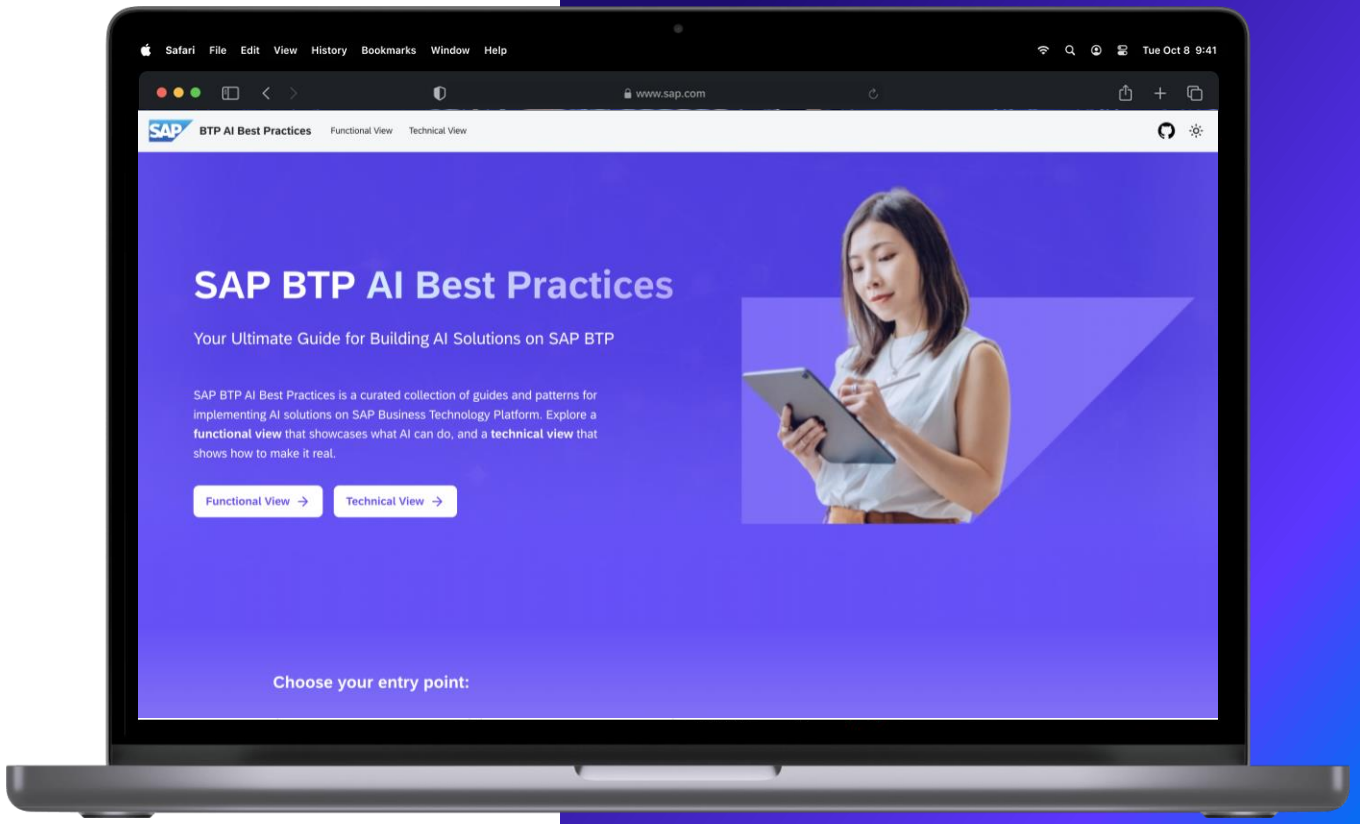
SAP BTP AI Best Practices

Anomaly Detection

Detect **unexpected patterns** and prevent costly failures with SAP BTP AI



BTP AI Services Center of Excellence

30.07.2025

Steps

1 Overview

2 Pre-requisites

3 Key Choices and Guidelines

4 Implementation

Anomaly Detection

Simple Identification of Anomalies

Anomaly detection is the process of identifying data points, events, or patterns that **deviate significantly from the expected or normal behavior** within a dataset through **SAP Hana ML**

