PESTICIDE LEACHABILITY ANALYSIS WITH MACHINE LEARNING AND THEORETICAL INDEX

Isaiah Guenther, Naaran Brindt, Tammo S. Steenhuis, Brian K. Richards, Anna L. Schatz, Steven Pacenka
Soil & Water Laboratory, Biological and Environmental Engineering Department - Cornell University
March 4, 2025 - With and for New York State Department of Environmental Conservation, Bureau of Pesticides Management
More project info: https://soilandwaterlab.cornell.edu/upstate-new-york-pesticides-in-water-2021-2025/

BACKGROUND

Pesticides pose a threat to groundwater in New York (NY) State and beyond. Funded by the NYSDEC, we are monitoring pesticide contamination in upstate NY groundwater and using the data to test if machine learning (ML) models can predict leaching events with chemical and site characteristics.

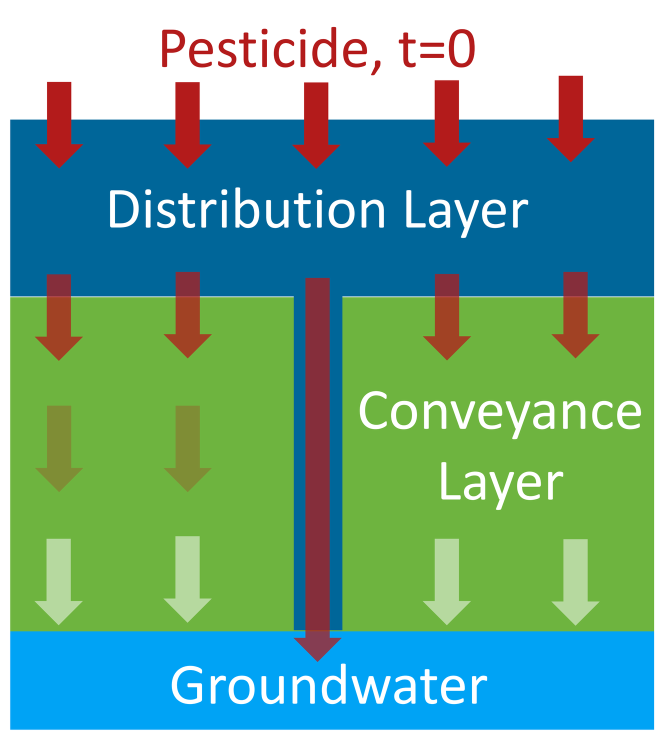
Objectives

- Develop ML model that predicts pesticide leaching
- Compare ML to theoretical leaching model

MODELS

Theoretical Groundwater Ubiquity Score (TGUS)

Developed by Steenhuis et al., 2024



$$t_{LRP} = 3.3 t_{\frac{1}{2}} log_{10}\left(\frac{\lambda\Phi}{K_{oc} f_{oc} \rho}\right)$$

$t_{\frac{1}{2}} = soil\ half\_life[days]$, $\lambda = preferential\ flow\ fraction$,

$\Phi, = \frac{application\ rate}{detection\ limit\ * distribution\ layer\ thickness}$,

$K_{oc} = organic\ carbon\ partition\ coef. \left[\frac{ml}{g}\right]$,

$f_{oc} = soil\ organic\ carbon\ fraction$, $\rho = soil\ bulk\ density\ [\frac{Mg}{m^3}]$

Fig 1: Visualization of TGUS model. The Leaching Risk Period, t_{LRP}, is the time window (days) after application that pesticides can potentially leach to groundwater.

