Difficulty-Directed Hybrid Sampling (DD-Hybrid)

# 1. Motivation

Imbalanced datasets often suffer from two intertwined issues: (i) class rarity, which biases learners toward the majority class, and (ii) class overlap, which produces brittle decision boundaries and noisy synthetics when oversampling indiscriminately. DD-Hybrid addresses both by (a) concentrating minority oversampling near informative regions (border) while avoiding outliers, and (b) reducing majority redundancy through iterative, threshold-driven pruning that preserves the boundary.

# 2. Notation

Let (X, y) = {(x\_i, y\_i)} with y\_i in {0,1}, where minority class y=1. Define M={i: y\_i=1}, G={i: y\_i=0}. Let kNN\_k(x) return the k nearest neighbors of x, and opp\_k(x) be the number of opposite-class neighbors within kNN\_k(x). The imbalance ratio is IR = |G|/|M|. Default hyperparameters: k=5, border/safe oversampling split = 75%/25%, target IR band IR\_target in [1.0, 2.0].

# 3. Minority Categorization (kNN-based)

Each minority instance is assigned to one of three categories using opposite-neighbor counts:  
- Safe: opp\_k(x) <= 1  
- Border: opp\_k(x) in {2,3}  
- Outlier: opp\_k(x) in {4,5}  
This discretization encodes local impurity: safe (≤0.2), border (0.4–0.6), outlier (≥0.8) for k=5.

# 4. Focused Oversampling (OS)

We oversample only from Safe ∪ Border (never from Outliers). The OS budget (implicitly determined by IR\_target or explicitly by a factor α) is allocated as: Border seeds: 75% of the budget; Safe seeds: 25% (used primarily when additional IR reduction is needed). Synthetic samples are generated via SMOTE-style interpolation using neighbors from the same class, with a preference for same-category neighbors.

## 4.1 Optional Distance-Aware Refinements

- Margin score: prefer seeds with reasonable/positive margins.  
- Density-adjusted impurity: prioritize denser border regions.  
- Distance-weighted impurity: weight neighbor votes by inverse distance.  
Spillover rule: if border cannot absorb 75%, spill remainder to safe. Tiny-data guard: cap per-seed quota to avoid overfitting.

# 5. Iterative Redundancy Pruning (US, majority only)

Rather than random under-sampling or boundary-aggressive cleaning, DD-Hybrid prunes only deep-interior, redundant majority points in iterations with no fixed removal quota.  
At iteration t:  
1. Compute redundancy signals (interiorness, distance to minority, local density).  
2. Eligibility: u\_j <= 1, d\_min(j) ≥ 60th percentile, ρ\_j ≥ 60th percentile. Guards: retain Tomek pairs, prototypes.  
3. Remove all eligible. Recompute IR.  
4. If IR not in target, relax thresholds slightly (quantiles down by 0.05) and repeat.

# 6. Stopping Criteria & Defaults

Stop when IR is in IR\_target or no eligible majority remains. Defaults: k=5, initial thresholds q\_d=q\_ρ=0.60, relaxation step 0.05, border/safe split=0.75/0.25. Optional safety cap: do not remove >30% of majority in one iteration.

# 7. Complexity

Neighbor queries dominate. With a standard kNN index, each iteration costs O(n log n + nk). OS is linear in synthetics generated. Overall adds modest overhead relative to standard SMOTE.

# 8. Rationale & Expected Behavior

- Prioritizing border strengthens the classifier near errors; safe keeps class cohesion.  
- Excluding outliers prevents synthetic noise.  
- Redundancy pruning targets only deep-interior majority, preserving decision frontier.  
- Iterative scheme adapts as OS changes geometry, unlike fixed-quota RUS/NCL.

# 9. Implementation Notes

- Use a fixed random\_state.  
- Report IR before OS, after OS, and after each US iteration.  
- Include ablations on OS split, refinements, US iterative vs single-pass, thresholds.

# 10. Limitations & Scope

- kNN stats can be noisy in high dimensions; consider scaling/PCA.  
- Very small minority sets (<20) remain fragile; use stronger guards or cost-sensitive learning.