FINAL PROJECT REPORT

Predicting Pesticide Contamination in New York Aquifers

ORIE 4741: Learning with Big, Messy Data
Taught by Dr. Haiyun He

Completed by:

Isaiah Guenther
B.S. in Biological Engineering from Cornell University - '24

With Assistance from:

Laura Hebrant, Tammo Steenhuis, Brian Richards,
Steve Pacenka, Naaran Brindt, Ani Schatz, Denise Nyangwechi

May 2024

Table of Contents

List of Figures & Tables

Abbreviations

Acc. = Accuracy; Aq. Vuln. = Aquifer Vulnerability; Avg. = Average; D. Class = Drainage Class; D. Limit = Detection Limit; GUS = Groundwater Ubiquity Score; P. Coeff. = Partitioning Coefficient; Sc. = Scenario; Soil H.L. = Soil Half-life; TGUS = Theoretical Groundwater Ubiquity Score; Tr. = Train; Tst. = Test.; Val. = Validation

Resources & More Project Information

GitHub Repository - https://github.com/izguenther6/orie4741-final
Cornell Soil & Water Lab - https://soilandwaterlab.cornell.edu/
New York State Department of Environmental Conservation - https://dec.ny.gov/

Abstract

Over the past three years, the Cornell Soil and Water Lab (SWL) has been working with the New York State Department of Environmental Conservation (DEC) to test for pesticide contamination in Upstate New York aquifers. This collaborative effort aims to better understand the effectiveness of current pesticide regulations and to explain leaching events through chemical transport theory. We used the test results to build three machine learning models that performed binary classification of whether or not a pesticide will leach with high accuracy and provided insight into important prediction features . The best model, boosted gradient classification, had an average prediction accuracy upwards of 98% . These models have also verified the prediction efficacy of a new, theoretical groundwater ubiquity score that SWL members recently derived. This report covers the model building process and subsequent results analysis. Note that this was also completed as a final project for *ORIE 4741: Learning with Big, Messy Data*.

Data

The dataset contained results from 528 pesticide tests conducted on groundwater samples from June 2022 to November 2023. The data was represented in a matrix with each row as a separate test for a singular pesticide and 30 feature columns containing information about the active ingredients, sample location, or test procedure. One additional column contained the pesticide concentration detected in the test, 96 of which were above zero. For this project, only 7 of the 30 features were used in different combinations and are defined below:

- *Soil Half-Life* is the amount of time it takes for a pesticide to degrade to half its initial concentration in soil (Gustafson, 1989). The longer a pesticide takes to degrade, the more likely it is to leach.

- *Partitioning Coefficient* is an indicator of how much a pesticide binds to the surrounding soil substrate (Gustafson, 1989). The less a pesticide binds to the soil, the more likely it is to leach.

- *Groundwater Ubiquity Score (GUS)* is an indicator of a pesticide's likelihood to leach into groundwater supplies. It was derived empirically by Gustavson et al. in 1989 using soil half-life and partitioning coefficient values and has since been a commonly used metric to assess pesticide leaching events. Pesticides with a GUS under 1.8 are considered to be non-leachers. The equation is as follows, where $t_{\frac{1}{2}}$ is the soil half-life and K_{OM} is the partitioning coefficient:

$$GUS = log(t_{\frac{1}{2}}) * (4 - log(K_{OM})) \qquad 1$$

- *Theoretical Groundwater Ubiquity Score (TGUS)* is an indicator of a pesticide's likelihood to leach into groundwater supplies. It was recently derived by SWL members in an attempt to theoretically explain the GUS equation. TGUS uses soil half-lives and partitioning coefficients, as well, but also includes other theoretical chemical transport values and yields some different results than GUS. Pesticides with a TGUS score less than 0.3 multiplied by the amount of days from first application to first significant water infiltration event are considered to non-leachers. The equation can be found below, where $t_{\frac{1}{2}}$ is the soil half-life and K_{OM} is the partitioning coefficient. Note that the other visible constants and operators are derived using chemical transport theory:

$$TGUS = t_{\frac{1}{2}} * (3.4 - log(K_{OM})) \qquad 2$$

- *Aquifer Vulnerability* is a categorical description of how susceptible an aquifer is to contamination from human activities. The categories are high, medium, and low, and were created by SWL members using DEC guidelines. It is expected that aquifers with higher vulnerability will experience more pesticide leaching events.