Machine Learning - Extreme Gradient Boosting (XGB)

Ensemble ML algorithm, Chen and Guestrin, 2016

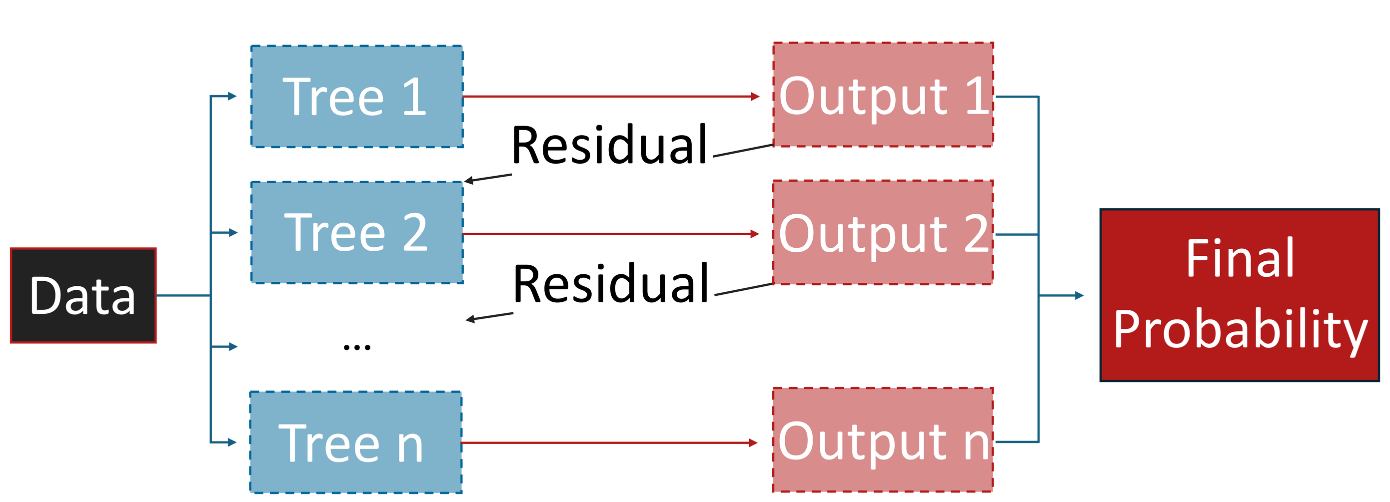


Fig 2: XGB classification flowchart. The final class probability is the sigmoid function applied to the sum of n log-odds outputs from each tree.

