

Adaptive Train-Test Partitioning (ATTP) Framework

Comprehensive Reference Architecture & Technical Specification | Version 2.4.0

1. EXECUTIVE SUMMARY & CONCEPTUAL OVERVIEW

Traditional machine learning workflows treat data partitioning as an unoptimized heuristic—applying static random sampling or standard class stratification. This introduces subtle covariate shift, variance inflation, and target distribution distortion between training and evaluation splits. The **Adaptive Train-Test Partitioning (ATTP)** framework redefines data partitioning as an explicit, multi-objective optimization problem over latent feature manifolds and continuous target spaces.

ATTP constructs joint class-cluster or continuous target-density strata, evaluates allocation candidate ratios, and verifies statistical parity using the Multivariate Central Limit Theorem (M-CLT) and Monte Carlo permutations. Across 21 specialized algorithms, ATTP guarantees strict distribution preservation for Classification, Regression, Time-Series, Active Learning, and Unsupervised Clustering domains.

2. MASTER FRAMEWORK TAXONOMY & PEP 8 IMPORT HIERARCHY

```
# =====
# ADAPTIVE TRAIN-TEST PARTITIONING (ATTP) MASTER IMPORT HIERARCHY
# =====

# 1. Classification Domain
from attp import (
    s_attp_clf,          # Statistical Baseline Classification ATTP
    m_attp_clf,          # Model-Based Flexible Classification ATTP
    b_attp_clf,          # Balanced Imbalanced-Target Classification ATTP
    mc_mcls_attp_clf,    # Monte Carlo Multivariate Central Limit Splitter (Clf)
    auto_attp_al_clf,    # Active Learning Pool Partitioning Engine (Clf)
    auto_attp_clf        # Automated Classification Meta-Engine
)

# 2. Regression Domain
from attp import (
    s_attp_reg,          # Statistical Binned-Target Regression ATTP
    m_attp_reg,          # Model-Based PLS/GMM Regression ATTP
    r_t_attp,            # Target-Aware Moment Matching ATTP
    r_f_attp,            # Feature-Aware Covariate Matching ATTP
    r_tf_attp,           # Joint Target + Feature Z-Space ATTP
    r_ra_attp,           # Residual-Aware Prediction Error ATTP
    r_d_attp,            # Distribution-Aware Non-Parametric MMD ATTP
    mc_mcls_attp_reg,    # Monte Carlo Multivariate Central Limit Splitter (Reg)
    auto_attp_al_reg,    # Active Learning Pool Partitioning Engine (Reg)
    auto_attp_reg        # Automated Regression Meta-Engine
)

# 3. Time Series & Clustering Domains
from attp import (
    s_attp_ts,           # Stratified Temporal Block ATTP
    m_attp_ts,           # Multivariate Lagged Time-Series ATTP
    auto_attp_ts,        # Automated Temporal Partitioning Engine
    s_attp_cluster,      # Density-Preserving Stratified Clustering ATTP
    m_attp_cluster       # Topology-Preserving Multivariate Clustering ATTP
)
```

Core Architectural Guarantees

PRODUCTION READY

- Distribution Parity:** Zero statistically significant covariate shift (M-CLT $p > 0.05$) across splits.
- Strict Isolation:** Resampling (e.g., SMOTE) is applied strictly within training folds, protecting test sets.
- Deterministic Reproducibility:** Fully seeded stochastic processes across clustering and permutations.

3. CLASSIFICATION DOMAIN SPECIFICATIONS

3.1 s_attp_clf – Statistical Classification Baseline

CLASSIFICATION

Purpose & Novelty: Standard baseline using Linear Discriminant Analysis (LDA) and K-Means to create joint categorical-structural strata ($Y \otimes C$), optimizing split ratios automatically.

$$Z = W_{LDA}^T X, \quad C_i = \operatorname{argmin}_k ||z_i - \mu_k||^2, \quad S_i = Y_i \otimes C_i$$

```
from attp import s_attp_clf
res = s_attp_clf(Input_X_df=X, Input_y_df=y, Splits=[0.6, 0.7, 0.8], Min_clusters=2, Max_Clusters=10, accuracy_model=None, metric="accuracy", random_state=42)
```

Parameter	Type	Default	Description
Input_X_df / Input_y_df	DataFrame / Series	Required	Feature matrix ($N \times p$) and target labels ($N \times 1$).
Splits	List[float]	[0.6, 0.7, 0.8]	Candidate training set allocation ratios evaluated during optimization.
Min_clusters / Max_Clusters	int	2 / 10	Search grid bounds for latent feature space k-Means segmentation.

