Characterizing Chemical Toxicity for Life Cycle Assessment Using Machine Learning and Deep Learning Models Based on Environmental Footprint – Methodological Comparison & textile case study

Tianran Ding[1]*, Gustavo Larrea-Gallegos[1], Federico Busio[1], Antonino Marvuglia[1] & Thomas Schaubroeck[1]

[1] Environmental Sustainability Assessment and Circularity (SUSTAIN) unit, Luxembourg Institute of Science and Technology (LIST), Esch-sur-Alzette, Luxembourg

*E-mail contact: tianran.ding@list.lu

Contents

1. Random Forest

The Random Forest (RF) method builds a collection of decision trees, where each tree trained on a different bootstrap sample from the original training dataset, with all trees generated under the same sampling distribution (Breiman 2001). It requires minimal data preprocessing prior to application without the need for explicit scaling, normalization, or standardization (Danieli et al. 2022). Owing to their ensemble structure and inherent feature subsampling, RF are relatively robust to overfitting,

computationally efficient to train, and versatile across a wide range of predictive tasks (Danieli et al. 2022), making them well suited as baseline models in chemical toxicity modeling.

In this study, RF models were applied in a single-target setting using Mordred descriptors or reduced GROVER features. Median imputation was used to handle missing values in Mordred descriptors. For GROVER features, no missing values are observed and thus no imputation was not applied. However, GROVER dataset contains more than four thousand of dimensions, which can substantially increase computational cost and exacerbate overfitting. A feature reduction was performed using an autoencoder-based approach.

For each target, RF regression was trained on non-missing continuous labels using a constrained forest to reduce overfitting: n_estimators=100, max_depth=10, min_samples_leaf=5, max_features='sqrt', n_jobs=-1, and a fixed random_state=seed. In hurdle mode, the classifier was a standard RandomForestClassifier(n_estimators=100, random_state=seed) with stratified splitting for class balance.

2. XGBoost

Extreme Gradient Boosting (XGBoost) (Chen and Guestrin 2016) is another widely used tree-based ensemble method. The core principle of this method is to construct classification or regression trees sequentially, where each new tree is trained to model the residual errors of the ensemble built so far and allow the model to iteratively refine its performance and progressively improve predictive accuracy. It handles sparse data and can be scaled to large datasets while requiring substantially fewer computational resources than RF. (Chen and Guestrin 2016) Like RF, this work adopted XGBoost for single-target configuration using tabular molecular, but no imputation was used because it handles missing feature values natively, or reduced GROVER features through NN autoencoder.

XGBoost regression employed histogram-based training (tree_method='hist') for computational efficiency on high-dimensional descriptor matrices. Hyperparameters were tuned via RandomizedSearchCV (10 iterations, 5-fold CV, scoring by R^2, parallelized with n_jobs=-1), exploring n_estimators $\in$ {100,300,500}, max_depth $\in$ {3,5,7}, learning_rate $\in$ {0.01,0.05,0.1}, subsample $\in$ {0.8,1.0}, and colsample_bytree $\in$ {0.8,1.0}.

3. Gaussian Process

Different from RF, and XGBoost, Gaussian Process (GP) regression is a probabilistic, kernel-based learning approach that defines a distribution over functions and provides both predictive means and uncertainty estimates (Rasmussen and Christopher 2006). We applied GP models to single-target regression tasks using Mordred descriptors or reduced GROVER features through NN autoencoder. Feature preprocessing was fitted on the training features and applied to the test features, including median imputation, standardization (zero mean and unit variance), and dimensionality reduction via principal component analysis (PCA) to ensure numerical stability and computational tractability. Target values were similarly standardized using a separate z-score scaler fitted on the training targets and applied to the test targets.

For each target, an exact GP model (gpytorch) was trained after tabular preprocessing: median imputation → standardization → PCA (95% variance). Targets were standardized using a dedicated transform_data() routine and inverse-transformed for reporting in original units. A GaussianLikelihood was used, and the model was optimized by maximizing the Exact Marginal Log Likelihood (ExactMarginalLogLikelihood) using fit_gpytorch_mll. Predictive uncertainty was extracted from the posterior variance and reported as an average standard deviation in original units via the scaler's scale parameter, enabling uncertainty-aware CF screening.

4. Deep Neural Network with Tabular Data

Deep Neural Network (DNN) with multiple hidden layers have been successfully applied to learn complex relationships between chemical structures and toxic effects that traditional ML models are unable to handle (Guo et al. 2023). In this work, we implemented DNN with DeepMTP python framework (Iliadis et al. 2023). The DeepMTP framework employs two separate neural network branches to encode chemical features and targets into low-dimensional embedding vectors. These two embeddings are then combined to produce a prediction for a specific instance-target pair.

To predict multiple targets at the same time, standardized ecotoxicity CFs across different environmental compartments (11 targets) were treated as one group of multiple related targets, while standardized human toxicity CFs across compartments (11 targets) were modeled as a separate multi-target set. In addition, for the purpose of comparison, DeepMTP was additionally trained in a single-target configuration for each target. The training data are either from curated Mordred descriptors with standard scaler or from original GROVER embeddings without feature reduction.

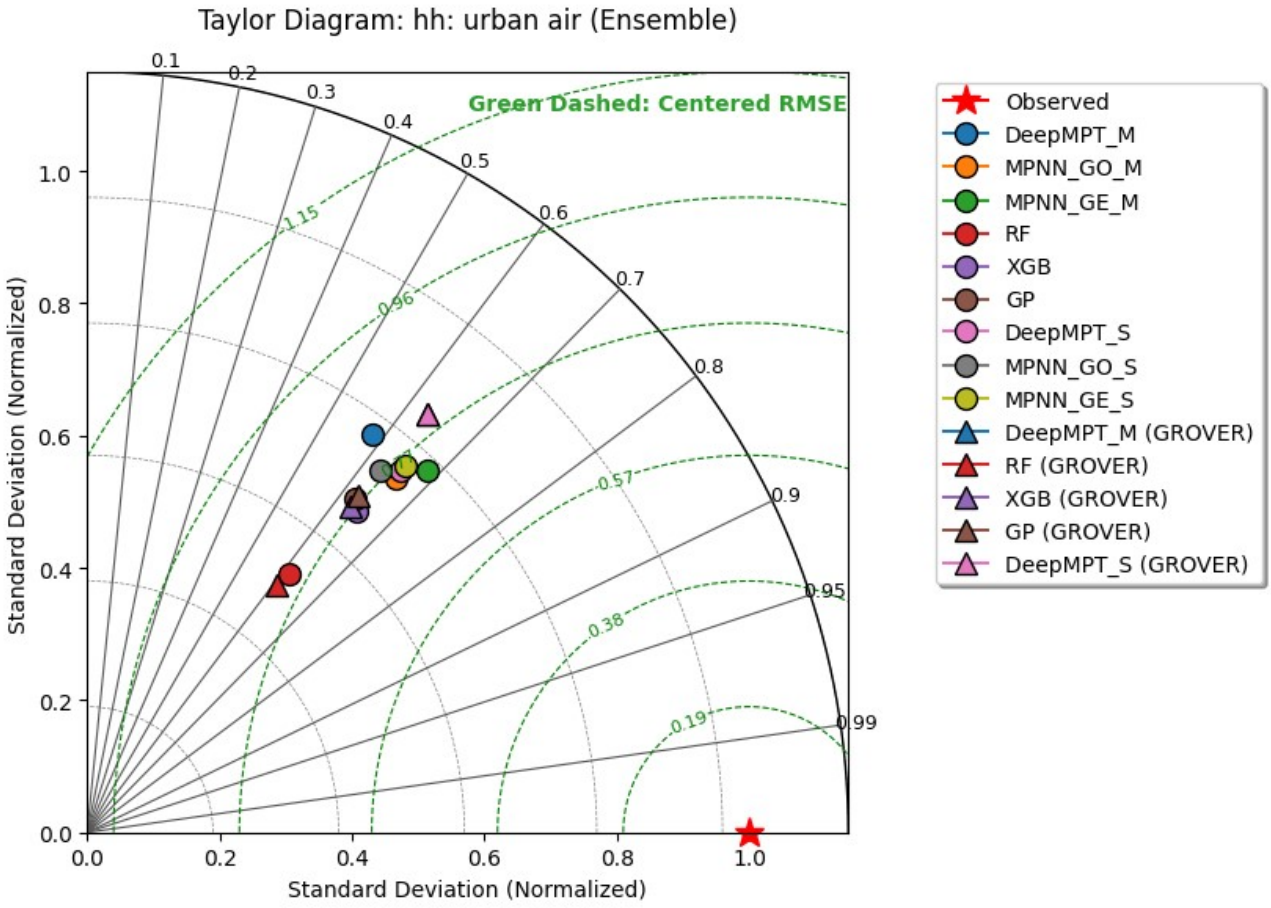
DeepMPT was implemented in both regression and classification modes, enabling multi-target learning over partially observed label matrices. Samples were retained if at least one target was available, supporting efficient use of sparse multi-endpoint toxicity data. For regression, targets were transformed using Yeo–Johnson PowerTransformer (standardize=True) to stabilize heavy tails and improve optimization. Inputs were processed via median imputation + standard scaling. The network used an MLP-based instance branch with two layers [256, 128], a compact target branch [32] (task embedding learned via one-hot), and a small combination head [64, 32] with embedding_size=32. Training used early stopping (patience=20) and monitored macro-averaged RMSE for regression (and macro AUROC for classification), with batch size 64 on cuda:0.

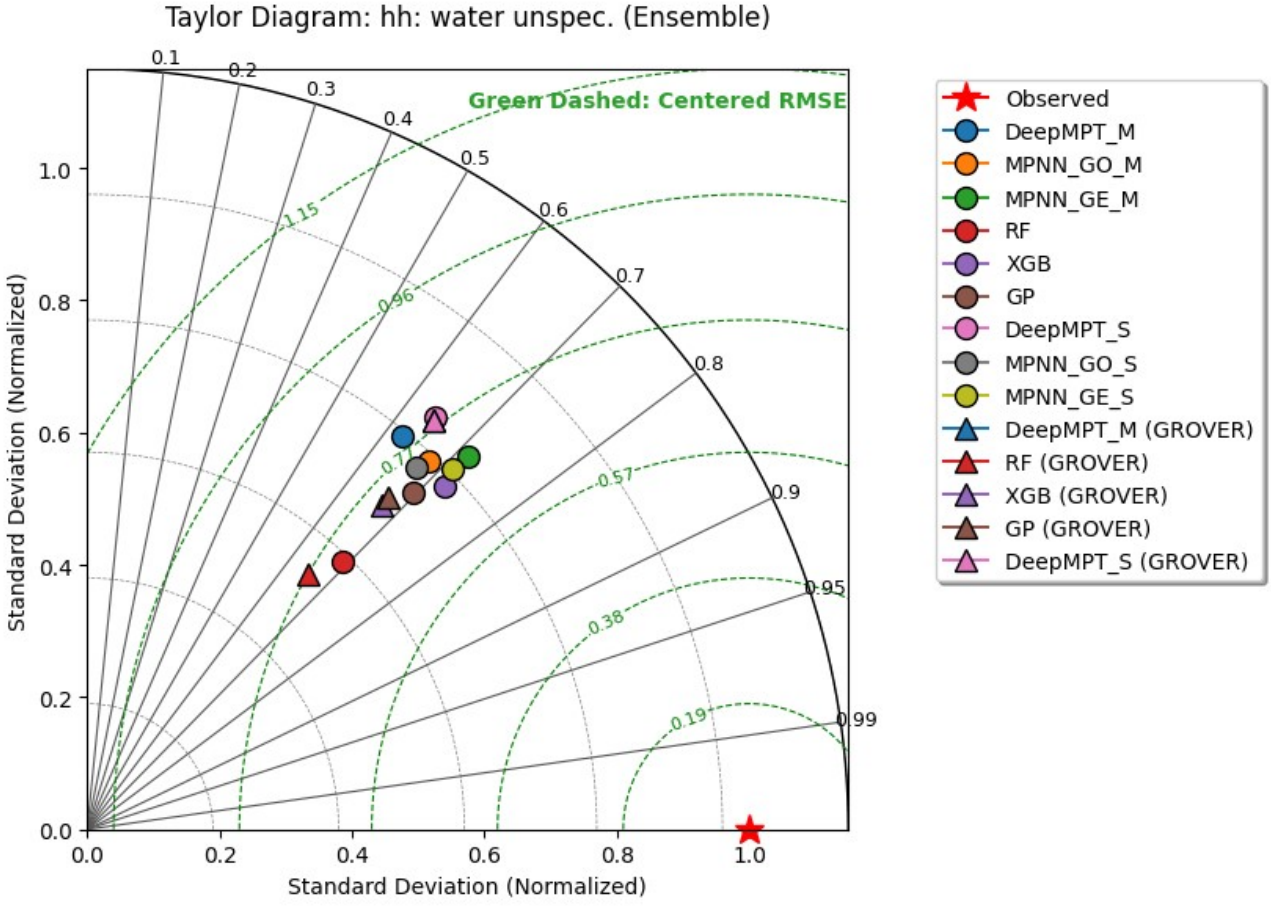
5. Graph Neural Network

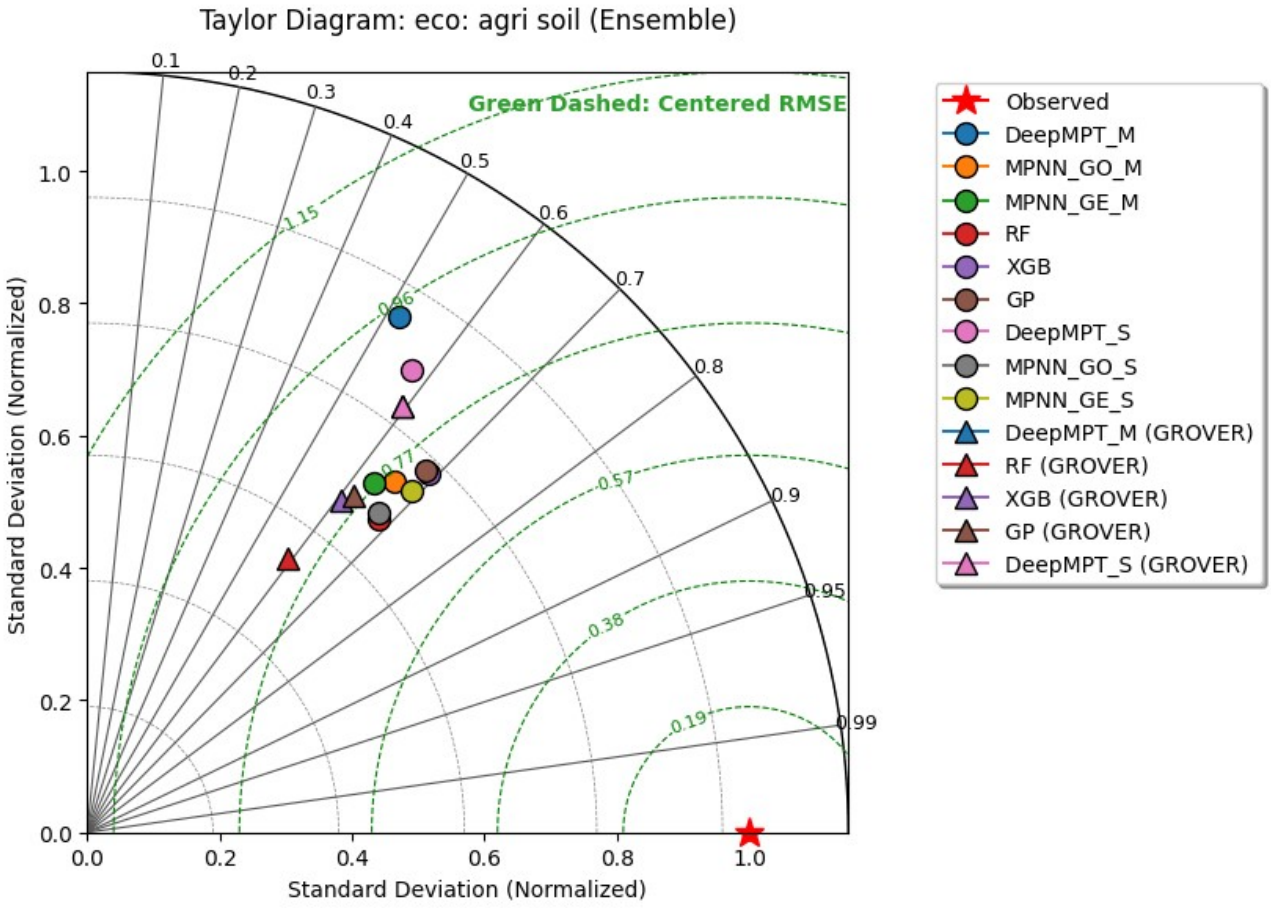
In contrast to above selected models that are fed by numerical tabular values, Graph Neural Network (GNN) treats molecules as graphs (image) with atoms as nodes and bonds as edges to predict chemical properties and toxicity. Unlike descriptor-based models, GNN eliminate the need for handcrafted features and are well suited for capturing complex structure–property relationships in chemical toxicity modeling. The architectural paradigm for GNNs is predominantly formulated as a Message-Passing Neural Network (MPNN) framework (Gilmer et al. 2017). In this representation, molecular geometry is mapped onto a three-dimensional graph where nodes encapsulate atomic chemical identities and edges parameterize the pairwise Euclidean distances between atoms (Cremer et al. 2023). Two variants were considered: (i) a graph-only MPNN relying solely on molecular topology and atom/bond features, and (ii) an extended MPNN incorporating additional mordred molecular descriptors as auxiliary inputs (same data treatment as in DNN). Same feature and target processing was applied as for Both single-target and multi-target training modes were explored. The implementation utilized the Chemprop (v2.0) framework (Heid et al. 2024).

