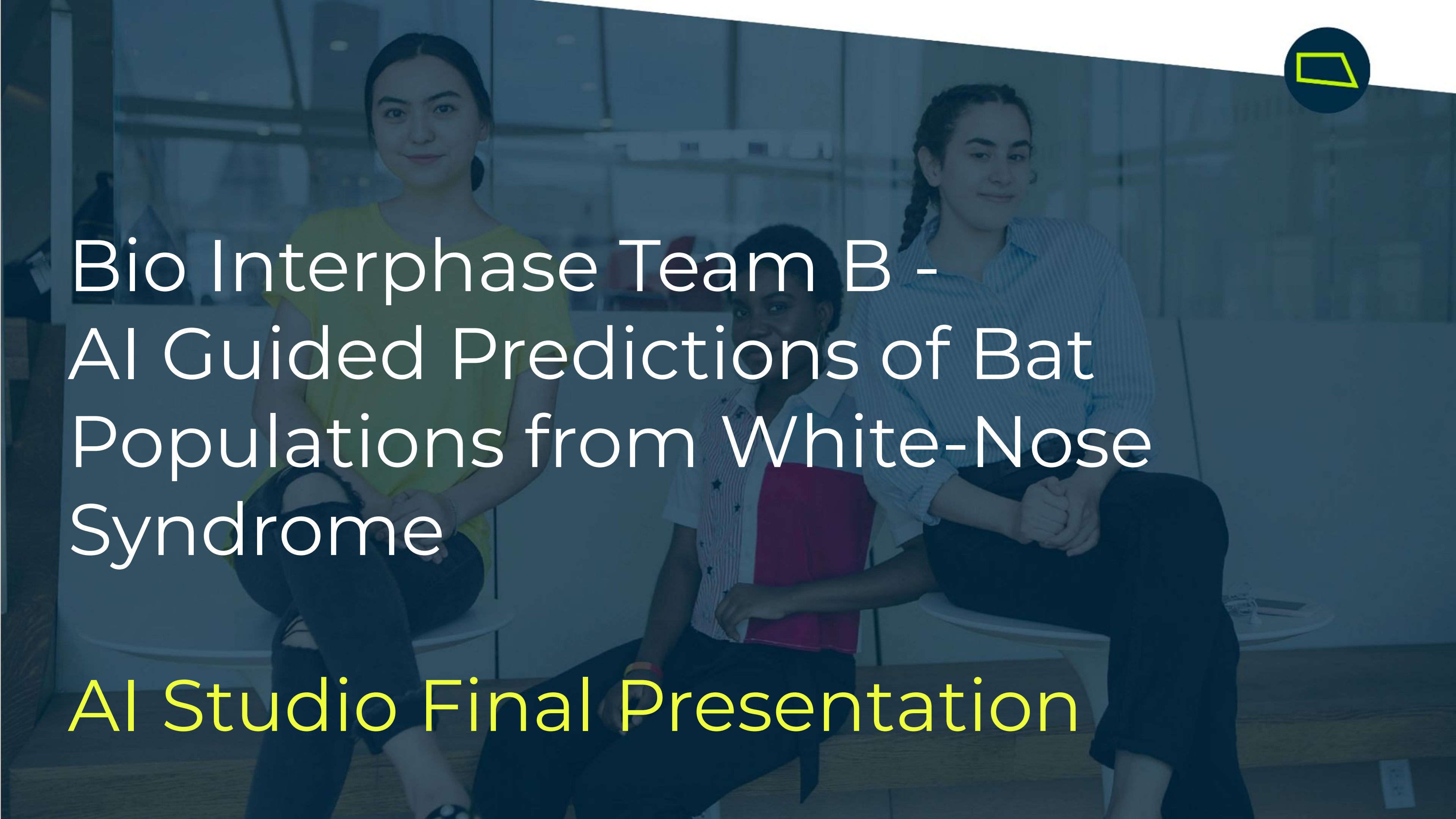


Bio Interphase Team B - AI Guided Predictions of Bat Populations from White-Nose Syndrome

AI Studio Final Presentation

Agenda

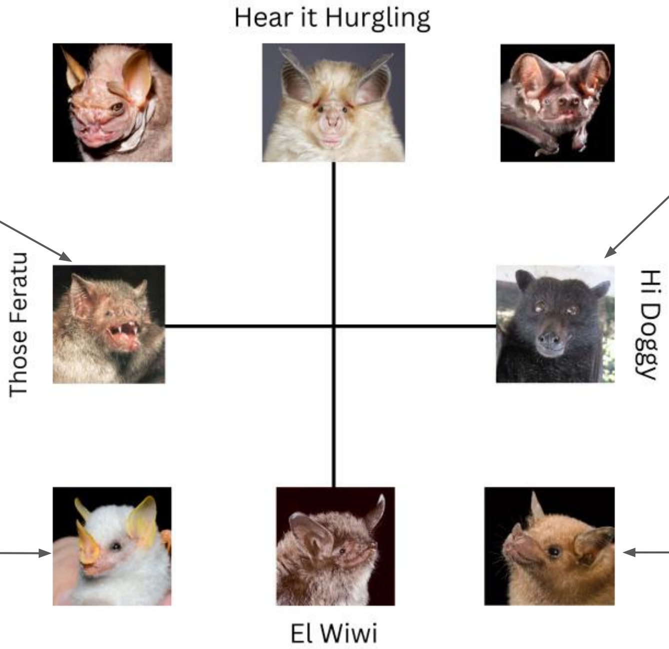
1. **Team Introductions**
   a. Meet Our Team
   b. AI Studio TA and Challenge Advisor
2. **Project Overview**
   a. Goals
   b. Business Impact
   c. Our Approach
   d. Resources Used
3. **Data Understanding & Data Preparation**
   a. Data Sources
   b. Feature Engineering
4. **Modeling & Evaluation**
   a. Model Selection
   b. Models Comparison
   c. Visualization
5. **Conclusions**

Introductions

Meet Our Team!

Margarita Kholostova

Dov Zipursky



Annissa Mu

Gwen Liu

Our AI Studio TA and Challenge Advisors

Leandra Marie Tejedor
AI Studio TA

Noah Snyder
Challenge Advisor

AI Studio Project Overview

Your mission, should you choose to accept it, is to create an ML predictive model for bat population decline throughout North America using publicly available datasets related to bat populations, bat acoustic information, and other easily accessible climate data.

