Notación Matemática en Modelos de Machine Learning

Técnicas de modelación y simulación

Conceptos Básicos de Notación Matemática

- **Variables y constantes**:
  - Escalares: x, y, α (letras minúsculas).
  - Vectores: **x**, **y** (negritas minúsculas).
  - Matrices: **X**,**Y** (negritas mayúsculas).
  - Constantes: π, e, c (letras griegas o minúsculas).
- **Funciones:**
  - *f(x):* Función de una variable.
  - *f*(**x**): Función de un vector.
  - f:$R^n \rightarrow R$: Función que mapea de R^n a R. Esta función toma un vector de *n* dimensiones y devuelve un escalar (un número real).
- **Conjuntos:**
  - R: Números reales.
  - Z: Números enteros.
  - N: Números naturales.

Ejemplo de función lineal

$$f(x) = \boldsymbol{w}^T \boldsymbol{x} + b$$

- **x** ∈ R^n: Es un vector de entrada de *n* dimensiones. Por ejemplo, **x**=$[x_1,x_2,\ldots,x_n]^T$.
- **w** ∈ R^n: Es un vector de pesos (también de *n* dimensiones). Por ejemplo, **w**=$[w_1,w_2,\ldots,w_n]^T$.
- *b* ∈ R: Es un escalar que representa el sesgo (bias).
- $\mathbf{w}^T$ **x**: Es el producto punto entre el vector de pesos y el vector de entrada.

Ejemplo de función lineal

$$\boldsymbol{w}^T\boldsymbol{x} = w_1x_1 + w_2x_2 + \cdots + w_nx_n$$

$$\boldsymbol{x} = [2,3,1]^T \qquad \boldsymbol{w} = [0.5, -1,2]^T \qquad b = 1$$

$$\boldsymbol{w}^T\boldsymbol{x} = [0.5 \quad -1 \quad 2]\begin{bmatrix} 2 \\ 3 \\ 1 \end{bmatrix} = (0.5 * 2) + (-1 * 3) + (2 * 1) = 0$$

$$f(x) = \boldsymbol{w}^T\boldsymbol{x} + b = 0 + 1 = 1$$

- El producto combina linealmente las componentes del vector **x** con los pesos **w**.
- El resultado es 1, que es un escalar en R.
- El resultado obtenido se suma al término *b*. Este término permite ajustar la función para que no necesariamente pase por el origen.

Notación en Machine Learning

- **Datos**: conjunto de entrenamiento

$$\mathcal{D} = \{(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), \cdots, (\boldsymbol{x}_n, y_n)\}$$

$\mathbf{x}_i$: Vector de características (features).
y_i: Etiqueta (target).

Matriz de datos: $\mathbf{X} \in R^{n \times d}$, donde *n* es el número de muestras y *d* es el número de características.

Notación en Machine Learning

- **Modelos**: regresión lineal

$$\hat{y} = \boldsymbol{w}^T \boldsymbol{x} + b$$

$$\mathcal{L}(\boldsymbol{w}) = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$$

Error cuadrático medio

Notación en Machine Learning

- **Optimización**: descenso de gradiente

$$\boldsymbol{w} \leftarrow \boldsymbol{w} - \eta \nabla \mathcal{L}(\boldsymbol{w})$$

η: Tasa de aprendizaje (learning rate).

$\nabla L(\mathbf{w})$: Gradiente de la función de pérdida.

El objetivo del descenso de gradiente es encontrar los valores de los parámetros de un modelo que minimicen una función de costo.

Notación en Machine Learning

- Ejemplo: gradiente

$\mathcal{L}(w) = w^2$ El objetivo es encontrar el valor de *w* que minimice L(*w*), siendo w un escalar.

