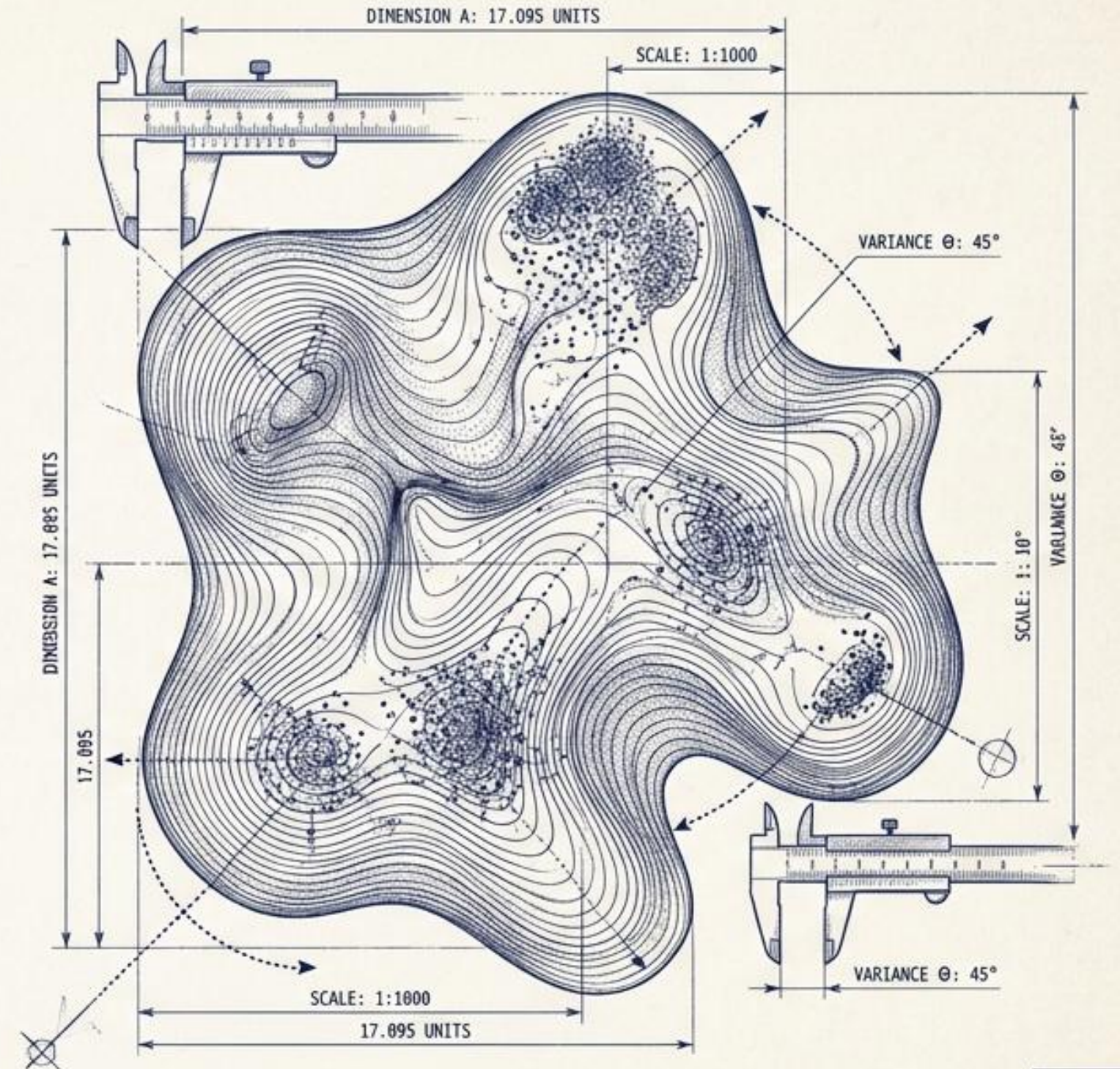


戦略と執念で掴んだ 17.095

Google Gemini DeepResearchから
始まったSVR×Deep Learning実装と
トラブルシューティングの全貌

PetFinder.my - Pawpularity Contest



ARCHITECTS BLUEPRINT / ENGINEERS LOGBOOK



スコアボード：執念の結実と「Trust Your CV」の証明

Private Score

17.09524

Local CV

17.1306

差わずか「0.035」。厳密なCV
構築による完璧な相関の証明。

1位解法

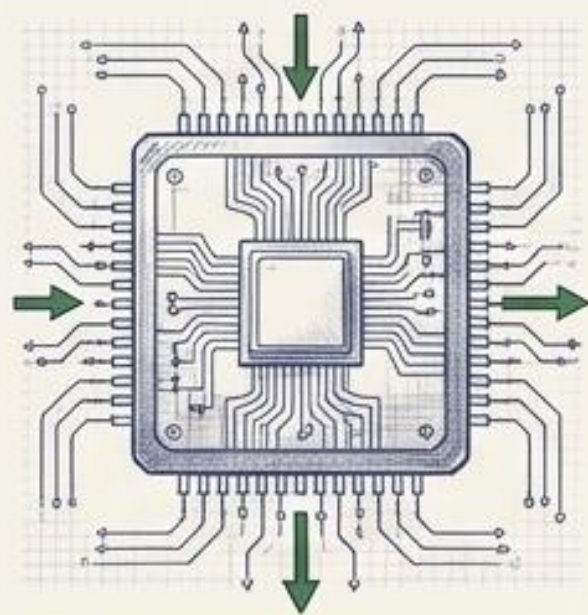
16.8225

Public Score

17.94270

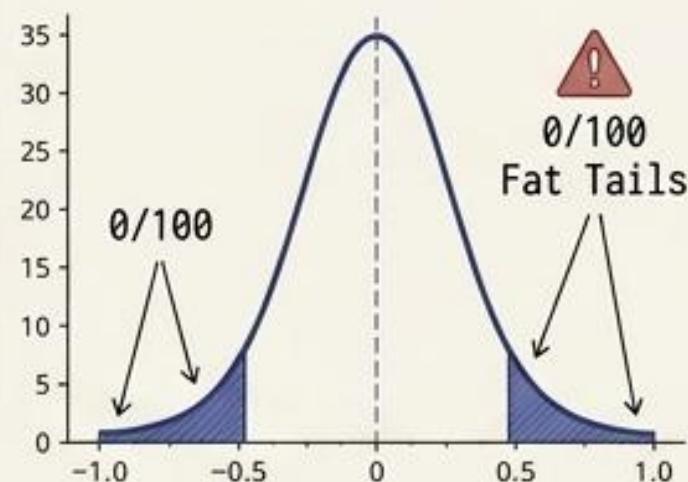
DeepResearchが示した4大戦略