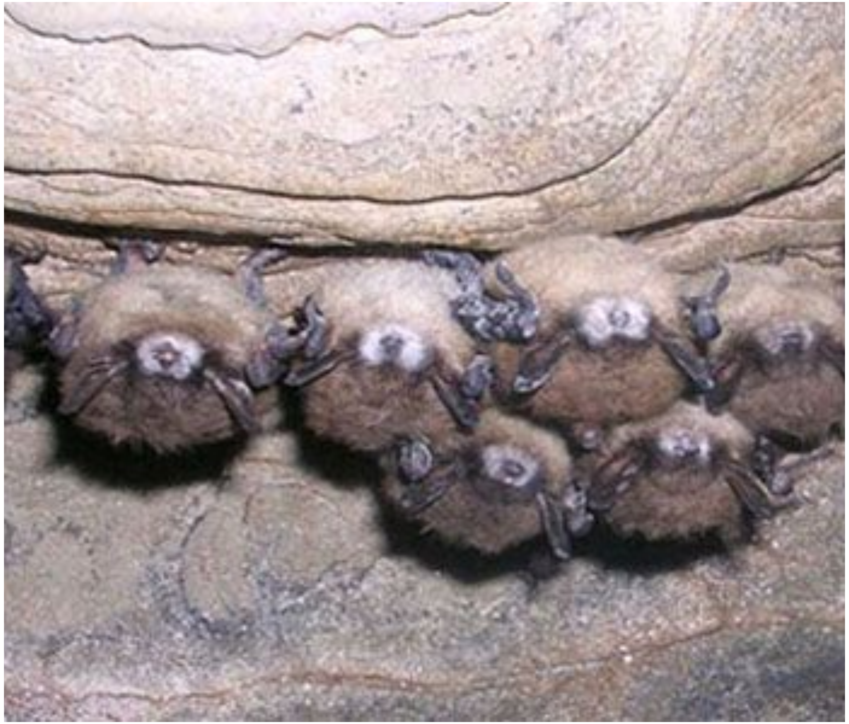
Bat Population Decline

White Nose Syndrome (WNS)

Disease of hibernating bats caused by the fungus *Pseudogymnoascus destructans* spreading across North America

First detected in New York in 2006

Killed over 90% of three NA bat species



Project Goal:

Purpose:

- To predict the presence, absence, or risk of White-Nose Syndrome (WNS) in various U.S. regions.

Model Focus:

- Identifying areas with active WNS in bats.
- Detecting regions where bats are positive for Pseudogymnoascus destructans (PD) fungus but do not show symptoms of WNS.

Data Utilization:

- Leveraging data from 2006 to 2023.
- Incorporating climate data and bat habitat information for accurate predictions.

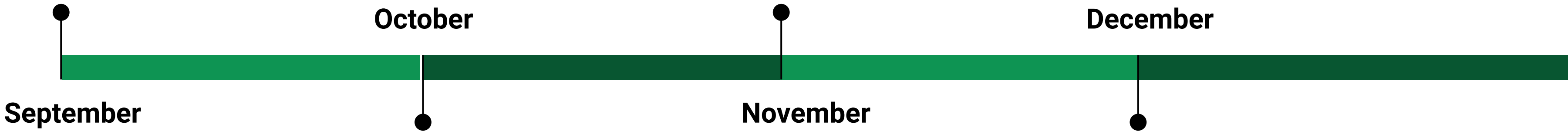
Potential Project Impact

- By improving how conservation groups use their resources, we can better protect bat populations. This can lead to more partnerships and funding for businesses that work in wildlife conservation, making their efforts more effective and impactful.

- Protecting bats can help farms and businesses that grow crops. Bats eat lots of pests naturally, so if there are more bats, farmers will not need to rely on as many chemicals to keep bugs away from their crops

- The findings from our model could be useful for biotech and pharmaceutical companies. They can use this information to develop new ways to treat or prevent White-Nose Syndrome in bats. This could also lead to new opportunities for research and development funding in these fields.

Our Approach



September

Data Hunt and Exploration

Finding data, both independently and from the sources our CA provided, and understanding the components of that data.

October

The Continued Search and the Beginning of the Merge

Still trying to find more data, we start to merge what we have into a useable set, select a label, and narrow down potential features and model algorithms.

November

The Beginning of the End

With a dataset finally intact, we start to build models while learning to handle spatial and temporal data

December

Model and Delivery

We finalize a model and a visualization for our predictions

Resources We Leveraged

- Workflow: Google Collab, Trello
- Data Preparation: Xarray, Pandas
- Models: Scikit Learn

Data Understanding & Data Preparation

Locating and Accessing Data Sources: Challenges

One of the major challenges of this project was finding relevant data.

Data tends to fall into one of these categories:

- Incomprehensive data sets
- Format Unfit for Analysis
- Restricted or Classified Information

The data we did have contained primarily locations, dates, species of bats affected, numbers related to the amount of bats affected, and whether or not there was WNS present at the site.

Locating and Accessing Data Sources: Successes

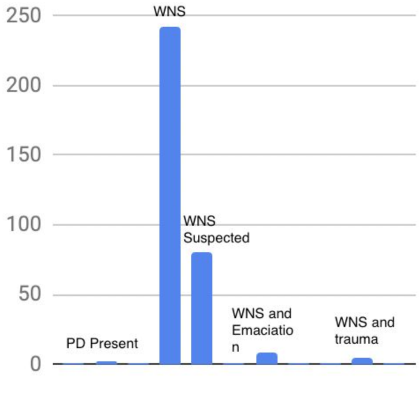
We decided to choose a label and merge our data, even though we didn't have the data we wanted.

