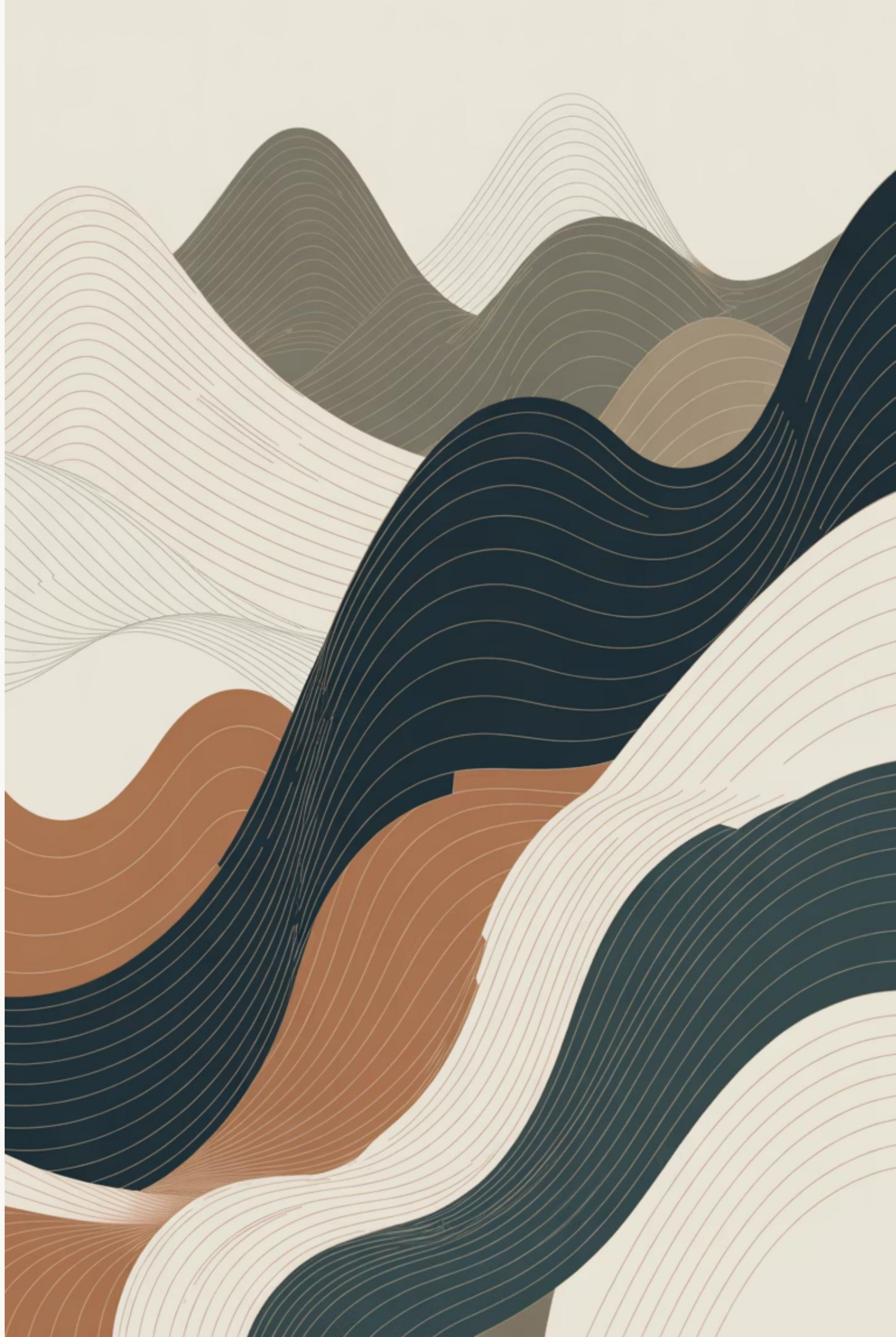
TSPred-IT: Time Series Prediction with Integrated Tuning

A comprehensive framework for automating time series forecasting through integrated preprocessing, augmentation, and model optimization

Eduardo Ogasawara
eduardo.ogasawara@cefet-rj.br
https://eic.cefet-rj.br/~eogasawara



The Challenge in Time Series Forecasting

Time series prediction remains a fundamental challenge across scientific research and industrial applications, from demand forecasting to financial modeling. While substantial progress has been made in developing sophisticated forecasting models, a critical gap persists in existing frameworks.