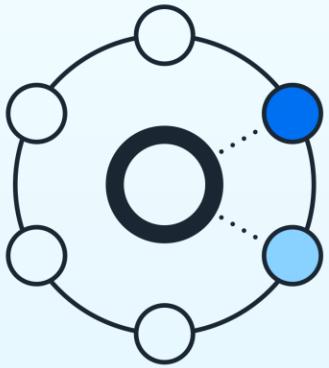
In the SAP ecosystem, this involves leveraging tools within SAP HANA ML (PAL, *hana-ml*) to find unusual patterns

Expected Outcome

- Successfully identify and flag unusual behavior or outliers in various types of data.
- Enable proactive responses to potential risks through early detection of fraud, system failures, or compliance violations.
- Optimize business processes by identifying operational inefficiencies and understanding unexpected variations.

Key Benefits

Why use SAP HANA ML for Anomaly Detection?



Algorithm Interchangeability

Easily switch between algorithms (DBSCAN, Isolation Forest, One-Class SVM, K-Means) to best suit your task.

Out-of-the-box Features

Supercharge your development with built-in capabilities for time series anomaly detection, clustering, and distance-based outlier scoring.

Security & SAP Ecosystem

It’s fully integrated into the SAP Ecosystem, leveraging the best of SAP technologies.

Pre-requisites

Supported Environments

- SAP HANA Platform 2.0 SPS 04 or higher
- SAP Hana Cloud (recommended for easier management)
- SAP Datasphere
- SAP Hana express edition (for development and testing)

Required Components

- Application Function Library (AFL) containing PAL and APL
- Script Server enabled for ML algorithms
- Required user authorizations and roles for PAL/APL