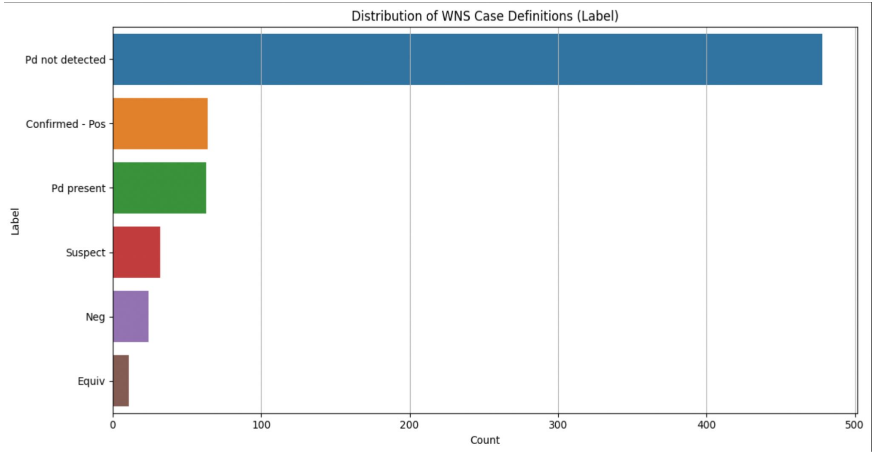
Having a label was helpful, since now we had a real target. We managed to track down a dataset with climate data, both real recorded data a predicted data for the future, with a variety of predictive models to choose from.

The key for this was to stop looking for specific, probably classified data, and shoot for general but still possibly correlated data.

Data Engineering

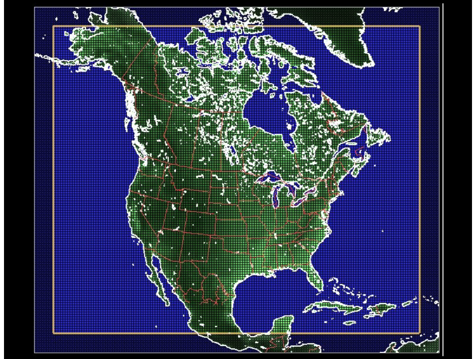
- Our sample data was fairly balanced between examples where WNS was detected and examples where it wasn't
- One of the decisions we had to make was which items should be in the three categories we decided on; WNS, Negative, and Fungus Detected.





Feature Engineering

- We used climate data from The North American Cordex Program
  - Utilizes regional climate models to simulate conditions (2006-2100)
  - Predictions at daily frequencies
  - Spatial resolution of 0.22 degrees/25 km
- The features we extracted:
  - Temperature: annual mean, max, min
  - Humidity: annual mean, max, min
  - Precipitation: annual mean, max, min
  - Surface Downwelling Shortwave Radiation (solar radiation): annual mean
- Feature engineering consisted of scaling the data to enhance compatibility with models employed and introducing a distance parameter to account for spatial aspect of our data
  - We calculated distance between points and the nearest previous disease detection using the Haversine formula