1. RAPIDS SVR Magic



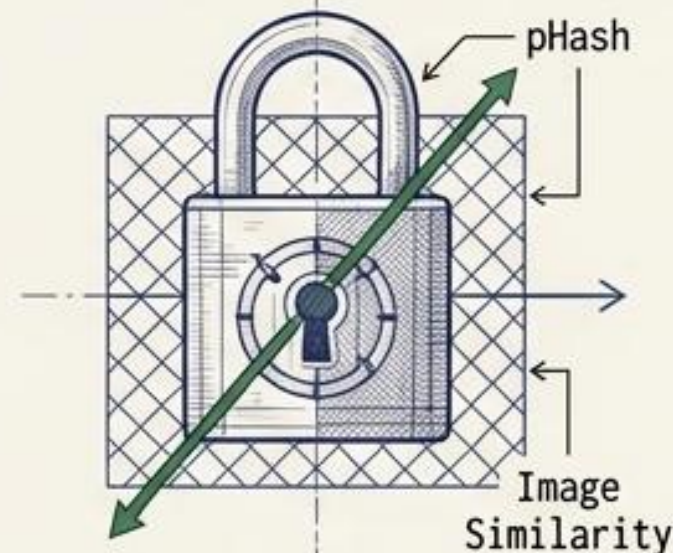
数百の事前学習済みモデルから特徴量を抽出。cuML SVRによる高速探索(山登り法)のアプローチ。

2. BCE Loss(Quasi-MLE)



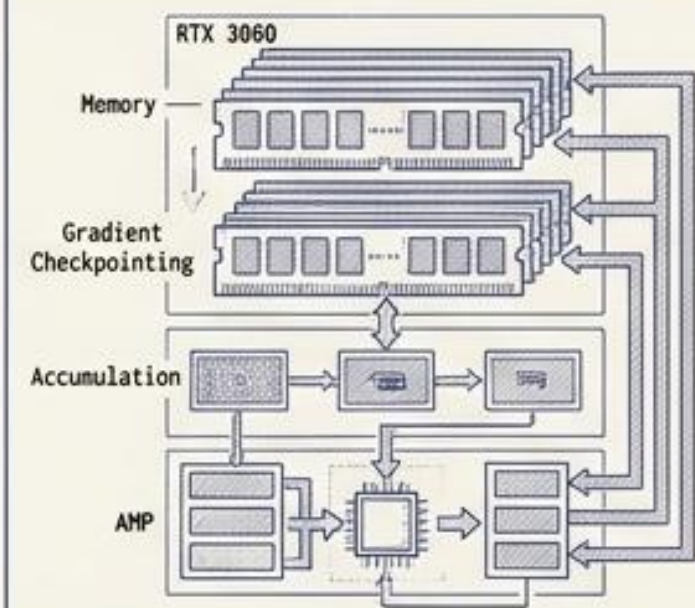
Pawpularity特有の0/100ファットテール分布に対する最適解。平均への回帰バイアスを打破。

3. 厳密な重複除去CV



pHash + 画像類似度による完全な隔離。ローカルCVとLBを高精度で一致させる基盤。

4. VRAM 12GB省メモリ化



RTX 3060環境を突破。Gradient Checkpointing / Accumulation / AMPによる大型モデル学習。

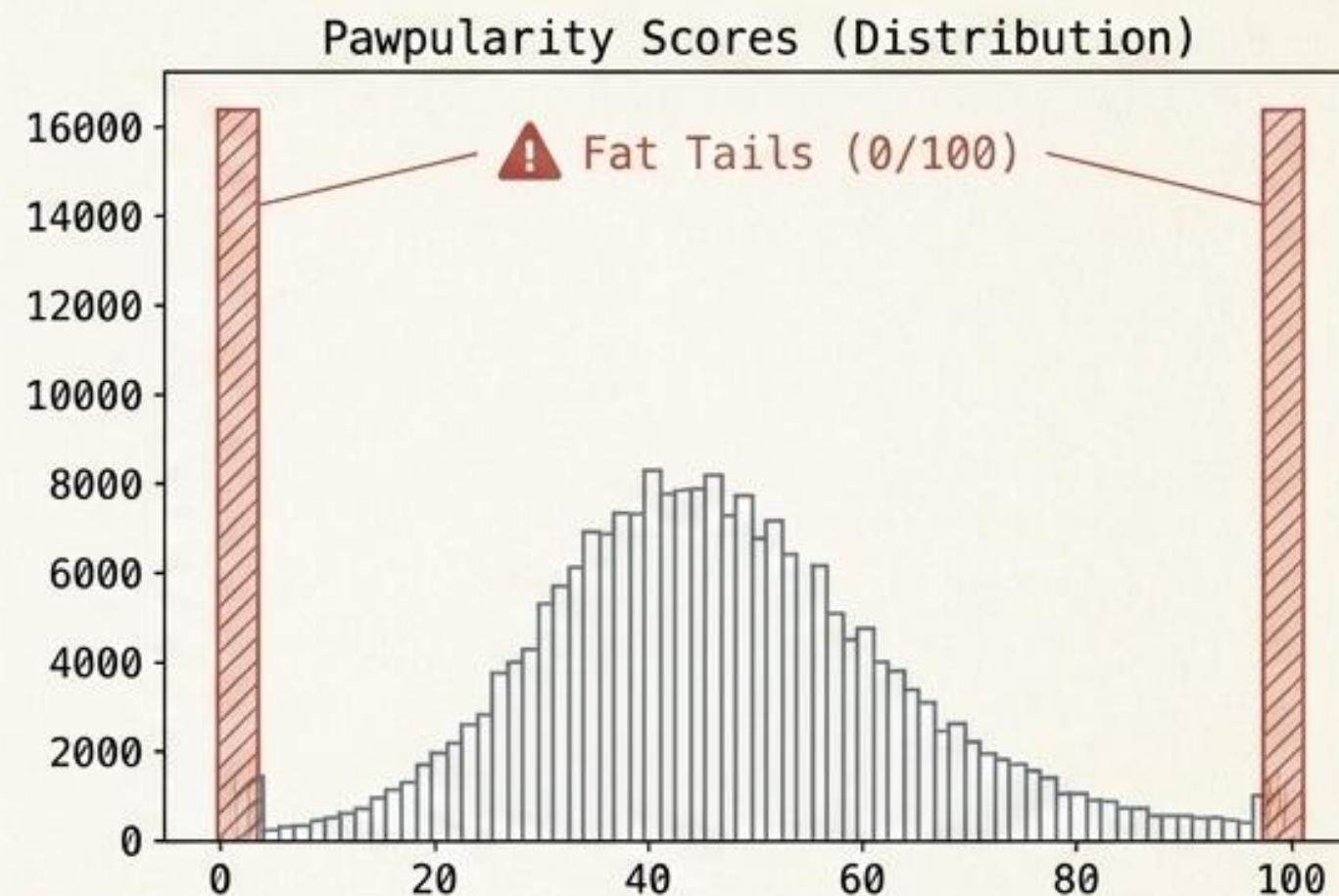
生データの現実と「ファットテール」の壁

学習画像: 9,912枚

平均値: 38.04 (±標準偏差 20.59)