3.2 m_attp_clf – Model-Based Flexible Classification ATTP

CLASSIFICATION

Purpose & Novelty: Grants full control over dimension reduction (LDA, PCA, UMAP), clustering (K-Means, GMM), and estimation models for specialized complex domains.

```
from attp import m_attp_clf
from sklearn.ensemble import RandomForestClassifier
res = m_attp_clf(Input_X_df=X, Input_y_df=y, DR_method="pca", clustering_method="gmm", accuracy_model=RandomForestClassifier(n_estimators=300), metric="macro_f1")
```

Parameter	Type	Default	Description
DR_method	str	"lda"	Dimension reduction pipeline: "lda", "pca", "umap", or "none".
clustering_method	str	"kmeans"	Latent segmentation algorithm: "kmeans" or "gmm".
accuracy_model	Estimator	LDA()	Classifier used to evaluate partition candidate generalization.

3.3 b_attp_clf – Balanced Classification ATTP

IMBALANCED CLF

Purpose & Novelty: Designed for highly imbalanced targets. Performs joint strata splitting, then applies resampling strictly to the training fold, keeping test sets uncorrupted.

$$D_{train}^{balanced} = R(D_{train}), \quad D_{test}^{original} = D_{test} \quad \text{where } R \in \{SMOTE, ADASYN, RandomUnder\}$$

```
from attp import b_attp_clf
res = b_attp_clf(Input_X_df=X, Input_y_df=y, balance_method="smote", validation_size=0.15, metric="balanced_accuracy")
```

4. ADVANCED DIAGNOSTIC & REGRESSION VARIANTS

3.4 mc_mcls_attp_clf — Monte Carlo Central Limit Diagnostic

DIAGNOSTIC

Purpose: Validates partition representative quality via Hotelling's T^2 multivariate CLT and 2,000 Monte Carlo permutations, confirming $p > 0.05$.

$$T^2 = n_{test} (\bar{x}_{test} - \mu)^T S_{pooled}^{-1} (\bar{x}_{test} - \mu) \sim \chi^2_p, \quad p_{MC} = (1 + \sum I(D_b \geq D_{obs})) / (B + 1)$$

```
from attp import mc_mcls_attp_clf
val = mc_mcls_attp_clf(X_full=X, y_full=y, X_test=X_te, y_test=y_te, n_permutations=2000)
p_mclt, p_mc = val.run_diagnostic()
```

3.5 & 3.6 Active Learning & Automated Meta-Engines

AUTOML

auto_attp_al_clf: Selects cold-start seed splits without labels and balances query batches against test sets using entropy-density scoring: $x^* = \arg\max [\alpha H(Y|x) + (1-\alpha) D_M(x, \mu)]$.

auto_attp_clf: Meta-engine that analyzes dataset geometry, selects the optimal ATTP variant, tunes cut-points, and emits a Pareto diagnostic report.

```
from attp import auto_attp_clf
X_tr, y_tr, X_te, y_te, report = auto_attp_clf(X, y, test_size=0.20, optimize_criterion="pareto")
```

5. REGRESSION DOMAIN SPECIFICATIONS

5.1 s_attp_reg — Statistical Binned-Target Regression ATTP

REGRESSION

Purpose & Novelty: Baseline regression engine converting continuous response $y \in \mathbb{R}$ into quantile bins $B(Y)$, combined with PCA clusters $C(X)$ into joint strata $S = B(Y) \otimes C(X)$.

```
from attp import s_attp_reg
res = s_attp_reg(Input_X_df=X, Input_y_df=y, Splits=[0.6, 0.7, 0.8], Target_Bins=10, metric="r2")
```

5.2 m_attp_reg & 5.3 r_t_attp — PLS Model-Based & Target-Aware ATTP

REGRESSION

m_attp_reg: Employs Partial Least Squares (PLS) regression to project joint X-y covariance before GMM clustering.

r_t_attp: Isolates continuous target moment matching $y \in \mathbb{R}$ via high-resolution quantile density stratification, ideal for sparse feature regimes.

```
from attp import r_t_attp
res = r_t_attp(Input_X_df=X, Input_y_df=y, Target_Bins=15, metric="rmse")
```

5. SPECIALIZED REGRESSION, TEMPORAL & CLUSTERING VARIANTS

5.4 r_f_attp & 5.5 r_tf_attp – Feature & Joint Z-Space ATTP

COVARIATE MATCHING

r_f_attp: Performs unsupervised covariate matching over $X \in \mathbb{R}^p$ ignoring y , forming an essential baseline for domain adaptation tests.

r_tf_attp: Performs joint optimization over the augmented manifold $Z = [X \ y] \in \mathbb{R}^n \times (p+1)$, preserving non-linear feature-target dependencies.

```
from attp import r_tf_attp
res = r_tf_attp(Input_X_df=X, Input_y_df=y, Target_Bins=10, Min_clusters=3, DR_method="pls")
```