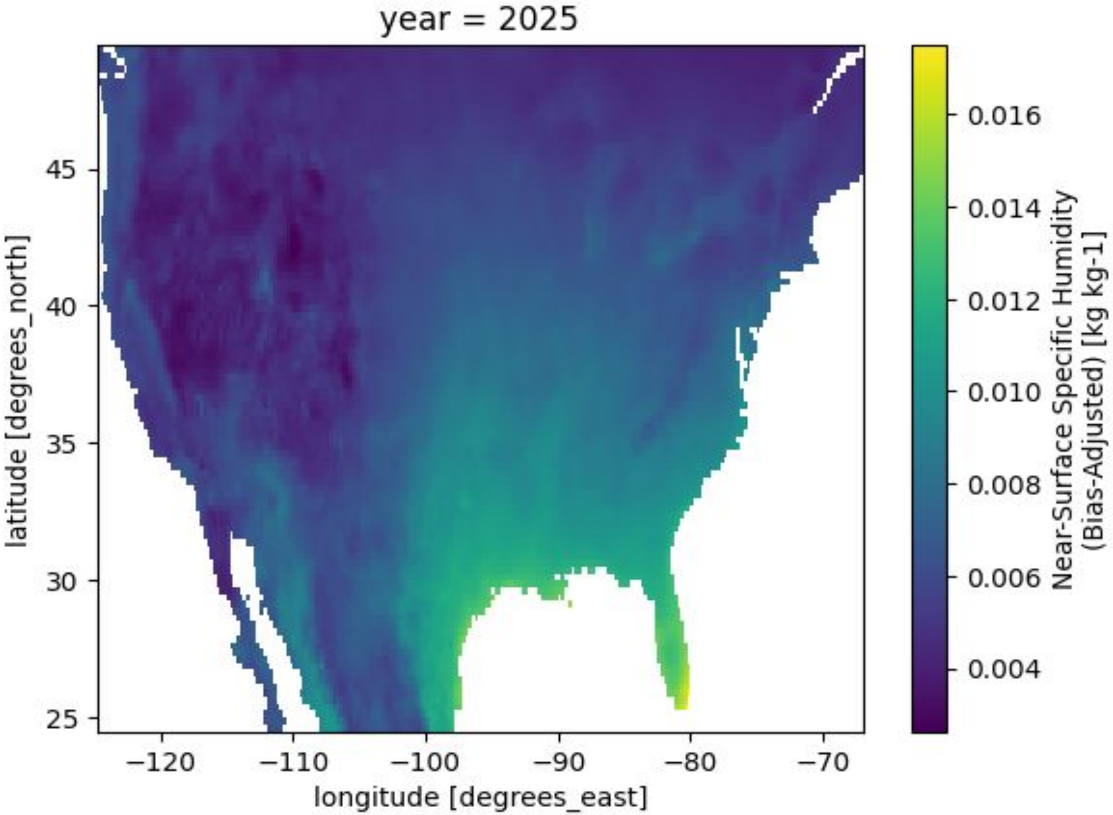


CORDEX-NA simulation domain, 0.44°/50km resolution

WCRP CORDEX

NA-CORDEX

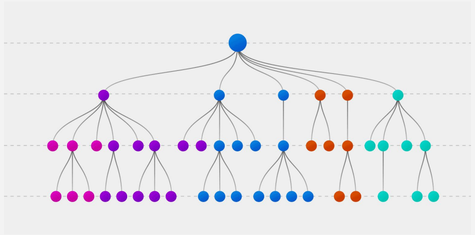
Humidity Trends: 2025



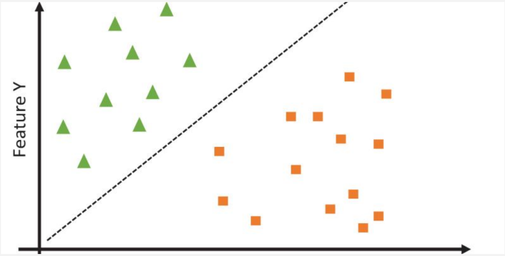
Modeling & Evaluation

Model Selection

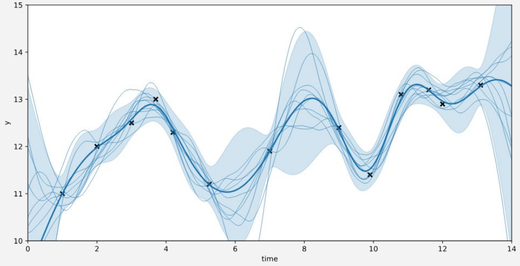
- Decision Trees



- Support Vector Machines

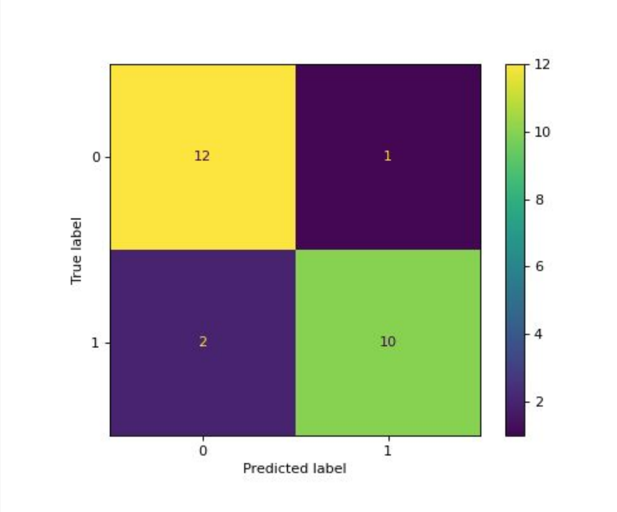


- Gaussian Process

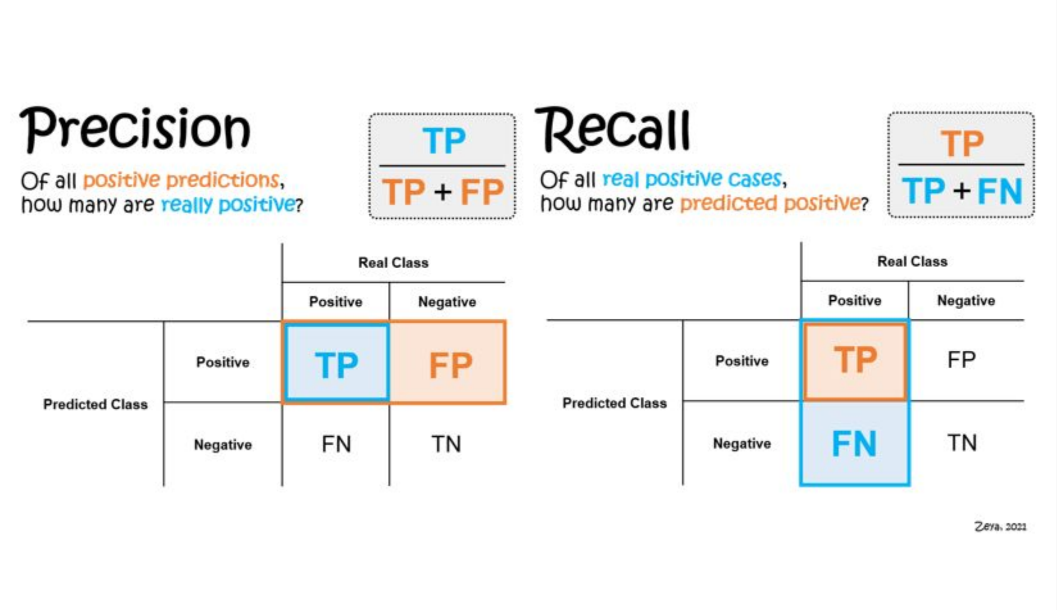


Key Terms for Evaluating Models

Confusion Matrix:



Precision and Recall:



Precision

Of all positive predictions, how many are really positive?

$$\frac{TP}{TP + FP}$$

		Real Class	
		Positive	Negative
Predicted Class	Positive	TP	FP
	Negative	FN	TN

Recall

Of all real positive cases, how many are predicted positive?

