NeutroHydro: A Python Package for Neutrosophic Chemometrics in Groundwater Analysis

Dickson Abdul-Wahab*[1] and Ebenezer Aquisman Asare†[2]

[1]University of Ghana, Ghana
[2]Nuclear Chemistry and Environmental Research Centre, National Nuclear Research Institute (NNRI), Ghana Atomic Energy Commission, Ghana

January 2, 2025

Abstract

Groundwater quality assessment relies on understanding the complex interactions between natural geogenic processes and anthropogenic perturbations. Traditional chemometric approaches often struggle to disentangle these sources due to the inherent uncertainty, ambiguity, and incomplete information in hydrogeochemical datasets. `NeutroHydro` introduces the Neutralization-Displacement Geosystem (NDG) framework with Stoichiometric Inversion, a novel neutrosophic chemometric approach that addresses these challenges by operating in absolute concentration space rather than compositional space. The package provides a mathematically rigorous workflow for decomposing groundwater chemistry into baseline (geogenic) and perturbation (anthropogenic) components, enabling quantitative source attribution and mineral inference.

Keywords: Python, groundwater, chemometrics, neutrosophic logic, hydrogeochemistry, partial least squares, water quality

1 Introduction

Groundwater chemometrics faces several persistent challenges: (1) distinguishing natural baseline chemistry from anthropogenic contamination, (2) handling uncertainty and ambiguity in concentration measurements, (3) performing statistically valid operations in non-compositional space, and (4) linking statistical patterns to physical hydrogeochemical processes through mineral stoichiometry. Existing approaches either use compositional data analysis (CoDa) which operates in log-ratio space and loses direct physical interpretability (Aitchison, 1986), or apply standard multivariate methods that fail to properly account for the inherent uncertainty structure in geochemical data (Reimann and Filzmoser, 2008).

`NeutroHydro` fills this gap by implementing a neutrosophic framework that explicitly represents each ion concentration as a triplet of truth (baseline), indeterminacy (uncertainty), and falsity (perturbation) values (Smarandache, 1998). This representation enables simultaneous modeling of multiple information channels while maintaining mathematical rigor through well-defined Euclidean space operations. The package is designed for hydrogeologists, environmental scientists, and water resource managers who need to:

*Corresponding author: dabdul-wahab@live.com; ORCID: 0000-0001-7446-5909
†ORCID: 0000-0003-1185-1479

- Quantitatively separate natural and anthropogenic contributions in groundwater samples
- Assess variable importance with channel-wise decomposition
- Infer plausible mineral sources through stoichiometric constraints
- Handle missing data and measurement uncertainty systematically
- Generate interpretable results aligned with domain knowledge

The framework has been applied to groundwater quality assessment and has demonstrated capability in identifying pollution sources, characterizing baseline water chemistry, and diagnosing hydrogeochemical processes (Abdul-Wahab and Asare, 2025).

2 Mathematical Framework

2.1 Neutrosophic Data Representation

`NeutroHydro` maps each standardized ion concentration x_{ij} (sample i, ion j) to a neutrosophic triplet (T_{ij}, I_{ij}, F_{ij}) where:

- **Truth** (T): Baseline component computed via robust operators (median, low-rank approximation, or robust PCA)
- **Indeterminacy** (I): Uncertainty/ambiguity channel quantifying measurement or epistemic uncertainty
- **Falsity** (F): Perturbation likelihood derived from standardized residuals

For the Truth channel, the baseline operator $\mathcal{B}$ is applied to the standardized predictor matrix:

$$X_T = \mathcal{B}(X^{\text{std}}) \tag{1}$$

The residuals are computed as $R = X^{\text{std}} - X_T$, and the Falsity channel uses a monotone map:

$$F_{ij} = 1 - \exp\left(-\frac{|R_{ij}|}{\sigma_j}\right) \tag{2}$$

where σ_j is the robust scale (median absolute deviation) of residuals for ion j.

2.2 Augmented Hilbert Space Regression

The three channels are combined into an augmented predictor matrix:

$$X^{\text{aug}} = [X_T \quad \sqrt{\rho_I} X_I \quad \sqrt{\rho_F} X_F] \in \mathbb{R}^{n \times 3p} \tag{3}$$

where ρ_I and ρ_F are channel weights. Elementwise precision weights derived from falsity downweight high-perturbation observations:

$$W_{ij} = \exp(-\lambda_F \cdot F_{ij}) \tag{4}$$

Probabilistic Neutrosophic PLS (PNPLS) regression is then performed on the weighted augmented matrix $\tilde{X}^{\text{aug}} = W \odot X^{\text{aug}}$ using the NIPALS algorithm (Wold, 1966; Mevik and Wehrens, 2007) to extract latent components that predict a target variable y (e.g., log total dissolved solids).