5.6 r_ra_attp & 5.7 r_d_attp – Residual & Distribution MMD ATTP

ADVANCED REG

r_ra_attp: Incorporates Out-Of-Fold (OOF) prediction errors $e_i = y_i - \hat{y}_i^{\text{OOF}}$ to balance easy, hard, and heteroscedastic error regions equally across splits.

r_d_attp: Non-parametric variant directly minimizing Maximum Mean Discrepancy (MMD) over kernel space:

$$MMD^2(D_{tr}, D_{te}) = E[k(z, z')] - 2E[k(z, w)] + E[k(w, w')]$$

6. Time Series Domain – s_attp_ts, m_attp_ts & auto_attp_ts

TEMPORAL

Purpose & Mechanics: Preserves chronological ordering ($t_1 < t_2 < \dots < t_n$) while applying blocked seasonal/regime stratification to eliminate temporal lookahead leakage.

```
from attp import s_attp_ts, auto_attp_ts
res = s_attp_ts(Input_X_df=X_ts, Input_y_df=y_ts, n_splits=5, block_size="monthly")
X_tr, y_tr, X_te, y_te, cfg = auto_attp_ts(X_ts, y_ts, max_lag=12)
```

7. Unsupervised Clustering – s_attp_cluster & m_attp_cluster

UNSUPERVISED

Purpose & Mechanics: Purely unsupervised sampling modules that preserve density and high-dimensional spatial topology when pulling representative sub-datasets.

```
from attp import m_attp_cluster
sampled_df = m_attp_cluster(Input_X_df=X_raw, sample_fraction=0.30, cluster_method="gmm")
```

8. MASTER ARCHITECTURAL COMPARISON TABLE

Complete comparison across all 21 ATTP core variants, their objective matrices, mathematical criteria, and primary application domains:

Domain	Variant Name	Objective Matrix	Mathematical Criterion	Primary Application
Classification	s_attp_clf	$X \in \mathbb{R}^{n \times p}, Y \in \{1..C\}$	LDA Latent Space + K-Means Strata	Standard tabular classification baseline
	m_attp_clf	Flexible X, Y	Custom DR + Custom Estimator Score	Complex features (PCA/UMAP/GMM)
	b_attp_clf	Imbalanced Y	Resampled $R(D_{tr})$; Pure Test D_{te}	Severe class imbalance (SMOTE/ADASYN)
	mc_mcls_attp_clf	Diagnostic Split	Hotelling's T^2 + Monte Carlo p_{MC}	Verification of partition representative quality
	auto_attp_al_clf	Unlabeled Pool D_{pool}	Uncertainty $H(Y x)$ + Density D_M	Active learning seed & query management
	auto_attp_clf	Automated X, Y	Multi-Objective Pareto Optimization	Production AutoML deployment pipelines
Regression	s_attp_reg	$X \in \mathbb{R}^{n \times p}, y \in \mathbb{R}$	Quantile Bins $B(Y) \otimes$ PCA Cluster $C(X)$	Standard continuous target baseline
	m_attp_reg	Flexible X, y	PLS Latent Scores + GMM Clustering	Multivariate continuous response tasks
	r_t_attp	Target $y \in \mathbb{R}$	Continuous Quantile Density $B(Y)$	Sparse/high-dimensional feature spaces
	r_f_attp	Covariates $X \in \mathbb{R}^p$	Unsupervised Feature Centroid Match	Domain adaptation & covariate shift tests
	r_tf_attp	Joint $Z = [X \ y]$	Augmented Z-Space Cluster Matching	Complex non-linear feature-target correlation
	r_ra_attp	Errors $e = y - \hat{y}^{OOF}$	OOF Residual Stratification $C(E)$	Heteroscedastic / non-uniform error data
	r_d_attp	Density $P(X, y)$	Kernel Maximum Mean Discrepancy (MMD)	Non-parametric global density preservation
	mc_mcls_attp_reg	Diagnostic Split	Augmented Z-Space M-CLT T^2 Test	Regression split statistical verification
	auto_attp_al_reg	Unlabeled Pool D_{pool}	Variance Informativeness + Density	Active regression labeling pipelines
	auto_attp_reg	Automated X, y	Multi-Objective Pareto Optimization	Automated continuous target partitioning
Time Series	s_attp_ts	Sequential X_t, y_t	Temporal Block Stratification	Stationary seasonal time-series tasks
	m_attp_ts	Multivariate Lagged	Lagged Rolling Latent-Space Segmentation	Non-stationary multivariate time series
	auto_attp_ts	Automated X_t	Expanding Window Block Optimization	Automated temporal model validation
Clustering	s_attp_cluster	Unsupervised X	Quantile Density Stratified Sampling	Fast density-preserving subsampling
	m_attp_cluster	High-Dimensional X	Spatial GMM Topology Preservation	Topology-aware representative sampling