欠損値: 0

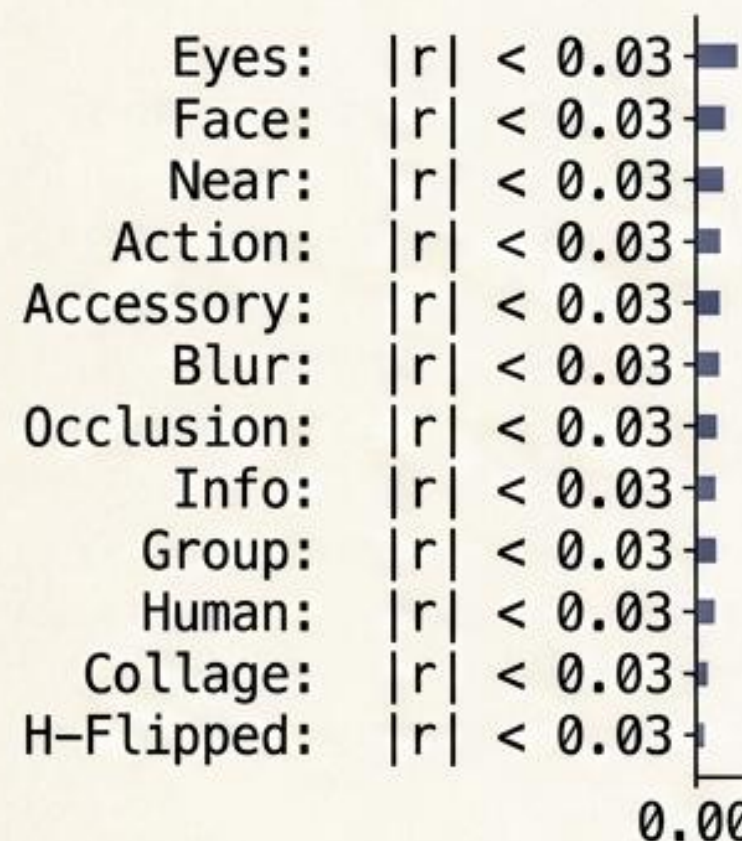
画像解像度: RGB・破損なし
(幅 mean 823px / 高さ mean 917px)



なぜRMSEではなくBCE Lossか？

0と100付近の不自然なピーク(ファットテール)に対し、RMSEは平均への回帰バイアスを生む。
BCE(Quasi-MLE)で確率分布として学習する戦略が必須。

メタデータの幻想：12項目の無相関証明

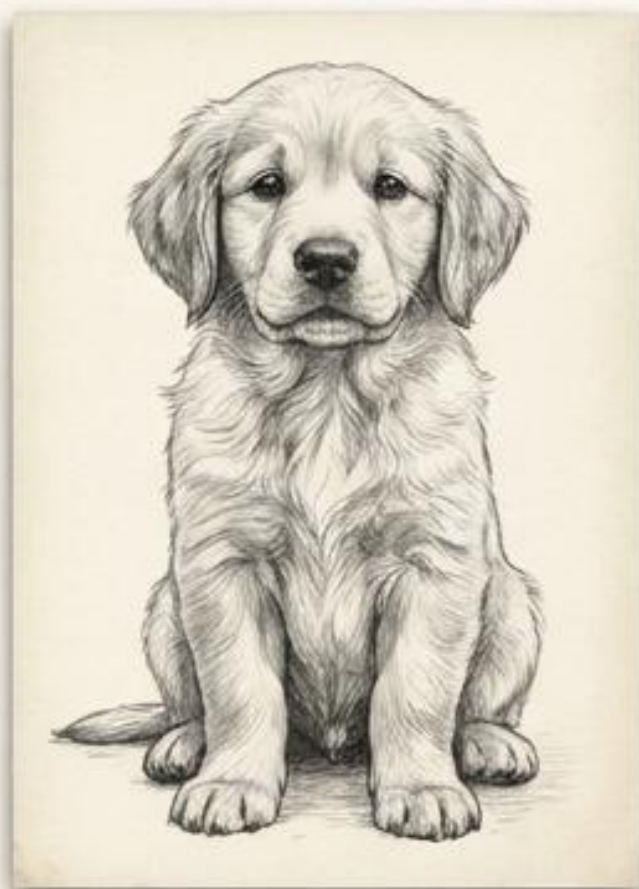


Decision: メタデータを漫然と混ぜる誘惑を捨てる。

全メタデータの相関係数は $|r| < 0.03$ 。ノイズを増やすことを避け、計算資源を「画像特徴量そのもの」へ全集中する。

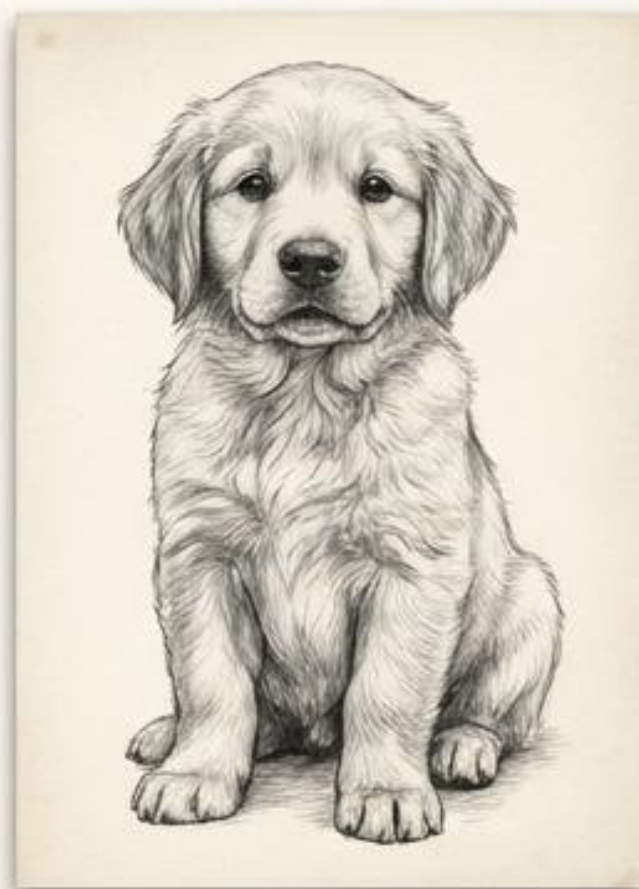
重複画像の罠と、完璧なCV基盤の構築

Problem



Id: A (Pawpularity 37)

