del Quiz

1) **Overfitting y Underfitting:** El *overfitting* ocurre cuando el modelo aprende demasiado bien el conjunto de entrenamiento, pero no logra generalizar en datos nuevos. El *underfitting*, en cambio, refleja que el modelo no logra captar la relación entre las variables, obteniendo bajo rendimiento en ambos conjuntos.
2) **k-Fold Cross-Validation:** Se divide el conjunto de entrenamiento en k subconjuntos. En cada iteración se entrenan $k-1$ y el restante se utiliza para validar. Al finalizar, se promedian los resultados.
3) **Mínimos locales y globales:** Un mínimo local es el valor más bajo dentro de una región reducida de la función. El mínimo global es el valor más bajo en todo el dominio.
4) **Derivada parcial de L con respecto a w:**

$$L = \frac{1}{N}\sum\big((wx_i+b)-y_i\big)^2, \quad \frac{\partial L}{\partial w} = \frac{2}{N}\sum\big((wx_i+b)-y_i\big)x_i.$$

II. Indicaciones de la Tarea I

- Realizar la tarea en equipos de tres personas. Fecha de entrega: 16 de septiembre.
- Solo un integrante debe subir el archivo comprimido con los nombres de todos los miembros.
- Se permite únicamente el uso de `numpy` y `pandas`.
- El informe no debe contener código, únicamente análisis, resultados y conclusiones.
- El notebook será evidencia del trabajo realizado.
- La función de pérdida y las gráficas deben hacerse de forma manual.
- El formato debe ser IEEE.
- Se debe comprobar si la relación entre las variables es lineal; si no, aplicar *feature engineering*.
- El método `describe()` ayuda a resumir los datos de forma estadística.
- Figuras deben colocarse en parte superior o inferior de columnas.

III. Actividad IEEE

En noviembre se llevará a cabo un evento IEEE en La Sabana. Este congreso reúne presentaciones sobre inteligencia artificial y biología molecular, y es una oportunidad para establecer conexiones con investigadores y profesionales.

IV. Regresión Logística

A. Definición

La regresión logística es un modelo de clasificación binaria que estima la probabilidad de que un dato pertenezca a una clase. A diferencia de la regresión lineal, que entrega un valor continuo, este modelo transforma la salida en una probabilidad entre 0 y 1, y se basa en la distribución de Bernoulli.

B. Función Sigmoide

$$\sigma(x) = \frac{1}{1+e^{-x}}$$

La función sigmoide transforma cualquier número real en un valor en $[0,1]$. Se define un umbral (generalmente 0.5) para decidir la clase asignada. Como se observa en la Figura 1, es la base para convertir salidas lineales en probabilidades.

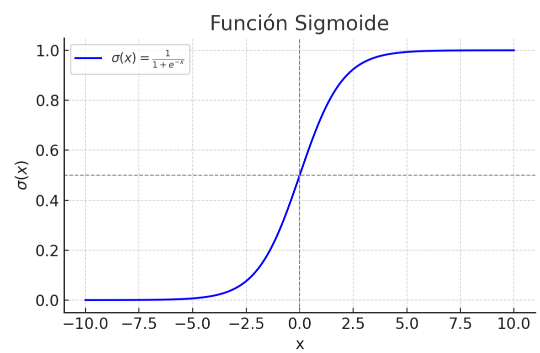

Fig. 1: Función sigmoide

C. Derivada de la Sigmoide

$$\sigma'(x) = \sigma(x)(1-\sigma(x)).$$

El hecho de que la derivada se exprese en función de la propia sigmoide la hace eficiente y práctica en optimización. La Figura 2 ilustra este comportamiento.

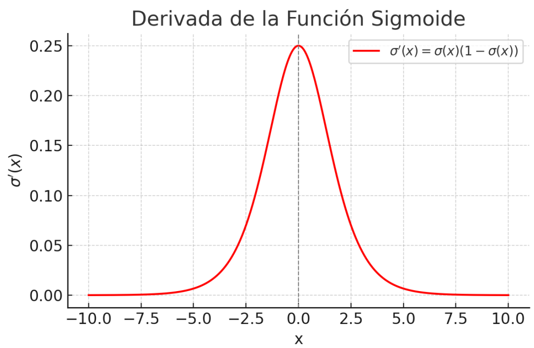

Fig. 2: Derivada de la función sigmoide

D. *Verosimilitud vs. Error cuadrático*

Mientras que el error cuadrático medio (MSE) es ideal para predecir valores continuos, la verosimilitud se utiliza cuando el resultado es una probabilidad. En regresión logística, se busca maximizar la probabilidad de que el modelo asigne la clase correcta a cada ejemplo. La Figura 3 compara ambos enfoques.

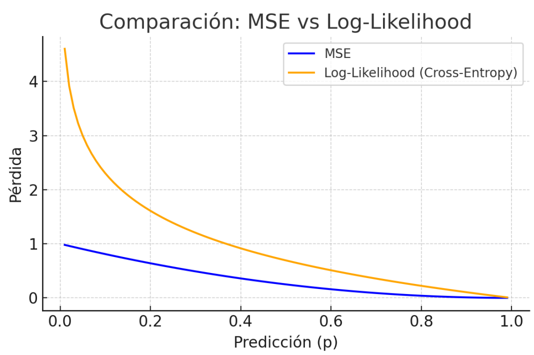

Fig. 3: Comparación entre MSE y verosimilitud

E. *Interpretación de la verosimilitud*

La verosimilitud se entiende como la probabilidad de observar los datos dados los parámetros actuales. Analicemos los casos:

- **Caso** $y_i = 1$**:** La probabilidad es $\sigma(wx+b)$. Ejemplo: $wx+b = 1.458 \implies \sigma(1.458) = 0.81$. Esto significa que hay un 81% de probabilidad de que la calabaza *no sea naranja*, como se muestra en la Figura 4.
- **Caso** $y_i = 0$**:** La probabilidad es $1-\sigma(wx+b)$. Ejemplo: $wx+b = -1.32 \implies \sigma(-1.32) = 0.21$. Entonces $1-0.21 = 0.79$, lo que se interpreta como un 79% de probabilidad de que la calabaza *sí sea naranja* (Figura 5).