Current tools focus almost exclusively on model hyperparameter optimization while treating data preprocessing as a separate, manual step. This fragmented approach overlooks the complex interplay between preprocessing choices and model performance.

Critical Need

Simultaneous optimization of both preprocessing pipelines and model architectures

Current Gap

Frameworks optimize models but neglect preprocessing parameter tuning

Key Question

Which preprocessing and model combination yields optimal forecasting accuracy?

Framework Architecture: Five Integrated Modules

Preprocessing

Transformations, augmentation, normalization strategies

Modeling

Statistical, ML, and deep learning architectures

Sampling

Train-test splits and cross-validation schemes

Tuning

Hyperparameter optimization with grid and random search

Evaluation

MSE, sMAPE, AIC, BIC performance metrics

TSPred-IT orchestrates these five modules into a cohesive pipeline, enabling end-to-end automation from raw data to validated predictions. The modular design ensures extensibility while maintaining tight integration between preprocessing decisions and model selection.

Core Innovation: Co-Optimization of Preprocessing and Modeling

Seamless Integration

Unified pipeline combining preprocessing, augmentation, training, tuning, and evaluation in a single framework

- Eliminates manual workflow gaps
- Ensures reproducibility
- Reduces error-prone transitions

First-of-Its-Kind

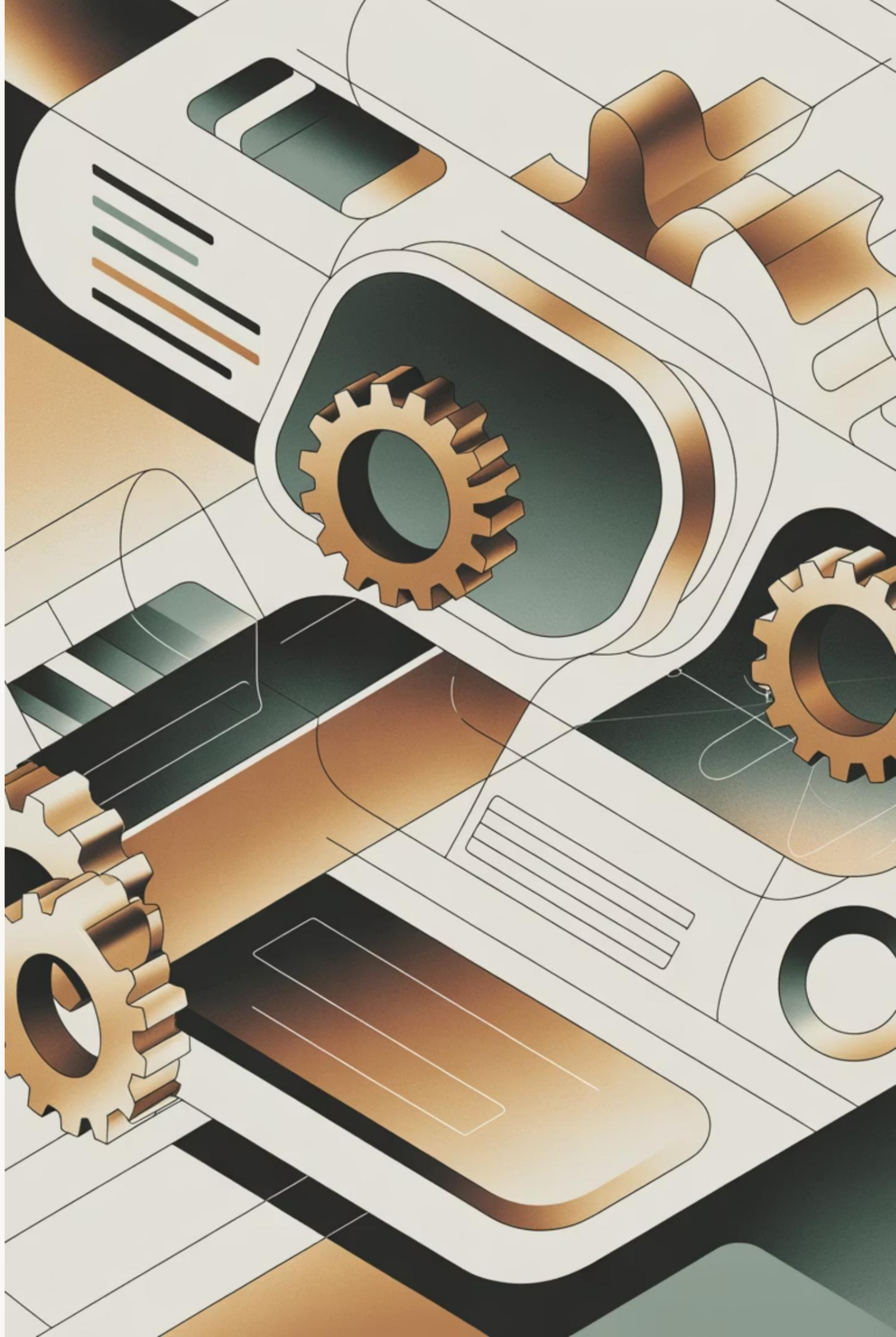
Only tool to simultaneously optimize both preprocessing parameters and model hyperparameters