PROCEDURE

Dataset

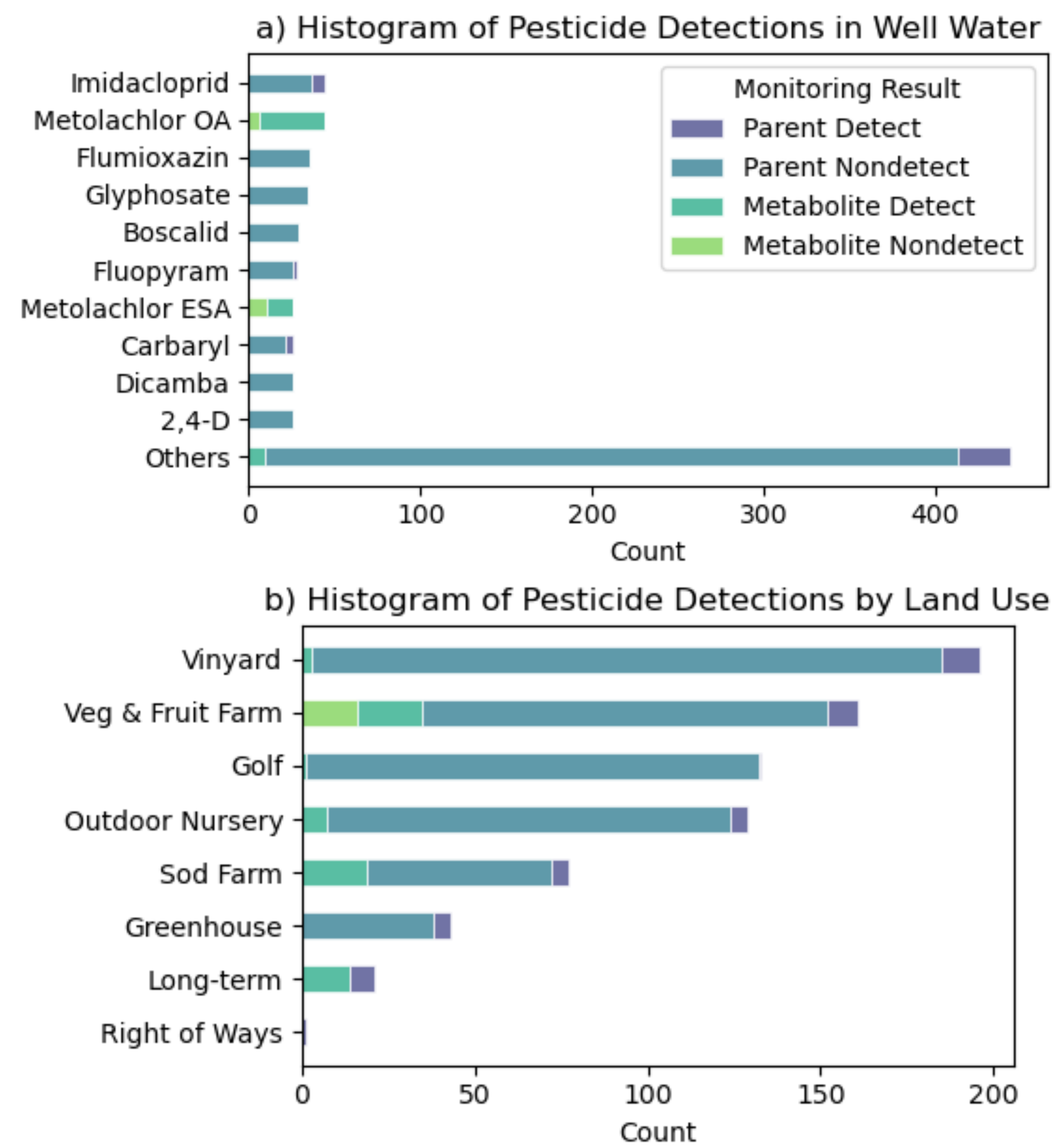


Fig 3: Dataset inspection. (a) Monitoring results across pesticides. (b) Monitoring results across land use. 107 pesticide detects in total, 654 nondetects.

Methods

TGUS	1. Predict leaching with t_{LRP} : $pred. = \begin{cases} detect, & t_{LRP} \geq 100\ days \\ nondetect, & t_{LRP} < 100\ days \end{cases}$ 2. Calculate prediction performance
XBG	1. Tune hyperparameters with grid search 2. CV train on 80% stratified data split 3. Calculate prediction performance on 20% held-out set, threshold = 0.05 4. Repeat 100 total iterations steps 2-3, record all metrics and outputs

Table 1: Model prediction steps. TGUS and XGB were used for binary classification, with targets as detect (leacher) or nondetect (nonleacher).

RESULTS

Model	Peformance Metrics		
	Precision [%]	Recall [%]	F_β [%]
TGUS	20.4	99.1	56.0
XGB, avg	58.4	87.1	79.0
XGB, best	71.0	100.0	92.4

$Precision = \frac{TP}{TP+FP}$, $Recall = \frac{TP}{TP+FN}$, $F_\beta = (1+\beta^2) * \frac{Precision\ * Recall}{(\beta^2 * Precision) + Recall}$ ↳ $\beta = 2$

Table 2: Performance metrics. XGB outperformed TGUS overall. Metrics are derived from model outcomes: true positive (TP), false positive (FP), true negative (TN), false negative (FN).

Fig 4: Best XGB model SHAP beeswarm plot, held-out set. Each feature's contribution to the final leaching probability.

Conclusions

- XGB outperformed TGUS in most metrics
- XGB learned patterns consistent with TGUS theory
- TGUS may benefit from additional variables or better incorporation of current variables

Considerations

- Limited dataset and scope of ML tools
- No analysis of feature dependences
- Key TGUS variables (λ, distribution layer thickness) not readily available