F. *Uso de logaritmos*

Multiplicar probabilidades pequeñas genera valores cercanos a cero, causando inestabilidad. Aplicando logaritmos se transforma en sumas:

$$\ln(L) = \sum \left[y_i \ln(f_{w,b}(x_i)) + (1-y_i)\ln(1-f_{w,b}(x_i)) \right].$$

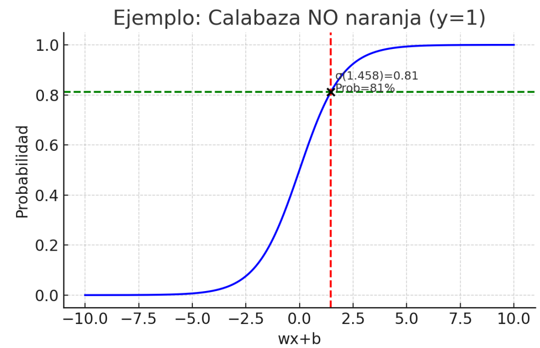

Fig. 4: Ejemplo: calabaza no es naranja

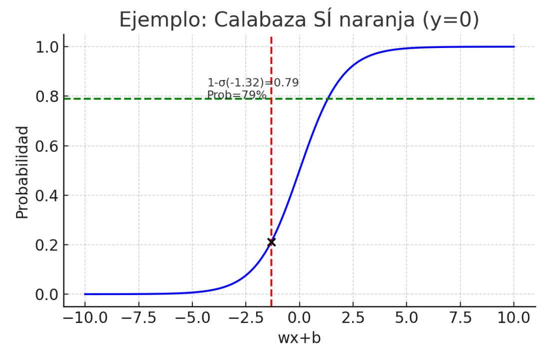

Fig. 5: Ejemplo: calabaza sí es naranja

G. *Composición de funciones y regla de la cadena*

El modelo puede expresarse como:

$$z(x) = wx + b, \quad a(z) = \sigma(z), \quad f_{w,b}(x) = a(z).$$

Entonces la función de costo es:

$$L = y_i \ln(a(z)) + (1-y_i)\ln(1-a(z)).$$

Aplicando la regla de la cadena:

$$\frac{\partial L}{\partial w} = \frac{\partial L}{\partial a} \cdot \frac{\partial a}{\partial z} \cdot \frac{\partial z}{\partial w}.$$

De manera análoga para b.

H. *Derivadas parciales paso a paso*

- Con respecto a $a(z)$:
$$\frac{\partial L}{\partial a} = -\frac{y_i}{a(x)} + \frac{1-y_i}{1-a(x)}.$$
- Con respecto a z:
$$\frac{\partial a}{\partial z} = \sigma(z)(1-\sigma(z)).$$
- Con respecto a w y b:
$$\frac{\partial z}{\partial w} = x, \quad \frac{\partial z}{\partial b} = 1.$$

I. Actualización de parámetros

Finalmente, los parámetros se actualizan con descenso de gradiente:

$$w = w - \alpha \frac{\partial L}{\partial w}, \quad b = b - \alpha \frac{\partial L}{\partial b}.$$

El valor de α (tasa de aprendizaje) es crucial. El flujo de cálculo se muestra en la Figura 6.

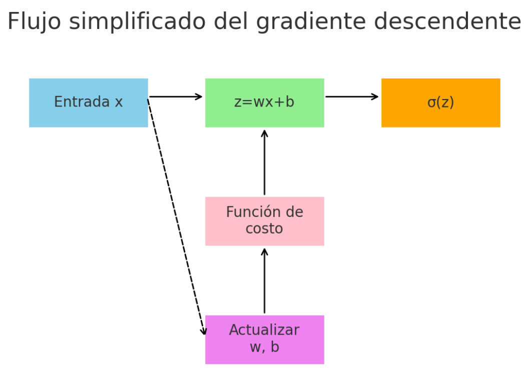

Fig. 6: Flujo de cálculo y actualización de parámetros

J. Aspectos prácticos

- **Epochs:** número de veces que el modelo recorre todo el dataset. Más epochs permiten aprender mejor, pero también aumenta el riesgo de overfitting.
- **Batch size:** cantidad de ejemplos procesados antes de actualizar parámetros. Un batch pequeño hace el entrenamiento más ruidoso pero puede mejorar la generalización.
- **Gradiente descendente estocástico (SGD):** actualiza parámetros con un ejemplo a la vez, lo que lo hace más rápido pero inestable.

V. Conclusiones

Durante esta semana se consolidaron los fundamentos de la regresión logística. Se estudiaron sus bases matemáticas, la función sigmoide y su derivada, la diferencia entre MSE y verosimilitud, el uso de logaritmos para simplificar expresiones y la actualización de parámetros mediante gradiente descendente. Los ejemplos prácticos de la calabaza facilitaron la interpretación de probabilidades, y la descomposición paso a paso de derivadas mostró cómo se aplica la regla de la cadena en la práctica. Con esto se sientan las bases para enfrentar algoritmos más avanzados en aprendizaje supervisado.