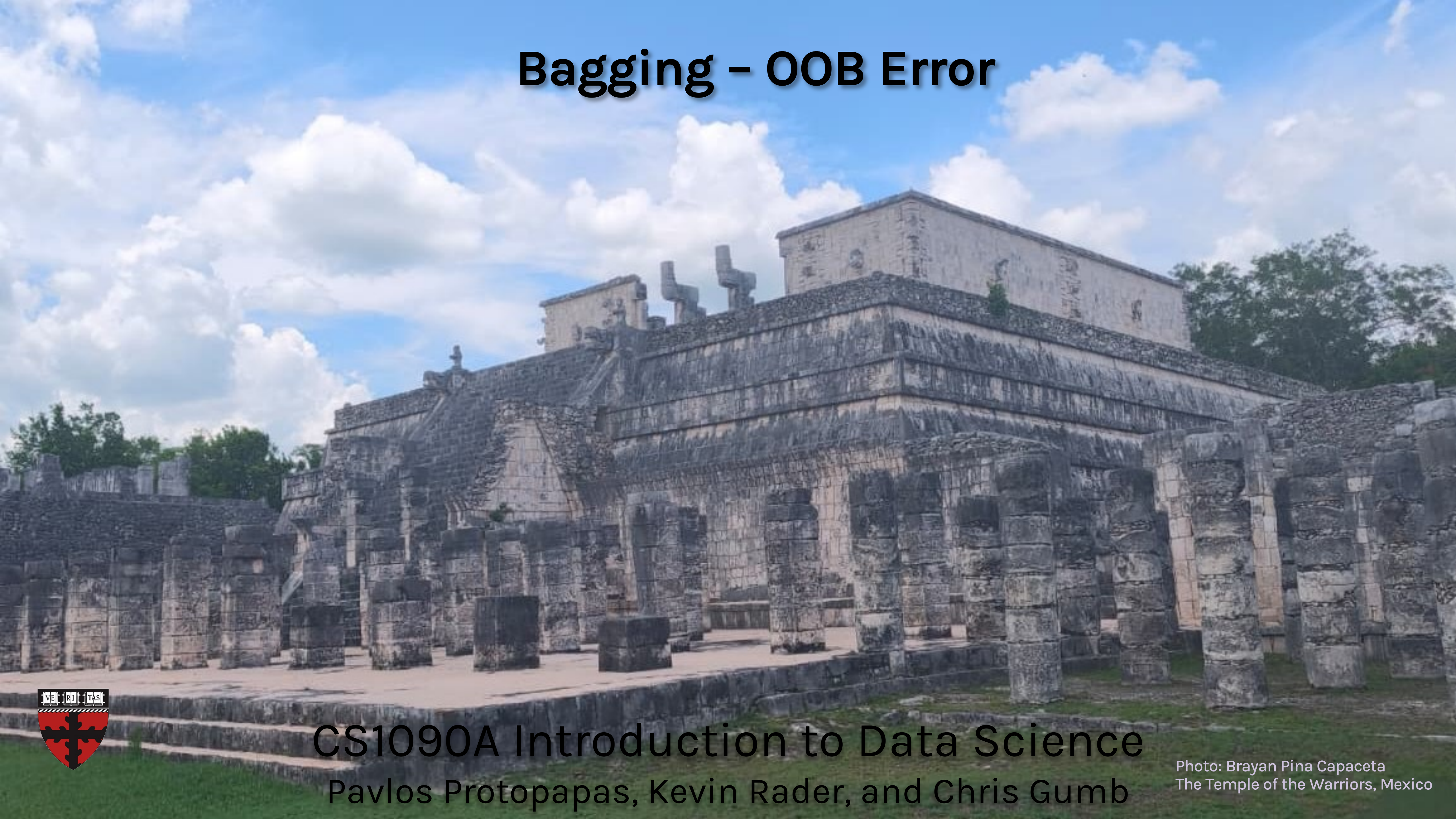
Bagging – OOB Error



CS1090A Introduction to Data Science

Pavlos Protopapas, Kevin Rader, and Chris Gumb