- *Drainage Class* is a soil parameter that categorically describes how its water table fluctuates, as defined by the Natural Resources Conservation Service in the United States Department of Agriculture. Six classes are used: Excessively Well Drained, Well Drained, Moderately Well Drained, Somewhat Poorly Drained, Poorly Drained, and Very Poorly Drained. More information on these categories can be found here.

- *Detection Limit* is the lowest pesticide concentration that the analysis equipment is able to measure in a groundwater sample. Although this information does not provide any theoretical insight, SWL members believe that more pesticides will be detected as the detection limit decreases.

Project questions:

1. How well can we predict pesticide leaching events?

2. Does TGUS perform as well as GUS for predicting pesticide leaching events?

3. What features are most important for predicting pesticide leaching events?

Preprocessing

Feature engineering & dataset configuration

Soil half-lives, partitioning coefficients, GUS, TGUS, and detection limits were all normalized for use in the algorithms, whereas aquifer vulnerability and drainage class were one-hot encoded. To set up the data for binary classification, a new column was created that contained '1' if a test concentration was above zero, and '-1' otherwise. An offset feature column was also appended to each dataset. Other feature engineering information can be found in the data_organization.ipynb file located in this project's GitHub repository.

To evaluate how well TGUS predicts leaching compared to GUS, different datasets were used in each machine learning model that isolated the respective ubiquity scores with the other features. Since both equations are derived from soil half-lives and partitioning coefficients, these raw values were also analyzed separately with the other features. Another dataset included both ubiquity scores, raw values, and other features to analyze how well everything predicted outputs together. Table 1 below summarizes the four different datasets used. Table 2 below shows a sample of the GUS-focused dataset.

Dataset	Feature						
	Soil H.L.	P. Coeff.	GUS	TGUS	Aq. Vuln.	D. Class	D. Limit
All	X	X	X	X	X	X	X
GUS Focus			X		X	X	X
TGUS Focus				X	X	X	X
Raw Focus	X	X			X	X	X

Table 1: *Summary of different datasets used in models*. These different datasets allowed for performance comparison between GUS and TGUS. We could see how well they worked together and separately. An 'X' in a cell indicates that the feature was used in the dataset. The dimension sizes were: 'All' = 14; 'GUS Focus' = 11; 'TGUS Focus' = 11; 'Raw Focus' = 12.

GUS-Focused Dataset					
Index	Moderately Well-Drained	Poorly Drained	...	Normalized GUS	Normalized Detection Limit
0	1	0	...	-0.143	-0.452
1	1	0	...	-0.465	-0.452
2	1	0	...	-0.143	-0.452
3	1	0	...	-0.465	-0.452

Table 2: *Sample from GUS-focused dataset.* This table represents what the first three rows of the GUS-focused dataset looked like. TGUS and the raw values replaced GUS in the other focused datasets. Note that the ellipses represent the other one-hot encoded data in the matrix.

Data visualization

Graphical representations were difficult to interpret, as many of the separate tests were for the same pesticide or ones with the same feature values, resulting in overlaid data. However, the test results plotted on graphs of the normalized detection limits versus the normalized ubiquity scores suggested a potentially high degree of linear separation, especially on the detection limit axis. This motivated us to investigate both linear and more complex algorithms.