2.3 Variable Importance Decomposition

A key theoretical contribution is the **L2 decomposition theorem** for Variable Importance in Projection (VIP). For each ion j, the aggregate VIP satisfies:

$$VIP^2_{\text{agg}}(j) = VIP^2_T(j) + VIP^2_I(j) + VIP^2_F(j) \tag{5}$$

This additive decomposition enables unambiguous attribution of prediction importance to baseline versus perturbation sources. The baseline fraction $\pi_G(j)$ for each ion is:

$$\pi_G(j) = \frac{VIP^2_T(j)}{VIP^2_T(j) + VIP^2_I(j) + VIP^2_F(j)} \in [0, 1] \tag{6}$$

Ions with $\pi_G(j) \geq 0.7$ are classified as baseline-dominant (geogenic), while $\pi_G(j) \leq 0.3$ indicates perturbation-dominant (anthropogenic).

2.4 Stoichiometric Mineral Inference

To link statistical patterns to physical processes, `NeutroHydro` implements stoichiometric inversion. Given ion concentrations $c \in \mathbb{R}^m$ in meq/L and a stoichiometric matrix $A \in \mathbb{R}^{m \times K}$ representing K candidate minerals, the weighted non-negative least squares (NNLS) problem is:

$$\hat{s} = \arg\min_{s \geq 0} \|D(c - As)\|_2^2 \tag{7}$$

where $D = \text{diag}(\pi_G^\eta)$ prioritizes baseline ions in the fit. The solution $\hat{s}$ represents plausible mineral contributions, validated through residual norms and contribution thresholds. The package includes a comprehensive mineral library and supports custom mineral definitions.

3 Features

`NeutroHydro` provides a complete pipeline implementation:

1. **Preprocessing**: Robust centering and scaling using median and median absolute deviation (MAD) to resist outliers
2. **NDG Encoder**: Multiple baseline operators (global median, hydrofacies-conditioned median, low-rank SVD, robust PCA)
3. **PNPLS Regression**: Augmented space regression with configurable channel weights and precision weighting
4. **NVIP Computation**: Channel-wise variable importance with L2 decomposition
5. **Attribution Analysis**: Ion-level baseline fractions (π_G) and sample-level attribution (G_i)
6. **Mineral Inference**: Weighted NNLS inversion with plausibility assessment, thermodynamic validation via saturation indices, and diagnostic indices (Simpson Ratio, Base Exchange Index, Chloro-Alkaline Indices)
7. **Visualization**: Gibbs diagrams, VIP decomposition plots, mineral fraction charts, and correlation matrices

8. **Quality Assessment**: WHO guideline compliance, redox zonation, and pollution fingerprinting

The package operates in absolute concentration space (mg/L, meq/L) rather than compositional space, preserving physical interpretability and enabling direct application of stoichiometric constraints. All operations occur in well-defined Euclidean spaces with rigorous mathematical guarantees.

4 Example Usage

The following code demonstrates the basic workflow:

```python
import numpy as np
from neutrohydro import NeutroHydroPipeline

# Prepare data: ion concentrations (mg/L or meq/L) and target
X = ...  # Shape: (n_samples, n_ions)
y = ...  # Target: e.g., log TDS
ion_names = ["Ca2+", "Mg2+", "Na+", "K+",
             "HCO3-", "Cl-", "SO42-"]

# Run complete pipeline
pipeline = NeutroHydroPipeline()
results = pipeline.fit(X, y, feature_names=ion_names)

# Access results
print(f"Model R2: {results.r2_train:.3f}")
print(f"Baseline fractions (pi_G): {results.nsr.pi_G}")
print(f"Baseline-dominant ions: {results.nsr.baseline_labels}")

# Sample-level baseline fraction
print(f"Sample baseline scores (G): {results.sample_attribution.G}")

# Optional: mineral inference (requires meq/L)
if results.mineral_result:
    print(f"Plausible minerals: {results.mineral_result.plausible}")
    print(f"Mineral fractions: {results.mineral_result.mineral_fractions}"
        )
```

Listing 1: Basic usage of NeutroHydro pipeline

5 Performance and Testing

The package includes comprehensive unit tests and integration tests covering all modules. Testing includes:

- Preprocessing transformations and inverse transforms
- NDG encoder with multiple baseline types
- PNPLS regression with synthetic and real datasets

- NVIP L2 decomposition verification
- Attribution metrics and classification
- Mineral inversion with stoichiometric constraints
- Thermodynamic validation via PHREEQC integration

The NVIP L2 decomposition theorem (Equation 5) is verified numerically to machine precision across diverse datasets, confirming the mathematical correctness of the implementation.

