Introduction

In this project we intend to look at a dataset on rice paddy yields to predict optimal configurations for maximal yield. As climate change progresses as a more prevalent threat along with the rising global population, there is a need to address our current global food system. Most North American and European diets commonly include meat as a staple; however, the current level at which we produce and consume it is unsustainable due to heavy carbon emissions, significant resource use, and general inefficiency. Rice, by contrast, is a staple food in many countries across the world–consumed by about half of the global population–and shows a lower environmental impact. However, like any crop, rice still has an environmental impact; growing rice is water-intensive and requires large land plots for large-scale farming. Therefore, maximizing the yield is the best way to reduce environmental impact. Increasing efficiency reduces the amount of resources required for the same output. We intend to answer what combination of growing conditions produce the highest rice paddy yield within a hectare.

Dataset Review

This dataset comes from the UC Irvine Machine Learning Repository, downloaded from their website. This dataset has 45 features and 2790 observations. Each observation correlates to a different plot/paddy. The observations are not from human data, so there is no population demographics, but we did some EDA to find statistics for some of the numerical features. For example, the mean number of hectares of each paddy (observation) is 3.72, with a standard deviation of 1.44. The categorical feature "Soil Types" contains two types of soil, alluvial and clay, with 54.5% of the paddies having clay soil and 45.5% having alluvial, showing a relatively even distribution.

Methods

For the regression task, ridge regression was performed. A pipeline was used containing a standard scaler and ridge regression function. A list of alpha values were curated to find an optimal value for the task. For the support vector machine we chose support vector regression, since the intention is to predict optimal rice paddy yields. Similarly to the regression task a parameter grid was created containing a range of hyperparameters that were then tested with gridsearch. For both methods the top features were listed and ordered based on coefficients for ridge regression and permutation importance for support vector regression. For the ensemble method, we chose a Random Forest Regressor. Though not a necessary step in Random Forest, we added a preprocessing step to our pipeline to standardize and one hot encode as per the instructions. For cross validation in RF, we used four hyperparameters (number of estimators/regression trees used, max tree depth, minimum number samples required to split a node, and minimum number of samples required to be in a leaf after a split. We used KFold and GridSearchCV objects to perform hyperparameter tuning and evaluated the test set on the best model (a similar workflow for all 4 methods).

For the NN, we built a feed-forward neural network in PyTorch, wrapped in a custom TorchRegressor to integrate with scikit-learn pipelines. The pipeline applied the same preprocessing as the other models to prevent data leakage. We used GridSearchCV to tune key hyperparameters (hidden units, learning rate, weight decay, epochs, and batch size). To assess feature contributions, we computed permutation importance on the validation set after preprocessing.All models were evaluated with 3 metrics: r^2, Mean Squared Error, and Mean Absolute Error.

Results

Random Forest performed the best of our 4 models, with the highest r^2 value, indicating almost perfect explained variance, and relatively negligent error due to the high scale of ‘y’ as yield numbers. The feature importance chart from our Random Forest model shows that the strongest predictors of yield are land preparation manure (amount of fertilizer), amount of potassium fertilizer used, and the amount of rice seed sown per unit area. The evaluation of the Random Forest approach on the test data partition found r^2=~0.992, mean absolute error of ~540kg, and mean squared error of ~568,669.

	R^2	**MAE**	**MSE**
Ridge Regression	0.989	698.7	859587
Support Vector Regression	0.988	698.6	905428
Random Forest Regressor (test set)	**0.992**	**539**	**568669**

Torch Regressor (NN)	0.989957	656	814571

Discussion

We found that Random Forest performs the best at successfully finding predictors of paddy yield, followed closely by the Torch Regressor Neural Network. When looking at feature importances across all 4 outputs, **Num_seedrate** and **Num_potash** appear the most frequently in our bar charts, representing the number of seeds sown per unit of land and amount of potassium fertilizer used. This suggests that both the amount of seed planted and the level of potassium supplementation play a major role in determining yield. These results are valuable in maximizing yield in that they show which growing factors to put the most resources into in order to increase production and add value.

While the models agree on several key predictive factors, there are some limitations to each approach. Ridge regression assumes a linear relationship between inputs and yield, which limits its flexibility. Support Vector Regression with nonlinear kernels performed better but still underperformed compared to Random Forest, which can fit complex data without heavy tuning. The neural network had similar accuracy to ridge but slightly higher errors, likely because its performance depends strongly on hyperparameter choices, number of epochs, and dataset size. Neural networks are best for large datasets (much larger than our paddy data). A key limitation of our project is that this dataset represents a single region with a specific set of environmental and soil conditions. Our findings may not generalize to different climates or rice varieties. Also, our dataset may not be representative of all paddy data. Because of this, our models could be overfitting to our dataset and not generalize well on a completely different dataset.