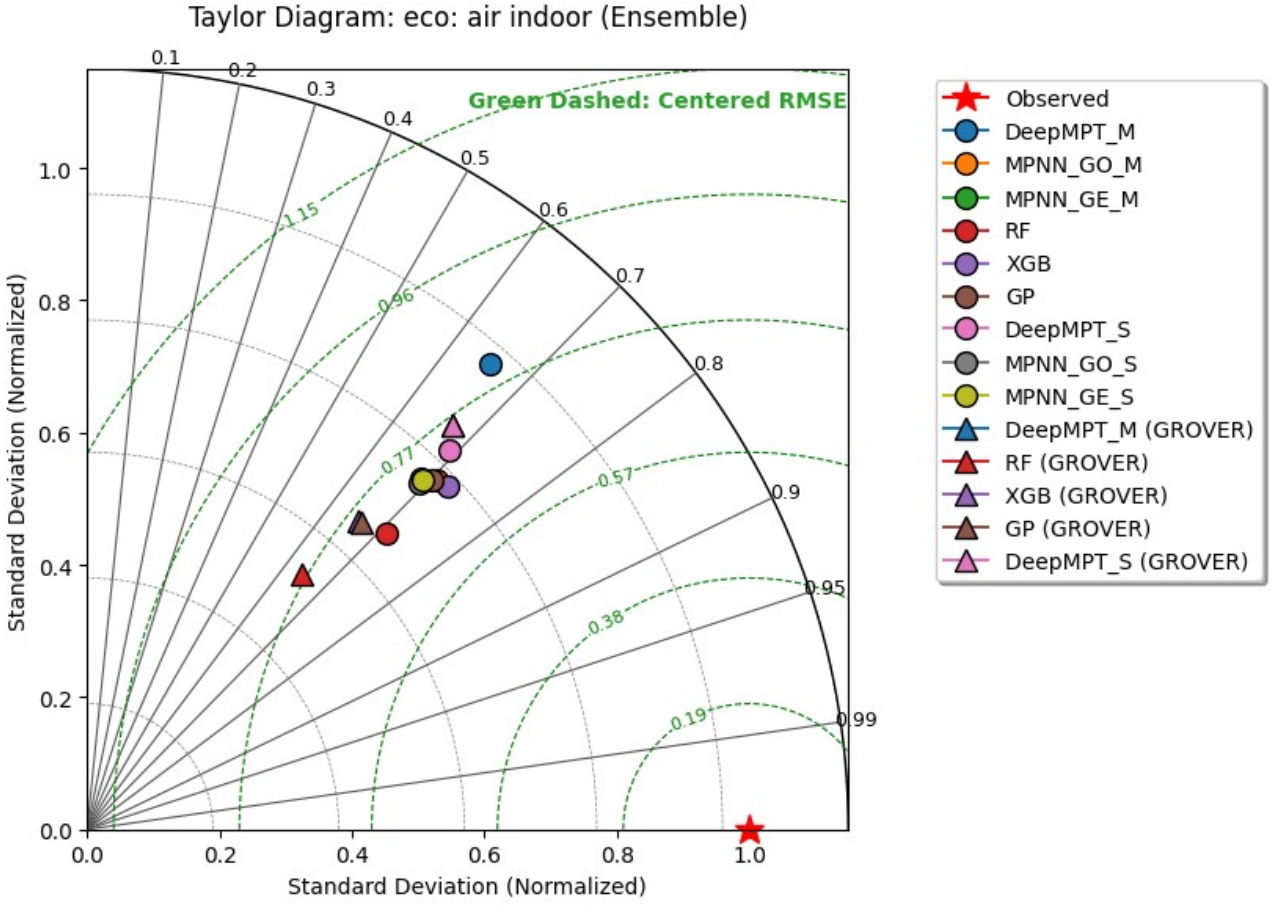
For regression, EvidentialFFN head was used to obtain both predictions and calibrated uncertainty proxies, with targets normalized by Chemprop's normalize_targets() and unscaled via UnscaleTransform. Optimization used PyTorch Lightning with ModelCheckpoint(monitor=val_loss) and EarlyStopping(monitor=val_loss, patience=10). Hyperparameters (e.g., depth, hidden size, dropout, batch size, learning-rate schedule) were optimized separately for regression and classification using Hyperopt/TPE on validation loss.

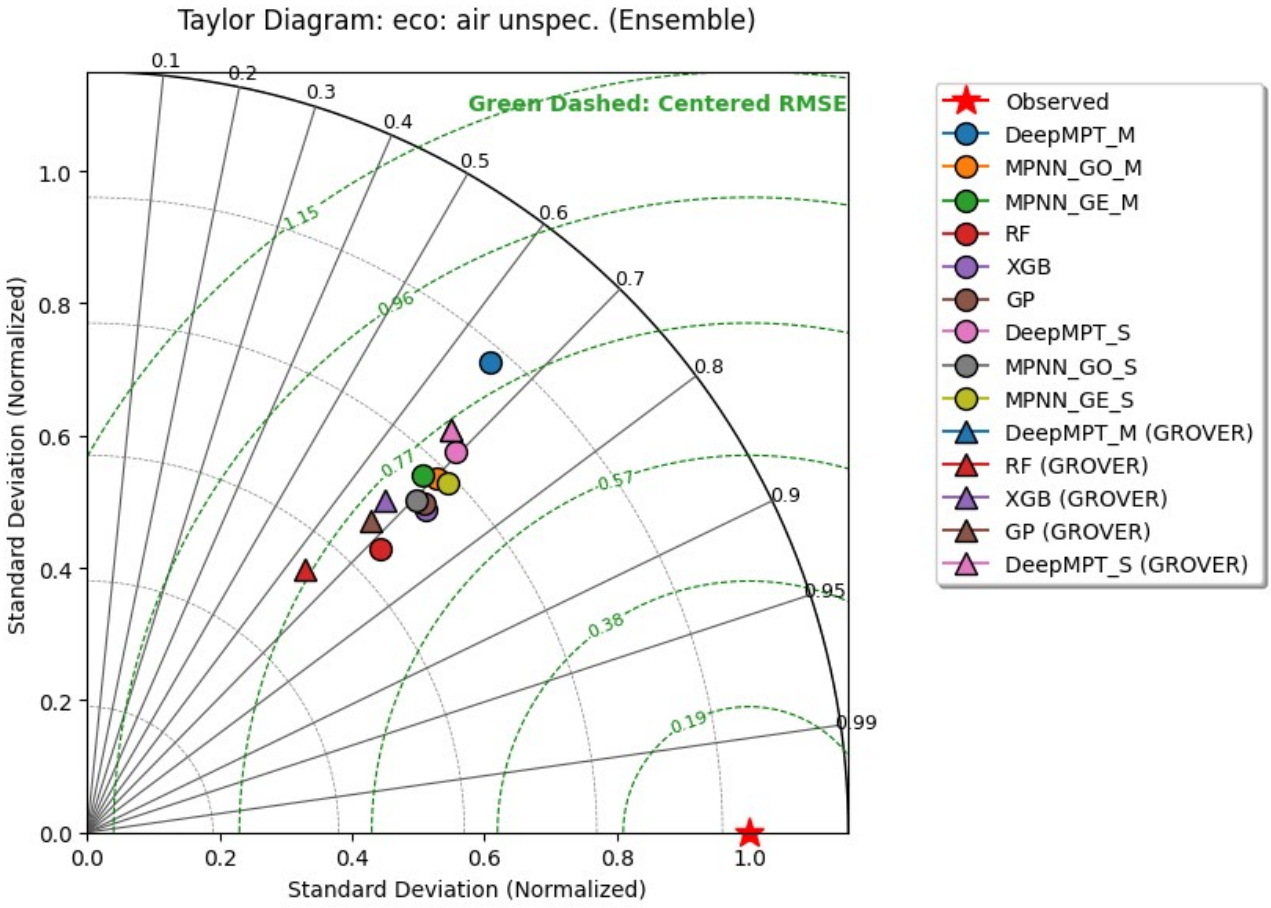
6. Taylor graps

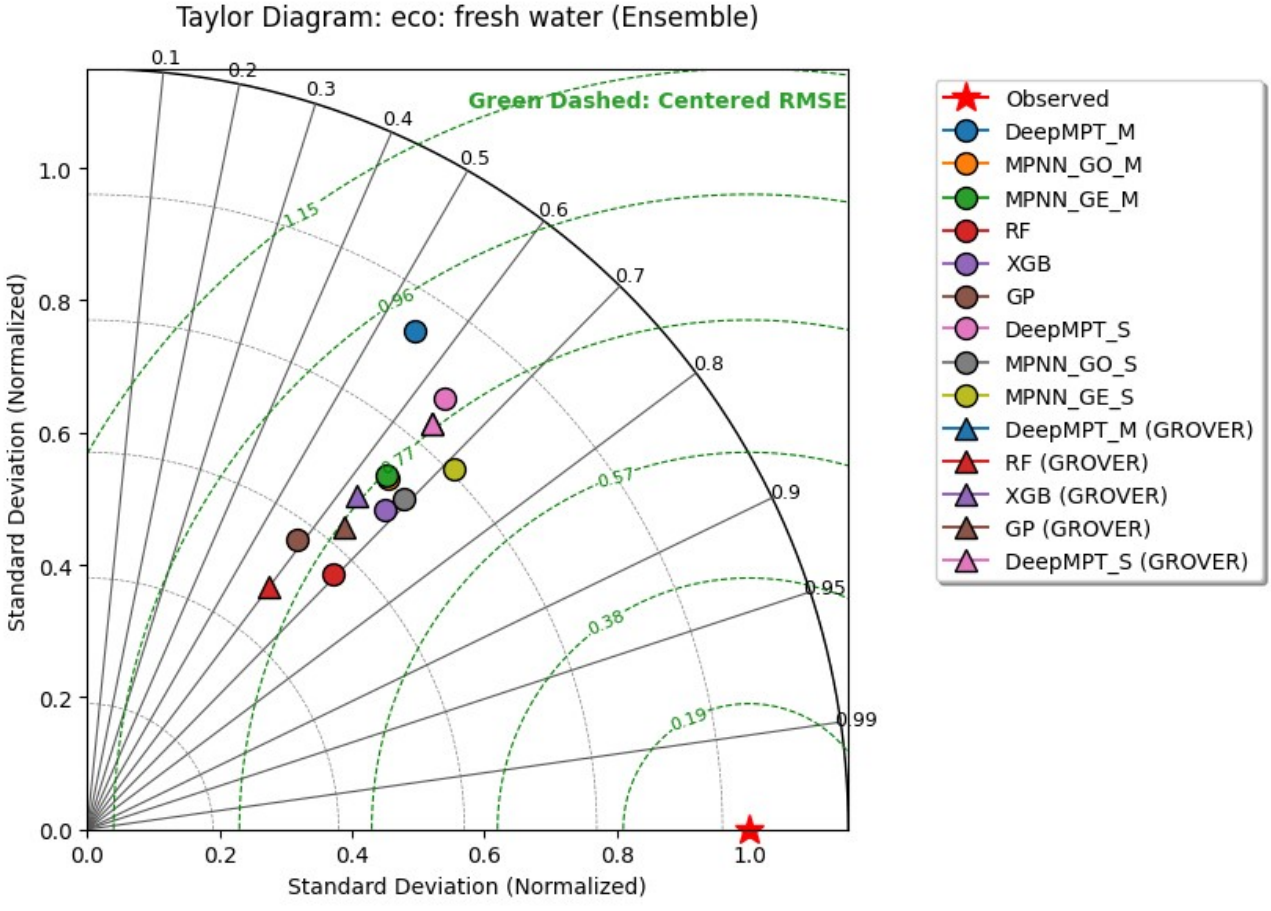

Taylor Diagram: hh: urban air (Ensemble)
Green Dashed: Centered RMSE
Standard Deviation (Normalized)
Standard Deviation (Normalized)
Observed
DeepMPT_M
MPNN_GO_M
MPNN_GE_M
RF
XGB
GP
DeepMPT_S
MPNN_GO_S
MPNN_GE_S
DeepMPT_M (GROVER)
RF (GROVER)
XGB (GROVER)
GP (GROVER)
DeepMPT_S (GROVER)

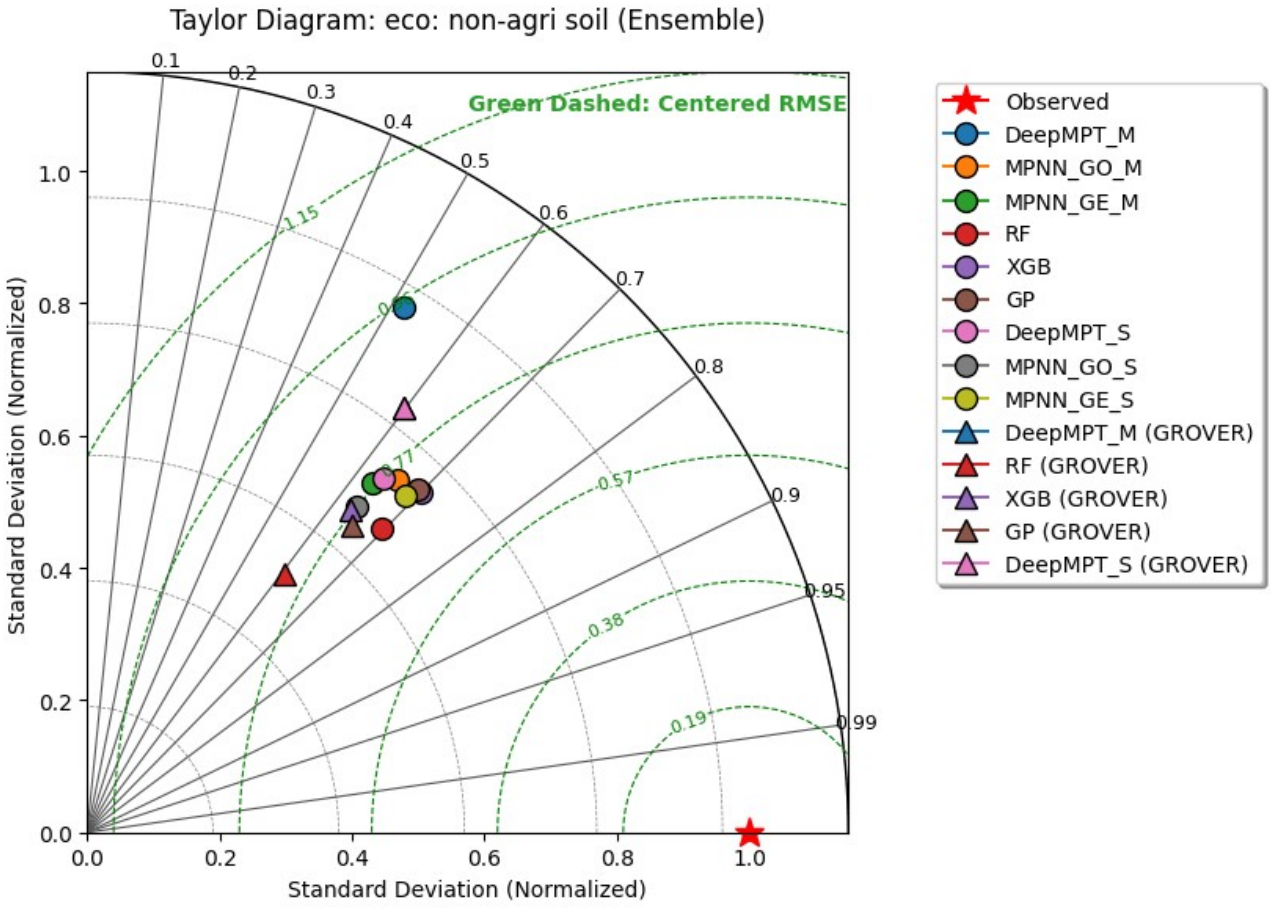

Taylor Diagram: hh: water unspec. (Ensemble)
Green Dashed: Centered RMSE
Standard Deviation (Normalized)
Standard Deviation (Normalized)
Observed
DeepMPT_M
MPNN_GO_M
MPNN_GE_M
RF
XGB
GP
DeepMPT_S
MPNN_GO_S
MPNN_GE_S
DeepMPT_M (GROVER)
RF (GROVER)
XGB (GROVER)
GP (GROVER)
DeepMPT_S (GROVER)