6 Availability and Documentation

`NeutroHydro` is released under the MIT License and is available on PyPI and GitHub:

- **Installation**: `pip install neutrohydro`
- **Repository**: https://github.com/dabdul-wahab1988/neutrohydro
- **Documentation**: https://github.com/dabdul-wahab1988/neutrohydro/tree/main/docs

The package requires Python $\geq$ 3.9 and depends on NumPy (Harris et al., 2020), SciPy (Virtanen et al., 2020), scikit-learn (Pedregosa et al., 2011), and pandas (McKinney, 2010). Optional visualization features require Matplotlib (Hunter, 2007) and Seaborn.

Comprehensive documentation includes:

- Mathematical framework and theory
- API reference for all modules
- Tutorial examples (basic and advanced)
- Interpretation guides for results
- Critical reviews addressing limitations

7 Conclusions

`NeutroHydro` provides the first open-source implementation of neutrosophic chemometrics for groundwater analysis. By operating in absolute concentration space with explicit uncertainty quantification, the framework enables rigorous separation of geogenic and anthropogenic contributions while maintaining physical interpretability. The L2 decomposition theorem for variable importance provides mathematically sound attribution, and stoichiometric inversion links statistical patterns to hydrogeochemical processes. The package is designed for both research applications and operational water quality assessment, with comprehensive testing and documentation supporting reproducible science.

Acknowledgements

We acknowledge the University of Ghana and the Ghana Atomic Energy Commission for institutional support. We thank the reviewers and the open-source Python scientific computing community, particularly the developers of NumPy (Harris et al., 2020), SciPy (Virtanen et al., 2020), scikit-learn (Pedregosa et al., 2011), pandas (McKinney, 2010), and Matplotlib (Hunter, 2007), upon which this package is built.

References

Abdul-Wahab, D. and Asare, E. A. (2025). Neutrosophic chemometrics for groundwater quality assessment: Application of the ndg framework. *In preparation*. Manuscript in preparation.

Aitchison, J. (1986). *The Statistical Analysis of Compositional Data*. Chapman and Hall, London.

Harris, C. R., Millman, K. J., van der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., Wieser, E., Taylor, J., Berg, S., Smith, N. J., Kern, R., Picus, M., Hoyer, S., van Kerkwijk, M. H., Brett, M., Haldane, A., del Río, J. F., Wiebe, M., Peterson, P., Gérard-Marchant, P., Sheppard, K., Reddy, T., Weckesser, W., Abbasi, H., Gohlke, C., and Oliphant, T. E. (2020). Array programming with numpy. *Nature*, 585(7825):357–362.

Hunter, J. D. (2007). Matplotlib: A 2d graphics environment. *Computing in Science & Engineering*, 9(3):90–95.

McKinney, W. (2010). Data structures for statistical computing in python. In van der Walt, S. and Millman, J., editors, *Proceedings of the 9th Python in Science Conference*, pages 56–61.

Mevik, B.-H. and Wehrens, R. (2007). The pls package: Principal component and partial least squares regression in r. *Journal of Statistical Software*, 18(2):1–23.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, É. (2011). Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, 12:2825–2830.

Reimann, C. and Filzmoser, P. (2008). Normal and lognormal data distribution in geochemistry: death of a myth. consequences for the statistical treatment of geochemical and environmental data. *Environmental Geology*, 39(9):1001–1014.

Smarandache, F. (1998). Neutrosophy: Neutrosophic probability, set, and logic. *American Research Press*, pages 1–105.

Virtanen, P., Gommers, R., Oliphant, T. E., Haberland, M., Reddy, T., Cournapeau, D., Burovski, E., Peterson, P., Weckesser, W., Bright, J., van der Walt, S. J., Brett, M., Wilson, J., Millman, K. J., Mayorov, N., Nelson, A. R. J., Jones, E., Kern, R., Larson, E., Carey, C. J., Polat, İ., Feng, Y., Moore, E. W., VanderPlas, J., Laxalde, D., Perktold, J., Cimrman, R., Henriksen, I., Quintero, E. A., Harris, C. R., Archibald, A. M., Ribeiro, A. H., Pedregosa, F., van Mulbregt, P., and Contributors, S. . (2020). Scipy 1.0: Fundamental algorithms for scientific computing in python. *Nature Methods*, 17:261–272.

Wold, H. (1966). Estimation of principal components and related models by iterative least squares. *Multivariate Analysis*, pages 391–420.