- Joint search space exploration
- Captures preprocessing-model interactions
- Maximizes end-to-end performance

Extensible Design

Architecture supports custom methods and multiple programming environments

- R and Python wrappers available
- Plugin system for new algorithms
- Community contributions enabled



Comprehensive Model Library

Statistical Models

- **ARIMA:** Auto-regressive integrated moving average
- **ETS:** Error, trend, seasonal decomposition
- **Holt-Winters:** Triple exponential smoothing
- **Theta:** Decomposition-based forecasting

Classical methods with proven track records in time series analysis.

Machine Learning

- **Random Forests:** Ensemble tree-based prediction
- **SVM:** Support vector machines with kernel tricks
- **MLP:** Multi-layer perceptron networks
- **ELM:** Extreme learning machines

Flexible algorithms capturing complex nonlinear patterns.

Deep Learning

- **Conv1D:** One-dimensional convolutional networks
- **LSTM:** Long short-term memory recurrent nets

PyTorch-powered architectures for sequential dependencies and temporal features.

Tight Coupling: All models are integrated with preprocessing modules, enabling joint optimization of transformation parameters alongside model-specific hyperparameters.

Advanced Data Preprocessing & Augmentation

Transformations

Stabilize variance and remove trends:

- Logarithmic and Box-Cox transformations
- Differencing for stationarity
- Empirical Mode Decomposition (EMD)
- Wavelet decomposition

Normalization

Scale features for model compatibility:

- Min-max scaling to [0,1] range
- Adaptive normalization techniques
- Context-aware rescaling

Augmentation

Expand training data and improve robustness:

- Jitter: Add controlled noise
- Warping: Time-axis distortion
- Flipping: Temporal reversal
- Stretching: Duration modification
- Wormhole: Segment manipulation

These preprocessing strategies address fundamental time series challenges: non-stationarity, small sample sizes, and distribution shifts. By tuning preprocessing parameters alongside model hyperparameters, TSPred-IT discovers optimal data representations for each forecasting task.

Typical Workflow: From Data to Predictions

Load Dataset

Import time series data (e.g., Brazilian fertilizer consumption 1961–2020) and perform initial exploration

Create Sliding Windows

Transform univariate series into supervised learning format with configurable lag features

Select Preprocessing

Choose transformation, normalization, and augmentation methods from available options

Define Candidate Models

Specify statistical, ML, or deep learning architectures to evaluate during tuning

Run Tuning

Execute cross-validation with grid or random search across preprocessing-model combinations

Evaluate Performance

Assess final models on held-out test set using sMAPE, MSE, and information criteria

Experimental Validation: Brazilian Fertilizer Consumption

Experimental Setup

Dataset: Annual fertilizer consumption in Brazil from 1961 to 2020 (60 observations)

Preprocessing methods tested:

- MM: Min-max normalization
- DIF: First differencing
- AN: Adaptive normalization

Augmentation strategies:

- None (baseline)
- Jitter (noise injection)
- Stretch (temporal scaling)

Models compared: MLP, ELM, SVM, LSTM, Conv1D

Key Findings

Preprocessing Matters

Choice of transformation and normalization significantly impacts forecasting accuracy across all model types