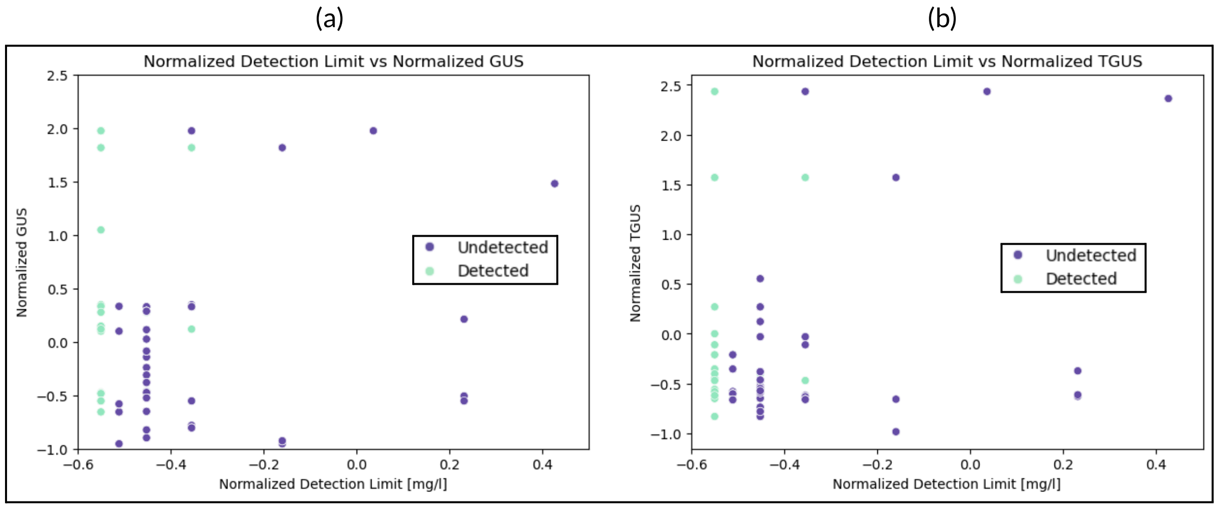

Figure 1: ***(a)*** *Results from pesticide tests on graphs of normalized detection limits versus normalized GUS.* ***(b)*** *Results from pesticide tests on graphs of normalized detection limits versus normalized TGUS.* The data showed there was a potentially high degree of linear separation along the detection limit axis, although not perfect. This motivated us to put the data into both linear and more complex algorithms.

Models and Results

For each model and dataset, the output space was a binary classification of whether or not a pesticide was detected in groundwater, and this procedure was followed:

1. Assign 80% of the data randomly as the training set
2. Perform 6-fold cross validation on the training set to obtain a final model and record the best training/validation accuracies
3. Test and record the final model's accuracy on remaining 20% of data
4. Repeat the previous steps 100 times and record all accuracies…keep the model with the best testing accuracy, as well as the associated training/validation accuracies

Note that 'testing accuracy' in reference to the models refers to the final model's accuracy in classifying the remaining data, and does not refer to the actual pesticide concentration test performed on the groundwater sample in the lab. All accuracies were calculated using zero-one loss.

Perceptron

The first model was built using the perceptron algorithm because of the suspected high degree of linear separability in the data. Results are summarized in Table 3 below.

Dataset	Perceptron Results [%]					
	Avg. Tr. Acc.	Avg. Val. Acc.	Avg. Tst. Acc.	Best Sc. Tr. Acc.	Best Sc. Val. Acc.	Best Sc. Tst. Acc.
All	93.1	95.9	90.7	93.0	95.7	98.1
GUS Focus	93.1	95.9	90.2	94.0	95.8	99.1
TGUS Focus	92.0	94.7	87.4	93.0	94.4	98.1
Raw Focus	93.2	95.7	89.3	93.0	95.7	98.1

Table 3: *Results from implementation of the perceptron algorithm.* All datasets led to very successful outcomes. The best case scenario columns all correspond to the same fitted perceptron model and were identified based on the highest test accuracy.

All dataset combinations resulted in highly accurate models, with balanced training, validation, and testing accuracies. The last three columns of the best scenario all corresponded to the same fitted perceptron model and were identified as described in the procedure above. It's important to consider the best scenario, as outliers in the data can significantly worsen perceptron's performance and misrepresent its efficacy. The average and best accuracies together showed that the data had a high degree of linear separation.

Table 3 shows that all average accuracies using the TGUS dataset were lower than the rest.. To investigate this further, we performed two-sided, independent T-tests between each dataset's average training, validation, and testing accuracies from the 100 perceptron iterations. A T-test is a statistical procedure with a null hypothesis that two sample means are not statistically different from each other (Kim, 2015). For this project, if the test's p-value was below an alpha level of 0.05, then the null hypothesis was rejected. The resulting p-values are shown in Table 4 below.

