DATA 144 Project
Global Agriculture Trends

Kiana Kazemi, Varsha Madapoosi,
Hamsavardhini “Anu” Thirunarayanan

INTRODUCTION

01

Introduction: Our Question

Can we predict terrestrial protected land areas from other agricultural features?

How do these differ between global North and global South countries? *(EDA)*

MOTIVATION

Environmentalism vs Profit

By using other features to understand how much land is protected, we can model how much land:

- should be *preserved*
- should be *conserved*
- can be *utilized* to meet economic needs

Justice

By looking for differences between Global North & Global South countries, we can understand how much land is in each category (outlined before), and infer what systems allow for/create these differences

Important for:

- Federal policymakers
- International NGOs
- Multinational corporations
- Farmers

DATASET

02

Global Environmental Indicators Data

Land and Agriculture

United Nations Statistics Division

UN environment programme

Available Features *(excluding terrestrial protected land areas in 2018)*:
Consumption of fertilizers per unit of agricultural land area (nitrogen, phosphate, potassium); Agricultural area (km^2), % change of agricultural area since 1990, % of total land area covered by agricultural area, Arable land (km^2), Permanent crops, Permanent meadows & pastures, Agricultural area actually irrigated (km^2))

* *unless otherwise indicated, data is from 2013*

FEATURE ENGINEERING

Merging datasets

Converting missing values

Trimming extra spaces

Converting to numeric/ categorical

METHODS

03

Prepping for Machine Learning

01

Identifying Columns

02

Train/Test Split

03

Importing Models

Preliminary Methods

Linear Regression

Training set error for linear model: **9.3**

Test set error for linear model: **15.8**

Decision Tree

Training set error for decision tree: **0.0**

Test set error for decision tree: **21.8**

Decision Trees

here's what the decision tree looks like

Random Forest: Feature Engineering

Max Depth

Min_Sample_Split

Min_Sample_Leaf

	max_depth_train	min_sample_split_train	min_sample_leaf_train
0	8.972898	inf	5.746999
1	7.558370	5.124146	7.112187
2	6.470150	4.995548	7.901202
3	5.581497	6.813188	8.444870
4	5.238292	6.206261	8.903052
5	5.547239	7.062362	9.641651
6	4.673209	7.193611	9.534466
7	5.179230	7.554978	9.885209
8	5.200921	8.240477	9.981191

	max_depth_test	min_sample_split_test	min_sample_leaf_test
0	15.707010	inf	15.897675
1	16.464007	17.904299	16.244752
2	19.264753	18.201843	15.608039
3	16.916625	19.263544	15.613896
4	18.085715	17.634644	16.404392
5	16.508575	16.062404	16.370769
6	14.391632	16.592227	14.808134
7	16.704709	16.072542	14.898676
8	17.143590	17.234341	15.967201

Random Forest: Feature Engineering

Max Depth = 6

Min_Sample_Split =5

Min_Sample_Leaf =6

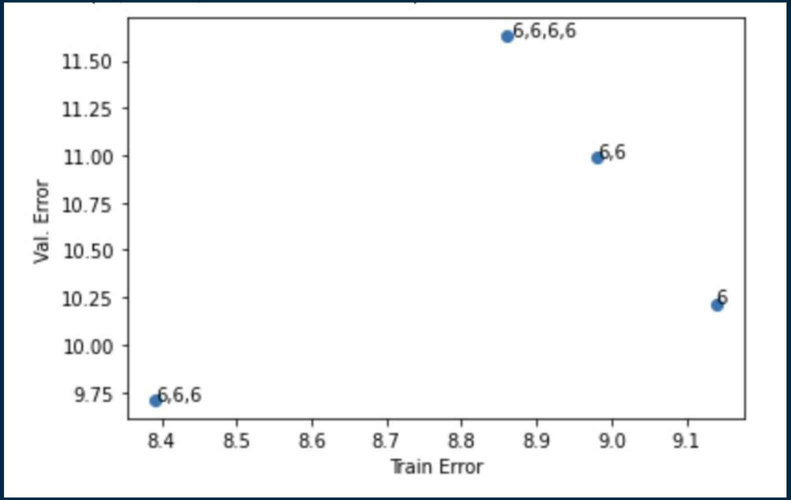
Training set error for random forest: **4.63**
Training set error for random forest with tuned parameters: **9.10**

Test set error for random forest: **16.08**
Test set error for random forest with tuned parameters: **15.72**

