
Algorithm 1 PC-SMOTE (versión binaria)

- 1: **Entrada:** $X \in \mathbb{R}^{n \times d}$, $y \in \{0, 1\}$ (1 = minoritaria), k , p_{dist} , p_{ent} (opcional), p_{den} (opcional), r (radio de densidad), *criterio_pureza* = entropía o proporción, *modo_espacial* $\in \{2d, 3d\}$, *max_sint* (opcional).
- 2: **Salida:** X', y' (sintéticas etiquetadas como 1).
- 3: Convertir X e y a arreglos; inicializar métricas internas.
- 4: Separar $X_{\min} \leftarrow X[y = 1]$, $X_{\text{maj}} \leftarrow X[y = 0]$. **Si** $|X_{\min}| < k + 1$ **entonces return** (X, y) .
- 5: Ajustar k NN global sobre X ; para cada $x_i \in X_{\min}$ obtener vecinos $\mathcal{N}_i^{\text{all}}$ (excluyendo x_i) y calcular el **riesgo** local

$$r_i \leftarrow \frac{1}{k} \sum_{x_j \in \mathcal{N}_i^{\text{all}}} \mathbb{I}[y_j = 0].$$

- 6: Ajustar k NN sobre X_{\min} ; obtener \mathcal{N}_i^{\min} y la **densidad** por intersección de esferas

$$\text{den}_i \leftarrow \frac{1}{k} \sum_{x_j \in \mathcal{N}_i^{\min}} \mathbb{I}(\|x_i - x_j\| \leq 2r),$$

usando sólo las tres primeras coordenadas si *modo_espacial* = 3d; en caso contrario, el vector completo.

- 7: **Pureza:** **si** el criterio es entropía, calcular

$$H_i = \text{entropy}(\text{freq}(y_{\mathcal{N}_i^{\text{all}})}),$$

y si p_{ent} está definido, fijar $\tau_{\text{ent}} = \text{percentil}(H, p_{\text{ent}})$; definir $m_i^{\text{pureza}} = \mathbb{I}[H_i \leq \tau_{\text{ent}}]$. **Si** el criterio es proporción, calcular $\pi_i = \frac{1}{k} \sum_{x_j \in \mathcal{N}_i^{\text{all}}} \mathbb{I}[y_j = 1]$ y definir $m_i^{\text{pureza}} = \mathbb{I}[0.4 \leq \pi_i \leq 0.6]$.

- 8: **Densidad:** **si** p_{den} está definido, fijar $\tau_{\text{den}} = \text{percentil}(\text{den}, p_{\text{den}})$ y $m_i^{\text{dens}} = \mathbb{I}[\text{den}_i \geq \tau_{\text{den}}]$; en caso contrario, $m_i^{\text{dens}} = \mathbb{I}[\text{den}_i > 0]$.
 - 9: **Filtrado:** $m_i = m_i^{\text{pureza}} \wedge m_i^{\text{dens}}$; $C = \{x_i \in X_{\min} : m_i = 1\}$. **Si** $|C| < k + 1$ **return** (X, y) .
 - 10: **Cantidad de sintéticas:** $n_{\text{sint}} = \text{max_sint}$ si fue provisto; en caso contrario $|X_{\text{maj}}| - |X_{\min}|$. **Si** $n_{\text{sint}} \leq 0$ **return** (X, y) .
 - 11: **for** $t = 1$ **to** n_{sint} **do**
 - 12: Muestrear $x_i \sim C$; recuperar r_i y $\mathcal{N}_i^{\text{all}}$.
 - 13: Calcular distancias $D_{ij} = \|x_i - x_j\|$ para $x_j \in \mathcal{N}_i^{\text{all}}$ y $\theta_i = \text{percentil}(D_i, p_{\text{dist}})$.
 - 14: Definir $V_i = \{x_j \in \mathcal{N}_i^{\text{all}} : D_{ij} \leq \theta_i\}$. **Si** $|V_i| = 0$, **continue**.
 - 15: Muestrear $z \in V_i$.
 - 16: **if** $0.4 \leq r_i < 0.5$ **then**
 - 17: $\delta \leftarrow \text{Uniform}(0.6, 0.8)$
 - 18: **else if** $0.5 \leq r_i \leq 0.6$ **then**
 - 19: $\delta \leftarrow \text{Uniform}(0.3, 0.5)$
 - 20: **else**
 - 21: $\delta \leftarrow \text{Uniform}(0.4, 0.6)$
 - 22: **end if**
 - 23: Generar $\tilde{x} = x_i + \delta(z - x_i)$ y apilar a X con etiqueta 1.
 - 24: **end for**
 - 25: **return** (X', y') con todas las sintéticas concatenadas al final.
-