Data Portion	T-Test p-values					
	All vs GUS	All vs TGUS	All vs Raw	GUS vs TGUS	GUS vs Raw	TGUS vs Raw
Test	0.079	0.001	0.072	0.111	0.871	0.173
Validate	0.628	< 0.001	0.083	< 0.001	0.187	0.009
Train	0.922	< 0.001	< 0.001	< 0.001	< 0.001	0.002

Table 4: *T-test p-values between 100 accuracies in each data portion after perceptron algorithm implementation.* The null hypothesis is that the means are not different. The alpha level is 0.05. Null hypotheses that are rejected are highlighted in gold. There is conflicting evidence on how TGUS compares to GUS.

The T-tests had conflicting outcomes. When comparing the GUS- and TGUS-focused datasets directly, the null hypothesis was rejected for the validation and training accuracies, but not for the testing accuracies. The null hypothesis was rejected in every data portion when comparing the all-values dataset to the TGUS-focused dataset, but not rejected when compared to the GUS-focused dataset. Both null hypotheses were rejected when comparing the TGUS-focused dataset and the GUS-focused dataset to the raw-focused dataset in the training data portion but only rejected for TGUS in the validation portion. With all of these outcomes considered, it was possible that TGUS had a lower predicting power than GUS or the raw values, but this is inconclusive.

Support Vector Classifiers

Support vector classification (SVC) with a polynomial kernel was chosen as the next algorithm due to its increased complexity and outlier robustness compared to perceptron. The slack and gamma values were set to 0.1 and 10, respectively. The results are shown below in Table 5.

Dataset	SVC results [%]					
	Avg. Tr. Acc.	Avg. Val. Acc.	Avg. Tst. Acc.	Best Sc. Tr. Acc.	Best Sc. Val. Acc.	Best Sc. Tst. Acc.
All	99.7	99.2	96.9	99.0	100.0	100.0
GUS Focus	99.2	98.9	96.4	99.0	100.0	100.0
TGUS Focus	99.0	99.2	96.8	98.9	98.6	100.0
Raw Focus	99.6	99.0	96.5	99.0	98.6	100.0

Table 5: *Results from implementation of the SVC algorithm.* All datasets led to very successful outcomes. The best case scenario columns all correspond to the same fitted SVC model and were identified based on the highest test accuracy.

The SVC model proved to be much better than perceptron on average. Perfect classification of the test data occurred twice for the all-value dataset, once for the GUS-focused dataset, thrice for the TGUS-focused dataset, and six times for the raw-focused dataset. The same T-test procedure was applied to the SVC accuracy datasets, and the resulting p-values are shown below in Table 6.

Data Portion	T-Test p-values					
	All vs GUS	All vs TGUS	All vs Raw	GUS vs TGUS	GUS vs Raw	TGUS vs Raw
Test	0.051	0.572	0.060	0.140	0.857	0.171
Validate	0.006	1.000	0.057	0.006	0.379	0.634
Train	< 0.001	3.257	0.193	< 0.001	< 0.001	5.002

Table 6: *T-test p-values between 100 accuracies in each data portion after SVC algorithm implementation.* The null hypothesis is that the means are not different, while the alternative is that they are. The alpha level is 0.05. Null hypotheses that are rejected are highlighted in gold. There is conflicting evidence on how TGUS compares to GUS.

Again, we saw conflicting evidence on how GUS compared to TGUS. In the validation and training data portions, the null hypothesis was rejected when comparing the GUS-focused dataset to the all-value dataset, suggesting that GUS performed worse than TGUS when the average accuracies are considered. The null hypothesis was also rejected when comparing the TGUS-focused dataset to the GUS-focused dataset in the validation and training data portions, but with opposite implications when the average accuracies are considered. Moreover, the null hypothesis was rejected when comparing the raw-focused dataset to the GUS-focused dataset in the training portion, but not for the TGUS-focused dataset, suggesting that GUS performs worse when the average accuracies are considered. This evidence overall suggested antithetical predicting impacts from TGUS and GUS and did not provide any conclusive insight.