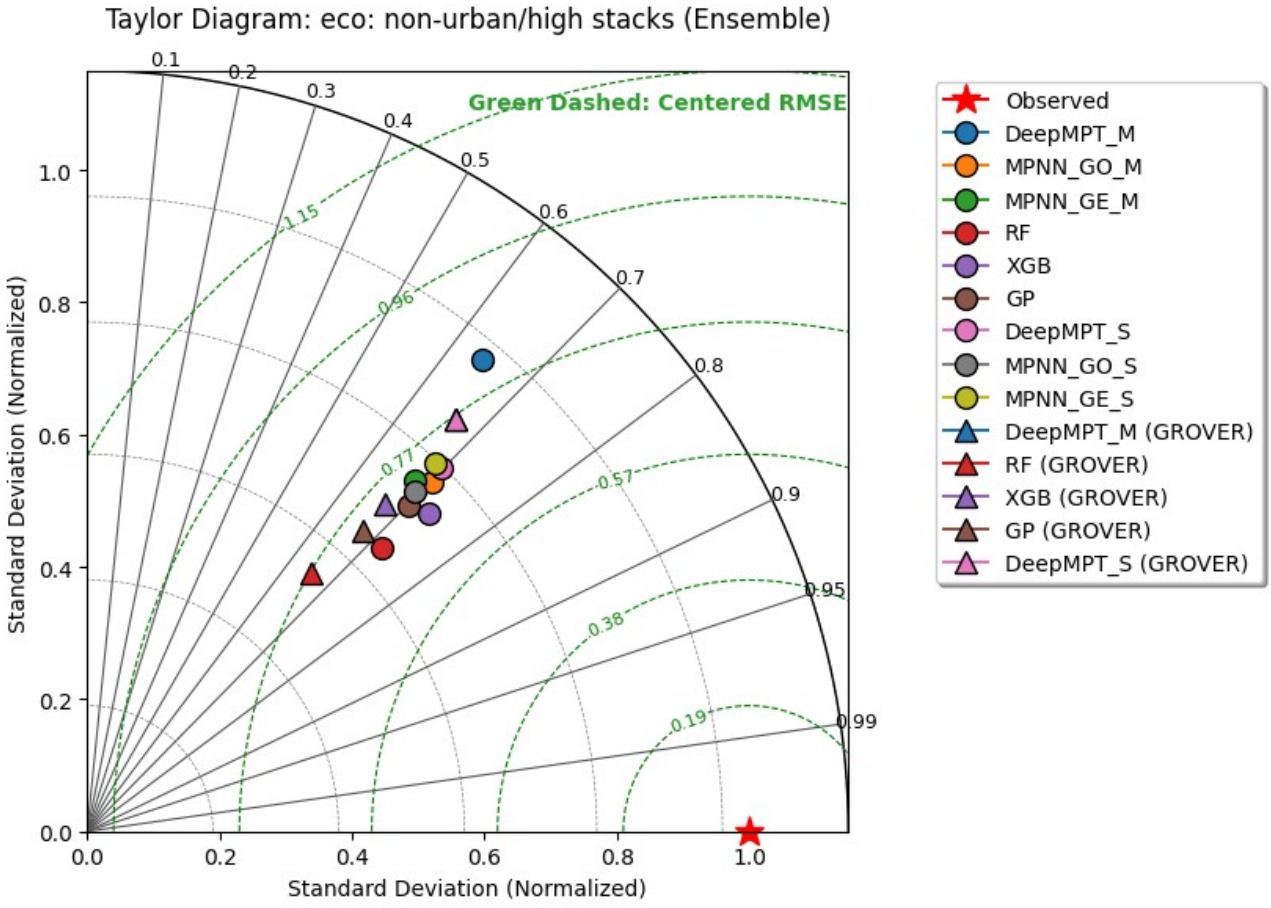

Taylor Diagram: eco: agri soil (Ensemble)
Green Dashed: Centered RMSE
Standard Deviation (Normalized)
Standard Deviation (Normalized)
Observed
DeepMPT_M
MPNN_GO_M
MPNN_GE_M
RF
XGB
GP
DeepMPT_S
MPNN_GO_S
MPNN_GE_S
DeepMPT_M (GROVER)
RF (GROVER)
XGB (GROVER)
GP (GROVER)
DeepMPT_S (GROVER)

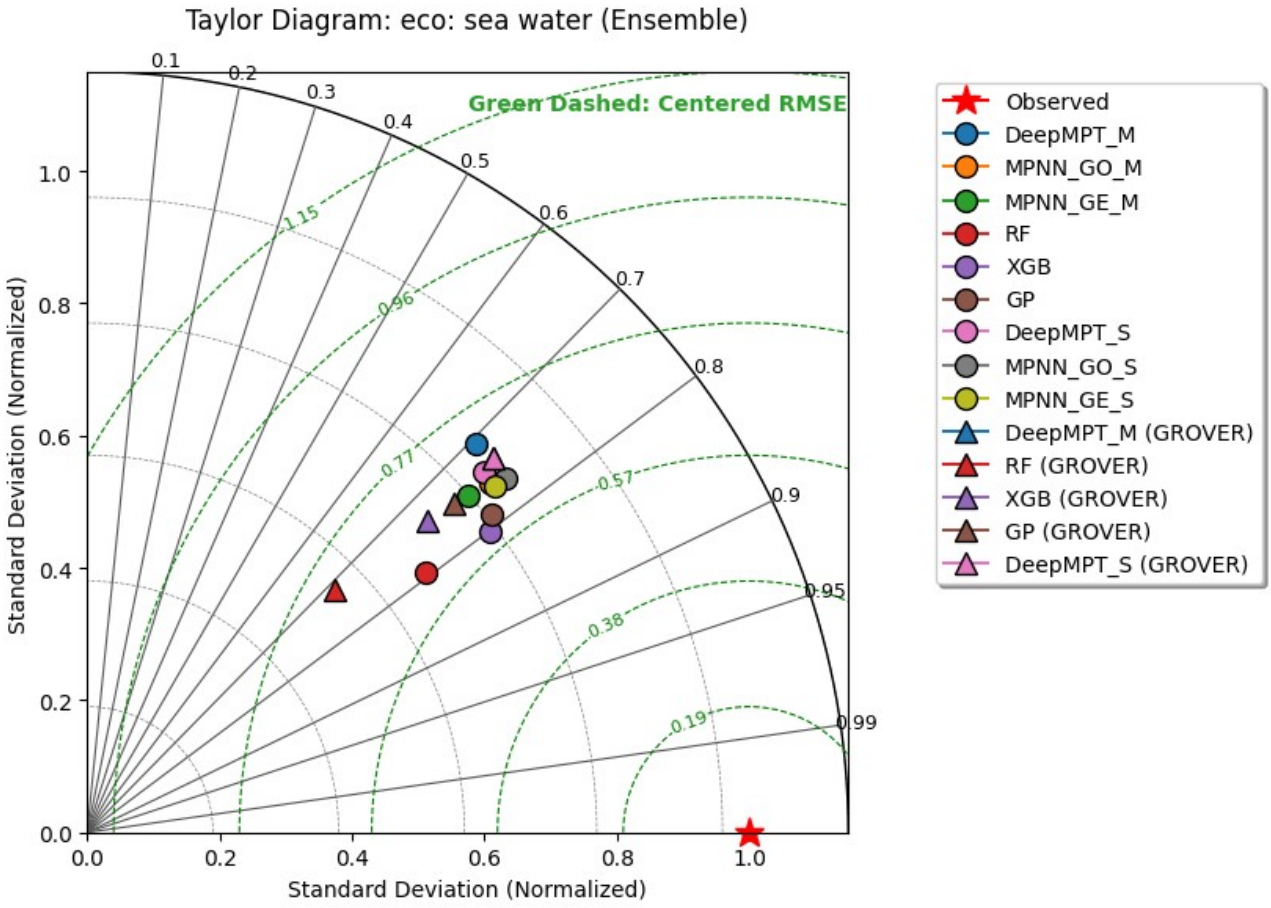

Taylor Diagram: eco: air indoor (Ensemble)
Green Dashed: Centered RMSE
Standard Deviation (Normalized)
Standard Deviation (Normalized)
Observed
DeepMPT_M
MPNN_GO_M
MPNN_GE_M
RF
XGB
GP
DeepMPT_S
MPNN_GO_S
MPNN_GE_S
DeepMPT_M (GROVER)
RF (GROVER)
XGB (GROVER)
GP (GROVER)
DeepMPT_S (GROVER)

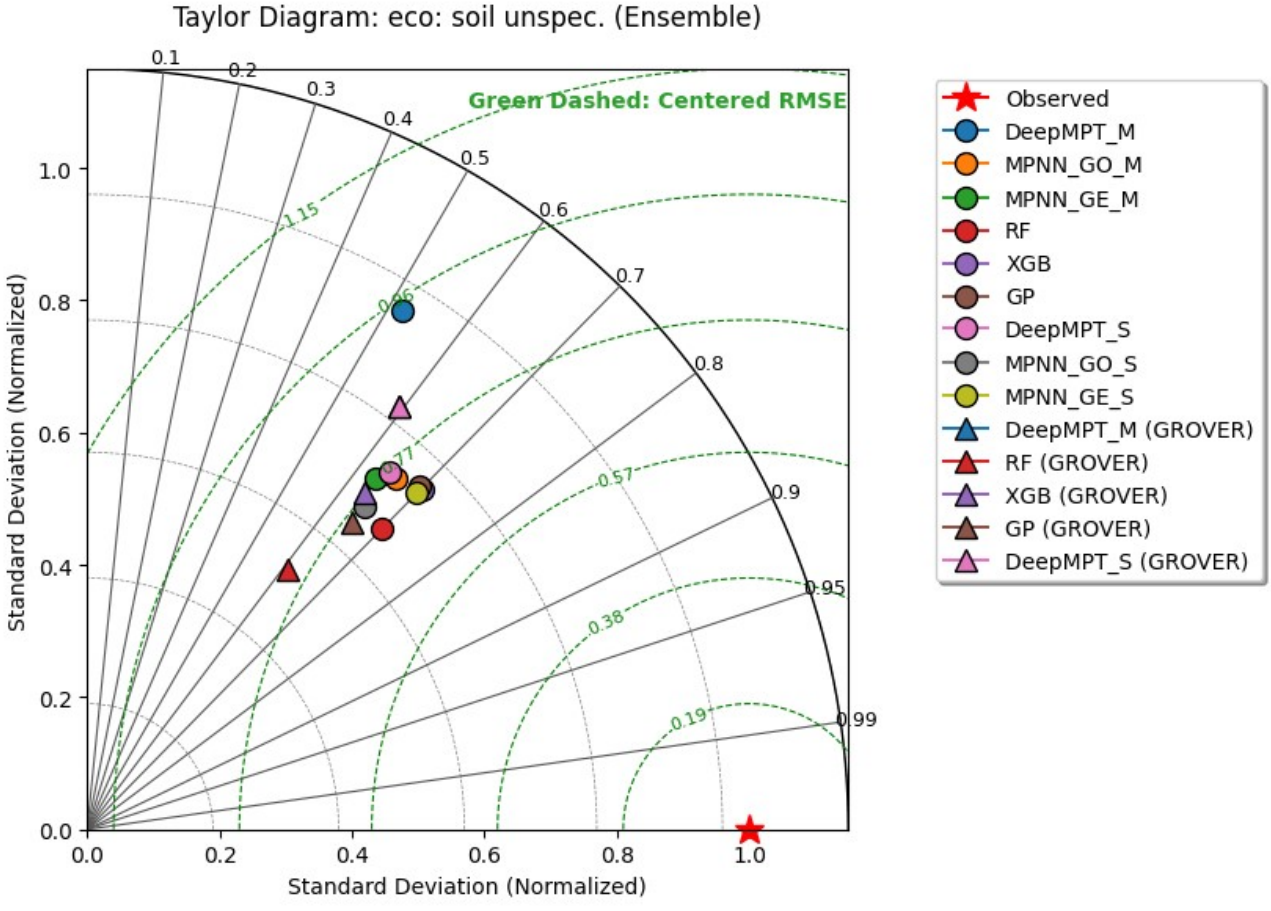

Taylor Diagram: eco: air unspec. (Ensemble)
Green Dashed: Centered RMSE
Standard Deviation (Normalized)
Standard Deviation (Normalized)
Observed
DeepMPT_M
MPNN_GO_M
MPNN_GE_M
RF
XGB
GP
DeepMPT_S
MPNN_GO_S
MPNN_GE_S
DeepMPT_M (GROVER)
RF (GROVER)
XGB (GROVER)
GP (GROVER)
DeepMPT_S (GROVER)

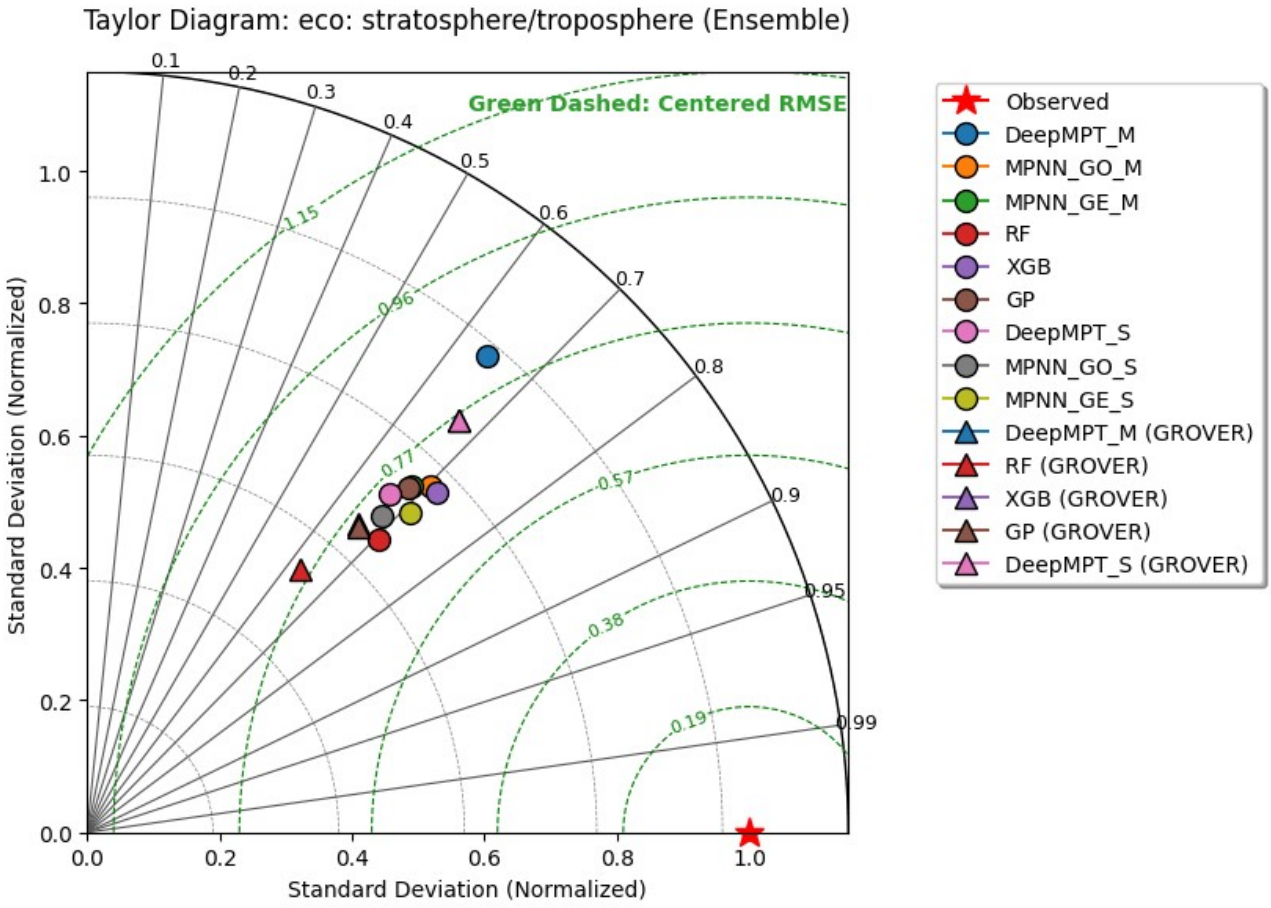

Taylor Diagram: eco: fresh water (Ensemble)
Green Dashed: Centered RMSE
Standard Deviation (Normalized)
Standard Deviation (Normalized)
Observed
DeepMPT_M
MPNN_GO_M
MPNN_GE_M
RF
XGB
GP
DeepMPT_S
MPNN_GO_S
MPNN_GE_S
DeepMPT_M (GROVER)
RF (GROVER)
XGB (GROVER)
GP (GROVER)
DeepMPT_S (GROVER)