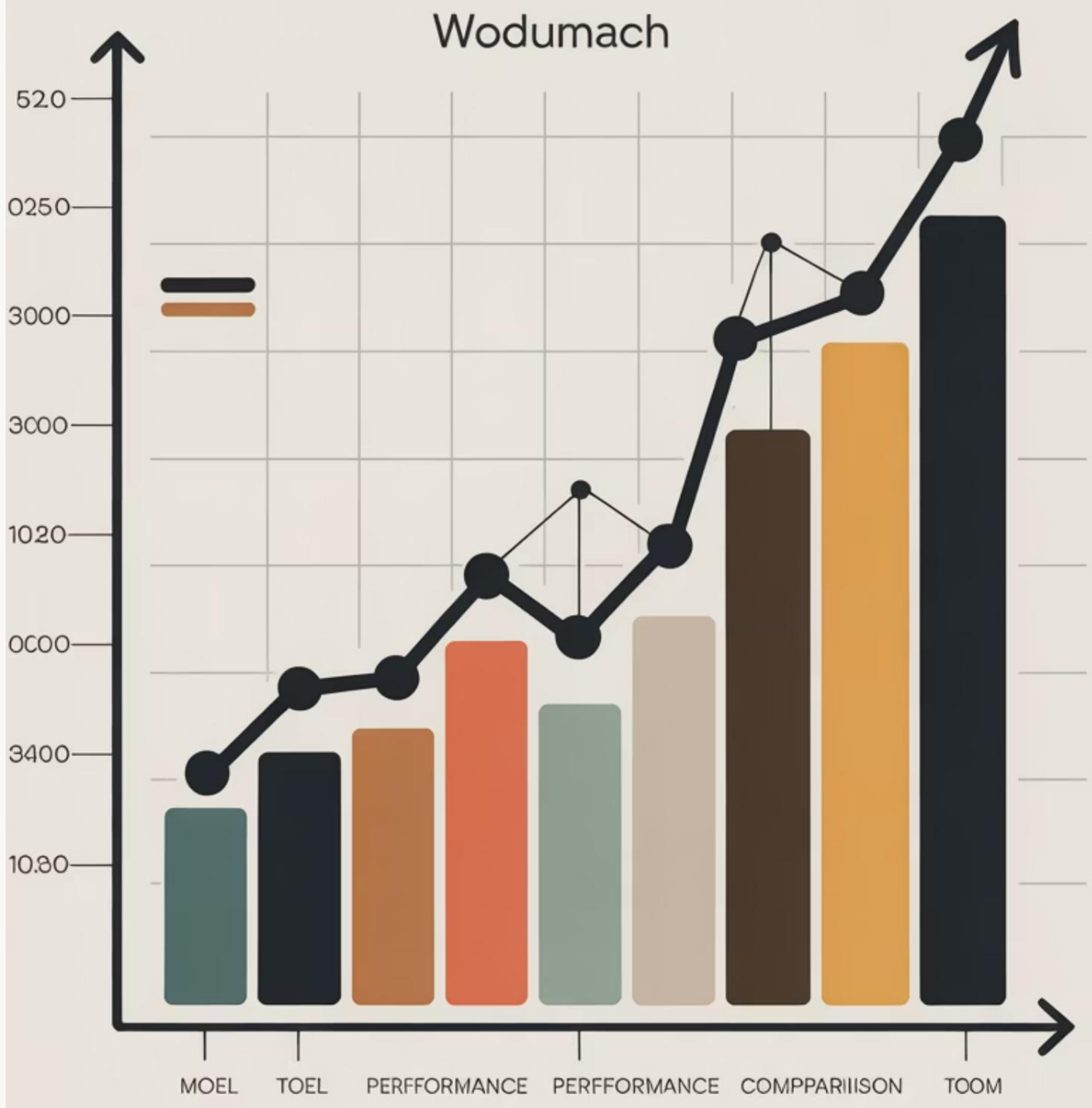
Context-Dependent Augmentation

Data augmentation helps in some configurations but can degrade performance if misapplied

LSTM Superiority

Long short-term memory networks achieved best overall performance when paired with optimal preprocessing

Results demonstrate that joint optimization of preprocessing and modeling yields substantial improvements over sequential, independent tuning of each component.



Why TSPred-IT Stands Out

Full Pipeline Automation

Eliminates manual interventions between preprocessing, training, tuning, and evaluation stages. End-to-end reproducibility ensures consistent results across experiments.

Novel Co-Optimization

First framework to treat preprocessing parameters and model hyperparameters as a unified search space, capturing critical interactions between data transformations and algorithms.

Extensible Architecture

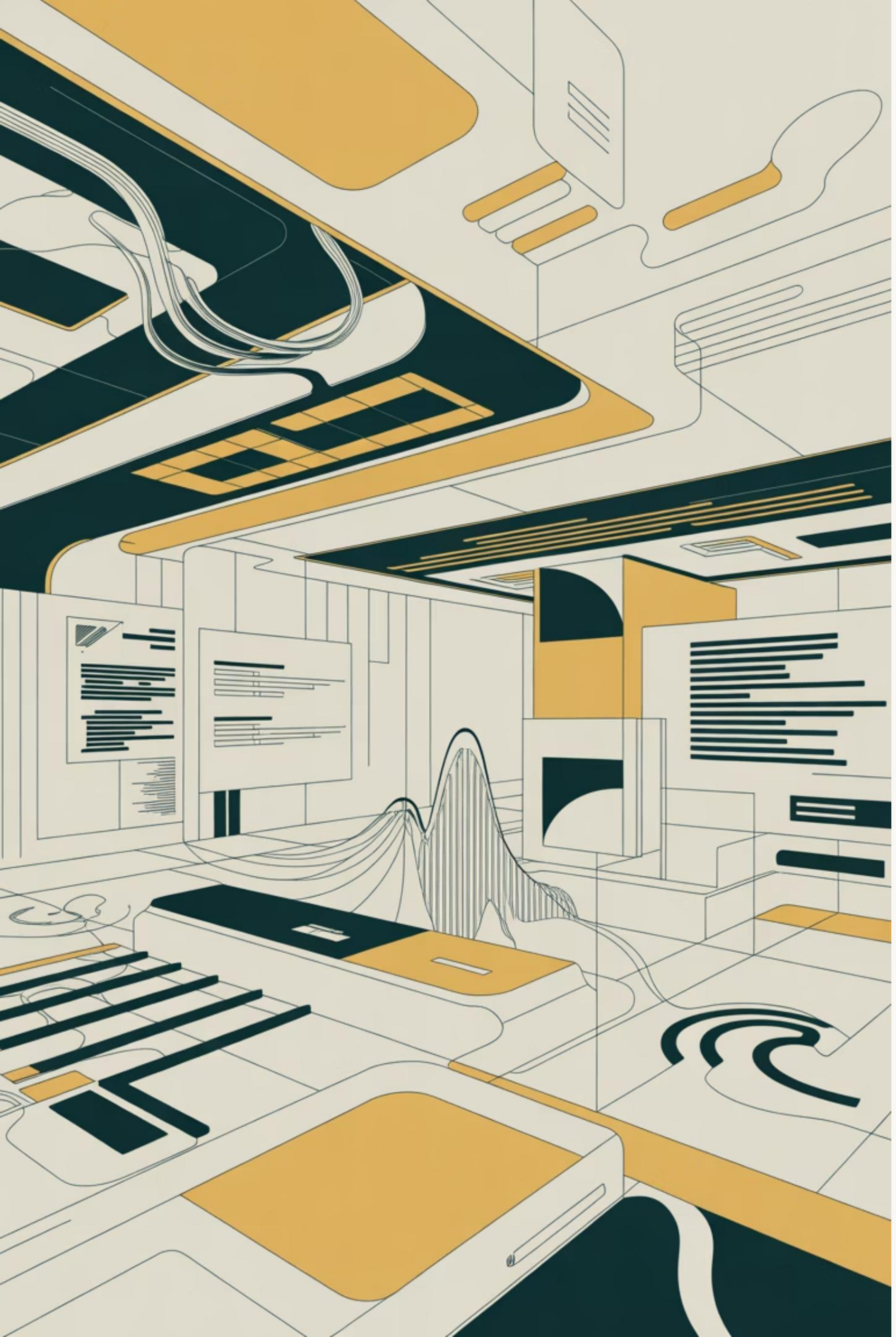
Plugin system supports custom preprocessing methods, models, and evaluation metrics. Available in both R and Python with straightforward APIs for community contributions.

Comprehensive Model Support

Unified interface for statistical classics, machine learning workhorses, and deep learning architectures. Mix and match algorithms without changing workflow.

Proven Real-World Performance

Validated on diverse datasets including economic indicators, environmental measurements, and industrial metrics. Competitive or superior accuracy compared to specialist tools.



Conclusion and Future Directions

> TSPred-IT bridges the critical gap between data preprocessing and model optimization in time series forecasting

Key Takeaways

- Integrated framework automating the full forecasting pipeline
- First tool enabling joint preprocessing-model optimization
- Extensible design supporting statistical, ML, and deep learning approaches
- Demonstrated accuracy improvements on real-world datasets
- Available on CRAN and GitHub for immediate use

Future Enhancements

- Expanded deep learning integration (Transformers, attention mechanisms)
- Advanced search strategies (Bayesian optimization, evolutionary algorithms)
- Multi-variate time series support
- Distributed computing for large-scale experiments
- AutoML capabilities for non-expert users

Access on CRAN

View on GitHub