Boosted Gradient Classification Forest

Boosted gradient classification (BGC) with a log-loss minimization was implemented next due to its high classification capability and feature importance interpretability. The optimal max tree depth was found to be 6, and the number of estimators was set to 100. The results are shown below in Table 7.

Dataset	BGC results [%]					
	Avg. Tr. Acc.	Avg. Val. Acc.	Avg. Tst. Acc.	Best Sc. Tr. Acc.	Best Sc. Val. Acc.	Best Sc. Tst. Acc.
All	99.5	99.8	98.2	99.0	100.0	100.0
GUS Focus	99.5	99.9	98.3	99.0	100.0	100.0
TGUS Focus	99.5	99.8	98.3	99.0	100.0	100.0
Raw Focus	99.0	99.9	98.5	99.0	100.0	100.0

Table 7: *Results from implementation of the BGC algorithm.* All datasets led to very successful outcomes. These were the best prediction results thus far.

The BGC resulted in the highest prediction accuracy of all the models. Perfect classification of the test data occurred 19 times for the all-value dataset, five times for the GUS-focused dataset, nine times for the TGUS-focused dataset, and 18 times for the raw-focused dataset. Moreover, when the same T-test procedure was performed as with the two previous models, the only null hypothesis rejected was when the all-value dataset was compared to the raw-focused dataset in the testing data portion. This suggested that GUS and TGUS were equally as effective in predicting pesticide leaching events. We looked at the relative feature importances (RFIs) for all datasets to investigate this further, as shown in Figure 2 below.