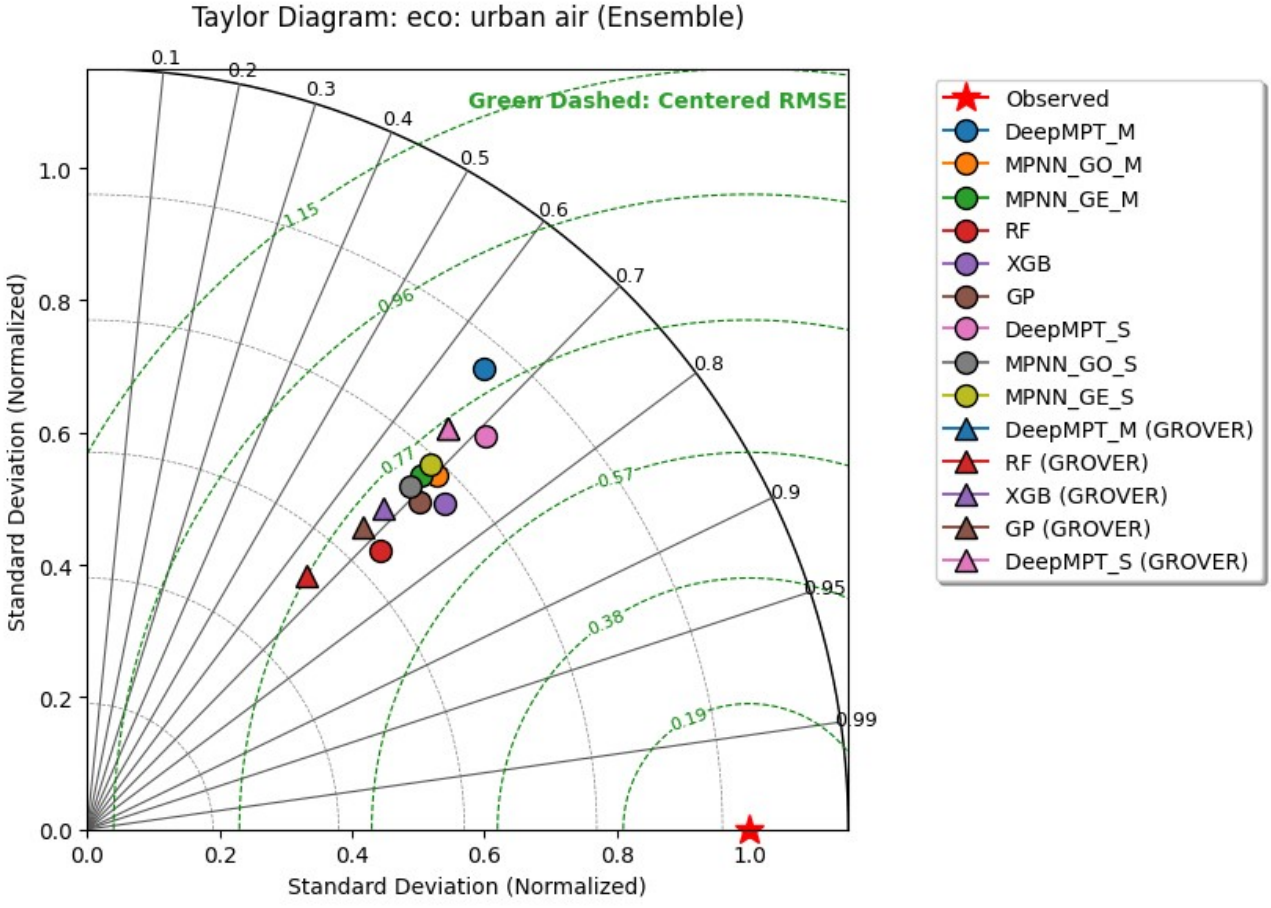

Taylor Diagram: eco: non-agri soil (Ensemble)
Green Dashed: Centered RMSE
Standard Deviation (Normalized)
Standard Deviation (Normalized)
Observed
DeepMPT_M
MPNN_GO_M
MPNN_GE_M
RF
XGB
GP
DeepMPT_S
MPNN_GO_S
MPNN_GE_S
DeepMPT_M (GROVER)
RF (GROVER)
XGB (GROVER)
GP (GROVER)
DeepMPT_S (GROVER)

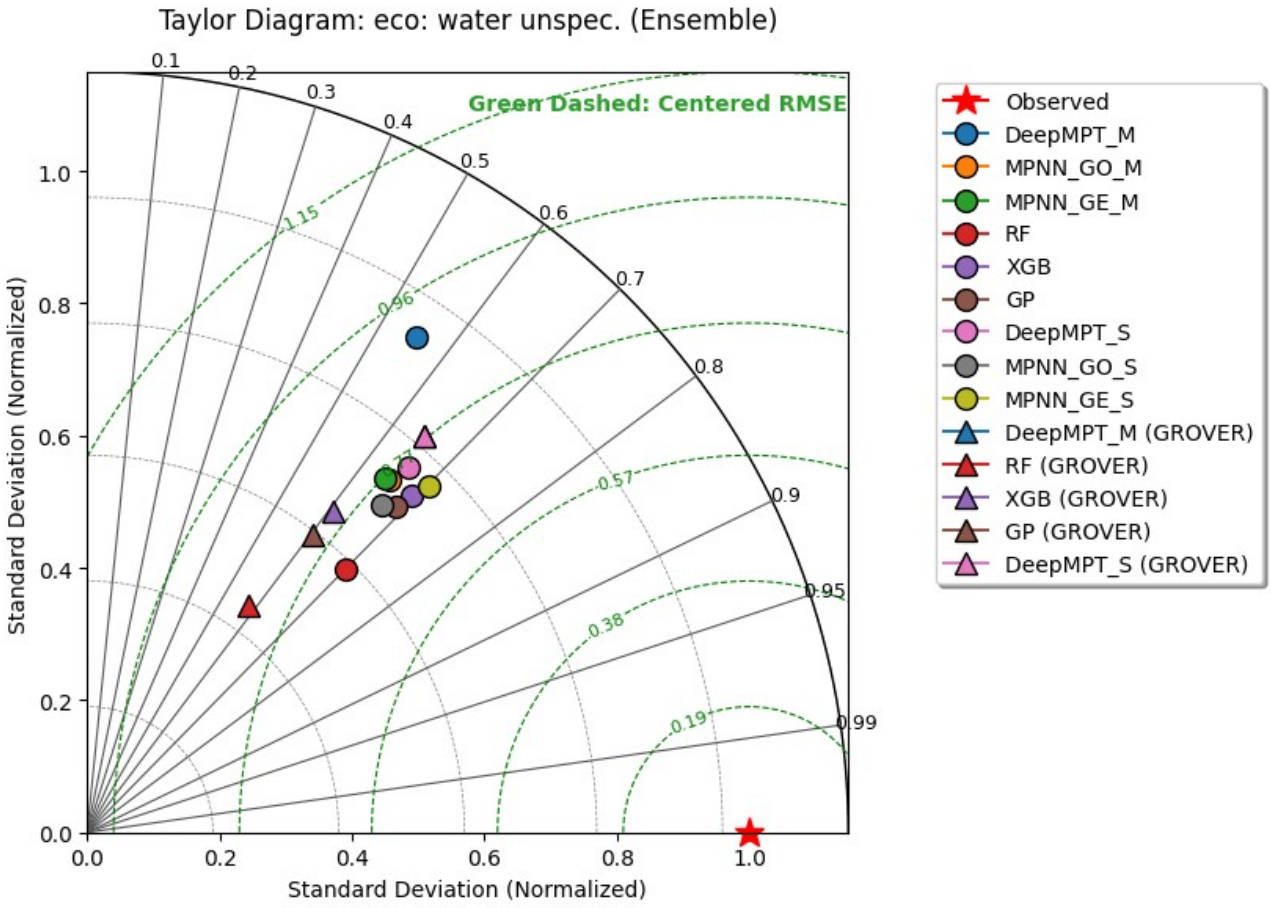

Taylor Diagram: eco: non-urban/high stacks (Ensemble)
Green Dashed: Centered RMSE
Standard Deviation (Normalized)
Standard Deviation (Normalized)
Observed
DeepMPT_M
MPNN_GO_M
MPNN_GE_M
RF
XGB
GP
DeepMPT_S
MPNN_GO_S
MPNN_GE_S
DeepMPT_M (GROVER)
RF (GROVER)
XGB (GROVER)
GP (GROVER)
DeepMPT_S (GROVER)

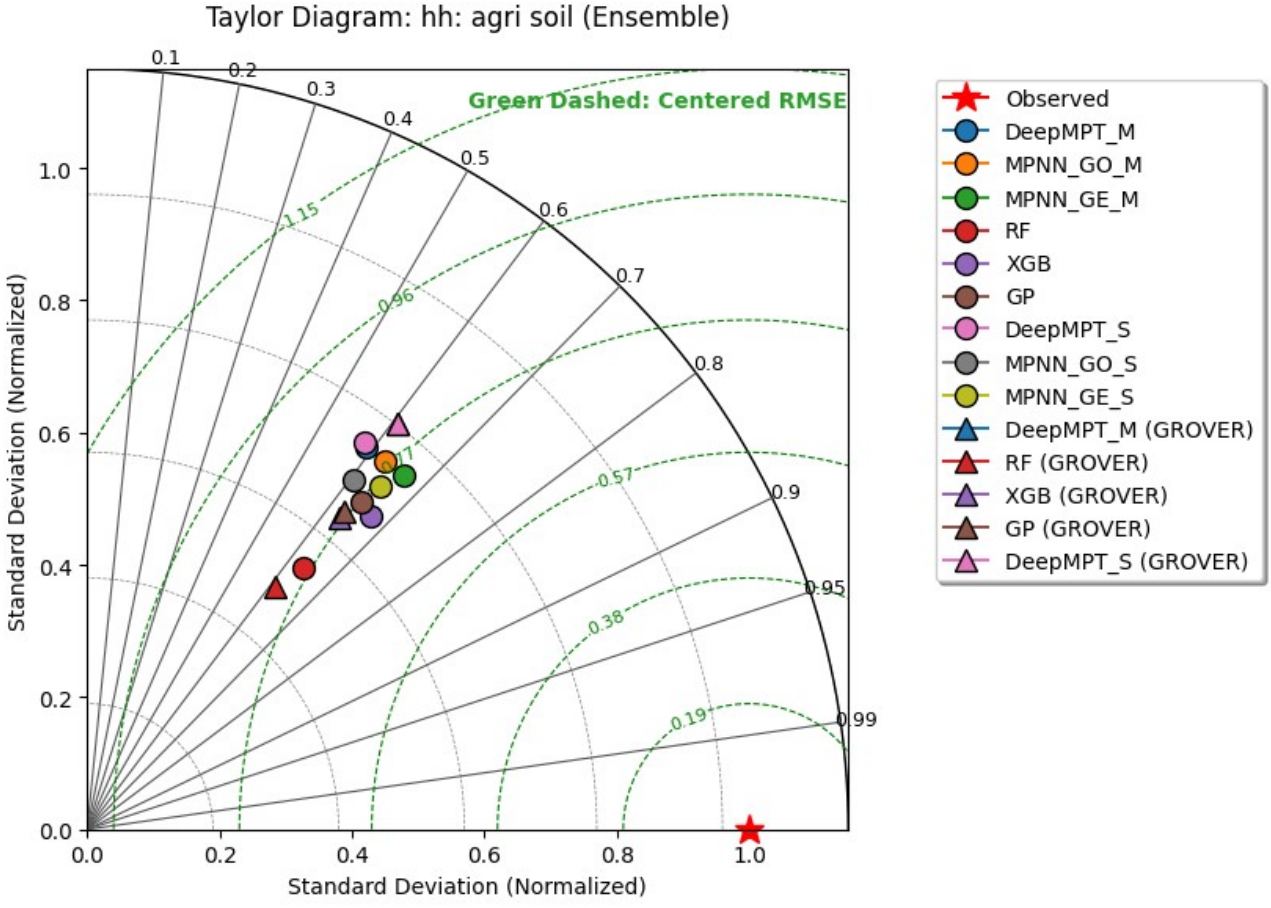

Taylor Diagram: eco: sea water (Ensemble)
Green Dashed: Centered RMSE
Standard Deviation (Normalized)
Standard Deviation (Normalized)
Observed
DeepMPT_M
MPNN_GO_M
MPNN_GE_M
RF
XGB
GP
DeepMPT_S
MPNN_GO_S
MPNN_GE_S
DeepMPT_M (GROVER)
RF (GROVER)
XGB (GROVER)
GP (GROVER)
DeepMPT_S (GROVER)

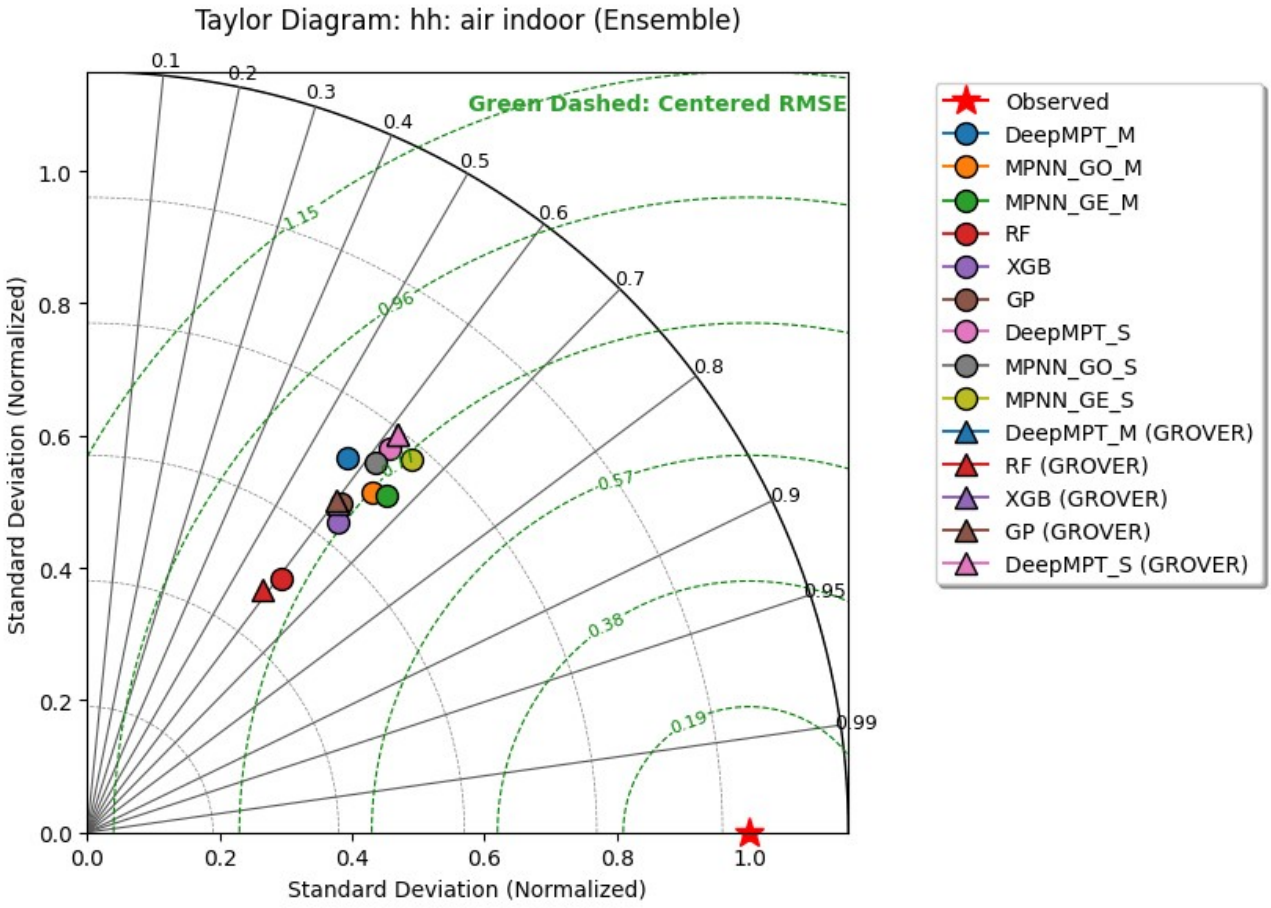

Taylor Diagram: eco: soil unspec. (Ensemble)
Green Dashed: Centered RMSE
Standard Deviation (Normalized)
Standard Deviation (Normalized)
Observed
DeepMPT_M
MPNN_GO_M
MPNN_GE_M
RF
XGB
GP
DeepMPT_S
MPNN_GO_S
MPNN_GE_S
DeepMPT_M (GROVER)
RF (GROVER)
XGB (GROVER)
GP (GROVER)
DeepMPT_S (GROVER)

