CSC 371 HW 1

Chetna Dawda and Ellora Devulapally

Due September 23, 2025

1 Background

Recreational water quality is a public health concern, especially at popular swimming sites like Presque Isle Beach 6 in Pennsylvania. High concentrations of E. coli in lake water can indicate fecal contamination and pose serious health risks to swimmers. Traditionally, water samples are collected and analyzed in labs, a process that can take up to 24 hours, delaying public advisories. To address this, the U.S. Geological Survey (USGS) developed predictive models using environmental data to estimate E. coli concentrations in near real-time.[1]

2 Exploring the Beach Dataset

To properly explore the data collected, we created scatter plots to visualize variables in the Beach 6 dataset to see whether they showed any relationship with E. coli concentrations.

The first plot looked at the number of birds observed near the beach compared to E. coli levels. While the data were fairly spread out, there was a slight indication that higher bird counts might be linked with increased bacterial concentrations.

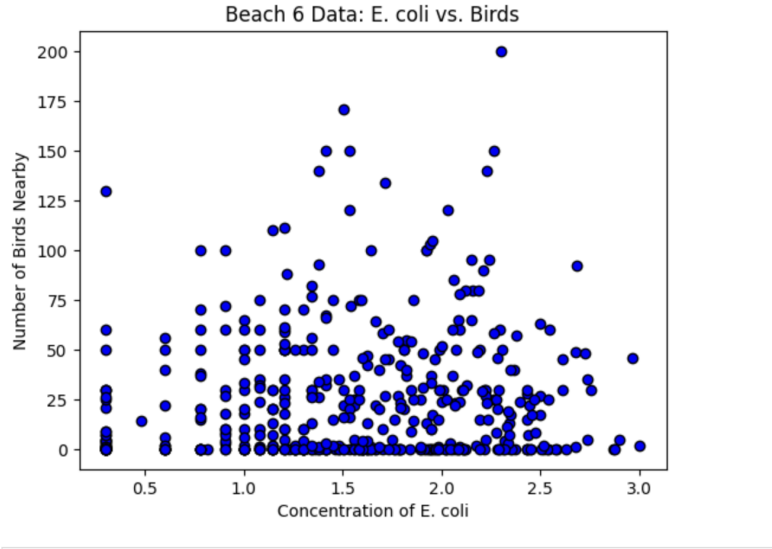

Figure 1: E. coli vs. Bird Counts

We then examined the relationship between lake level change over 24 hours and E. coli concentrations. This scatter plot did not suggest a strong pattern, but there appeared to be some clustering that hinted at a weak positive association when lake levels were rising.

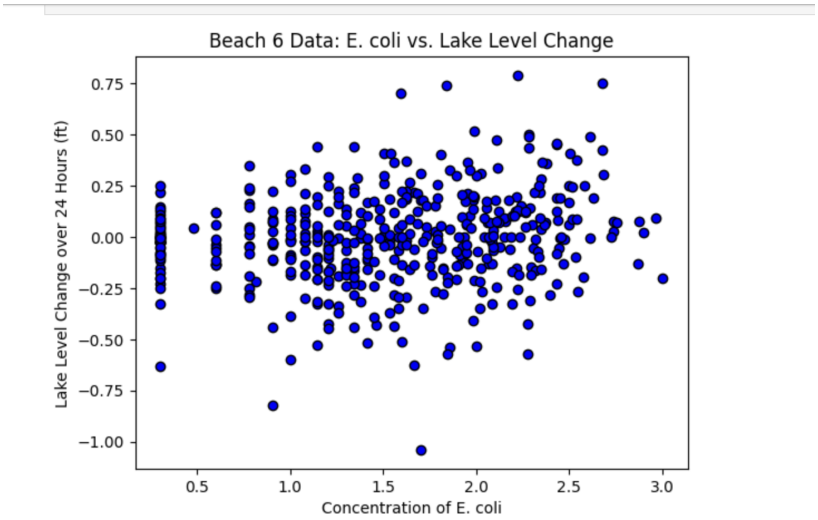

Figure 2: E. coli vs. Lake Level Change

The next feature was relative humidity. Here the points were more scattered, with most humidity values clustered between 60 and 80 percent. There was no clear linear trend, though the spread suggested that humidity might still play a role when combined with other predictors.