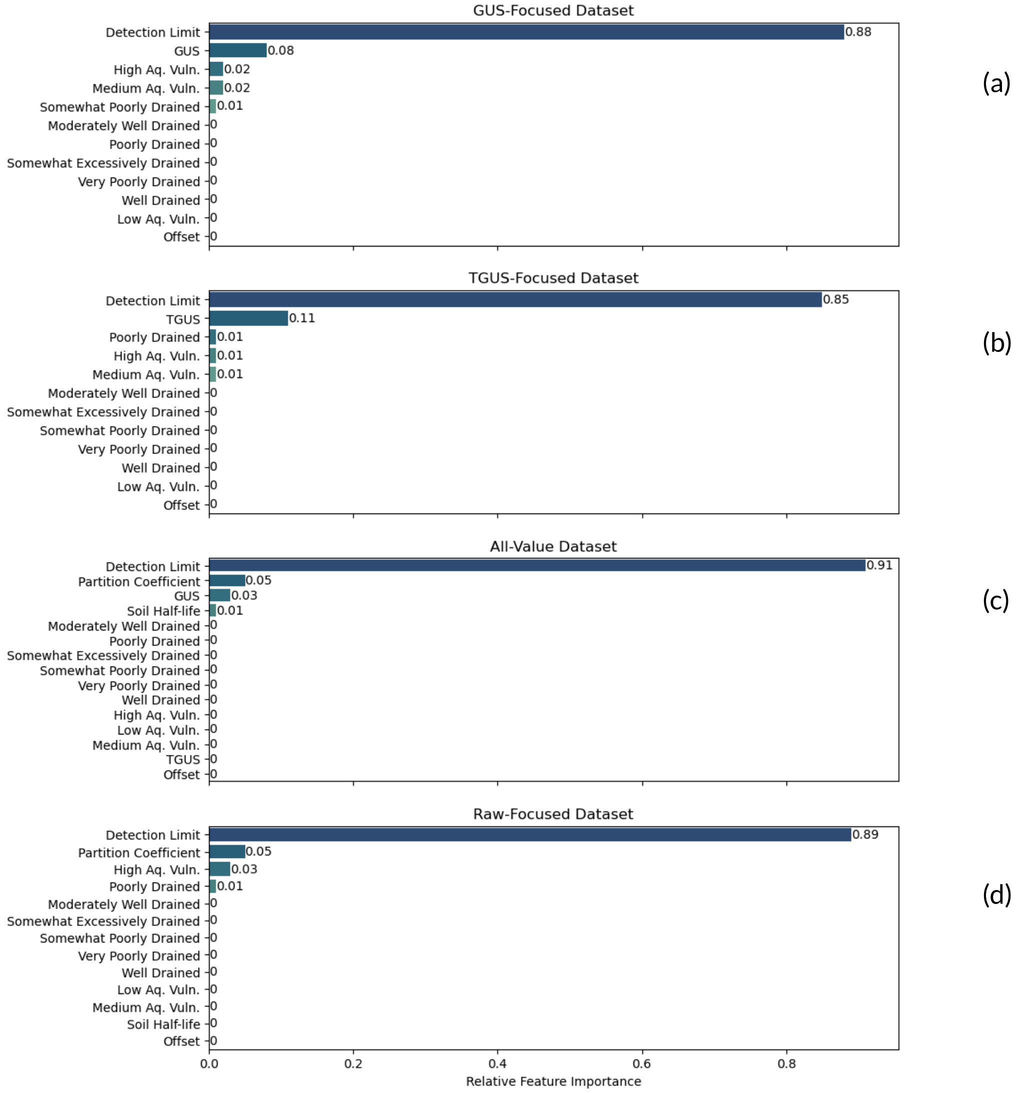

Figure 2: ***(a)*** *Relative feature importance for GUS-focused dataset in BGC algorithm.* ***(b)*** *Relative feature importance for TGUS-focused dataset in BGC algorithm.* ***(a)*** *Relative feature importance for all-value dataset in BGC algorithm.* ***(d)*** *Relative feature importance for raw-focused dataset in BGC algorithm.* Detection limit was disproportionately the most important classification feature in each dataset. TGUS and GUS seem to perform similarly when isolated, but TGUS had no importance in the all-value dataset. This could perhaps be due to redundancy and the massive detection limit importance.

Although many features had an RFI of 0, this does not necessarily mean that they are useless for classification. Because the detection limit had such a strong importance, only a few other features were needed afterwards to predict outcomes. This is a critical concept for considering the impacts of TGUS and GUS. In the all-value dataset, GUS had an RFI of 0.03, while TGUS had none. However, TGUS and GUS had similar RFIs in each of their respective focused datasets. To get a better understanding of the other features, the same BGC process was repeated using the same datasets without the detection limits. The number of estimators was kept the same, but the optimal max depth was changed to 8. The results are shown in Table 8 and Figure 3 below.

Dataset	BGC results [%]					
	Avg. Tr. Acc.	Avg. Val. Acc.	Avg. Tst. Acc.	Best Sc. Tr. Acc.	Best Sc. Val. Acc.	Best Sc. Tst. Acc.
All	96.4	96.9	93.3	96.0	98.6	98.1
GUS Focus	96.4	97.0	93.7	96.0	94.3	99.1
TGUS Focus	96.3	96.9	93.2	96.0	97.1	98.1
Raw Focus	96.4	96.9	92.9	96.0	97.1	98.1

Table 8: *Results from implementation of the BGC algorithm without the detection limit feature.* All datasets still led to very successful outcomes, however, average accuracies decreased.

Average accuracy decreased after removing the detection limit feature, and the model could never perfectly classify the testing data portion. Applying the same T-test process as before resulted in only one rejected null hypothesis between the GUS-focused and raw-focused datasets in the testing data portion, suggesting that GUS was a better predictor than the raw values.