≠



Id: B (Pawpularity 40)

脅威: 類似画像がTrain/Valに跨ると致命的なCVリークが発生する。

Solution

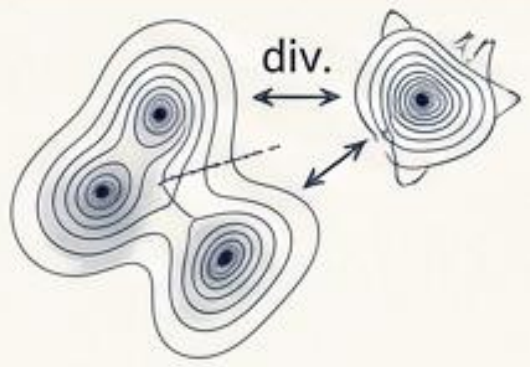
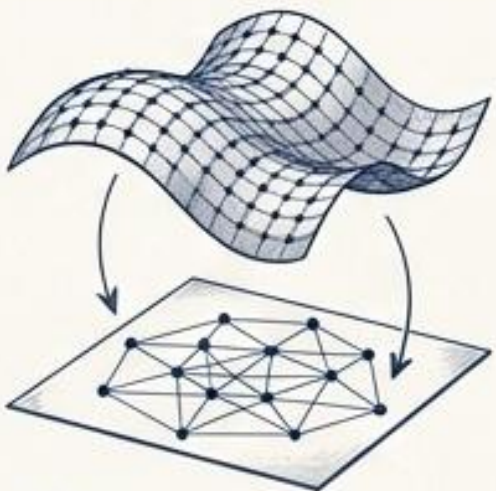
検出: pHash + EfficientNet-B0コサイン類似度で走査。

結果: 9,912枚中、38組の近似重複画像を特定。

隔離 (StratifiedGroupKFold): 重複ペアを同一グループ化 + Sturges則(14ビン)によるターゲット層化。

この厳密なCV設計が、後の「CVとLBの差 0.035」という驚異的な一致(Trust Your CV)を保証した。

特徴量空間の解釈：UMAP vs t-SNEの数理

次元削減手法	数理的アプローチ	読み方のポイント
t-SNE	KLダイバージェンス / 局所確率 	局所的なまとまりの再現に強い。クラスタ間の距離(大域構造)に意味を持たせにくい。
UMAP	多様体学習 (Manifold Learning) 	局所構造を保ちつつ、大域的な配置も保持。全体の構造的関係性を読むのに適する。

Rule for the next slides:
2つの異なるアルゴリズムを見比べ、両方に共通して現れる構造こそが、真に信頼できる強力な潜在表現(Latent Representation)である。

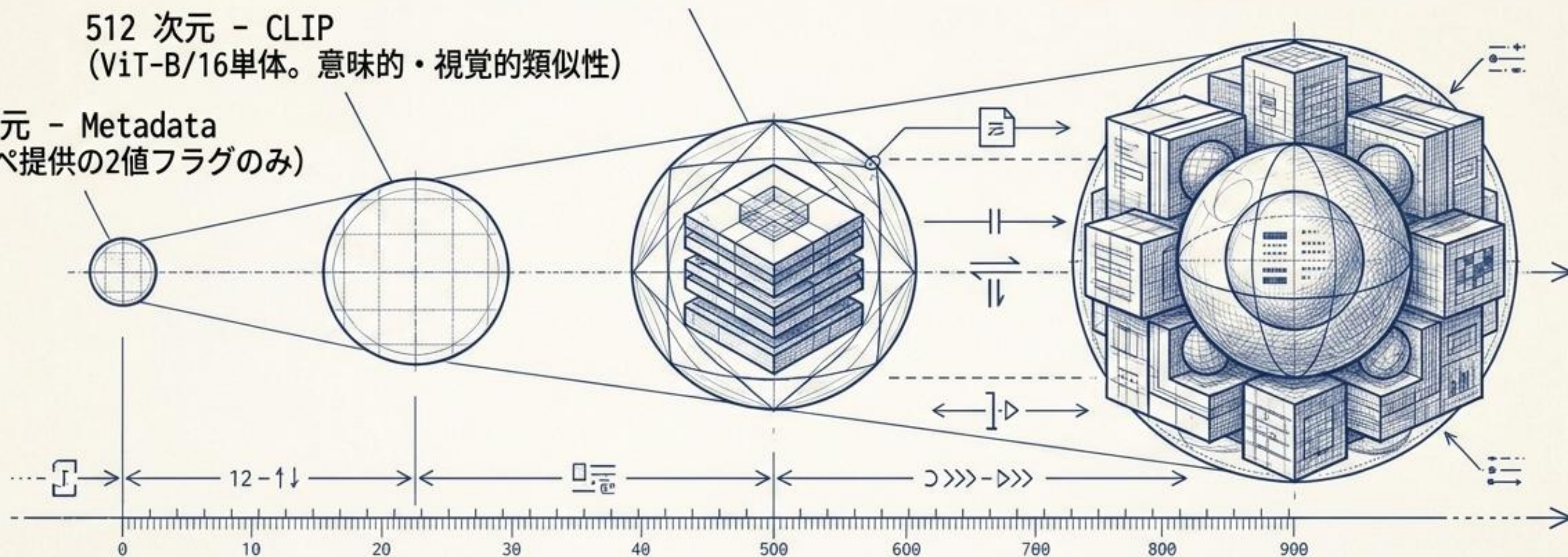
採用した特徴量埋め込み (Embeddings) のスケール

5,504 次元 - tf_efficientnet_l2_ns_475
(単体SVR最強モデル)