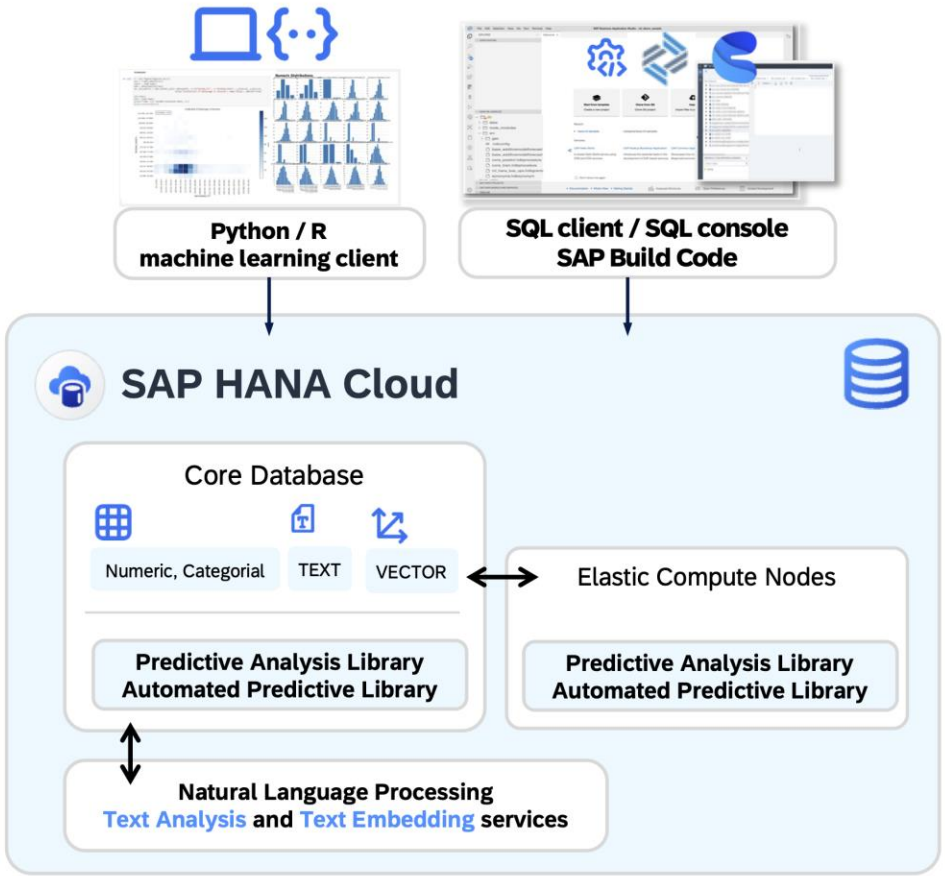
SAP HANA Platform

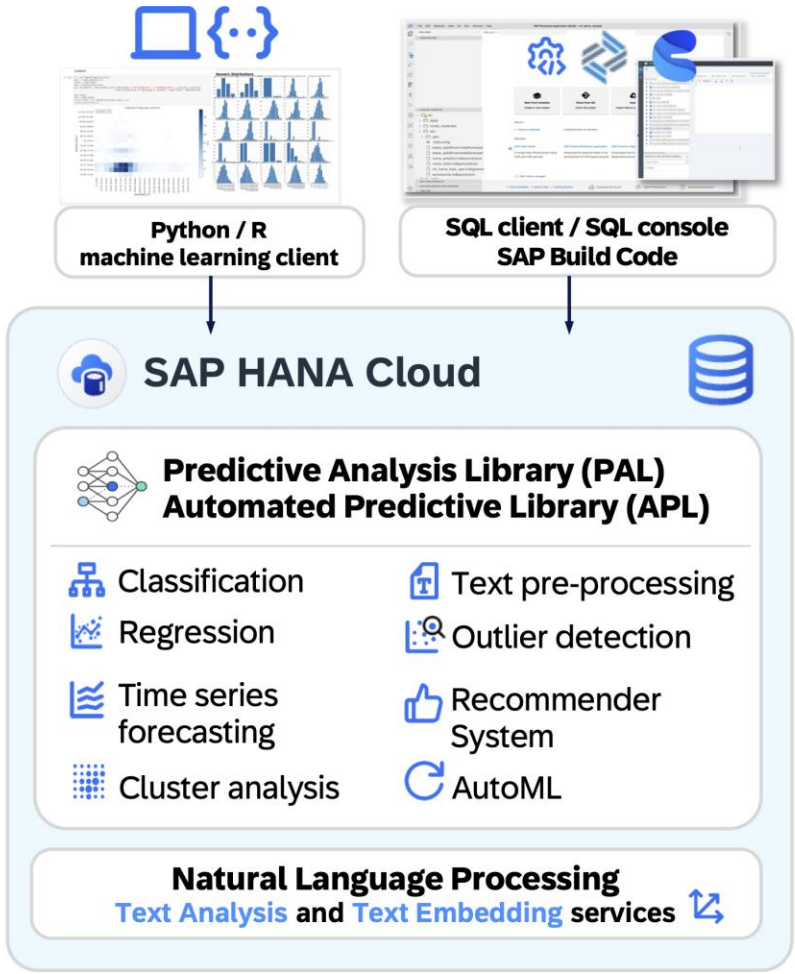
- SAP HANA is an in-memory database that enables real-time analytics and applications
- The HANA ML libraries (PAL, APL) provide native in-database functions for predictive analysis and Machine Learning.

Script Server

- Auxiliary SAP HANA process responsible for executing application function libraries
- Critical component that must be enabled for PAL and APL functionality
- Serves as execution environment for Machine Learning procedures

High-level reference architecture





Key Choices and Guidelines

Decisions that impact the performance and utility of the application

Algorithm Selection

- The choice of algorithm is critical to detection accuracy, computational efficiency, and the type of anomalies you want to find.
- Different algorithms excel in different scenarios:

General anomalies

- **DBSCAN**: For irregular, non-convex shapes and explicit noise labeling
  - *Example*: For vessel GPS tracking, detecting illegal fishing outside designated zones
- **Isolation Forest**: For high-dimensional data and isolated anomalies
  - *Example*: Credit card fraud detection with hundreds of transaction variables
- **One-Class SVM**: For complex decision boundaries and moderate-sized datasets
  - *Example*: Network intrusion detection trained on months of clean traffic
- **K-Means:** For well-formed clusters and when explanation clarity matters
  - *Example*: Call center metrics analysis where outliers indicating agents requiring coaching

Time Series Anomalies

- **Time Series Anomaly Detection**: For seasonal data with pronounced trends
  - *Example*: For retail demand with 7-day cycles, detect unusual spikes

Regression Models Anomalies

- **Regression Outlier Detection:** To detect outliers in regression models
  - *Example*: Real estate pricing models identify properties with abnormal prices based on features like size and location, flagging potential data errors or unique market conditions.