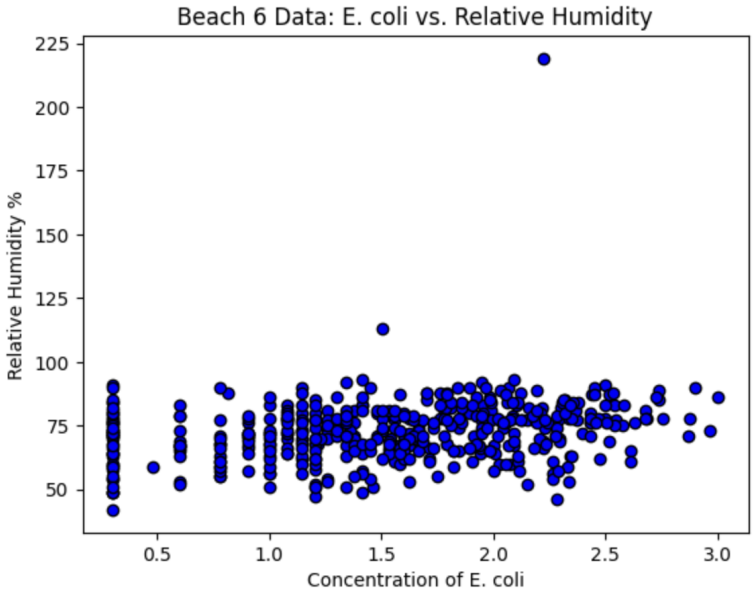

Figure 3: E. coli vs. Relative Humidity

For comparison, we also included the turbidity and water temperature plots from my own analysis. These show similar results to what was described earlier: turbidity had a weak positive trend with E. coli, and water temperature did not display a strong direct relationship but showed some clustering at higher concentrations.

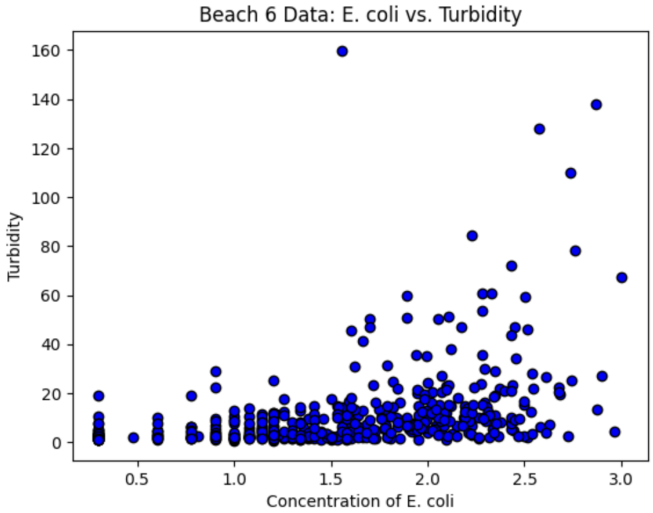

Figure 4: E. coli vs. Turbidity

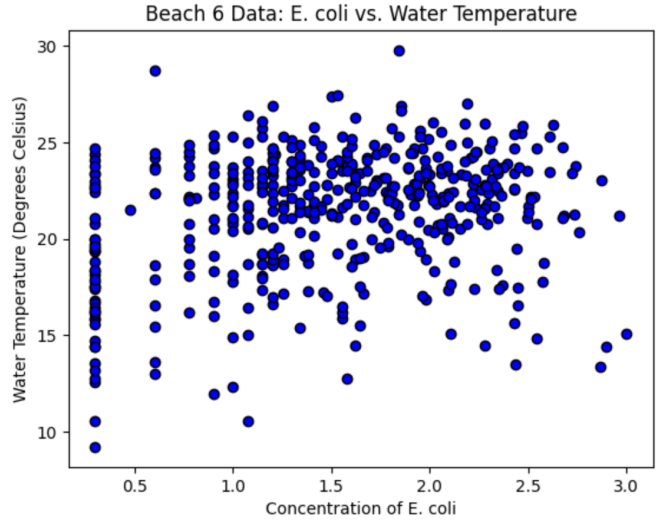

Figure 5: E. coli vs. Water Temperature

Overall, none of the scatter plots revealed a very strong linear relationship on their own, but rainfall, bird counts, turbidity, and lake level change all showed slight positive trends. These results suggest that while each variable individually may not be a perfect predictor, together they provide useful information for building regression models.