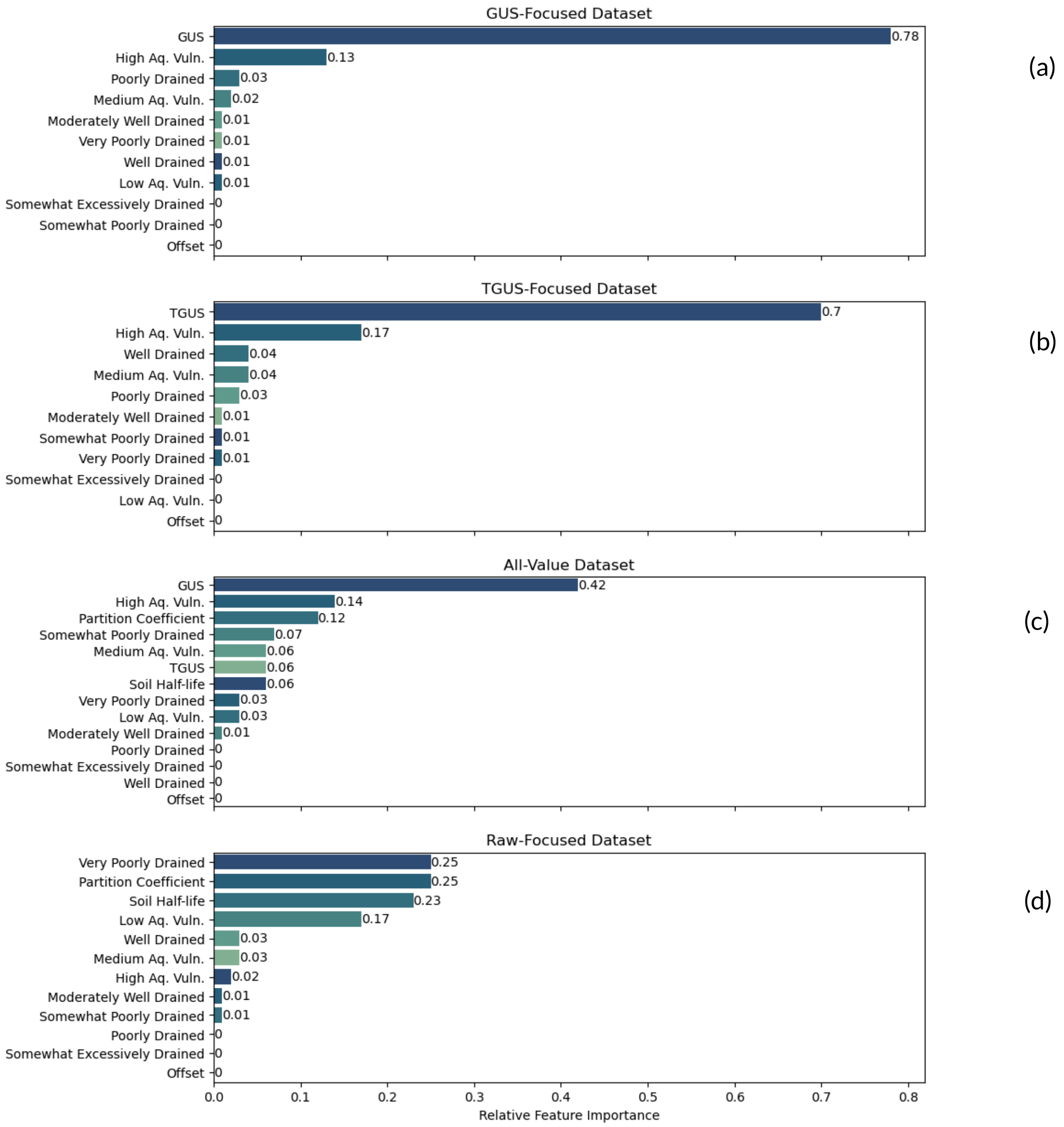

Figure 3: ***(a)*** *Relative feature importance for GUS-focused dataset with detection limit removed in the BGC algorithm* ***(b)*** *Relative feature importance for TGUS-focused dataset with detection limit removed in the BGC algorithm.* ***(c)*** *Relative feature importance for all-value dataset with detection limit removed in the BGC algorithm.* ***(d)*** *Relative feature importance for raw-focused dataset with detection limit removed in the BGC algorithm.* More features have nonzero importances with the detection limit removed. TGUS and GUS became the most significant features within their focused datasets, and GUS became the most important feature in the all-value dataset. The other features showed interesting RFI changes between the datasets.

The algorithm required more features in each dataset to classify outcomes, resulting in more nonzero RFIs. TGUS and GUS similarly became the most important feature in their respective datasets; however, TGUS had a slightly lower value, potentially suggesting that TGUS is a worse predictor. GUS became the most important feature in the all-value dataset and had a much higher RFI than TGUS. However, this could simply reflect their redundancy. The other features had varying importance between datasets.