****13,952 次元 - svr9combo**
(本番SVRに採用した9モデルの結合特徴量)

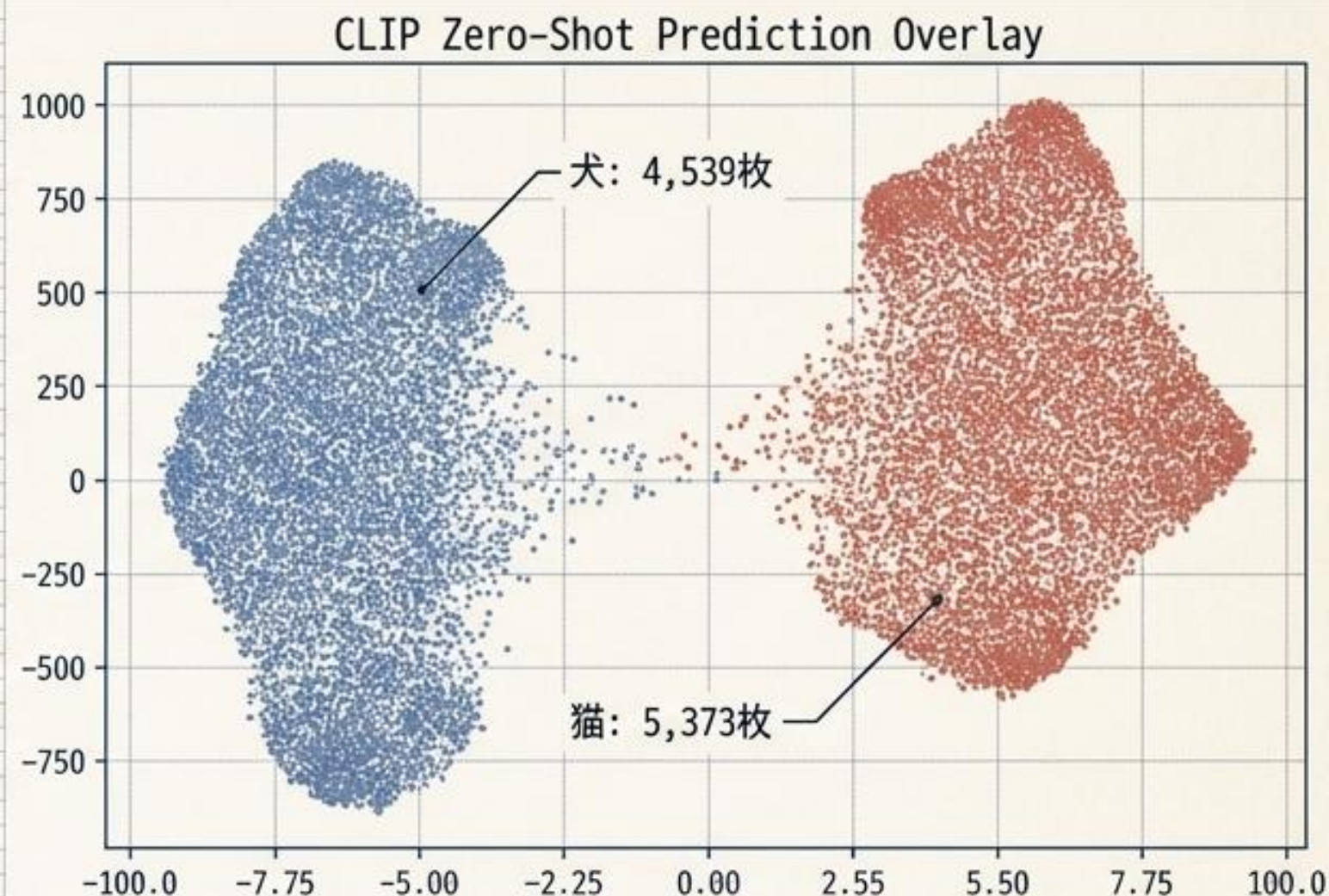
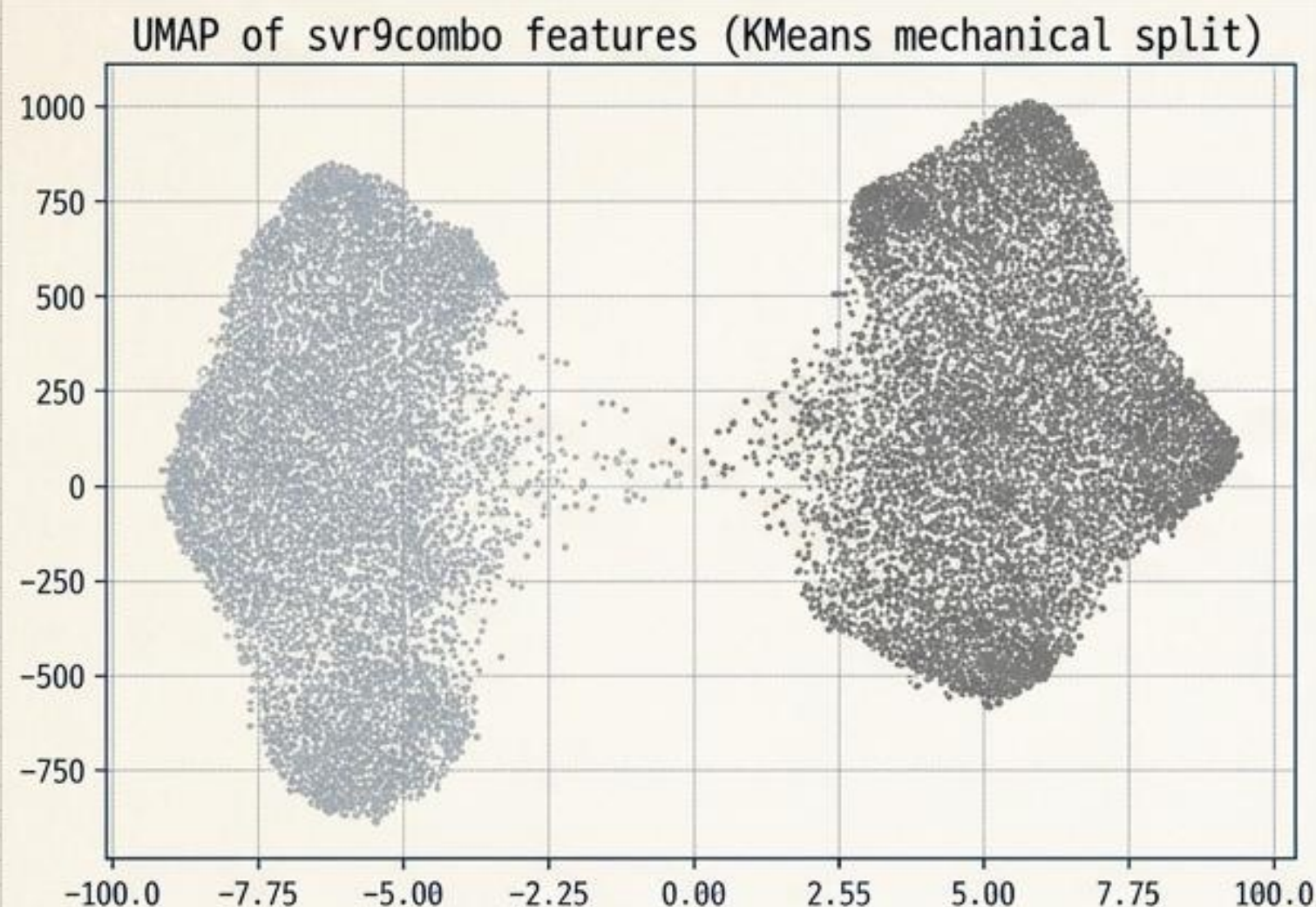
512 次元 - CLIP
(ViT-B/16単体。意味的・視覚的類似性)