3 Exploring the Huntington Dataset

Before building the actual model, we began with some exploratory analysis. We first loaded the dataset and examined its features to understand the types of values in each column (e.g., negative, 0–1, etc.). Our main focus was on investigating whether any of the features showed a linear relationship with the E. coli concentration. To explore this, we created a series of scatter plots using NumPy and Matplotlib.

Our first plot (Figure 1) compared Lake Temperature to E. coli concentration to check for any visible relationship between the two. The most noticeable feature of the plot was a major outlier at the top of the graph. Most of the data points clustered toward the bottom, with E. coli concentrations below 4000. At first glance, the scatter plot does not suggest a clear linear relationship between lake temperature and E. coli concentration.

To determine whether the outlier(s) were skewing the results, we re-plotted the data after removing them. Since the majority of points fell below an E. coli concentration of 4000, we used that as the cutoff. The graph below shows the updated visualization:

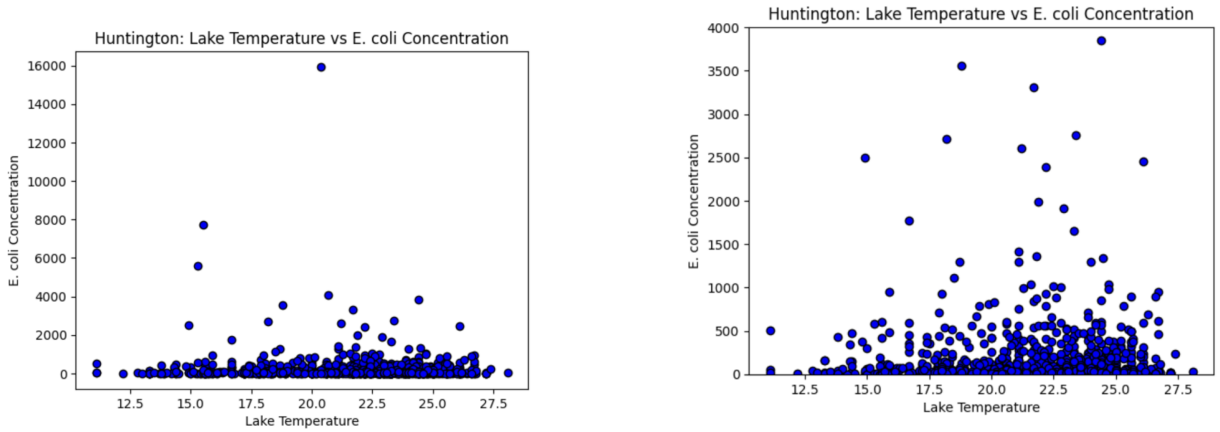

Figure 6: Lake Temperature vs. E. coli (left) and with cutoffs (right)

In this adjusted graph (Figure 2), a slight positive linear relationship between lake temperature and E. coli concentration becomes more apparent but still not very clear.

The next feature we examined was Lake Turbidity. Using the original dataset, we plotted the data and obtained the following graph: Figure 3

Similar to the previous plot, this graph does not reveal a clear relationship, as most of the data points are clustered in the bottom-left corner. To better observe the potential relationship between lake turbidity and E. coli concentration, we applied cutoffs to both axes. We set the x-axis cutoff at 150 and the y-axis cutoff at 2000, since the majority of the data fell within those ranges.