Iniciar con un valor aleatorio para *w*. $w_0 = 3$

Definir la tasa de aprendizaje η (un valor pequeño que controla el tamaño del paso). η = 0.1

$$\frac{d\mathcal{L}}{dw} = 2w$$

El gradiente de L(w) con respecto a w

Calcular el Gradiente

$$\frac{d\mathcal{L}}{dw}|w = 3 = 2 * 3 = 6$$

Primera iteración (w0=3)

Notación en Machine Learning

- Ejemplo: gradiente

$$\boldsymbol{w} \leftarrow \boldsymbol{w} - \eta \nabla \mathcal{L}(\boldsymbol{w})$$

$$w_{nuevo} = w_{actual} - \eta \frac{d\mathcal{L}}{dw}$$

Actualizar parámetro

$$w_1 = 3 - 0.1 * 6 = 2.4$$

- Si η es muy pequeña, el algoritmo converge lentamente.
- Si η es muy grande, el algoritmo puede divergir (no encontrar el mínimo).
- Es crucial elegir un valor adecuado de η.

Notación en Machine Learning

$$\boldsymbol{w} \leftarrow \boldsymbol{w} - \eta \nabla \mathcal{L}(\boldsymbol{w})$$

$$\frac{d\mathcal{L}}{dw}|w = 2.4 = 2 * 2.4 = 4.8$$

Segunda iteración (w1=2.4)

$$w_2 = 2.4 - 0.1 * 4.8 = 1.92$$

$$\frac{d\mathcal{L}}{dw}|w = 1.92 = 2 * 1.92 = 3.84$$

Tercera iteración (w2=1.92)

$$w_3 = 1.92 - 0.1 * 3.84 = 1.536$$

$$\frac{d\mathcal{L}}{dw}|w = 1.536 = 2 * 1.536 = 3.072$$

Cuarta iteración (w3=1.536)

$$w_4 = 1.536 - 0.1 * 3.072 = 1.2288$$

Notación en Machine Learning

- Después de 10 iteraciones, w será aproximadamente 0.016.
- Después de 20 iteraciones, w será aproximadamente 0.0001.

w se acercará cada vez más al valor que minimiza la función de costo (w=0)

- La función $L(w)=w^2$ es una parábola con su mínimo en w=0.
- El gradiente $dL/dw=2w$ indica la pendiente de la función en cada punto.
- En cada iteración, el descenso de gradiente "da un paso" hacia el mínimo, ajustando w en la dirección opuesta al gradiente

Notación en Machine Learning

- En problemas más complejos (como regresión lineal con múltiples características), **w** es un vector, y el gradiente es un vector de derivadas parciales:

$$\nabla\mathcal{L}(\boldsymbol{w}) = \left(\frac{\partial\mathcal{L}}{\partial w_1}, \frac{\partial\mathcal{L}}{\partial w_1}, \cdots, \frac{\partial\mathcal{L}}{\partial w_n}\right)$$

$$\boldsymbol{w}_{nuevo} = \boldsymbol{w}_{actual} - \eta\nabla\mathcal{L}(\boldsymbol{w})$$

Notación en Machine Learning

- **Capa oculta**: redes neuronales

$$\boldsymbol{h} = \sigma(\boldsymbol{W}\boldsymbol{x} + \boldsymbol{b})$$

W: Matriz de pesos.

σ: Función de activación (e.g., ReLU, sigmoide).

Ejemplos

- **Regresión logística:**

$$\sigma(z) = \frac{1}{1 + e^{-z}}$$ Función sigmoide

$$\hat{y} = \sigma(\boldsymbol{w}^T \boldsymbol{x} + b)$$ Predicción

Ejemplos

- **Clasificación con SVM:**

$$\boldsymbol{w}^T \boldsymbol{x} + b = 0$$

Hiperplano

$$\frac{2}{\|\boldsymbol{w}\|}$$

Margen

Ejemplos

- **Redes neuronales convolucionales (CNN):**

$$(\boldsymbol{I} * \boldsymbol{K})(i, j) = \sum_m \sum_n \boldsymbol{I}(i-m, j-n)\boldsymbol{K}(m, n)$$

Operación de convolución

I: Imagen de entrada.

K: Kernel (filtro)