Key Choices and Guidelines

Decisions that impact the performance and utility of the application

Parameter Optimization

- Picking correct values for algorithm-specific parameters is crucial

DBSCAN Parameters:

- **minPts**: Higher values create more robust clusters but may miss outliers
- **eps**: Controls density connectivity; critical for defining what's "close”

Isolation Forest Parameters:

- **n_estimators**: More trees increase accuracy but with diminishing returns
- **contamination**: Affects sensitivity to outliers

One-Class SVM Parameters

- **kernel**: RBF or polynomial kernels create effective non-linear boundaries
- **nu:** controls the upper bound on the fraction of training errors

Time Series Parameters:

- **window_size**: Affects smoothing and sensitivity to recent changes
- **seasonality detection**: Critical for data with cyclic patterns

Key Choices and Guidelines

Decisions that impact the performance and utility of the application

Data Preparation

Clear Null Policy

- Define a strategy for **handling missing values** (e.g., imputation, deletion).
- **Why**: Essential as many PAL procedures abort with Nulls, ensures data integrity
- **HANA ML Tool**: Use the Imputer (PAL) for various imputation strategies

Feature Importance Analysis

- Identify and select the **most relevant features** to distinguish anomalies
- **Why**: Reduces noise, combats the “curse of dimensionality”, improves model accuracy and interpretability
- **HANA ML Tool**: Use *FeatureSelection* or *permutation_importance* (PAL)

Dimensionality Reduction

- **Reduce features** using techniques like PCA while preserving key data variance
- **Why**: Reduces noise, combats the “curse of dimensionality”, improves model accuracy
- **HANA ML Tool**: Use *PCA* (PAL) for principal component analysis

Key Choices and Guidelines

Decisions that impact the performance and utility of the application

Model Evaluation

Labeled Anomalies

- **Confusion Matrix**: TP, TN, FP, FN breakdown
- **Precision, Recall, F1-Score**: Balance false alarms and missed anomalies
- **ROC AUC**: Discriminative power across thresholds

Unsupervised Evaluation

- **Domain Expert Validation**: Review high-scoring anomalies
- **Score Distribution Analysis**: Examine breaks/clusters in anomaly scores
- **Interpretability**: Understand *why* points are flagged
- **Cluster Validation**: Using Silhouette Score for clustering-based methods

Time Series Specific Evaluation

- **Visual Inspection**: Plot series and flagged anomalies
- **Residual Review**: Check fit for models like *OutlierDetectionTS*
- **Labelled Metrics** (if available): apply standard metrics from Labeled Anomalies analysis

Implementation

Programming Model Selection Guidelines

Data Science Workflows

Utilize the **Python** hana-ml library for a streamlined, intuitive experience aligned with standard data science practices, including convenient data manipulation and integration with machine learning workflows

Alternative Approaches

Use **SQLScript** to directly call PAL procedures when tight integration with SAP HANA artifacts is needed.

Python

SDK

- Hana_ml

Reference Code

- SAP BTP AI Best Practices - Sample Code

Learning Journeys

- Learning from Labeled Anomalies for Efficient Anomaly Detection using Python Machine Learning Client for SAP HANA
- Outlier Detection with One-class Classification using Python Machine Learning Client for SAP HANA

SQLScript

SDK

- SAP HANA Predictive Analysis Library (PAL)

Learning Journeys

- SAP HANA PAL quick start
- SAP HANA PAL – K-Means Algorithm or How to do Customer Segmentation for the Telecommunications Industry

Code Sample

Python – Isolation Forest

```python
from hana_ml.algorithms.pal.preprocessing import IsolationForest

# --- Training the Isolation Forest Model ---
iforest = IsolationForest(
    n_estimators=None,   # let PAL pick the number of trees
    max_samples=None,    # let PAL pick samples per tree
    thread_ratio=1       # let PAL decide on compute threads
)

print("\nTraining the Isolation Forest model...")
iforest.fit(
    data=hdf_input,
    key=KEY_COL,
    features=feature_cols
)

# --- Predicting Anomalies ---
print("\nPredicting anomalies using the trained Isolation Forest model...")
results_hdf = iforest.predict(
    data=hdf_input,
    key=KEY_COL,
    features=feature_cols
)
print("Prediction completed.")
```

Contributors

RF Robledo, Francisco

PS Pacheco-Sanchez, Sergio

 Marques, Luis

Thank you