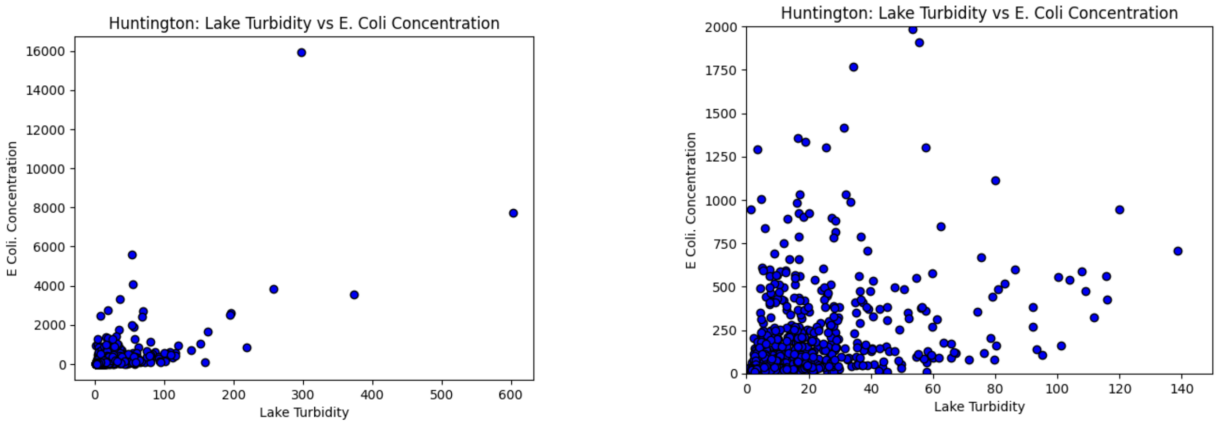

Figure 7: Lake Turbidity vs. E. coli (left) and with cutoffs (right)

Even with the cutoffs applied, the graph did not make the relationship completely clear. However, there does appear to be some indication of a potential positive linear trend between lake turbidity and E. coli concentration. While the relationship is weak and somewhat noisy, the clustering of points suggests that higher turbidity values may be associated with higher E. coli levels, making turbidity a feature worth considering for modeling.

Overall, our exploratory analysis showed that while neither lake temperature nor turbidity displayed a strong or obvious linear relationship with E. coli concentration, both features exhibited slight positive trends once outliers were removed and cutoffs were applied. While we did not explore the impacts of other features in this initial analysis, we assumed they displayed similar patterns, which we investigate in more detail in our modeling process. These results suggest that temperature and turbidity may still hold some predictive value, even if the relationships are not perfectly clear, and set the stage for a more formal assessment of all features in our models.

4 Beach Data Preprocessing

Before building the regression models, we had to preprocess the Beach 6 dataset so that it would match the structure of the published USGS model. The raw data contained several variables that were skewed or on very different scales, which required transformation to make the relationships more linear and to allow the model to treat each feature fairly.

The first step was to transform the E. coli measurements. Following the USGS approach, we applied a $\log_{10}$ transformation to the bacterial counts. Turbidity values were also $\log_{10}$ transformed. This helps reduce the influence of extreme values and makes the distribution closer to normal.

For rainfall, instead of a log transform, we applied a square root transformation to the 48-hour rainfall totals. We also did so here to match the data collection of the USGS.

After transforming the data, we shuffled the dataset to remove any time effects. The original records were collected chronologically, but shuffling ensured that training and test sets would not be biased by patterns. This step helped optimize the model by preventing it from simply learning time-based trends rather than the true relationships between variables. We then split the dataset into 80% training and 20% testing using the train-test split method. This split allowed us to optimize the model on the training set while keeping the test set completely separate, so that it provided an unbiased measure of predictive performance.

Finally, we scaled all the predictor variables to a 0-1 range using MinMax scaling. This step was important because the regression methods used in this project are sensitive to the magnitude of the input values. Scaling put rainfall, turbidity, temperature, bird counts, lake level changes, and relative humidity all on the same footing.

These preprocessing steps produced a dataset that could be directly compared to the USGS model and allowed the regression methods to perform without being dominated by the largest-scale variables.

5 Huntington Data Preprocessing

After exploring the data, we noticed numerous outliers and extreme values that could disproportionately impact our model. This is common with metrics that have a wide range, such as income. In such cases, simple min-max scaling is insufficient because extremely high values can compress the more representative lower values toward 0. In our dataset, the `['EcoliAve_CFU']` and `['Lake_Turb_NTRU']` columns exhibited a similarly large range.