$$\frac{TP}{TP + FN}$$

		Real Class	
		Positive	Negative
Predicted Class	Positive	TP	FP
	Negative	FN	TN

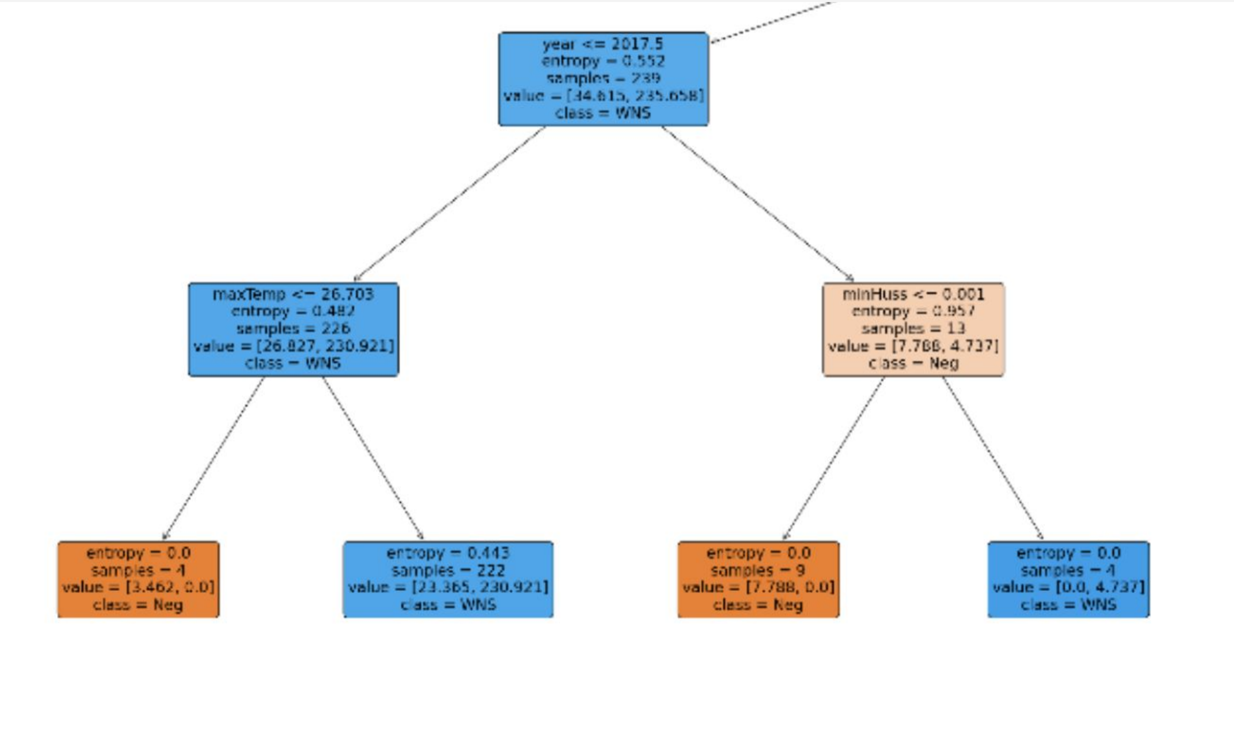
Zeya, 2021

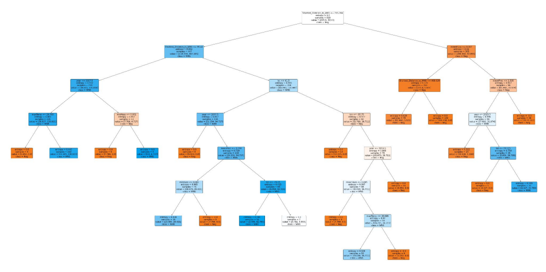
*We want to predict conservatively, so reducing False Positives (cases where WNS is predicted when actually there is no presence of WNS)

Decision Trees - Model Tuning

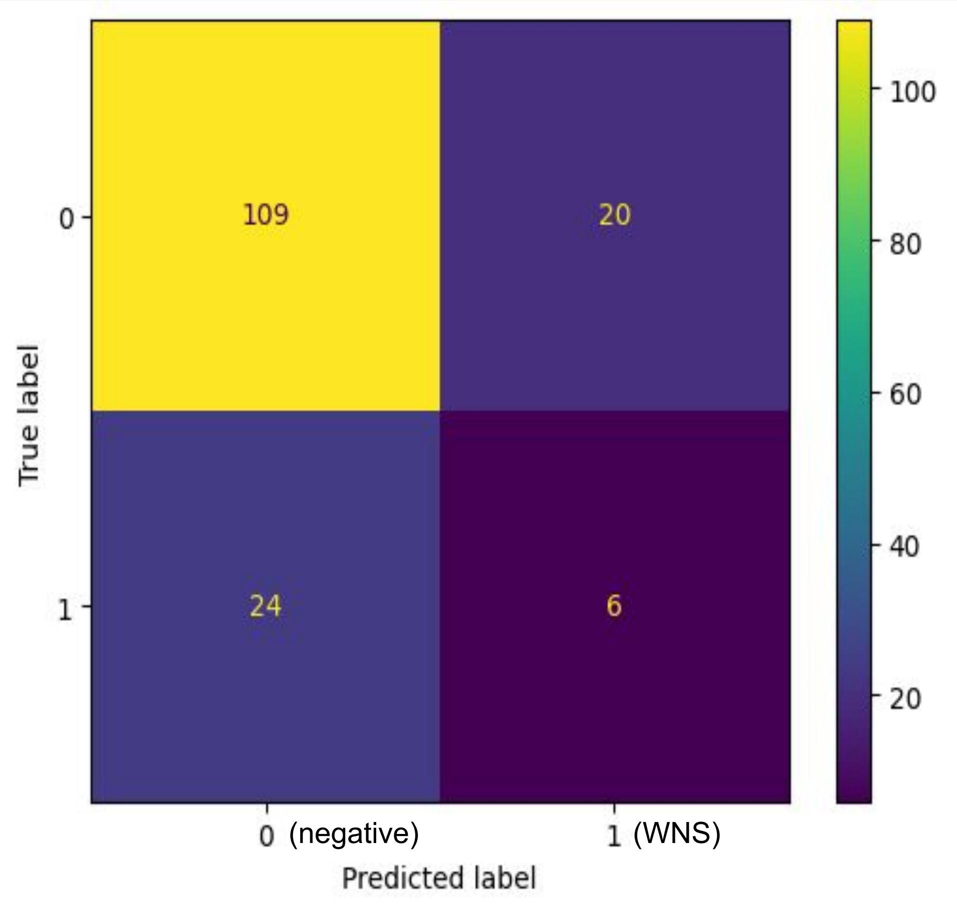
- Step 1: Split training and test data by year 2020
- Step 2: Decision Tree trained and evaluated (precision, recall)
- Step 3: Hyperparameter tuning (GridSearchCV)
    - Notably the model with highest accuracy used:
        - Classweight set to ‘balanced’ to handle imbalance of positive and negative cases
        - Depth of tree set to ‘None’ →risk of overfit especially on small dataset
        - Minimum samples per leaf set to 1 - suggests highly specific tree