12 次元 - Metadata
(コンペ提供の2値フラグのみ)



モデルはこれらの超高次元ベクトルから、人間には言語化不可能なPopularityの「視覚的法則」を読み解く。

埋め込み空間の「二つの島」：機械分割と意味的分割の一致



99.7%の一致 (9,885 / 9,912枚)

ラベルを与えていないにも関わらず、svr9combo空間の最大分散軸はCLIP由来の「種の違い(犬/猫)」を完全に保持している。Pawpularityスコアは、この軸に直交する微細な構造として存在する。

高スコアクラスタ「C2」の発見と暗黙のルール



Cluster C2

平均スコア: 60.9

(全8クラスタ中、突出して最高値。他は32~43台)



The Secret AI Discovered:

メタデータ単体では相関ゼロだった。しかし埋め込み空間は「中央構図」「単体被写体」「シンプルな背景」という人間の暗黙の好みを、構造的クラスタとして明確に捉え分離していた。

RAPIDS SVR & 貪欲法(Hill Climbing)による限界突破

Start: 単体最良モデル
(CV: 17.52)

環境: Windows / Anaconda + WSL2 (RAPIDS 26.02)
利点: cuMLによる圧倒的演算速度。
全55回の候補評価を約13分で完走。

+model X

CV: 17.45

CV: 17.46

+model Y

CV: 17.38

CV: 17.39

+model Z

CV: 17.26

CV: 17.24

改善幅: -0.375

End: 9モデル結合
「svr9combo」13,952次元
(CV: 17.1410)