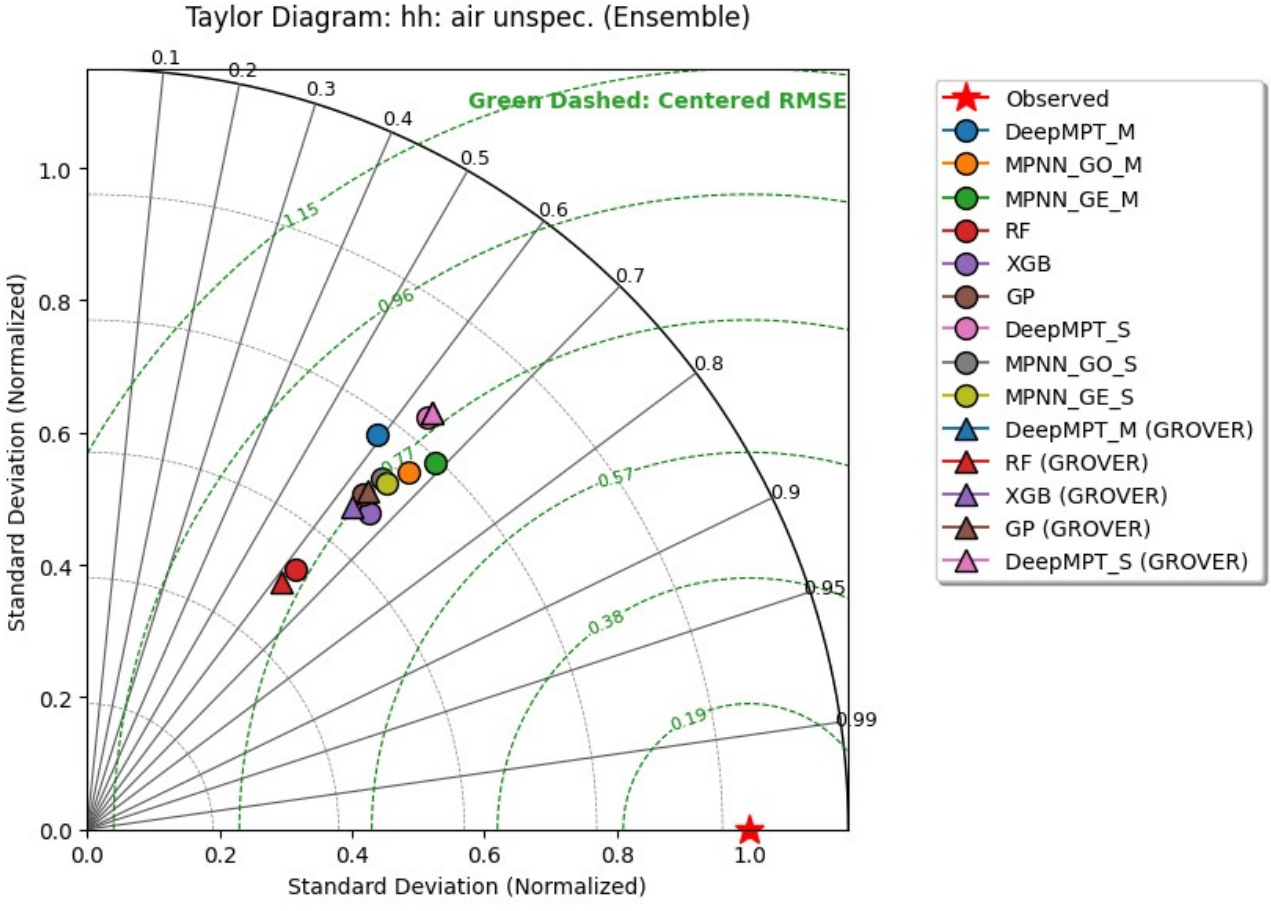

Taylor Diagram: eco: stratosphere/troposphere (Ensemble)
Green Dashed: Centered RMSE
Standard Deviation (Normalized)
Standard Deviation (Normalized)
Observed
DeepMPT_M
MPNN_GO_M
MPNN_GE_M
RF
XGB
GP
DeepMPT_S
MPNN_GO_S
MPNN_GE_S
DeepMPT_M (GROVER)
RF (GROVER)
XGB (GROVER)
GP (GROVER)
DeepMPT_S (GROVER)

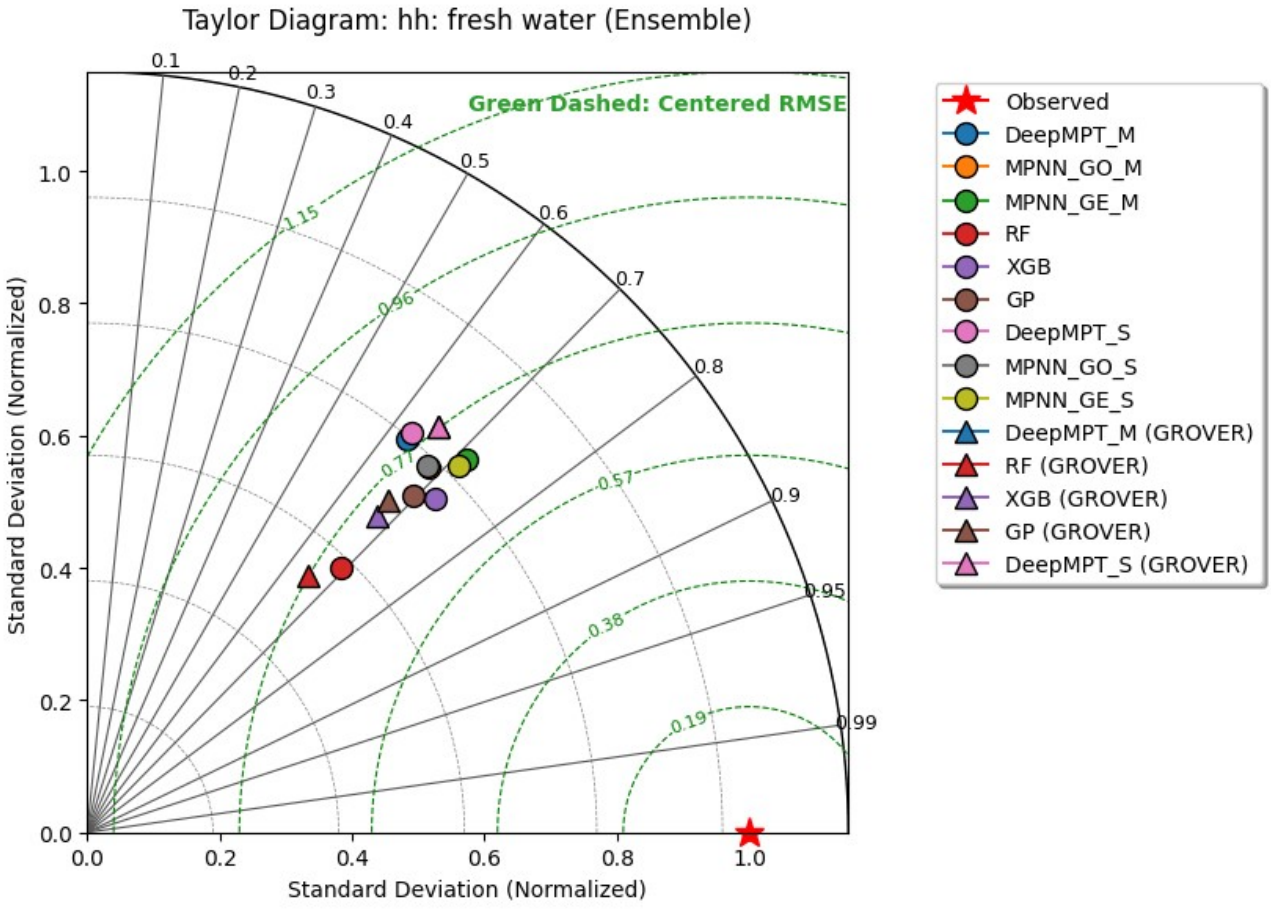

Taylor Diagram: eco: urban air (Ensemble)
Green Dashed: Centered RMSE
Standard Deviation (Normalized)
Standard Deviation (Normalized)
Observed
DeepMPT_M
MPNN_GO_M
MPNN_GE_M
RF
XGB
GP
DeepMPT_S
MPNN_GO_S
MPNN_GE_S
DeepMPT_M (GROVER)
RF (GROVER)
XGB (GROVER)
GP (GROVER)
DeepMPT_S (GROVER)

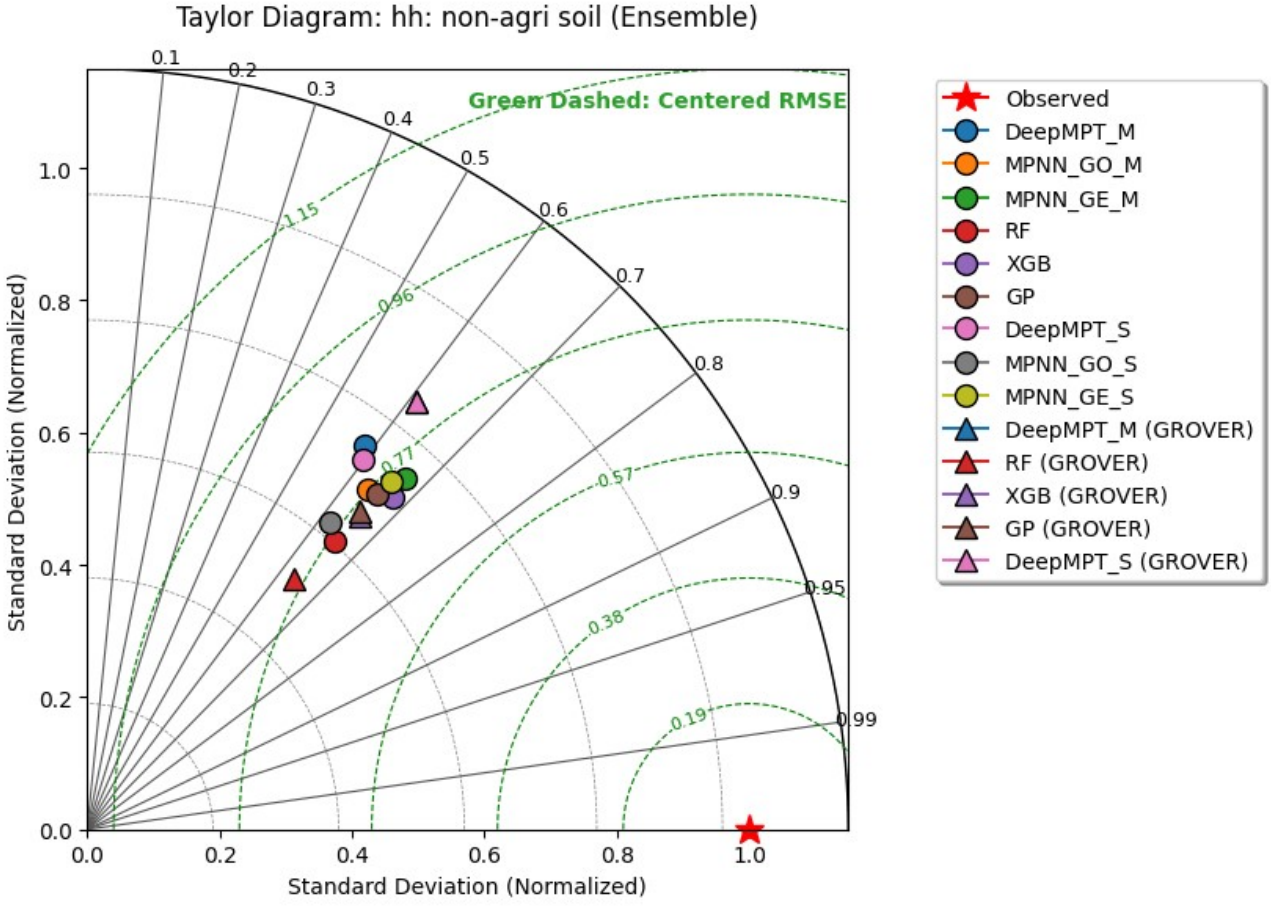

Taylor Diagram: eco: water unspec. (Ensemble)
Green Dashed: Centered RMSE
Standard Deviation (Normalized)
Standard Deviation (Normalized)
Observed
DeepMPT_M
MPNN_GO_M
MPNN_GE_M
RF
XGB
GP
DeepMPT_S
MPNN_GO_S
MPNN_GE_S
DeepMPT_M (GROVER)
RF (GROVER)
XGB (GROVER)
GP (GROVER)
DeepMPT_S (GROVER)

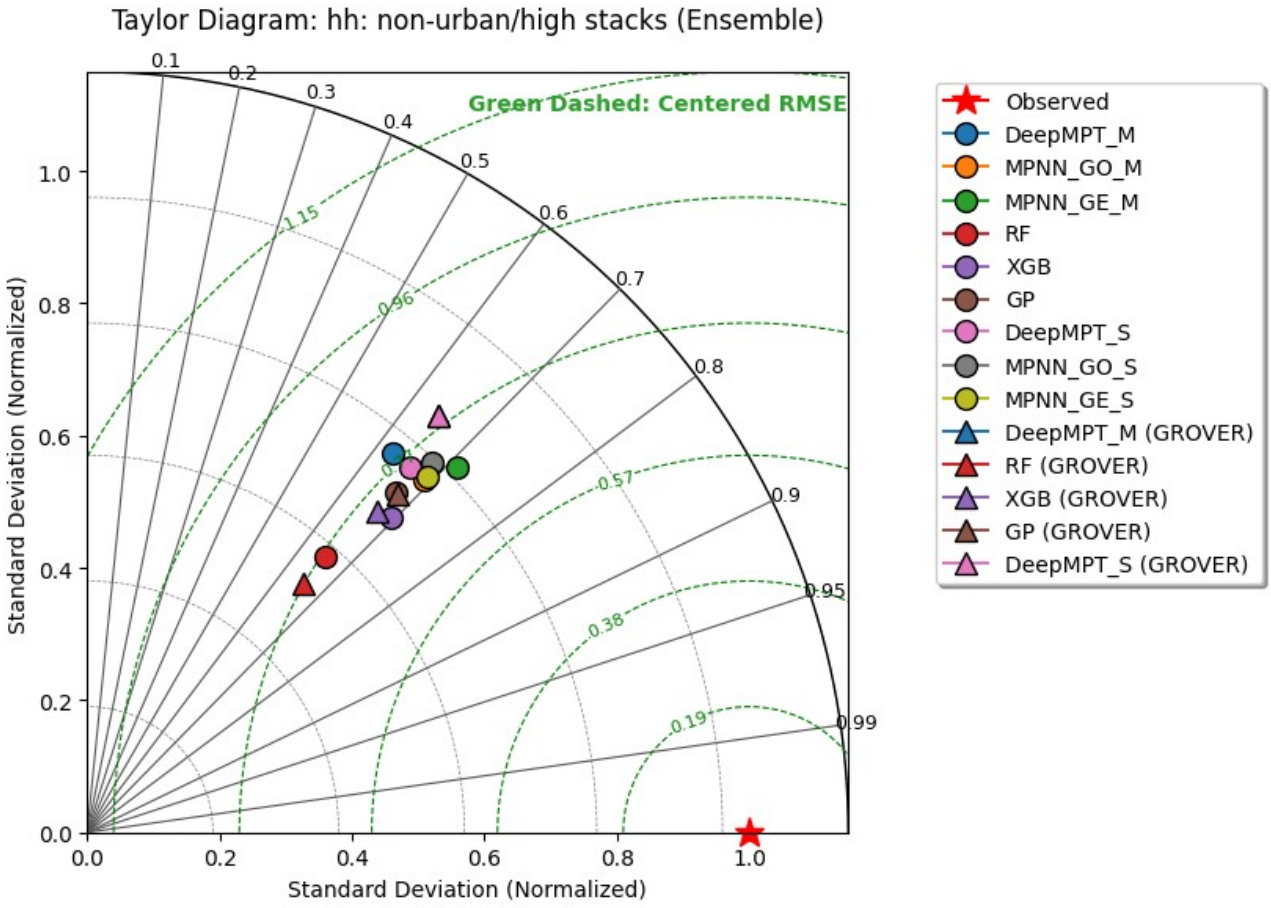

Taylor Diagram: hh: agri soil (Ensemble)
Green Dashed: Centered RMSE
Standard Deviation (Normalized)
Standard Deviation (Normalized)
Observed
DeepMPT_M
MPNN_GO_M
MPNN_GE_M
RF
XGB
GP
DeepMPT_S
MPNN_GO_S
MPNN_GE_S
DeepMPT_M (GROVER)
RF (GROVER)
XGB (GROVER)
GP (GROVER)
DeepMPT_S (GROVER)