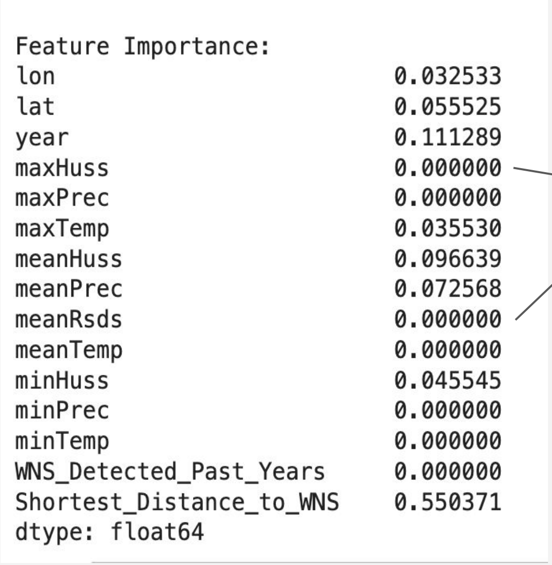
Decision Tree Plot





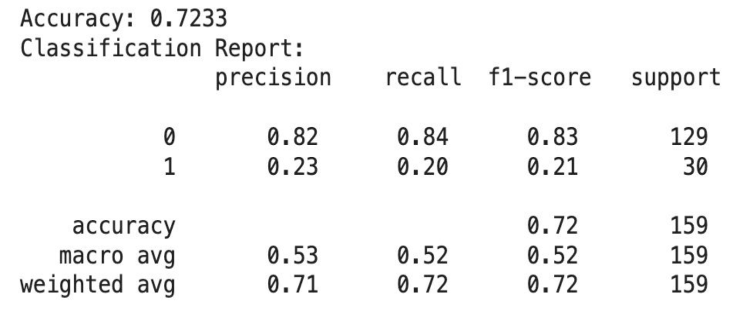
Confusion Matrix





```
Feature Importance:
lon                          0.032533
lat                          0.055525
year                         0.111289
maxHuss                      0.000000
maxPrec                      0.000000
maxTemp                      0.035530
meanHuss                     0.096639
meanPrec                     0.072568
meanRsds                     0.000000
meanTemp                     0.000000
minHuss                      0.045545
minPrec                      0.000000
minTemp                      0.000000
WNS_Detected_Past_Years      0.000000
Shortest_Distance_to_WNS     0.550371
dtype: float64
```

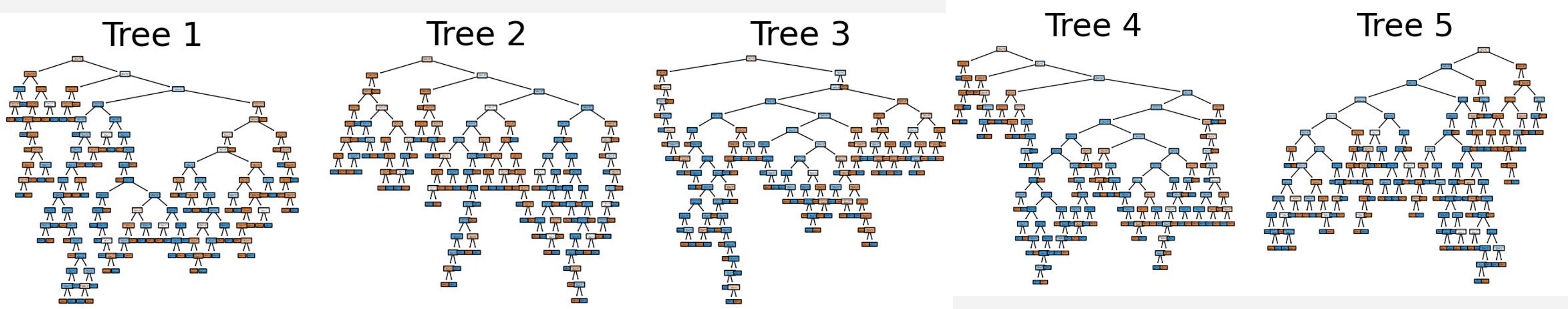
Some variables assigned 0 significance in individual decision tree



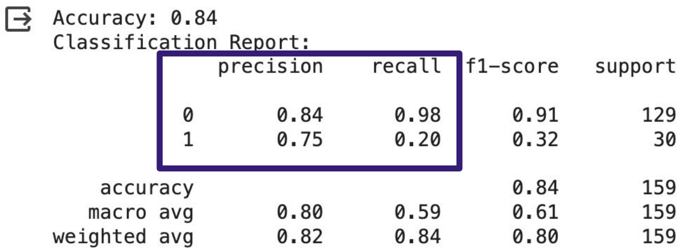
```
Accuracy: 0.7233
Classification Report:
              precision    recall  f1-score   support

           0       0.82      0.84      0.83       129
           1       0.23      0.20      0.21        30

    accuracy                           0.72       159
   macro avg       0.53      0.52      0.52       159
weighted avg       0.71      0.72      0.72       159
```

Decision Trees - Random Forest

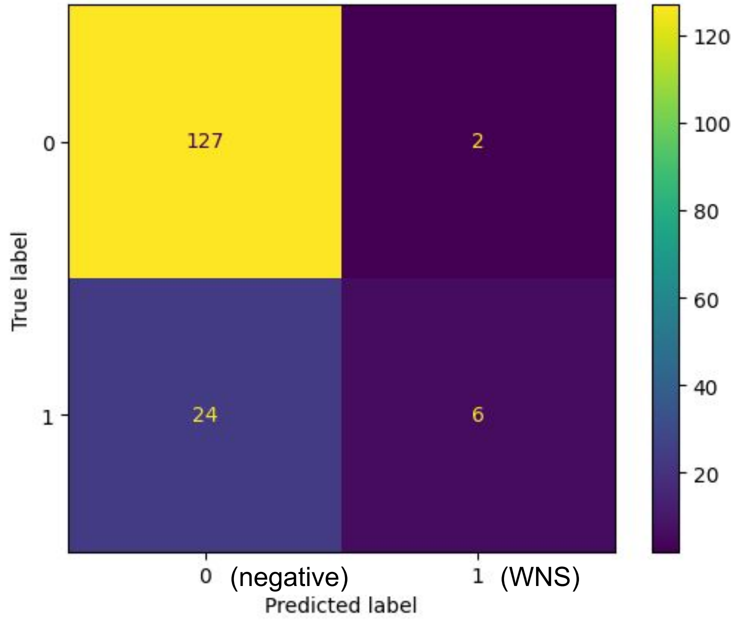
- We transitioned to a random forest model due to concern of overfitting with the small dataset and high number of features
  - Better performance than individual decision tree
  - Robust to outliers and noise
  - Better at handling high-dimensional data (high number of features)



Feature	Importance Score (0-1)
Distance to previous detection	0.224
Longitude	0.127
Mean Precipitation	0.085
Mean Humidity	0.069
Mean Solar Radiation	0.066
Mean Temperature	0.064