End-to-Endで巨大DLモデルを回すより遥かに軽い計算コストで、
極めて強力なベースライン(17.1台)を構築。これが1位解法のコア。

Deep Learning (beit_large) の2大理論バグと修正

Bug 1

Flaw

BCEの境界値発散。連続ターゲットの最適ロジットが $\text{target} \rightarrow 1$ で無限大に発散する数理的欠陥。



Symptom

Pawpularity=100の288件が境界に直撃し、学習が崩壊 (Val RMSE 36超)。



Fix

ターゲットを $[1e-3, 1-1e-3]$ に厳密にクリッピング。

Bug 2

Flaw

新規追加した出力ヘッドの過大な初期スケール。



Symptom

sigmoid関数が1エポック目から即座に飽和し、勾配消失。



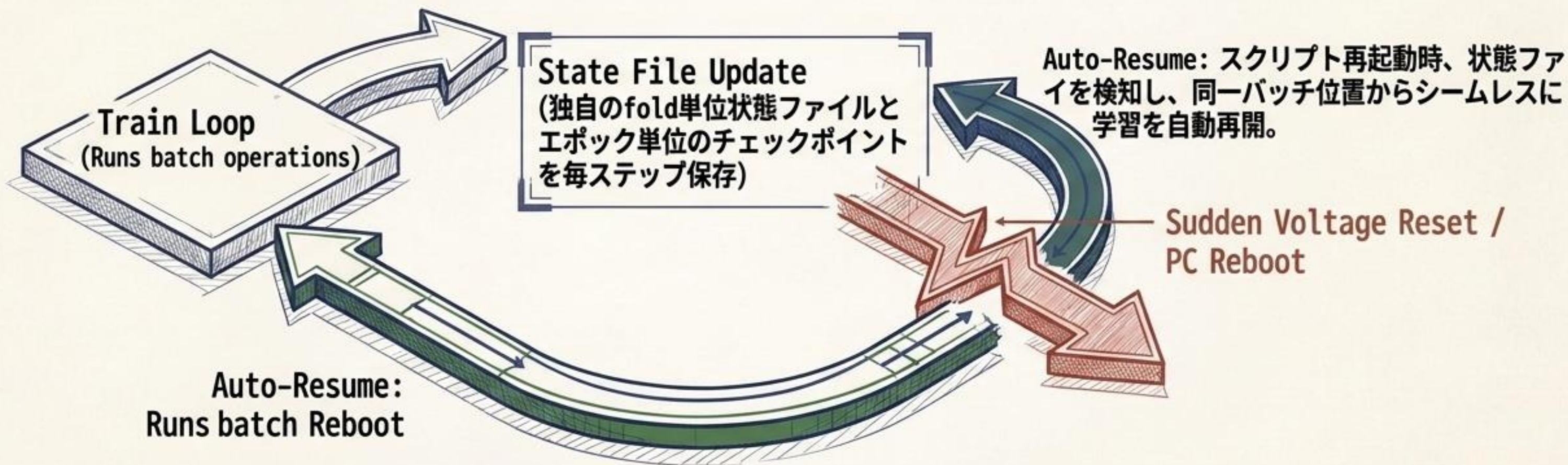
Fix

出力ヘッドを縮小再初期化 + バイアスを「グローバル平均のlogit」に設定。

ハードウェア不安定性を克服する耐障害ループ

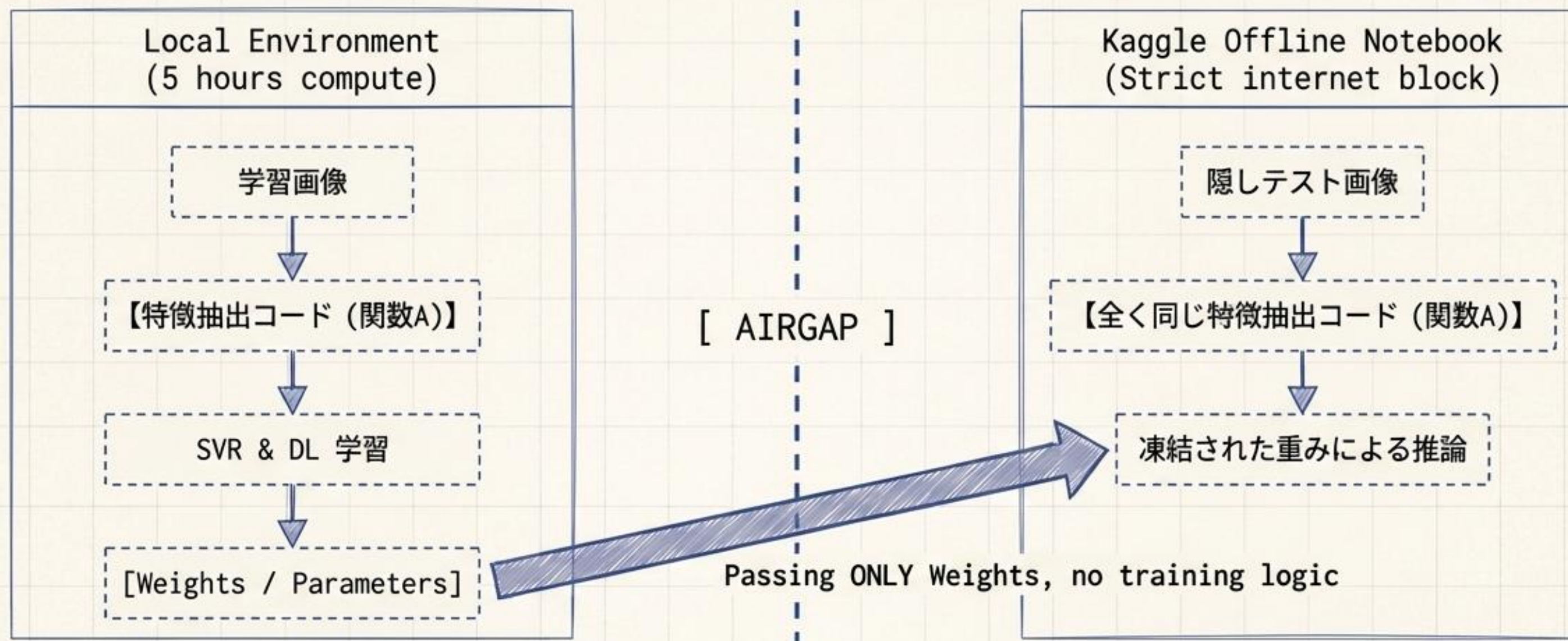


Intel Core i7-11700K環境特有のCステート電圧・クロック変動。
学習中の**不定期な完全フリーズと再起動**という物理的障害。



意図的にプロセスをkillするスモークテストを繰り返し、ハードウェアの死に一切依存しない完全な堅牢性をDLパイプラインに付与。

オフライン提出パイプライン：究極のスキュー回避設計

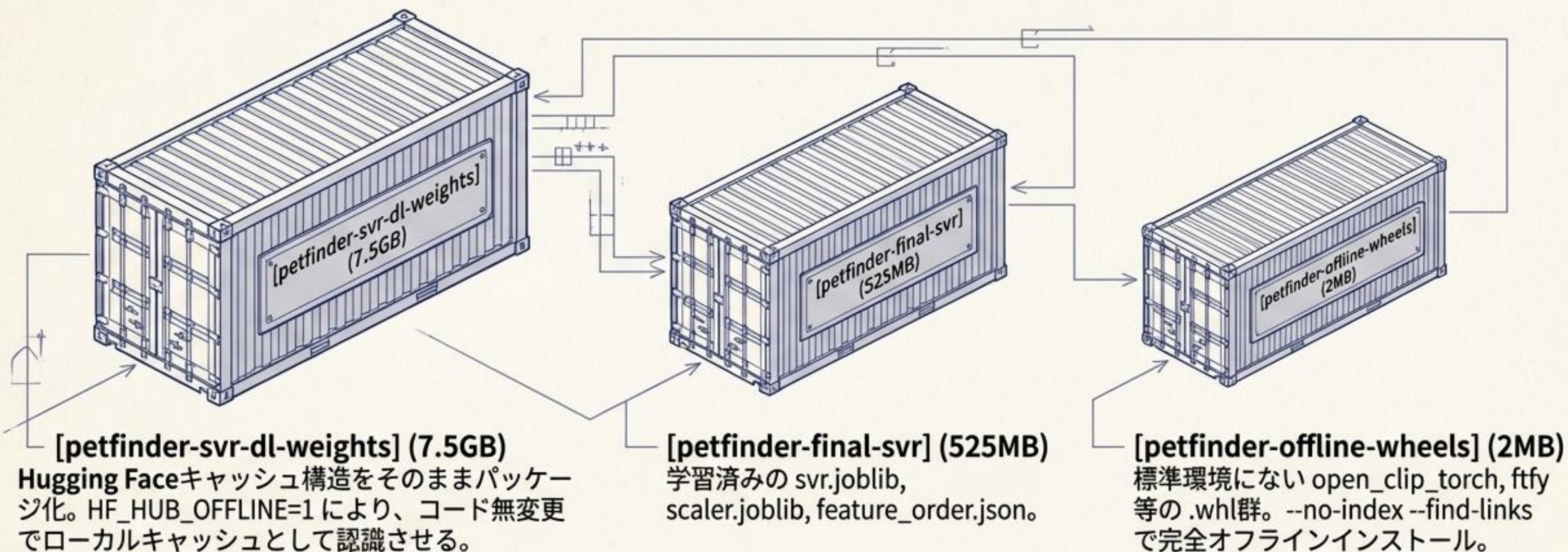


Anti-Skew Design:

ローカルとKaggleで「関数A」を2回走らせる。学習時と推論時の特徴量が完全に同じ処理経路(同一モデル・同一前処理)を通ることを構造的に保証し、前処理の食い違い(スキュー)を完全に排除。

ネットワーク遮断を突破するパッケージング戦略

Kaggleの完全オフライン採点環境へ、必要な全てをDatasetとして密輸する。



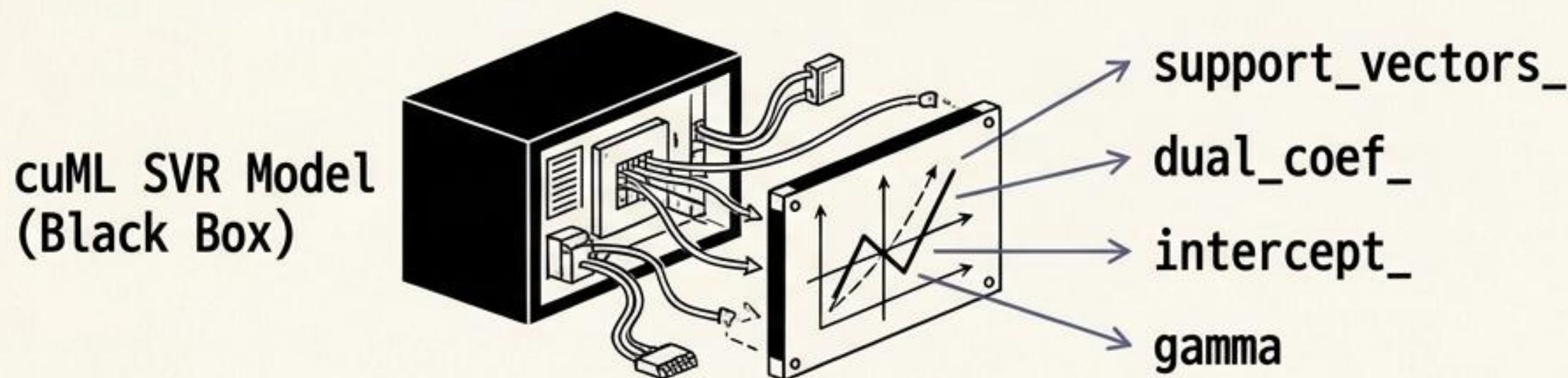
Path Resolution Hack: Kaggleのマウントパス規則変動に対し、決め打ちを避け find_input_dir() (再帰的glob検索) を実装。動的パス解決により環境差異を吸収。