Neural Network Tuning - Layers, LR

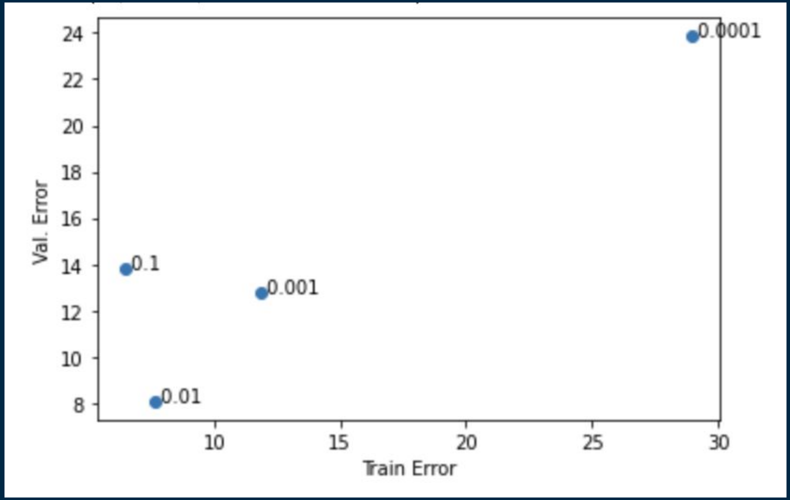
Num Neurons & Layers

The 6,6,6 layered model performed better than any other NN model



Learning Rate

A learning rate of 0.01 ended up leading to the best performance



- Note this is a subset of the tuning, we tried many other values for these, and other hyperparameters

Neural Network: Hyperparameters

Layer Size(s) = (6,6,6)

After experimenting with layer sizes, the optimal layer size was 6, with 3 layers performing best.

Learning Rate = 0.01

For this problem, with fewer data points and features, a larger learning rate than usual worked best.

Num. Epochs = 200

200 epochs was on average how long it took for the neural network to converge it's training error.

Linear Regression: Random Features

Adds more features through linear combinations of features; increases complexity but loses interpretability

```
def sigmoid(x):
    return 1 / (1 + np.exp(-x))

def add_random_feature(train_data, test_data):
    # Returns the modified train_data and test_data
    coeffs = np.random.uniform(-1,1,2)
    # This code gives the feature a convenient name
    feat_name = f"σ({coeffs[0]:0.2f}x1 + {coeffs[1]:0.2f}x2)"

    for dataset in (train_data, test_data):
        linear_combination = np.dot(dataset[["% change of agricultural area since 1990",
                                              '% of total land area covered by agricultural area in 2013']], coeffs)

        feature = sigmoid(linear_combination)
        dataset[feat_name] = feature
    return train_data, test_data

train_feats = train.copy()
test_feats = test.copy()
for i in range(10):
    train_feats, test_feats = (
        add_random_feature(train_feats, test_feats)
    )
train_feats.head()
```

With Random Features

Training Error = 0 | Validation Error = 0

RESULTS & CONCLUSION

04

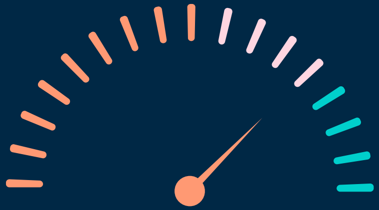
EDA Results

	type	Agricultural area in 2013 (km2)	% change of agricultural area since 1990	% of total land area covered by agricultural area in 2013	Arable land in 2013 (km2)	Permanent crops in 2013 (km2)	Permanent meadows and pastures in 2013 (km2)	Terrestrial protected areas
N = 21	Global North	25.333333	3.591667	4.611111	34.588235	14.058824	24.888889	18.252381
N = 181	Global South	29.584071	3.124138	4.102564	22.425532	17.601626	17.533333	16.733702
	Two-tailed P-value	0.6258	0.5190	0.4524	0.0721	0.5618	0.2040	0.5969
	Statistically significant?	No	No	No	No	No	No	No

Machine Learning Method Results

	Rank	Training Error	Validation Error
Linear Regression w/ Random Features	1st	0.0	0.0
Neural Network	2nd	7.0	7.6
Random Forest	3rd	4.5	14.6
Linear Regression	4th	9.3	15.8
Decision Trees	5th	0.0	21.8

Conclusions



Mostly Effective for the real-world

The model does help **improve understanding** of country's decisions around agricultural management and future predictions for land protection.

Caveats

- Small number of data points
- Interpretability
- Singular time-period

Implications:
Countries can predict terrestrial protection for other countries based on their current agricultural practices. This can then be used to determine allies/leaders in the field and be helpful in policy-making spaces.