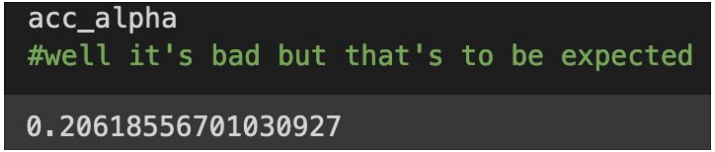
```
Accuracy: 0.84
Classification Report:
              precision    recall  f1-score   support

           0       0.84      0.98      0.91       129
           1       0.75      0.20      0.32        30

    accuracy                           0.84       159
   macro avg       0.80      0.59      0.61       159
weighted avg       0.82      0.84      0.80       159
```



Support Vector Machine - Model Tuning

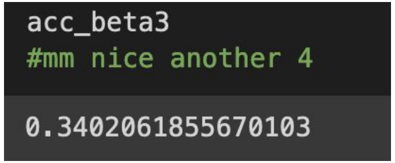
The process of attempting to improve the SVM was simple, if a bit fruitless.

- Step 1: Scaling. SVMs require their data to be scaled to work well.
- Step 2: Seeing what it does with no special settings:
    - It did not do anything impressive →



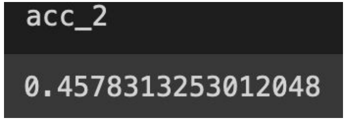
```
acc_alpha
#well it's bad but that's to be expected

0.2061855670103927
```

- Step 3: Do some grid searching and give it better parameters. Also changed the data a little. →



```
acc_beta3
#mm nice another 4

0.3402061855670103
```

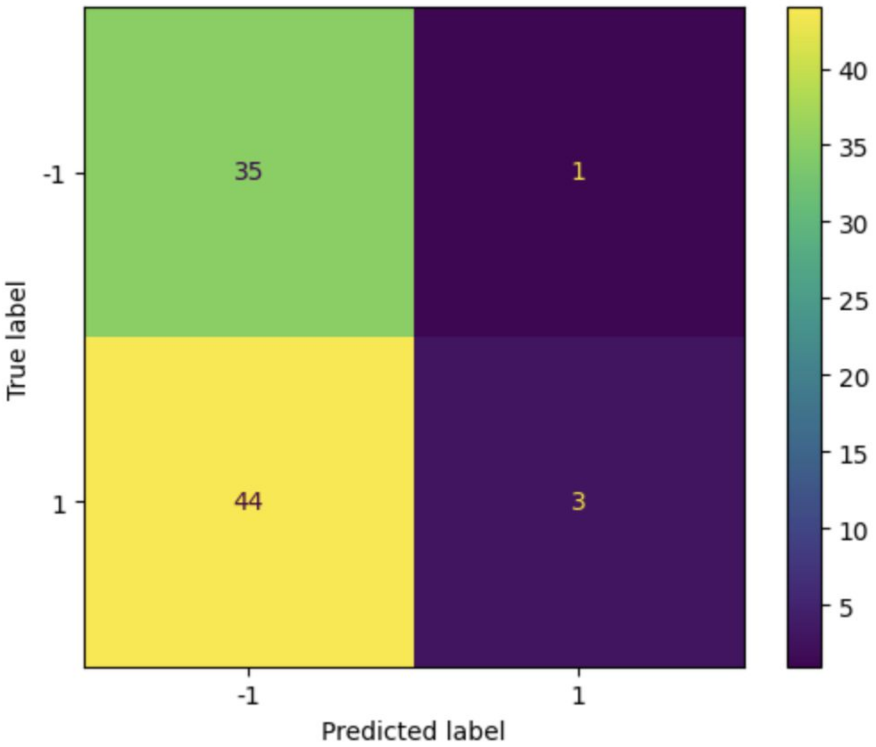
- Step 4: Remove a category to make it binary, creating this improvement →



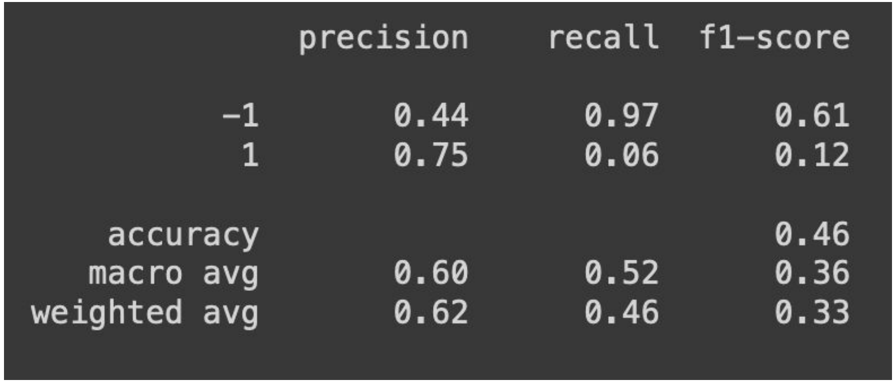
```
acc_2

0.4578313253012048
```

- Step 5: Realize that this number is suspicious…

-1 = negative, 1 = WNS



*This model is flipping a coin, which is why the accuracy is roughly ⅓ with three classes and ½ with two classes.



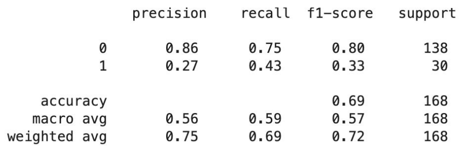
```
              precision    recall  f1-score

          -1       0.44      0.97      0.61
           1       0.75      0.06      0.12

    accuracy                           0.46
   macro avg       0.60      0.52      0.36
weighted avg       0.62      0.46      0.33
```

*The ideal version of this diagram has a yellow diagonal from the top left to the bottom right.