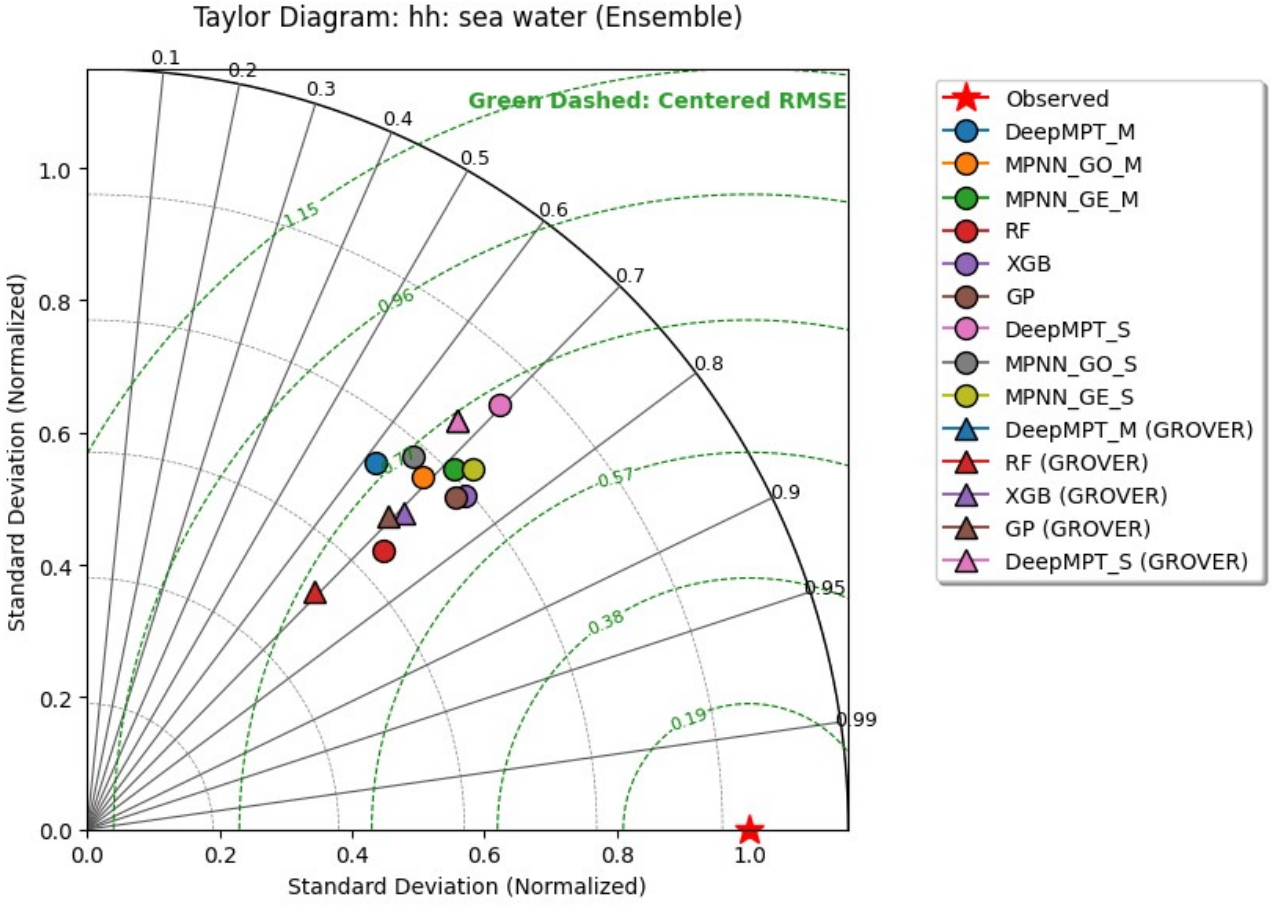

Taylor Diagram: hh: air indoor (Ensemble)
Green Dashed: Centered RMSE
Standard Deviation (Normalized)
Standard Deviation (Normalized)
Observed
DeepMPT_M
MPNN_GO_M
MPNN_GE_M
RF
XGB
GP
DeepMPT_S
MPNN_GO_S
MPNN_GE_S
DeepMPT_M (GROVER)
RF (GROVER)
XGB (GROVER)
GP (GROVER)
DeepMPT_S (GROVER)

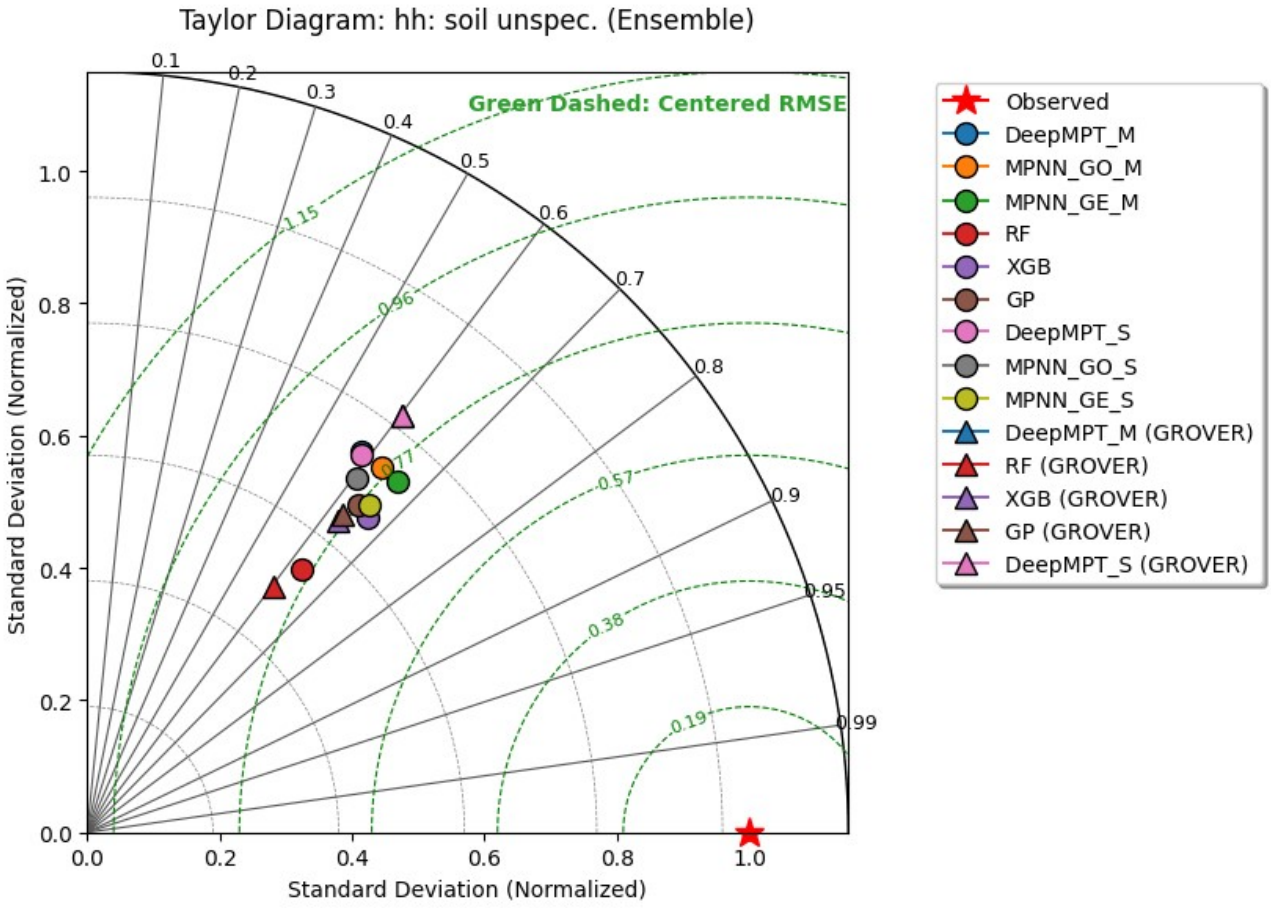

Taylor Diagram: hh: air unspec. (Ensemble)
Green Dashed: Centered RMSE
Standard Deviation (Normalized)
Standard Deviation (Normalized)
Observed
DeepMPT_M
MPNN_GO_M
MPNN_GE_M
RF
XGB
GP
DeepMPT_S
MPNN_GO_S
MPNN_GE_S
DeepMPT_M (GROVER)
RF (GROVER)
XGB (GROVER)
GP (GROVER)
DeepMPT_S (GROVER)

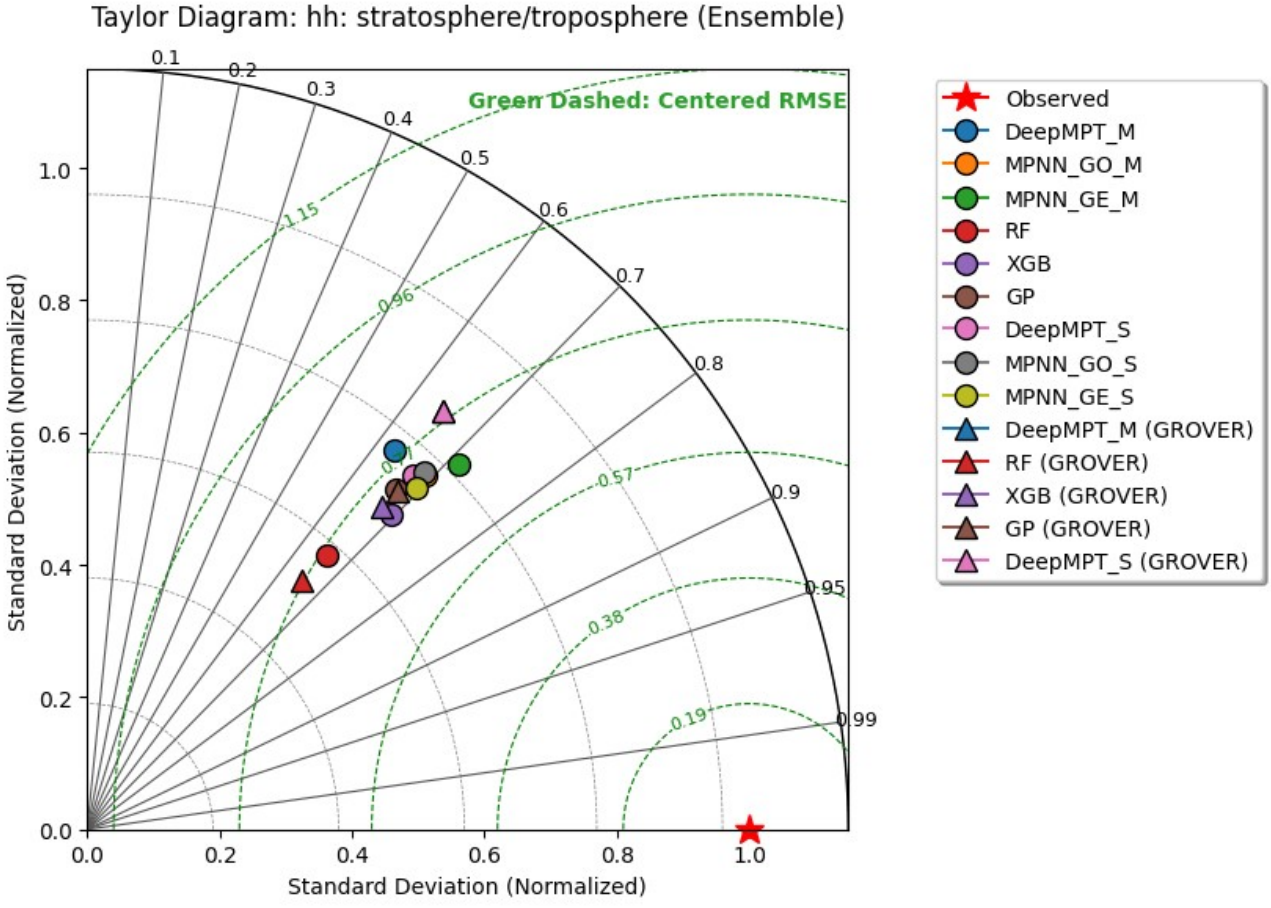

Taylor Diagram: hh: fresh water (Ensemble)
Green Dashed: Centered RMSE
Standard Deviation (Normalized)
Standard Deviation (Normalized)
Observed
DeepMPT_M
MPNN_GO_M
MPNN_GE_M
RF
XGB
GP
DeepMPT_S
MPNN_GO_S
MPNN_GE_S
DeepMPT_M (GROVER)
RF (GROVER)
XGB (GROVER)
GP (GROVER)
DeepMPT_S (GROVER)


Taylor Diagram: hh: non-agri soil (Ensemble)
Green Dashed: Centered RMSE
Standard Deviation (Normalized)
Standard Deviation (Normalized)
Observed
DeepMPT_M
MPNN_GO_M
MPNN_GE_M
RF
XGB
GP
DeepMPT_S
MPNN_GO_S
MPNN_GE_S
DeepMPT_M (GROVER)
RF (GROVER)
XGB (GROVER)
GP (GROVER)
DeepMPT_S (GROVER)


Taylor Diagram: hh: non-urban/high stacks (Ensemble)
Green Dashed: Centered RMSE
Standard Deviation (Normalized)
Standard Deviation (Normalized)
Observed
DeepMPT_M
MPNN_GO_M
MPNN_GE_M
RF
XGB
GP
DeepMPT_S
MPNN_GO_S
MPNN_GE_S
DeepMPT_M (GROVER)
RF (GROVER)
XGB (GROVER)
GP (GROVER)
DeepMPT_S (GROVER)


Taylor Diagram: hh: sea water (Ensemble)
Green Dashed: Centered RMSE
Standard Deviation (Normalized)
Standard Deviation (Normalized)
Observed
DeepMPT_M
MPNN_GO_M
MPNN_GE_M
RF
XGB
GP
DeepMPT_S
MPNN_GO_S
MPNN_GE_S
DeepMPT_M (GROVER)
RF (GROVER)
XGB (GROVER)
GP (GROVER)
DeepMPT_S (GROVER)


Taylor Diagram: hh: soil unspec. (Ensemble)
Green Dashed: Centered RMSE
Standard Deviation (Normalized)
Standard Deviation (Normalized)
Observed
DeepMPT_M
MPNN_GO_M
MPNN_GE_M
RF
XGB
GP
DeepMPT_S
MPNN_GO_S
MPNN_GE_S
DeepMPT_M (GROVER)
RF (GROVER)
XGB (GROVER)
GP (GROVER)
DeepMPT_S (GROVER)


Taylor Diagram: hh: stratosphere/troposphere (Ensemble)
Green Dashed: Centered RMSE
Standard Deviation (Normalized)
Standard Deviation (Normalized)
Observed
DeepMPT_M
MPNN_GO_M
MPNN_GE_M
RF
XGB
GP
DeepMPT_S
MPNN_GO_S
MPNN_GE_S
DeepMPT_M (GROVER)
RF (GROVER)
XGB (GROVER)
GP (GROVER)
DeepMPT_S (GROVER)

7. Differentiating Cancer and Noncancer Effects

To account for zero inflation and mixed discrete–continuous target distributions observed for selected toxicity CFs, a data-driven hurdle modeling strategy was implemented. For each target, the empirical distribution was first diagnosed to determine the presence of a hurdle structure (Figure 7-1 shows one target example.). When identified, the modeling task was decomposed into two sequential stages. In the first stage, a binary classification model was trained to predict the occurrence of non-zero toxicity values using a random forest classifier, thereby modeling the probability that a compound exhibits a measurable effect. In the second stage, conditional on a positive outcome, different algorithms were trained to predict continuous toxicity values (log-transformed where appropriate). If no hurdle structure was detected, a single regression model would fit directly to the continuous target. Median imputation was applied to molecular features prior to modeling, and model performance was evaluated separately for the classification (ROC–AUC) and regression (R^2, RMSE) components, with additional cluster-wise analysis used to assess local predictive behavior across chemically defined subspaces.

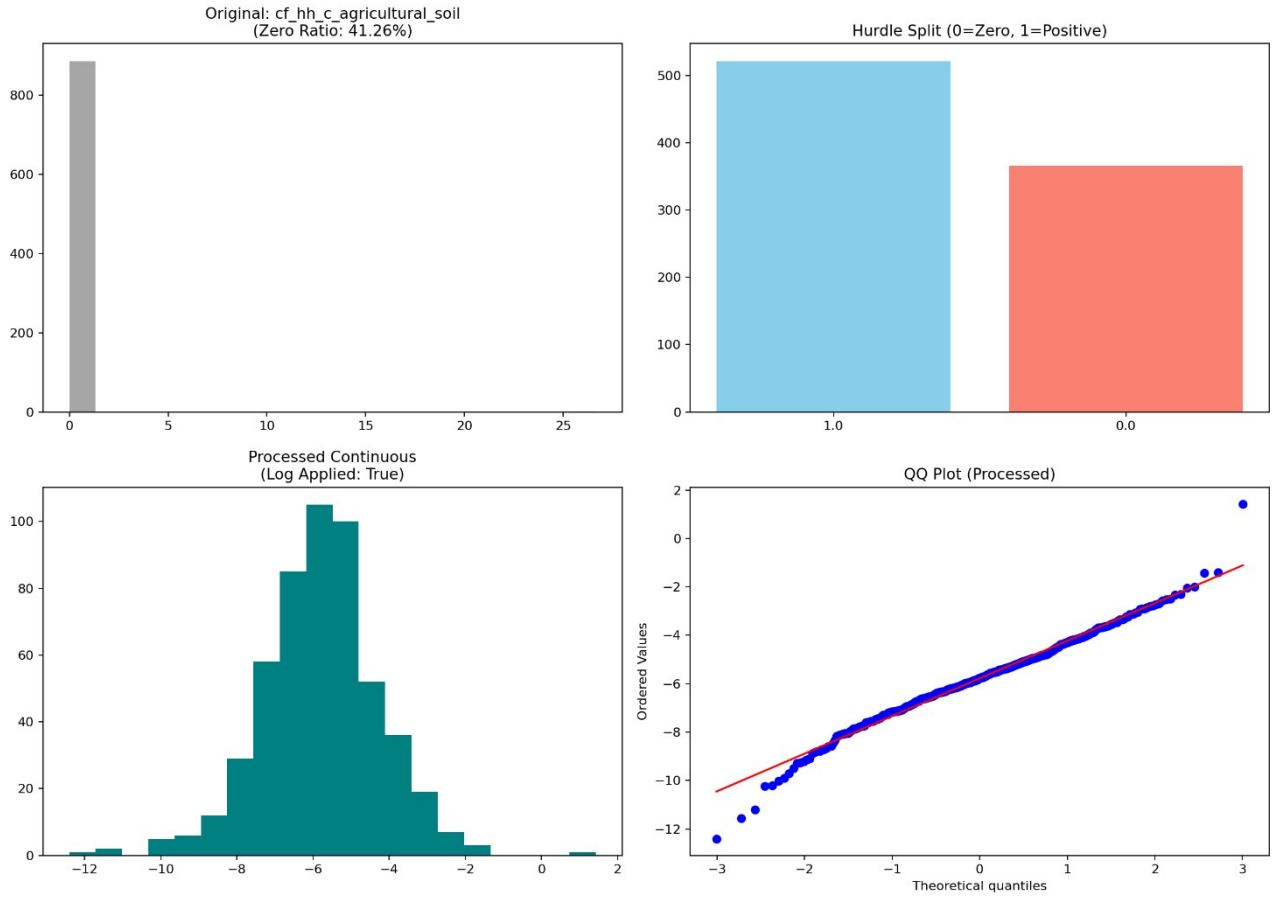


Figure 7-1 Diagnostic assessment and hurdle transformation of a zero-inflated toxicity characterization factor (up-left: distribution before transformation; up-right: zero and non-zero (indicate as 1) distribution; bottom-left: distribution of non-zero data after log-transformed; bottom-right: QQ plot for non-zero log-transformed data)

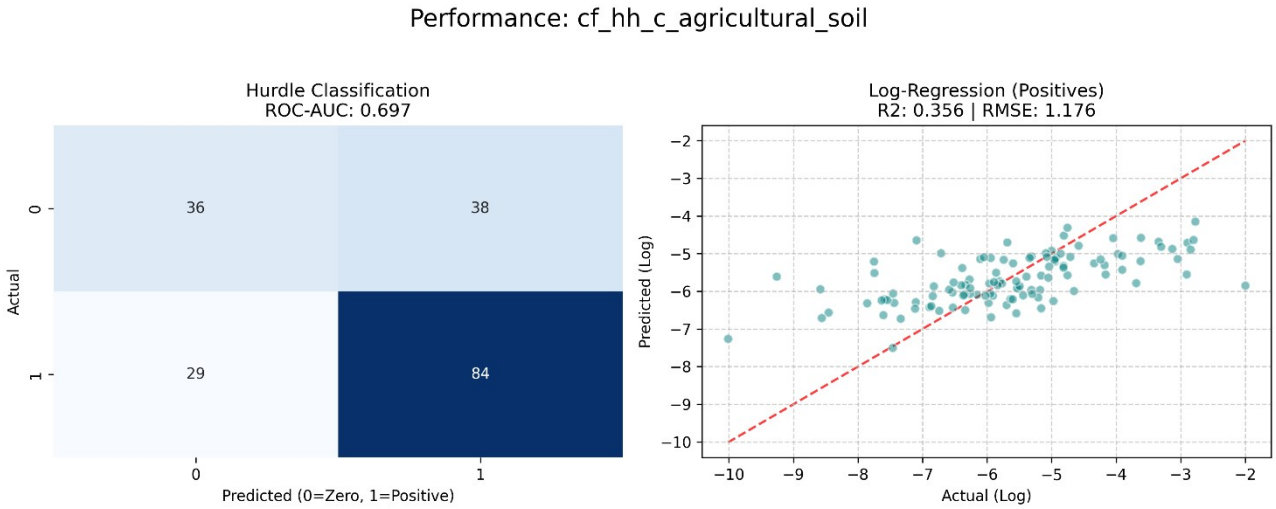


Figure 7-2 Performance of the hurdle modeling framework for human toxicity CF in agricultural soil (cf_hh_c_agricultural_soil)

The hurdle modeling approach achieves moderate but meaningful performance (Figure 7-2) for the human toxicity CF. The binary classification component discriminates between zero and non-zero CF values with a ROC–AUC of 0.697, indicating that the model can capture systematic differences between compounds with and without reported toxicity contributions. Conditional on positive observations, the regression model trained on log-transformed CF values yields an R^2 of 0.356 and an RMSE of 1.176, reflecting substantial variability in predicted magnitudes but preserving the overall trend across the observed value range. While uncertainty remains high for individual predictions, particularly at the extremes, the combined hurdle framework effectively separates occurrence from magnitude and provides non-zero toxicity estimates where conventional databases report zeros, supporting its use for screening and gap-filling applications in LCA.

Figure 7-3 heatmap compares R2 for human toxicity when cancer and non-cancer effects are separated. The results show that the best performance algorithms and data combinations reach 0.35-0.65. Comparing to modeling total cancer and noncancer effects in the main text, no significant improvement can be observed.

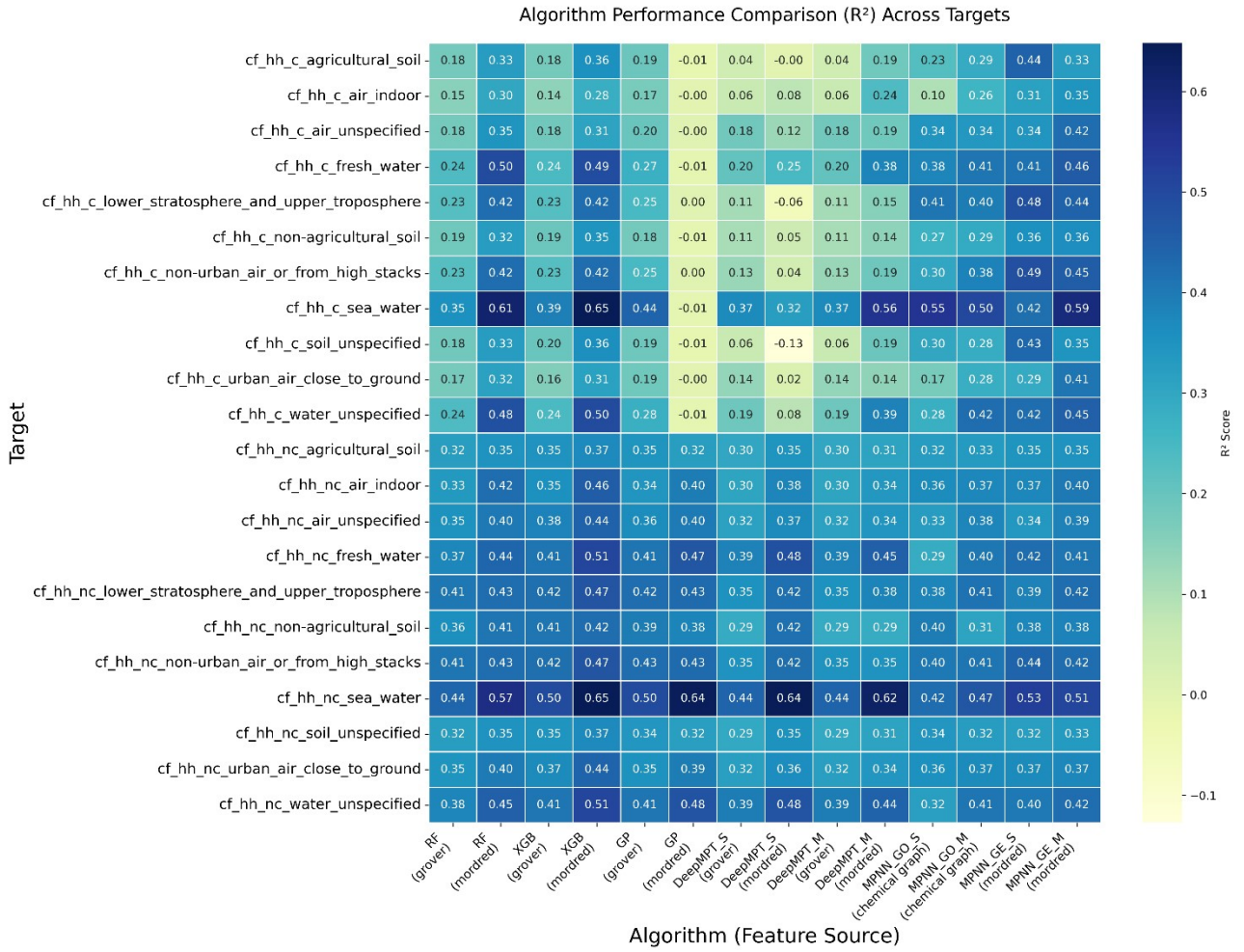


Figure 7-3 Heatmap of predictive performance (R^2) across multiple characterization factor (CF) targets for different learning algorithms and feature sources. Rows represent individual CF targets spanning ecotoxicity and human toxicity compartments, while columns correspond to model–feature combinations (e.g., DeepMPT, GP, MPNN, RF, and XGBoost using molecular descriptors or GROVER embeddings). Colour intensity and annotated values indicate R^2 scores, enabling a comparative assessment of algorithm robustness and feature representations across targets. (MPNN: Message-passing neural networks; GO: Graph only; GE: Graph and extra Mordred descriptors; M: Multi-target regression; S: Single-target regression; XGB: XGBoost; eco: ecotoxicity; hh: human health)

8. Life Cycle Assessment Case Study – Full Elaboration

The assessed case refers to a specific step of denim garment manufacturing, “garment washing”, which is performed to obtain esthetical and vintage looking effects on the finished denim jeans. Of the latter, an LCA study aligned as much as possible with PEF methodology has been executed. Figure S 2 shows the simplified foreground system for the two washing processes. The process is quasi equivalent for the two types of stone, with only some variations in the quantity and type of chemicals used. The functional unit (FU) is “1 pair of jeans washed”.

Included in the product system is also the onsite wastewater treatment plant which is used to recycle and reuse 90% of the washing output wastewater and releases treated effluents to the environment and sludge waste (in the pumice stone scenario only) to disposal.

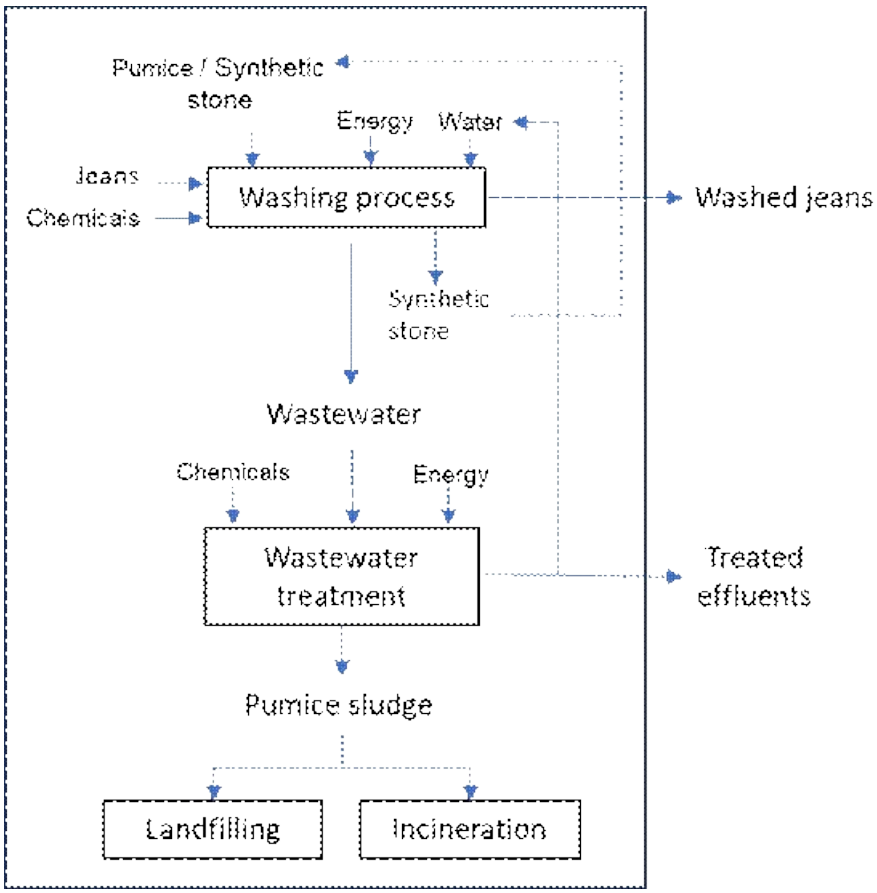


Figure S 2. EREKS washing process and product system (with shown foreground processes) considered in the case study LCA. The dashed arrow refers to the synthetic stone washing process only and indicates the reuse of the stones after the washing cycle.

When it comes to the inventory, foreground data were collected from the company EREKS. Background process amounts were estimated using the ecoinvent v3.10 (cut-off version) database. For consumables besides chemicals and machinery, respective ecoinvent processes were found as much as possible representative for Turkey. Building infrastructure was not covered. Table S 1 shows the chemical substances contained in the chemicals used in the product system and the corresponding ecoinvent v3.10 activity used to model their production impacts (i.e. supply impacts). The chemical composition of each chemical product was retrieved from their Material Safety Data Sheet (MSDS) and the mass of each chemical substance calculated according to its maximum concentration value.

Table S 1. Chemical substances used in the two washing processes and corresponding ecoinvent v3.10 (cut-off) activity to model the chemical product inventory.

Process	Chemical substance name	CAS	% Conc.	Ecoinvent activity
Washing	Phosphoric acid	7664-38-2	100	Phosphoric acid, industrial grade, without water, in 85% solution state {GLO}| market for phosphoric acid, industrial grade, without water, in 85% solution state | Cut-off, U
Washing	Polyester copolymer	139755-78-5	[35-40]	Methyl methacrylate {RER}| market for methyl methacrylate | Cut-off, U
Washing	Alcohols,C16-18,	68439-49-	[1-5]	Ethoxylated alcohol (AE>20)

	ethoxylated	6		{GLO}| market for ethoxylated alcohol (AE>20) | Cut-off, U
Washing	Sulfuric acid	7664-93-97	[1-5]	Sulfuric acid {RER}| market for sulfuric acid | Cut-off, U
Washing	Ceric sulfate tetrahydrate	10294-42-5	[1-3]	Cerium oxide {GLO}| market for cerium oxide | Cut-off, U
Washing	Cellulase	9012-54-8	2	Enzymes {GLO}| market for enzymes | Cut-off, U
Washing	2-Phenoxyethanol	122-99-6	<0.006	Phenoxy-compound {GLO}| market for phenoxy-compound | Cut-off, U
Washing	Poly(oxy-1,2-ethanediyl), α-tridecyl-ω-hydroxy-, branched / Alcohol, C12-14, ethoxylated	69011-36-5	[20-28]	Ethoxylated alcohol (AE11) {GLO}| market for ethoxylated alcohol (AE11) | Cut-off, U
Washing	Methylisothiazolinone	2682-20-4	[0-0.005]	-
Washing	Potassium Permanganate	7722-64-7	99	Potassium permanganate {GLO}| market for potassium permanganate | Cut-off, U
Washing	Poly(oxy-1,2-ethanediyl),α-hydro-ω-hydroxy- Ethane-1,2-diol, ethoxylated	25322-68-3	20	Ethylene glycol {GLO}| market for ethylene glycol | Cut-off, U
Washing	Isotridecanol, ethoxylated	69011-36-5	<10	Ethoxylated alcohol (AE11) {GLO}| market for ethoxylated alcohol (AE11) | Cut-off, U
Washing	Benzenesulfonic acid, C10-13-alkyl derivs., sodium salts	68411-30-3	< 2.5	Alkylbenzene sulfonate, linear, petrochemical {GLO}| market for alkylbenzene sulfonate, linear, petrochemical | Cut-off, U
Washing	Sodium chloride	7647-14-5	< 90	Sodium chloride, powder {GLO}| market for sodium chloride, powder | Cut-off, U
Washing	Sodium carbonate	497-19-8	[0-10]	Neutralising agent, sodium hydroxide-equivalent {GLO}| soda ash, light, crystalline, heptahydrate, to generic market for neutralising agent | Cut-off, U
Washing	Sodium hypochlorite	7681-52-9	[10-14]	Sodium hypochlorite, without water, in 15% solution state {RER}| market for sodium hypochlorite, without water, in 15% solution state | Cut-off, U
Washing	Disodium disulphite	7681-57-4	> 97	Disodium disulphite {GLO}| market for disodium disulphite | Cut-off, U
Washing	Quicklime	1305-78-8	< 3	Quicklime, milled, loose {CH}| market for quicklime, milled, loose | Cut-off, U
Washing	Sodium oxide	12401-86-4	< 1	Sodium oxide {RER}| market for sodium oxide | Cut-off, U
Washing	Hydrogen peroxide	7722-84-1	[30-70]	Hydrogen peroxide, without water, in 50% solution state {RER}| market for hydrogen peroxide,

				without water, in 50% solution state | Cut-off, U
Washing	Isotridecanol, ethoxylated	69011-36-5	[1-10]	Ethoxylated alcohol (AE11) {GLO}| market for ethoxylated alcohol (AE11) | Cut-off, U
Washing	Fatty alcohol ethoxylated	78330-20-8	[5-15]	Ethoxylated alcohol (AE11) {GLO}| market for ethoxylated alcohol (AE11) | Cut-off, U
Washing	(2-methoxymethylethoxy) propanol	34590-94-8	[1-5]	Dipropylene glycol monomethyl ether {RER}| dipropylene glycol monomethyl ether production | Cut-off, U
WWT	Polyaluminium chloride	1327-41-9	[40-60]	Polyaluminium chloride {GLO}| market for polyaluminium chloride | Cut-off, U
WWT	Anionic polyacrylamide	9003-0-8	NA	Polyacrylamide {GLO}| market for polyacrylamide | Cut-off, U
WWT	Octadecane-1ol ethoxylated	9005-00-9	[1-3]	-
WWT	Caustic soda	1310-73-2	[100]	Sodium hydroxide, without water, in 50% solution state {RER}| market for sodium hydroxide, without water, in 50% solution state | Cut-off, U
WWT	Hydrochloric acid	7647-01-0	[30]	Sodium hydroxide, without water, in 50% solution state {RER}| market for sodium hydroxide, without water, in 50% solution state | Cut-off, U
WWT	Sodium hypochlorite	7681-52-9	[10-14]	Sodium hypochlorite, without water, in 15% solution state {RER}| market for sodium hypochlorite, without water, in 15% solution state | Cut-off, U

Following the work of Roos et al.,[6] we then matched each chemical substance identified from the MSDS with the corresponding degradation/transformation product. For substances for which a transformation product was not found, it was assumed that this would be released as such after the washing process (Table S 2). The final quantity of each chemical substance used in the processes was calculated from the total chemical product consumption for one year of washing and its concentration in the chemical product as reported in the MSDS.

The modelling of the final mass of each chemical substance released in the environment (i.e. elementary flow) was based on the work of Roos et al.[6]: a 90% degradation efficiency from the wastewater treatment plant was applied for emissions to water; 0.1% emissions to air of chemical substances reported as volatile only; and 0.01% if volatile and reactive. However, since EREKS does not discharge directly in the environment but in a second municipal wastewater treatment plant, to account for this in the final quantities ultimately released in the environment, a further 90% degradation rate was applied to water emissions. The final LCA results are presented in Table S 3 as generated using Simapro version 9.6.0.1.

Table S 2. The output elementary flows considered for both pumice and synthetic stone washing considered as part of the product system inventory

Input chemical	Emissions to air	Emissions to water
Phosphoric acid	-	Phosphoric acid
Polyester copolymer	-	-
Alcohols, C16-18, ethoxylated	-	Alcohols, C12-14, ethoxylated
Sulfuric acid	Sulfuric acid	Sulfuric acid
Ceric sulfate tetrahydrate	-	Ceric sulfate tetrahydrate
Cellulase	-	Cellulase
2-Phenoxyethanol	-	2-Phenoxyethanol
Poly(oxy-1,2-ethanediyl), α-tridecyl-ω-hydroxy-, branched / Alcohol, C12-14, ethoxylated	-	Alcohols, C12-14, ethoxylated
Methylisothiazolinone	-	2-methyl-4-isothiazolin-3-one
Potassium Permanganate	-	Potassium Permanganate
Poly(oxy-1,2-ethanediyl),α-hydro-ω-hydroxy-Ethane-1,2-diol, ethoxylated	-	Poly(oxy-1,2-ethanediyl),α-hydro-ω-hydroxy-Ethane-1,2-diol, ethoxylated
Isotridecanol, ethoxylated	Isotridecanol, ethoxylated	Isotridecanol, ethoxylated
Benzenesulfonic acid, C10-13-alkyl derivs., sodium salts	-	Benzenesulfonic acid, C10-13-alkyl derivs., sodium salts
Sodium chloride	-	Sodium chloride
Sodium carbonate	-	Sodium carbonate
Sodium hypochlorite	-	Sodium hypochlorite
Disodium disulphite	-	Disodium disulphite
Quicklime	-	Calcium oxide
Sodium oxide	-	-
Hydrogen peroxide	Hydrogen peroxide	Hydrogen peroxide
Fatty alcohol ethoxylated	Alcohols ethoxylated	Alcohols ethoxylated
(2-methoxymethylethoxy)propanol	-	(2-methoxymethylethoxy) propanol
Polyaluminium chloride	-	Polyaluminium chloride
Anionic polyacrylamide	Acrylamide	Acrylamide
Octadecane -1ol ethoxylated	-	Octadecane -1ol ethoxylated
Caustic soda	-	Sodium hydroxide
Hydrochloric acid	-	Hydrochloric acid
Sodium hypochlorite	-	Sodium hypochloride

9. Procedure to include novel characterization factors (CFs) of chemical substances used in an industrial process

Overall approach:

To evaluate the potential environmental impacts of a chemical substance used in a process, it is necessary to: (**1) quantify the mass being released of this chemical substance and that of related transformation/degradation products into the environment; (2) quantify the associated environmental impact using a specific CF**.[1]

(1) Quantify the compounds and mass being released

9.1. Identifying the compounds (chemical substance & transformation products) emitted

Often, chemical substances used in the production process (e.g. washing of a textile or glueing of plywood) are not the same as the chemicals emitted by the process, because the initial chemical will change in different reaction products (transformation/degradation products). The proper identification of those transformation products will allow to precisely characterize the environmental impacts associated with the use of an input chemical.

The list of chemicals should be first screened for any corresponding transformation products. For the textile case studies this was done using the list of transformation products provided in the Supplementary Material of Roos et al.[6] (! This article is open access and you can find its supplementary material in the same link).

Similarly, a literature search for a certain relevant industrial sector could allow for the identification of potential transformation products associated with other sectors.

If the modelled chemical substance does not generate transformation products but it is directly emitted as such to the environment, or no literature sources can be identified, then the basic assumption could be that the initial input chemical substance is also present in the process' emissions.

!If you have no idea on transformation products, then just consider the chemical substance!

9.2. Identifying the amounts emitted to certain compartments (air, water etc.)

After having identified the types of compounds emitted, the emission's amounts to certain compartments should be estimated. The mass of chemical emitted to the environment will depend on the industrial sector and production process under study. For example, for textile there are both air emissions and water emissions of chemical substances; while for woodworking, the emissions are mainly air emissions from the pressing process. Assumptions must be made to model the mass of chemical substance being emitted to the environment. The textile case study uses the assumptions from Roos et al.[6] See Figure 1. !Yet, the factors of Roos et al.[6] could also be used for other sectors on a case by case basis!.

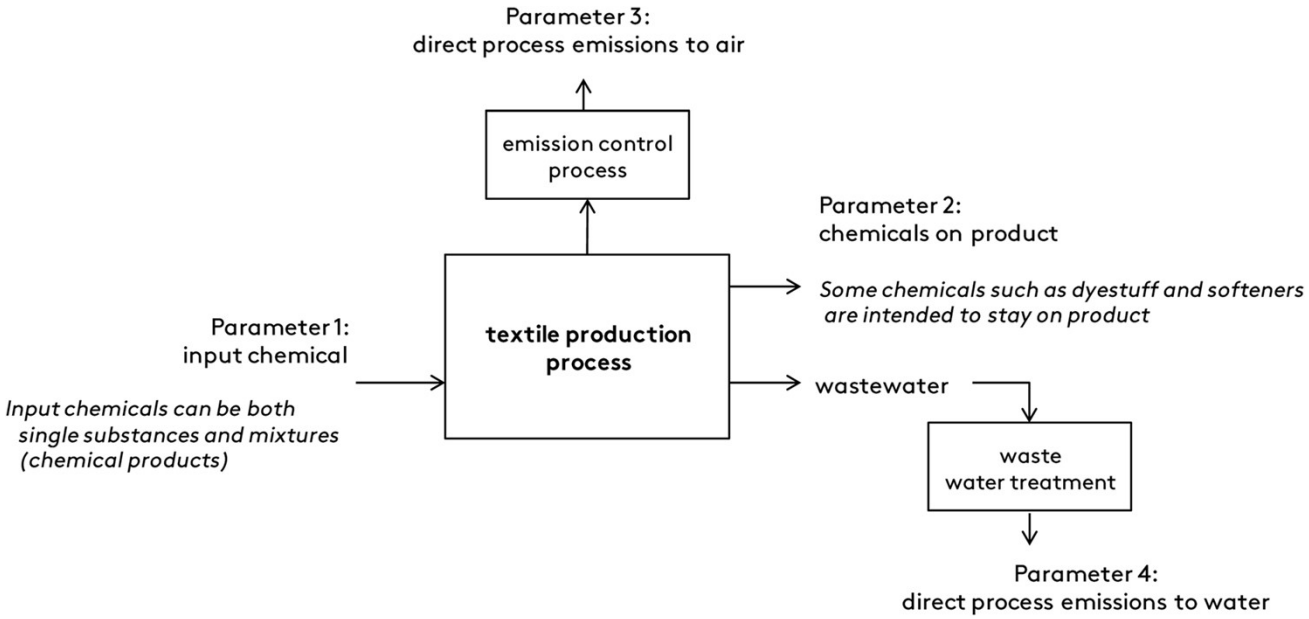


Figure 1. Overview of emission parameters considered for chemicals in the textile sector (with implicit consideration of transformation products), copied from Roos et al.[6], *serving as an example.*

(2) define the associated environmental impact using CF.

To define the impact:

emitted amount/mass of compound X to compartment Y * CF of compound X for compartment Y

!This does not necessarily need to be done in LCA software, one can also just do this simple multiplication in Excel!

For the chemical substance and transformation products, the CF should either be provided by the impact method or can be predicted as done in this work.

OPERATIONAL PROCEDURE FOR SOFTWARE

1) In your software, add to your product system or to the specific production process, the type and quantity of chemical substances that have been identified as being emitted. Those elementary flows should be added as direct outputs to the biosphere. Search for either the transformation products or the initial input chemical substance. The chemical of interest should be already included in the biosphere dataset of your software.
2) If the chemical substance is not present, then this must be added as a new elementary flow to the biosphere. Create a new elementary flow by adding the chemical name and the CAS number of the chemical in the biosphere dataset in your software.
3) After having added the chemical substance to the biosphere as elementary flow, add the CF for the given impact category. Those should be added to the method of your software so to quantify the impacts in terms of *Ecotoxicity, freshwater* and *Human toxicity, non-cancer*. The new CFs must be added directly to the impact assessment method (e.g. Environmental Footprint) in your software.

Operational Procedure specific for SimaPro modelling

1) Open the process activity in which the chemicals are being used, or the product system of your activity. In the **outputs section** you will have to **add** all the direct elementary flows to the environment, indicated as *Emissions to air*, *Emissions to water*, *Emissions to soil*.
2) Type to search or scroll the list of substances shown. Check the matching substance by verifying the CAS number. If the substance is already present in the list, click on "unspecified" for the sub-compartment selection and then click on "select" to add the elementary flow in the process activity.
3) Assign the corresponding mass to the elementary flow according to the emissions scenarios that have been defined by the optimization modelling or by literature search as discussed above.
4) Run your impact assessment

If the substance is not already included in the elementary flows list, then this means that it is not present and must be manually added.

1) In the **Methods tab** make a copy of the EF3.1 method (make sure to use a different name for the copy).
2) Open the new copy using the **Edit button**.
3) Click on the **Characterization tab**. In this tab you will see the list of all elementary flows to all the environmental compartments for each environmental impact category. In our case, we are only interested in the **Toxicity impacts** (eco and human).
4) Select the impact category, e.g. *Ecotoxicity, freshwater – part 2*, this will open a list of all elementary flows related to it. Select **Add** (the one in the middle of the page) to add a new substance to the list of elementary flows, then select the main compartment of emissions, e.g. **waterborne emissions**, then select **New**.
5) Add the new substance name and CAS number. This will add the substance to the list, double click on it, this should point you to where to insert the CF value, e.g. for freshwater ecotoxicity from water emissions.

[1] If the CF is omitted during the life cycle assessment stage, then its impacts on the environment and people will not be characterized. Furthermore, if the given chemical substance (elementary flow) does not have an associated characterization factor (CF) for the given impact category, the associated potential environmental impact will not be characterized.

Reference

Breiman L (2001) Random Forests. Mach Learn 45:5–32. https://doi.org/10.1023/A:1010933404324

Chen T, Guestrin C (2016) XGBoost: A Scalable Tree Boosting System. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, San Francisco California USA, pp 785–794

Cremer J, Medrano Sandonas L, Tkatchenko A, et al (2023) Equivariant Graph Neural Networks for Toxicity Prediction. Chem Res Toxicol 36:1561–1573. https://doi.org/10.1021/acs.chemrestox.3c00032

Danieli MG, Tonacci A, Paladini A, et al (2022) A machine learning analysis to predict the response to intravenous and subcutaneous immunoglobulin in inflammatory myopathies. A proposal for a future multi-omics approach in autoimmune diseases. Autoimmun Rev 21:103105. https://doi.org/10.1016/j.autrev.2022.103105

Gilmer J, Schoenholz SS, Riley PF, et al (2017) Neural Message Passing for Quantum Chemistry

Guo W, Liu J, Dong F, et al (2023) Review of machine learning and deep learning models for toxicity prediction. Exp Biol Med 248:1952–1973. https://doi.org/10.1177/15353702231209421

Heid E, Greenman KP, Chung Y, et al (2024) Chemprop: A Machine Learning Package for Chemical Property Prediction. J Chem Inf Model 64:9–17. https://doi.org/10.1021/acs.jcim.3c01250

Iliadis D, De Baets B, Waegeman W (2023) DeepMTP: A Python-based deep learning framework for multi-target prediction. SoftwareX 23:101516. https://doi.org/10.1016/j.softx.2023.101516

Rasmussen CE, Christopher KIW (2006) Gaussian Processes for Machine Learning. MIT Press