Conclusions

These results showed that machine learning models can predict pesticide leaching events with a potential accuracy upwards of 98%. While simpler models work well, more complex models, such as support vector classification and boosted gradient classification, can greatly improve accuracy. These results also suggested that the newly-derived TGUS equation could be as effective as the standard GUS equation for predicting pesticide leaching. If this is the case, the TGUS equation's use of chemical transport theory could help SWL members better understand these pesticide leaching events.

The detection limit was the most important feature for predicting leaching events, with GUS and TGUS following behind. It was unclear how important other features were in prediction. Referring back to Figure 2(c), the all-value dataset RFIs suggest that drainage class and aquifer vulnerability categories were not needed to successfully predict outcomes. This could potentially help the SWL reduce the amount of information needed for each pesticide concentration test. It's possible that these soil characteristics could be reflected in the TGUS equation, which would imply indirect reflection in the GUS equation, as well. However, Figure 3 shows this same categorical data having nonzero RFI values for each dataset.

Future Suggestions

Add more data - These models could be improved with more pesticide test results. The SWL will continue this project through 2025, so more data will be available to use in these models. Other data sources on pesticide leaching events could be used, as well.

Further investigate GUS vs TGUS, and other features - Some results suggested that TGUS may be a slightly worse predictor than GUS. More analysis needs to be done before that conclusion is reached. If this is the case, the SWL could look to refine the equation. Also, the prediction power of the drainage class and aquifer vulnerability features should be investigated further.

Consider time for TGUS - The TGUS equation's ability to predict pesticide leaching depends on the time between application and the first significant water infiltration event. However, for simplicity, no time frames were incorporated into this project. Although difficult to document, this addition could improve model accuracy.

Consider pesticide application rate and other legal parameters - These features could be imperative for tracking how pesticide regulations affect leaching events over time. Since these models purely rely on chemical transport theory, they would predict the same outcome for the same pesticide applied at different rates. We assumed that farmers all applied the same legal amounts based on current regulations.

Keep track of false positives and negatives - Although the average model accuracies are very high, it would be valuable to know where errors in predictions are happening and if there are any trends.

Multiclass output - Since these models were highly successful at predicting the binary output space of detected or undetected, they should also be tried on multiclass outcomes for varying levels of leaching. The SWL has already defined different classifications based on the concentration of detected pesticide, so this could quickly be implemented.

Regression - Models beyond classification should also be explored. Logistic regression could provide probabilities for binary and multiclass outcomes. Other algorithms should be tested to predict the actual pesticide concentration in the sample.

Fairness and Ethical Considerations

The models built in this project are fair. No protected attributes were used in predicting outcomes. All the data used in the project was gathered with the consent of the pesticide-appliers, and the output space is measurable. While the original data does include specific locations and farms where pesticides are applied, these were omitted from the project entirely to protect their privacy.

These models could be used as a Weapon of Math Destruction (WMD) in the wrong hands. The results could influence pesticide regulations that significantly harm the agricultural community or the environment. Moreover, these models could harm the relationship between researchers and pesticide-appliers, and subsequently the models themselves. They could see these results and become weary to share information about their pesticide use.

There are many hypothetical scenarios in which these models could lead to negative consequences or feedback loops. However, these would only occur through misuse of the results. The SWL will ensure that these models will only be used for the betterment of society as a whole.

Bibliography

Gustafson, D. I. "Groundwater Ubiquity Score: A Simple Method for Assessing Pesticide Leachability." *Environmental Toxicology and Chemistry* 8, no. 4 (1989): 339–57. https://doi.org/10.1002/etc.5620080411.

"Http://Npic.Orst.Edu/Factsheets/Bindingaffinity.Pdf." Accessed May 10, 2024. http://npic.orst.edu/factsheets/bindingaffinity.pdf.

Kim, Tae Kyun. "T Test as a Parametric Statistic." *Korean Journal of Anesthesiology* 68, no. 6 (December 2015): 540–46. https://doi.org/10.4097/kjae.2015.68.6.540.