Gaussian Process- Model Tuning

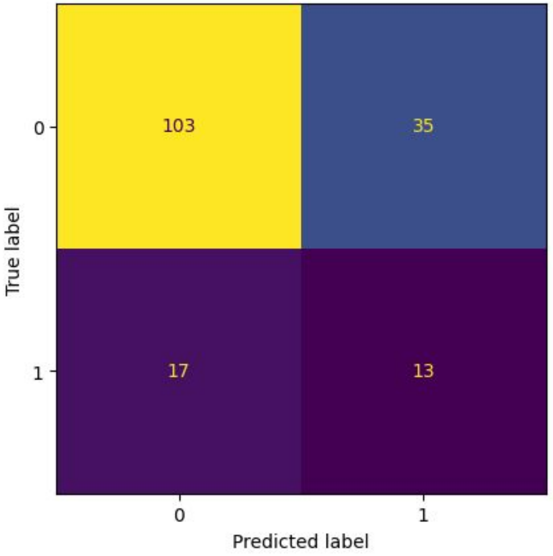
Goal: Identify areas in our dataset that have high probabilities of WNS

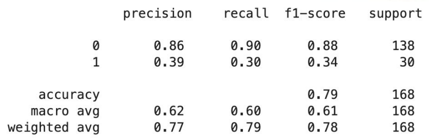
- Step 1: Split training and test data by year 2020
- Step 2: Gaussian Process Model created with spatial and temporal Kernels used to capture spatial and temporal aspects of data
- Step 3: Training model
- Step 4: Transforming Predictions (probabilities) to binary classifications with a threshold
    - 0.5 threshold means probabilities above 0.5 are classified as 1 (WNS detection), while those below are classified as the negative class 0 (No WNS)
    - Scaling this parameter changes precision/recall balance...



```
              precision    recall  f1-score   support

           0       0.86      0.75      0.80       138
           1       0.27      0.43      0.33        30

    accuracy                           0.69       168
   macro avg       0.56      0.59      0.57       168
weighted avg       0.75      0.69      0.72       168
```

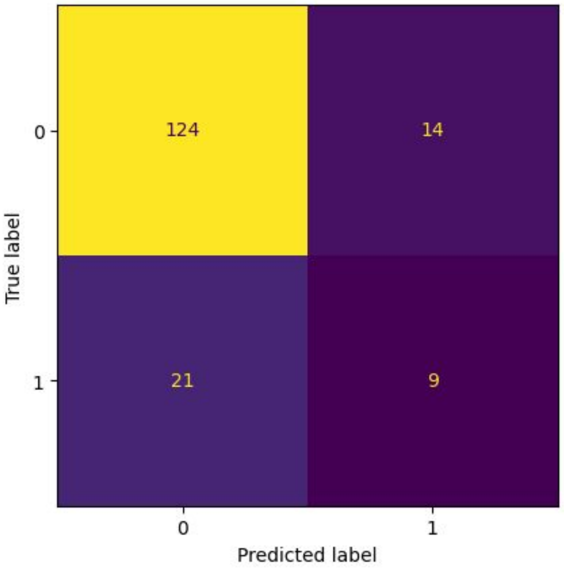
Threshold = 0.3

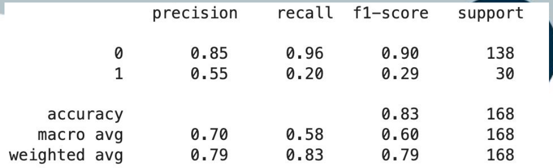




```
              precision    recall  f1-score   support

           0       0.86      0.90      0.88       138
           1       0.39      0.30      0.34        30

    accuracy                           0.79       168
   macro avg       0.62      0.60      0.61       168
weighted avg       0.77      0.79      0.78       168
```

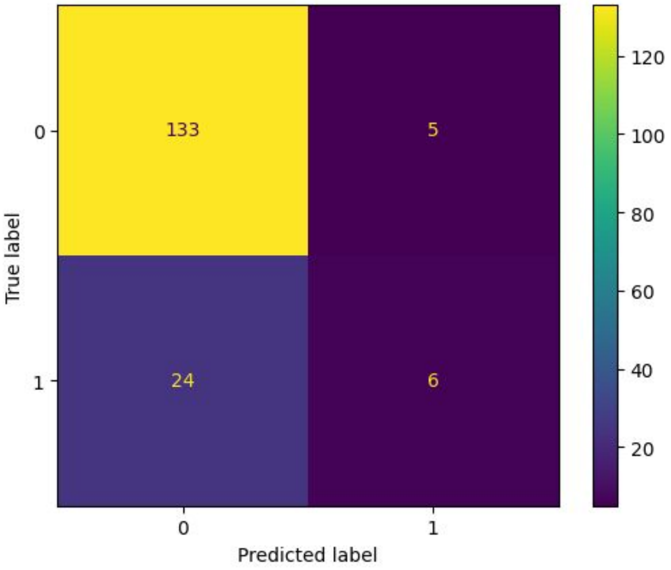
Threshold = 0.5





```
              precision    recall  f1-score   support

           0       0.85      0.96      0.90       138
           1       0.55      0.20      0.29        30

    accuracy                           0.83       168
   macro avg       0.70      0.58      0.60       168
weighted avg       0.79      0.83      0.79       168
```

Threshold = .7



*Not enough data for interpretability, scaling threshold higher allows more conservative predictions

Model Comparison

Model Name	Description	Results	Pros	Cons
Decision Trees	Regression Classification	Accuracy: 0.84	Interpretable Flexible as data updates	Overfitting risk Computationally demand
Support Vector Machines	Specialize in Binary Classification	Accuracy: 0.43	Binary classification Memory efficient	Not convenient to update Interpretability
Gaussian Process	Regression Probability Classifications	Accuracy: 0.83	Confidence levels or predictions Clusters	Interpretability

Final Thoughts

What We Learned

- Sourcing federally gathered natural data
- Merging and Cleaning Datasets
- Working with high dimensional climate data: features with coordinates in space and time
- Time Series predictions
- Using X-Arrays

Potential Next Steps

- **Detailed Bat Population Analysis:**

  Acquire more precise data on bat populations. Examine White-Nose Syndrome effects in depth.

- **Season-Specific Environmental Focus:**

  Prioritize environmental variables specific to the winter season. Winter is critical as it's when bats are most impacted by WNS.

- **Risk Assessment Model for Bat Populations**

  Create a second model that assesses the risk to local bat populations based on the output of the WNS spread model, incorporating factors like species susceptibility, population density, and health status

- **Future Development: User Interface**

  Future work will focus on developing an intuitive interface for public use, allowing non-experts to easily access and understand information about WNS status and bat population risks.

Questions?

Reserve slides: