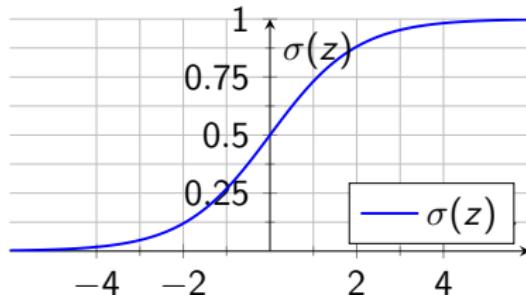


Motivación: Decisiones Probabilísticas

- Etiqueta binaria con probabilidad $p(y = 1 | x)$, no solo clase dura.
- Puntaje $z = b + w^\top x$ es no acotado; se enlaza a $[0, 1]$.
- **Puntaje vs probabilidad:** z solo ordena; $\sigma(z)$ es calibrada.
- Ejemplo: $z = 0.2 \Rightarrow p \approx 0.55$; $z = -2 \Rightarrow p \approx 0.12$.

Enlace Sigmoide y Geometría

- Sigmoide: $\sigma(z) = \frac{1}{1+e^{-z}}$ comprime cualquier puntaje a $[0, 1]$.
- $z = b + w^\top x$ es distancia con signo al hiperplano (escalada por $\|w\|$).
- La pendiente es máxima en $z = 0$; cerca de la frontera hay mayor incertidumbre.
- Mini chequeo:
 $x = (1, 2), w = (0.3, 0.4), b = -0.5 \Rightarrow z = 0.6, p \approx 0.65$.



Log-Odds Lineales

- Transformación logit: $\log \frac{p}{1-p} = b + w^\top x$.
- Logit afín en $x \Rightarrow$ frontera $w^\top x + b = 0$ es un hiperplano (modelo lineal).
- Monotonía: ordenar por z o por p es equivalente.

Verosimilitud Bernoulli

- Para datos i.i.d., $p_i = \sigma(b + w^\top x_i)$.
- Verosimilitud:

$$\mathcal{L}(w, b) = \prod_i p_i^{y_i} (1 - p_i)^{(1-y_i)}.$$

- Maximizar \mathcal{L} elige w, b que hacen más probables las etiquetas observadas.

MLE y Pérdida Logarítmica

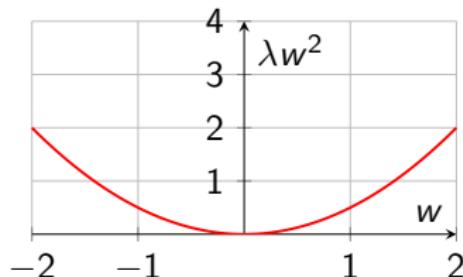
- Log-verosimilitud negativa:

$$\text{NLL} = - \sum_i [y_i \log(p_i) + (1 - y_i) \log(1 - p_i)].$$

- Minimizar NLL equivale a minimizar cross-entropy binaria (log-loss).
- Lectura: los errores confiados se penalizan fuertemente; los inciertos, menos.

Regularización (L2)

- Se agrega $\lambda\|w\|^2$ a la NLL para limitar pesos grandes.
- Vista geométrica: L2 mantiene w dentro de una hiperesfera; penaliza por igual en todas las direcciones.
- Reduce sobreajuste y estabiliza características correlacionadas.
- En scikit-learn, 'C' es inverso de la fuerza: $\lambda = \frac{1}{2C}$; 'C' menor \Rightarrow más shrinkage.



Desbalance de Clases

- 'class_weight="balanced"' define peso para clase c : $\text{peso}_c = \frac{N}{2N_c}$ (inversa de frecuencia).
- Cada término de pérdida se multiplica por su peso, dando más impacto a errores en la minoría.
- Ayuda a alinear umbral efectivo con recall/precision cuando los positivos son raros.

Predictión, Umbral, Ranking

- 'predict_proba' entrega $p = \sigma(b + w^\top x)$; decisión por defecto usa 0.5.
- Bajar el umbral reduce FN pero sube FP; subirlo hace lo contrario.
- ROC AUC evalúa el ranking por p (o z) sin fijar umbral.
- Mini diagrama: features $\rightarrow z = b + w \cdot x \rightarrow \sigma(z) \rightarrow p \rightarrow$ umbral \rightarrow clase.