提出環境デバッグ (1) : インフラの死闘

インフラの罨 (Symptom) ⚠	根本原因 (Root Cause)	突破口 (Bypass) 
Dataset更新が81%で無限停止	7.5GBの巨大Datasetへの微小な差分(wheels)追加によるKaggle側スタック。	Wheelsを2MBの別Datasetとして完全分離。更新頻度による設計の分離。
P100 GPU 'no kernel image'	割り当てGPU(Pascal P100) のCompute Capability不足により演算完全失敗。	モデル特徴抽出をtry-except で囲み、CPUへの自動フォールバックを実装。
GPU T4x2がCommitに反映されない	Web UIのSession Optionsのバグ。Save & RunがP100に強制固定される。	ライブセッションでT4x2を直接起動・確認後、そこから直接Save Versionを実行。

提出環境デバッグ (2) : cuMLサイレントバグと最強の数式バイパス

Fatal Error: cuMLの SVR.predict() が例外を出さず、全サンプルに対しサイレントに「0.0」を返すバグに遭遇(GPUカーネル起動失敗時のゼロ初期化バッファの返却)。



1. Extract Parameters:

support_vectors_, dual_coef_, intercept_, gamma を抽出。

2. NumPy Math Implementation:

距離計算: $\text{cdist}(X, \text{support_vectors}, 'sqeuclidean')$
RBFカーネル: $K = \text{np.exp}(-\text{gamma} * \text{distances})$
予測値: $(K @ \text{dual_coef.T}) + \text{intercept}$

3. Result:

cuMLのpredict()と
5~6桁の精度で完全一致。
恒久的に堅牢なCPU推論経路が完成。



The Blend : SVR × Deep Learning の統合

[RAPIDS SVR] (17.1410)

Weight: 0.93

+

= [Final Ensemble CV: 17.1306]

[DL beit_large] (18.6791)

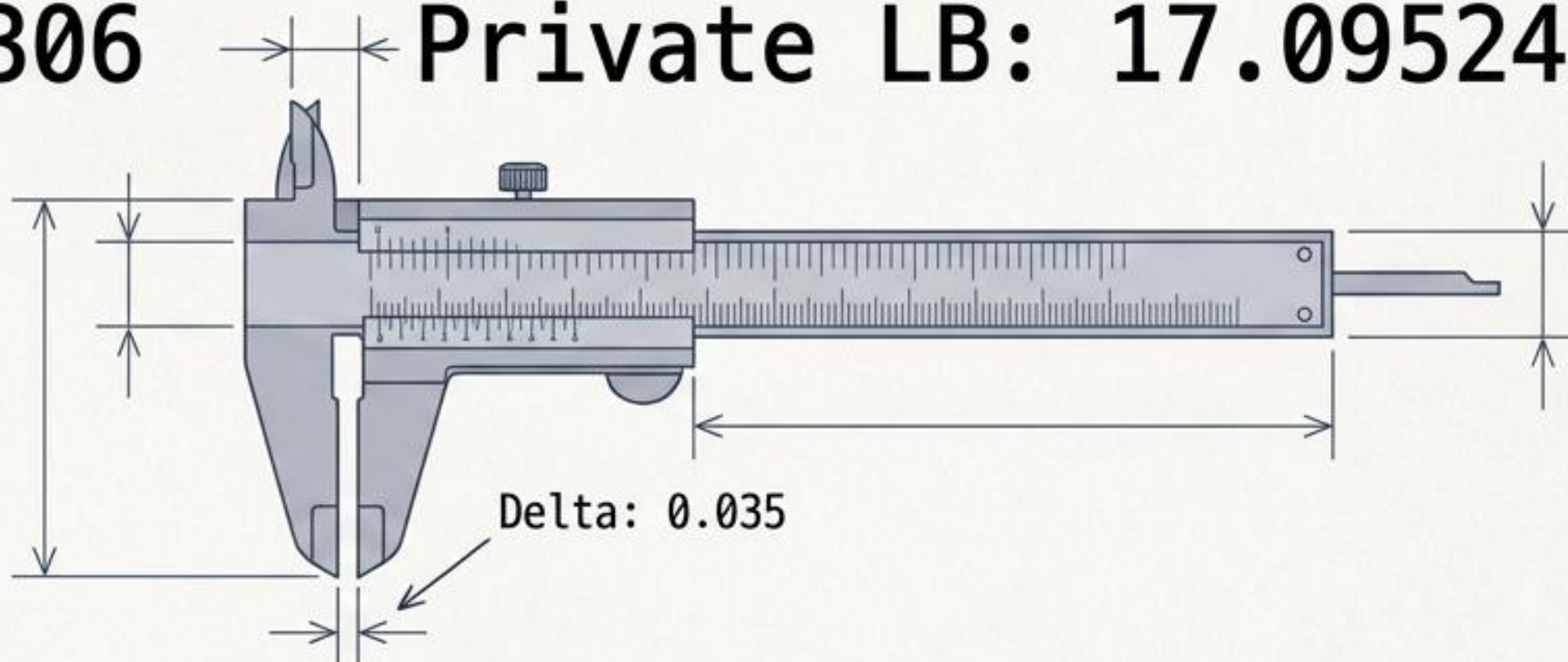
Weight: 0.07

「DL単体の精度がSVRに及ばないため、最適化の結果ブレンド寄与率は7%に留まった(+0.0104の改善)。しかしこれは、戦略レポートで分析した「1位解法(SVR主体)がDLアンサンブルを凌駕する」というコンペ全体のパラダイムと完全に一致する結果である。」

最終結果：証明された「Trust Your CV」

Local CV: 17.1306

Private LB: 17.09524



数百のモデルや16-48エポックを回した1位解法(16.8225)に対し、計算リソースの制約からSVRを9モデル、DLを1-fold/5epochに大幅スケールダウンした。にもかかわらず、その差はわずか0.27ポイント。厳格な重複除去(pHash + SGKF)によるアーキテクチャは完璧に機能し、コンペにおける至上命題「Trust Your CV」を見事に証明した。

Uncharted Territory：未開拓の伸びしろと次なる戦略

今回EDAで示唆されながら、実装を見送った最有力候補たち：