To address this, we applied log transformations. This approach smooths the data, reducing skewness and helping the model perform more accurately. The USGS team also used this technique. To avoid issues with taking the logarithm of zero, we added a small epsilon ($\epsilon = 1e{-}10$) to each value in the relevant columns.

Additionally, the USGS team applied a square root transformation to the `['AirportRain48W_in']` column. While the exact reasoning for this is unclear, we applied the same transformation to maintain comparability with their model. By using the same preprocessing steps, we can more reliably compare results and potentially replicate their findings.

Finally, we separated our data into X and y, where X contains all the features and y contains the E. coli concentration, which we aim to predict. We split the dataset so that 80% would be used for training and 20% for testing. We used scikit-learn to perform the split, which also allowed us to shuffle the data to ensure that both sets are representative of the overall dataset. This approach not only helps prevent overfitting but also ensures that the test set provides an unbiased estimate of the model's predictive performance. After splitting, we scaled all feature data to a 0–1 range using min-max scaling, which puts all variables on the same scale and allows them to be fairly compared by the model.

6 Ridge Regression on Beach Dataset

Once the data preprocessing was done, we proceeded by running ridge regression on the beach dataset. Scikit greatly simplified this process and we were able to see the coefficients that we got for our model (Figure 4).

```
Coefficients: [0.72650767 1.94181004 0.44182652 0.47921663 1.37442744]
Intercept: 0.16280334213527659
```

Figure 8: Initial Coefficient Results (Ridge Regression on Beach Dataset)

These coefficients correspond directly to the order of the columns in the dataset. While they were similar, they did not exactly match the coefficients reported by the USGS researchers. To fine-tune the model, we performed a grid search using five different alpha values: [0.01, 0.1, 1.0, 10.0, 100.0]. Using R^2 as the scoring metric, we found that alpha = 1.0 (the default value in scikit-learn) produced the best results. With this optimal alpha in mind, we then compared our model's coefficients to those from the USGS model (Figure 9).

	Ridge Coeff	Study Coeff
Lake_Temp_C	0.7265	0.0360
Lake_Turb_NTRU	1.9418	0.6896
WaveHt_Ft	0.4418	0.1942
LL_PreDay	0.4792	0.3781
AirportRain48W_in	1.3744	0.4537

Figure 9: Comparing Beach Ridge Results to USGS model

The column labeled "Ridge Coeff" shows the coefficients obtained from our model, while "Study Coeff" shows the coefficients from the USGS model. Some values are quite similar—for example, the coefficient for `"LL_PreDay"` differs by only about 0.1. However, other coefficients, such as `"AirportRain48W_in"`, show substantial differences between the two models. To fully evaluate the accuracy of our model, we ran it on the test data and measured performance using two metrics: R^2 and RMSE (root mean squared error). These metrics provide insight into how well the model can predict E. coli concentrations based on the given features. Our model achieved an R^2 value of 0.427 and an RMSE of 0.503. The R^2 value indicates that approximately 42.7% of the variance in the test set is explained by our model, meaning that over half of the variability (57.3%) remains unexplained. This suggests that while the model captures some meaningful patterns in the data, there are other factors affecting E. coli concentrations that are not accounted for in our current feature set or model structure. The RMSE value of 0.503 indicates that, on average, our predictions deviate from the actual values by about 0.5 units. Combined with the R^2 score, this shows that the model provides moderate predictive accuracy but could likely be improved by incorporating additional relevant features, applying more sophisticated modeling techniques, or addressing remaining data noise and outliers.

7 Ridge Regression on Huntington Dataset

The process for applying ridge regression to the Huntington dataset was similar to that used for the Beach dataset. We began by training our model on the training set and obtained the coefficients shown below.

```
Coefficients: [1.59926557 0.94365842 0.92540601 0.33775339 0.55609362 0.49868741
 0.54557869]
Intercept: -0.43393736088105794
```

Figure 10: Initial Coefficient Results (Ridge Regression on Huntington Dataset)

To further improve the model, we implemented a grid search to fine-tune the alpha hyperparameter and ensure the most accurate results. This process identified an optimal alpha value of 0.1, which differs from the default value of 1.0. Using this optimized alpha, we retrained the model and subsequently compared our coefficients to those from the USGS model.

	Ridge Coeff	Study Coeff
TURB_NTRU	1.7005	0.6835
RHUM_PCT	1.3391	0.0079
WTEMP_CEL	1.0171	0.0526
BIRDS_NO	0.3426	0.0018
CHANGELL_FT	0.6072	0.3176
AirportWindSpInst_mph	0.5297	0.0248
AirportRain48W_in	0.5080	0.1980

Figure 11: Comparing Huntington Ridge Results to USGS model

For the most part, our coefficients still differ from those of the USGS model. To evaluate how well our model performed, we tested it on the Huntington dataset using the same approach as with the Beach dataset. Our model achieved an R^2 value of 0.427 and an RMSE of 0.503.

The R^2 value indicates that approximately 42.7% of the variance in the test set is explained by our model, meaning that a majority of the variability (57.3%) remains unexplained. This suggests that while the model captures some meaningful patterns in the data, other factors affecting E. coli concentrations are not fully accounted for by our current features or model structure. The RMSE of 0.503 indicates that, on average, our predictions deviate from the actual values by about 0.5 units. Overall, these metrics show that the model provides moderate predictive accuracy but could likely be improved with additional features, refined preprocessing, or more sophisticated modeling techniques.

8 Lasso on Beach Dataset

We then studied how Lasso regression (L1) performed on the Beach 6 dataset to compare the effect of different types of regularization.

Lasso regression (Least Absolute Shrinkage and Selection Operator) is a type of linear regression that adds an L1 penalty to the size of the coefficients [0]. This penalty encourages the model to shrink some coefficients to exactly zero, which has the effect of both regularizing the model and performing feature selection. Unlike Ridge regression, which reduces the magnitude of coefficients but retains all predictors, Lasso produces simpler models that highlight only the most important variables. This is particularly useful for environmental data, where many features may be weakly correlated or redundant.

Before applying the model, we completed several preprocessing steps. We shuffled the dataset using sklearn.utils.shuffle [0] to remove any temporal ordering that might bias the model. We then applied transformations to match the USGS study: log10 transformations to E. coli and turbidity, and a square root transformation to rainfall. After this, we used traintestsplit [2] to partition the dataset into training and testing subsets (80/20). Finally, we applied MinMax scaling [0] so that all predictors were on the same scale.

The model itself was implemented using a scikit-learn Pipeline consisting of three main steps:

1. `MinMaxScaler` to scale the predictors,
2. `PolynomialFeatures` to automatically expand the dataset with polynomial and interaction terms[0],
3. `Lasso` regression to fit the model and shrink irrelevant coefficients to zero.

To optimize the model, we used GridSearchCV to tune two key hyperparameters: the polynomial degree and the regularization strength, or alpha. Testing multiple combinations allowed us to identify the parameter set with the lowest mean squared error during cross-validation.

The final model retained several predictors as important drivers of bacterial concentration. Log_{10} turbidity had the strongest positive coefficient (1.4653), indicating that water clarity is a key factor in predicting E. coli levels. Water temperature (0.7377), rainfall (0.4649), and bird counts (0.2607) also contributed positively. Lake level change had a smaller effect (0.1418), while relative humidity was shrunk to zero, suggesting it did not add predictive value for this site. Compared with the published USGS model, our coefficients differed in scale but emphasized similar trends, especially the strong role of turbidity and rainfall.

	Your Lasso Coeff	Study Coeff
LOG10_TURB	1.4491	0.6835
RHUM_PCT	0.0000	0.0079
WTEMP_CEL	0.4460	0.0526
BIRDS_NO	0.0784	0.0018
CHANGELL_FT	0.0000	0.3176
SQRT_RAIN	0.3977	0.1980
AirportWindSpInst_mph	NaN	0.0248

Figure 12: Comparison of Lasso Coefficients and Published USGS Coefficients

To evaluate performance, we plotted predicted values against actual $\log_{10}$ E. coli values. The scatterplot shows that while the model captured overall trends, it tended to underpredict at higher observed concentrations.

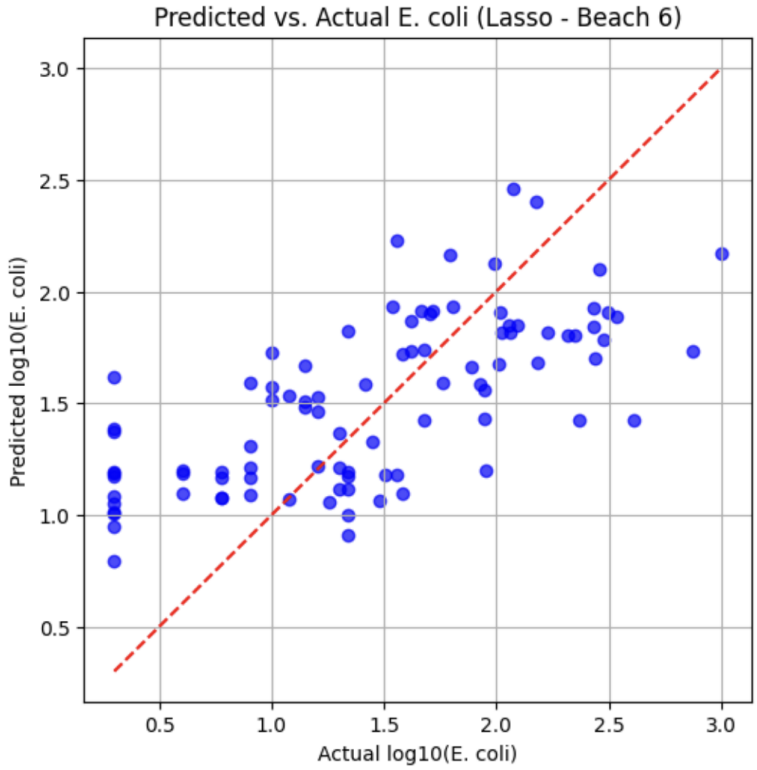

Figure 13: Predicted vs. Actual E. coli Concentrations (Lasso, Beach 6)

9 Lasso on Huntington Dataset

We next applied the same Lasso pipeline to the Huntington dataset. The preprocessing followed the same structure as before: we shuffled the data to remove temporal ordering, transformed the features using $\log_{10}$

for E. coli and turbidity, and square root transformations for wave height and rainfall. We then split the dataset into training and testing sets and scaled all predictors using MinMax scaling.

For Huntington, we again built a pipeline with scaling, polynomial feature expansion, and Lasso regression. Using GridSearchCV, we tuned the polynomial degree and the regularization parameter alpha. The best model was found at degree 1 with $\alpha = 0.01$, which produced the lowest cross-validation mean squared error.

The final Lasso model highlighted several predictors as important. Turbidity (1.2598), rainfall (1.0742), and wave height (0.5867) all had strong positive coefficients, while water temperature (0.2788) also contributed positively. Lake level change, however, was shrunk to zero, suggesting it did not provide predictive value in this dataset. Compared with the published Huntington model, our results agreed on the importance of turbidity, rainfall, and wave height, but our coefficients were generally larger in scale.

	Your Lasso Coeff	Study Coeff
LOG10_TURB	1.2598	0.6896
Lake_Temp_C	0.2788	0.0360
SQRT_WAVE	0.5867	0.1942
LL_PreDay	0.0000	0.3781
SQRT_RAIN	1.0742	0.4537

Figure 14: Comparison of Lasso Coefficients and Published USGS Coefficients (Huntington)

To evaluate predictive accuracy, we plotted predicted versus actual $\log_{10}$ E. coli values on the test set. The scatterplot shows that the model captured general patterns in the data, though as with Beach 6, it tended to underpredict at higher observed concentrations.

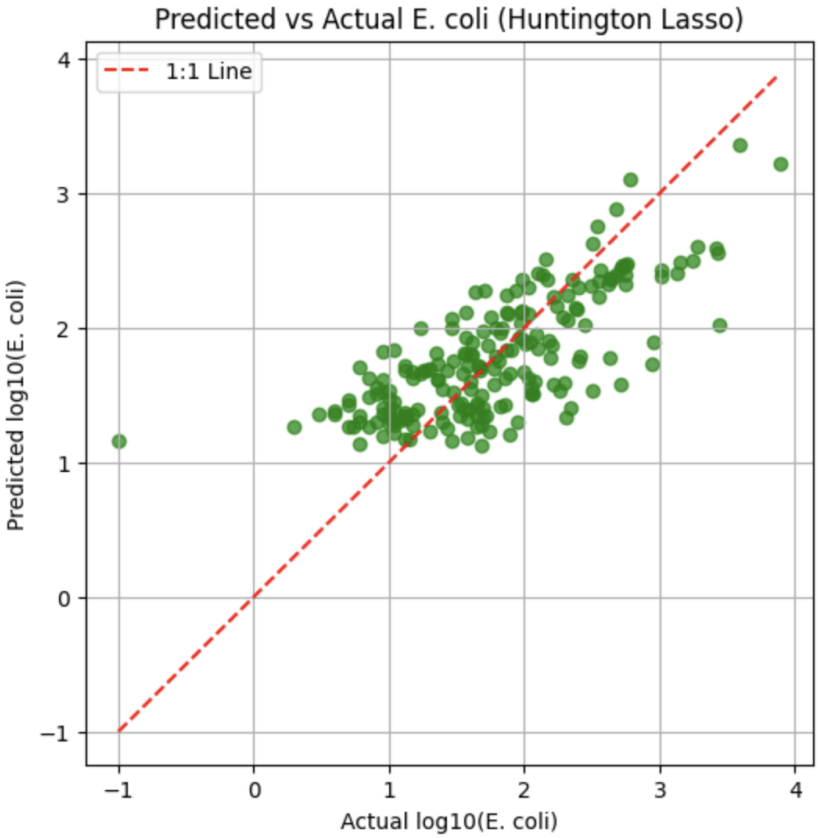

Figure 15: Predicted vs. Actual E. coli Concentrations (Lasso, Huntington)

10 Conclusion

In this project, we explored the relationship between environmental features and E. coli concentrations, applied data preprocessing techniques, and built regression models to predict bacterial levels. Our exploratory analysis revealed weak linear relationships with E. coli that still held potential predictive value once outliers and scale issues were addressed. To improve model performance, we applied log and square-root transformations, introduced an epsilon adjustment, and scaled features, closely following the preprocessing steps used in the USGS study to ensure comparability.

In general, our models achieved moderate predictive accuracy. They successfully replicated many of the trends observed in the USGS studies and demonstrated the value of regularization in simplifying models. These findings suggest that combining all these environmental predictors with regularization can improve quality forecasting.

Between the two methods, Lasso proved more useful for identifying the most influential predictors which led to a more interpretable model. Ridge regression retained all features as did the USGS model itself; the choice between the two depends on the choice between highlighting the most important parameters or maximizing accuracy.

Future improvements could involve expanding the feature set, applying more advanced models, and exploring feature engineering techniques that capture nonlinear effects or interactions. Improved handling

of outliers and temporal dependencies may also enhance predictive power.

Overall, our work demonstrates that environmental predictors contain valuable information for estimating E. coli concentrations, and that even relatively simple models with appropriate preprocessing can achieve moderate accuracy. While our results did not replicate the USGS models exactly, they highlight both the promise and the challenges of predictive modeling in water quality research. With additional data, refined preprocessing, and more sophisticated techniques, these models can be further improved to provide more accurate real-time estimates of bacterial risk in recreational waters.

References

[1] Centers for Disease Control and Prevention. About e. coli. https://www.cdc.gov/ecoli/about/index.html, 2025. Accessed: 2025-09-23.

[2] Scikit learn Developers. sklearn.model$_s$*election.train_test_split.*. Accessed: 2025-09-23.

Scikit learn Developers. sklearn.preprocessing.minmaxscaler. https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.MinMaxScaler.html. Accessed: 2025-09-23.

Scikit learn Developers. sklearn.preprocessing.polynomialfeatures. https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.PolynomialFeatures.html. Accessed: 2025-09-23.

Scikit learn Developers. sklearn.utils.shuffle. https://scikit-learn.org/0.15/modules/generated/sklearn.utils.shuffle.html. Accessed: 2025-09-23.

Scikit learn Developers. sklearn.linear_model.lasso. https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.Lasso.html, 2025. Accessed: 2025-09-23.