Photo: Brayan Pina Capaceta
The Temple of the Warriors, Mexico

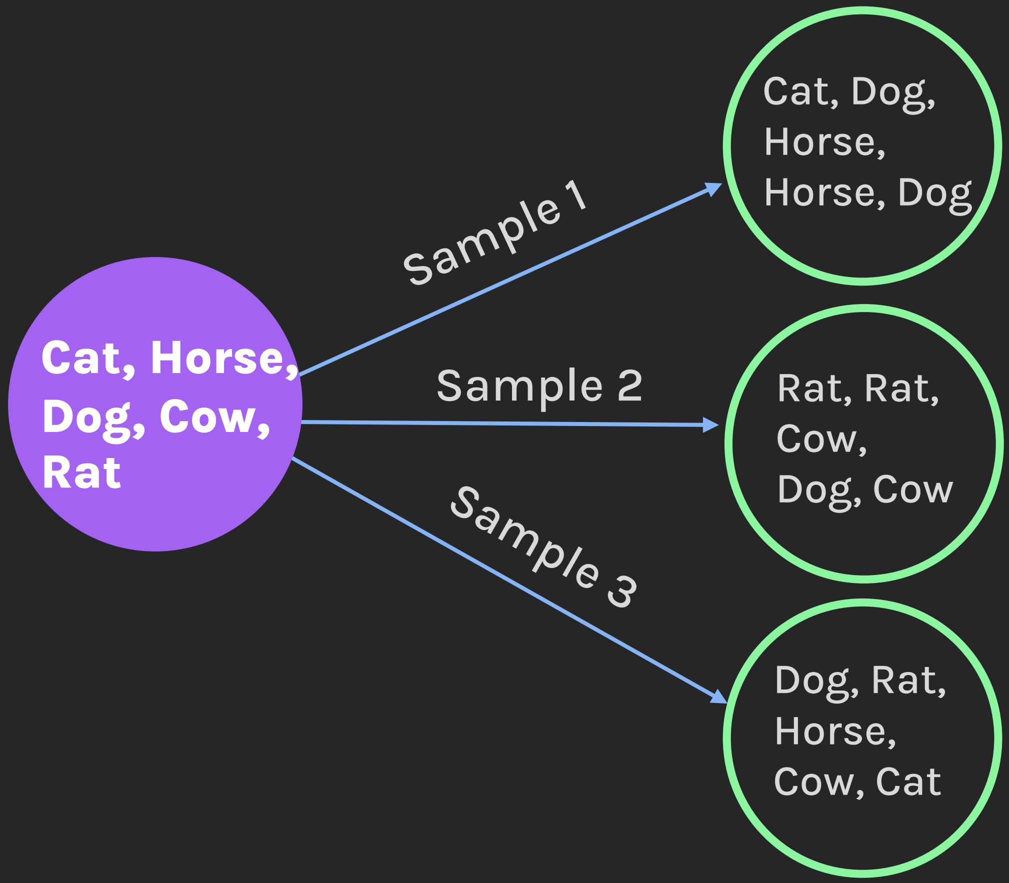
Outline

- Motivation
- Bagging
- **Out-of-bag Error**

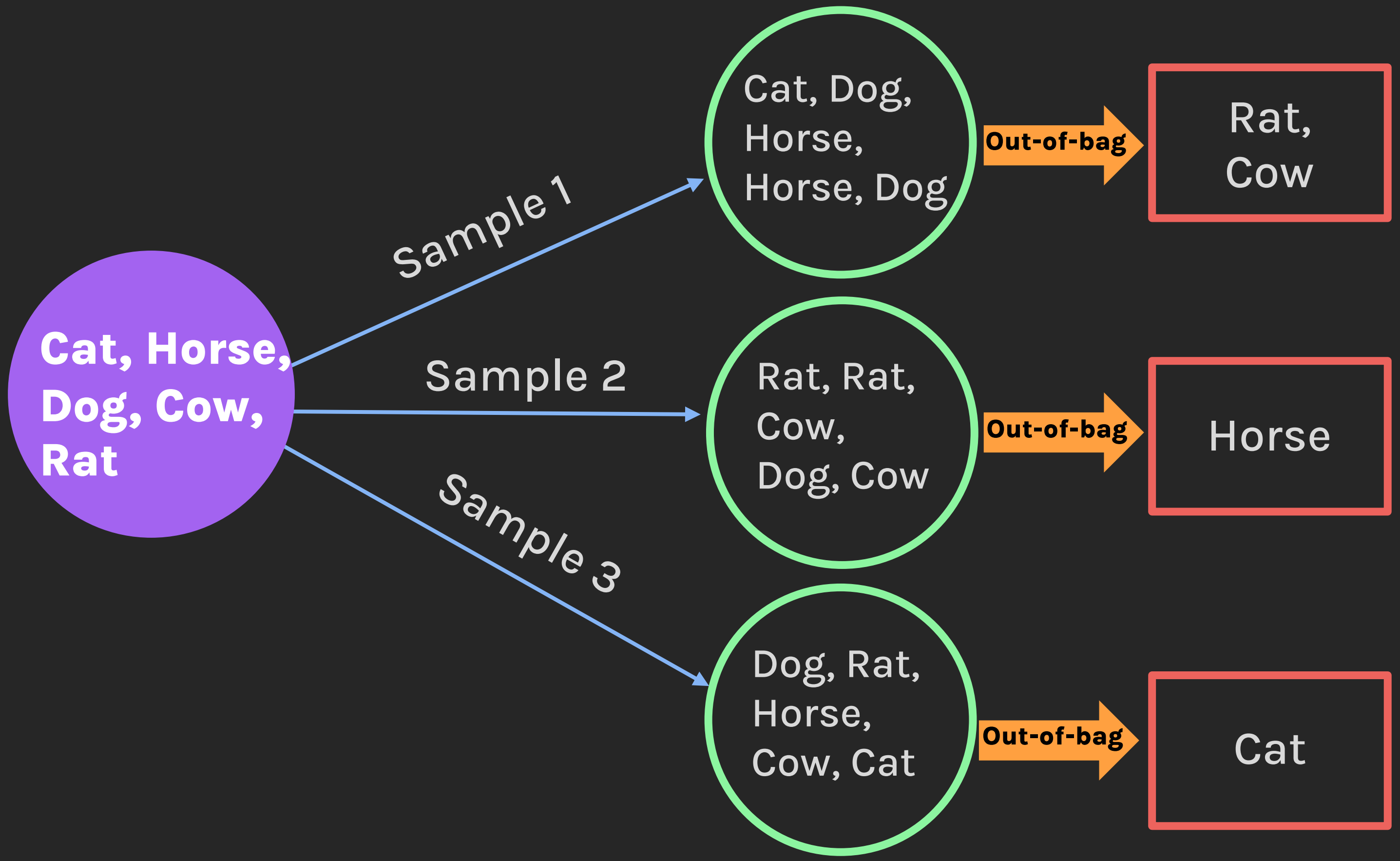
What is OOB?

Cat, Horse, Dog, Cow, Rat

What is OOB?

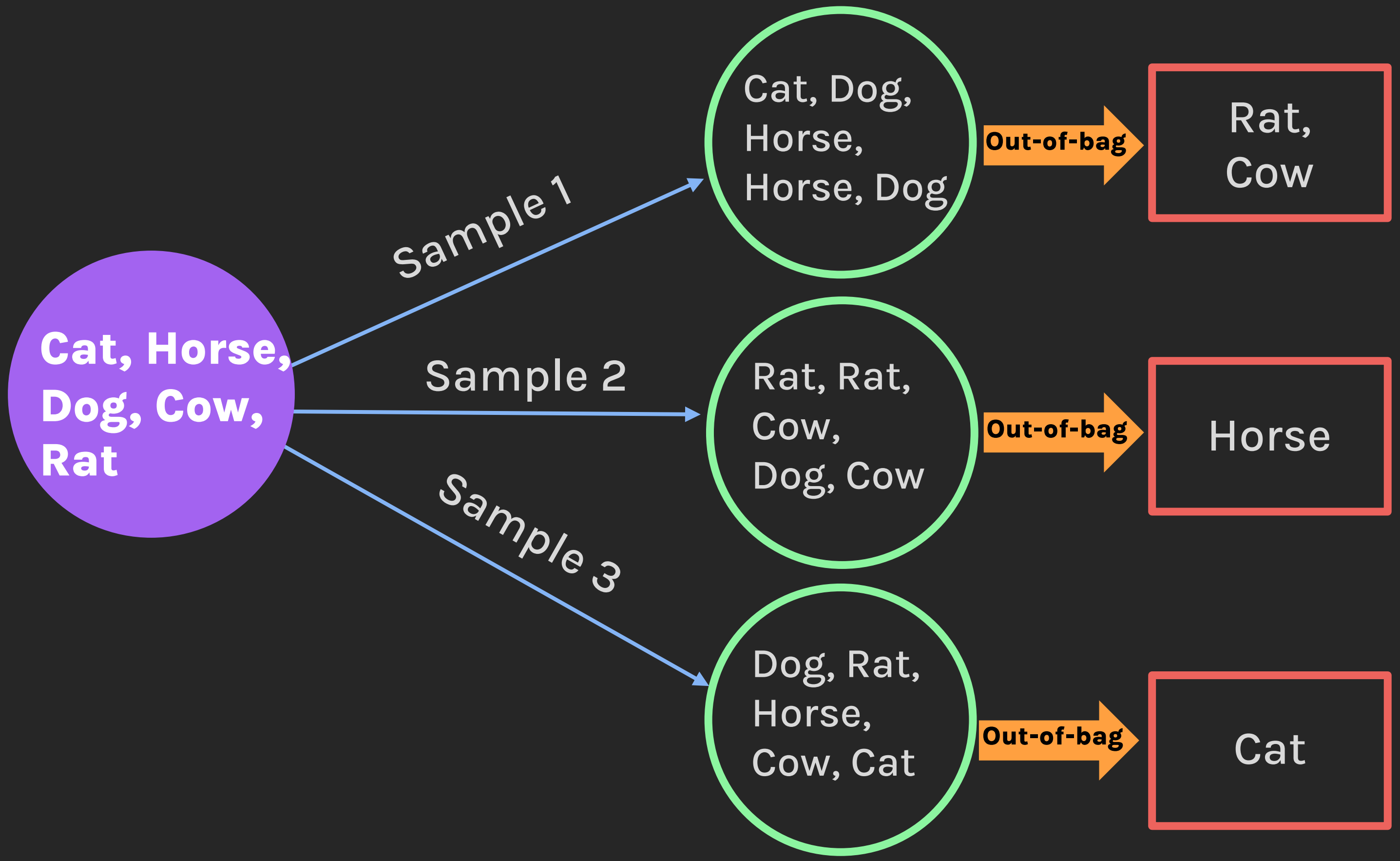

Cat, Horse, Dog, Cow, Rat
Sample 1
Sample 2
Sample 3
Cat, Dog, Horse, Horse, Dog
Rat, Rat, Cow, Dog, Cow
Dog, Rat, Horse, Cow, Cat

What is OOB?



Out-of-bag estimate is a method of determining the prediction error whilst being trained.

What is OOB?



Out-of-bag estimate is a method of determining the prediction error while being trained.

Why?

- To measure generalizability.
- To replace the need for a separate measurement of performance for a validation-set performance.

Let us explore this in more details with another example

Out-of-bag Error (OOB)

Original Data

X	Y
x_1	y_1
x_2	y_2
x_3	y_3
x_4	y_4
x_5	y_5
⋮	⋮
x_n	y_n

Predictor/Feature

Response/Target

Out-of-bag Error (OOB)

Original Data

X	Y
x_1	y_1
x_2	y_2
x_3	y_3
x_4	y_4
x_5	y_5
⋮	⋮
x_n	y_n

Predictor/Feature

Response/Target

Bootstrap Sample 1

X	Y
x_1	y_1
x_3	y_3
x_5	y_5
x_{21}	y_{21}
x_{35}	y_{35}
⋮	⋮
x_k	y_k

Out-of-bag Error (OOB)

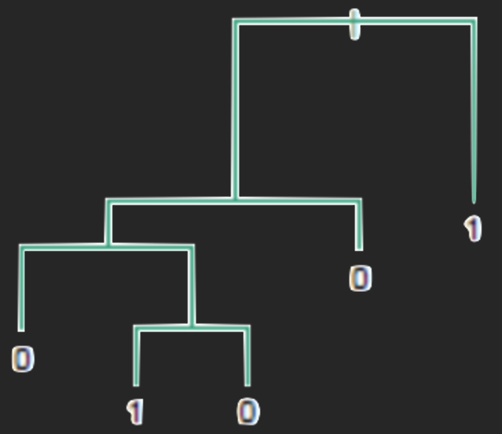
Original Data

X	Y
x_1	y_1
x_2	y_2
x_3	y_3
x_4	y_4
x_5	y_5
⋮	⋮
x_n	y_n

Bootstrap Sample 1

X	Y
x_1	y_1
x_3	y_3
x_5	y_5
x_{21}	y_{21}
x_{35}	y_{35}
⋮	⋮
x_k	y_k

Decision Tree 1



Predictor/Feature

Response/Target

Out-of-bag Error (OOB)

Original Data

X	Y
x_1	y_1
x_2	y_2
x_3	y_3
x_4	y_4
x_5	y_5
⋮	⋮
x_n	y_n

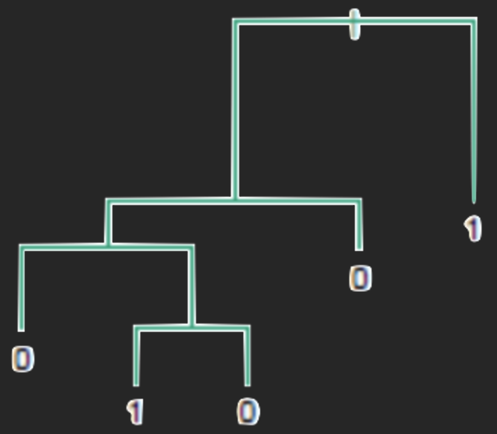
Predictor/Feature

Response/Target

Bootstrap Sample 1

X	Y
x_1	y_1
x_3	y_3
x_5	y_5
x_{21}	y_{21}
x_{35}	y_{35}
⋮	⋮
x_k	y_k

Decision Tree 1



Used and unused data

X	Y
x_1	y_1
x_2	y_2
x_3	y_3
x_4	y_4
x_5	y_5
⋮	⋮
x_n	y_n

Out-of-bag Error (OOB)

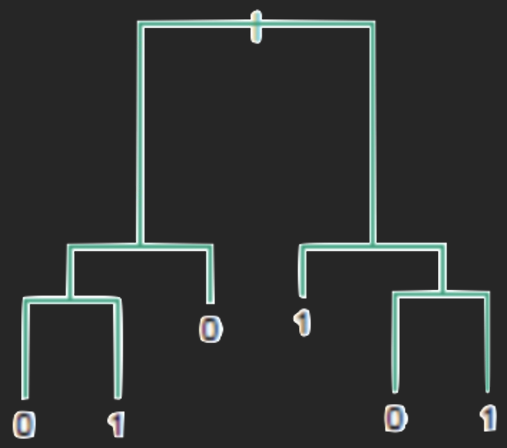
Original Data

X	Y
x_1	y_1
x_2	y_2
x_3	y_3
x_4	y_4
x_5	y_5
⋮	⋮
x_n	y_n

Bootstrap Sample 2

X	Y
x_5	y_5
x_7	y_7
x_{13}	y_{13}
x_{27}	y_{27}
x_{32}	y_{32}
⋮	⋮
x_k	y_k

Decision Tree 2



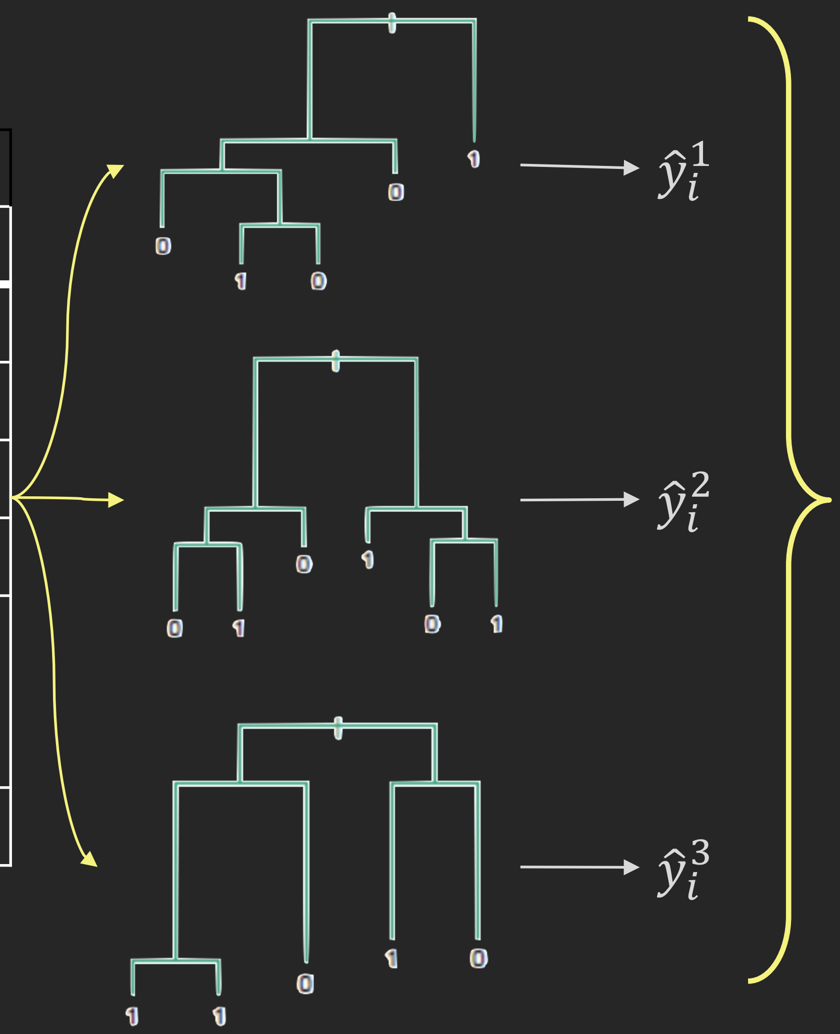
Used and unused data

X	Y
x_1	y_1
x_2	y_2
x_3	y_3
x_4	y_4
x_5	y_5
⋮	⋮
x_n	y_n

Point-wise out-of-bag error

B Trees that did not see $\{X_i, y_i\}$

X	Y
X_1	y_1
X_2	y_2
X_3	y_3
X_4	y_4
X_5	y_5
.. X_i ..	.. y_i ..
X_n	y_n



- Identify observations the trained models have not seen
- Get the predictions for these observations from the models

Point-wise out-of-bag error

Take majority for classification and average for regression tasks as the validation prediction for that observation

Point-wise prediction

Point-wise out-of-bag error

Classification

$$\hat{y}_{i,pw} = majority(\hat{y}_i^j)$$

$$e_i = \mathbb{I}(\hat{y}_{i,pw} \neq y_i)$$

Regression

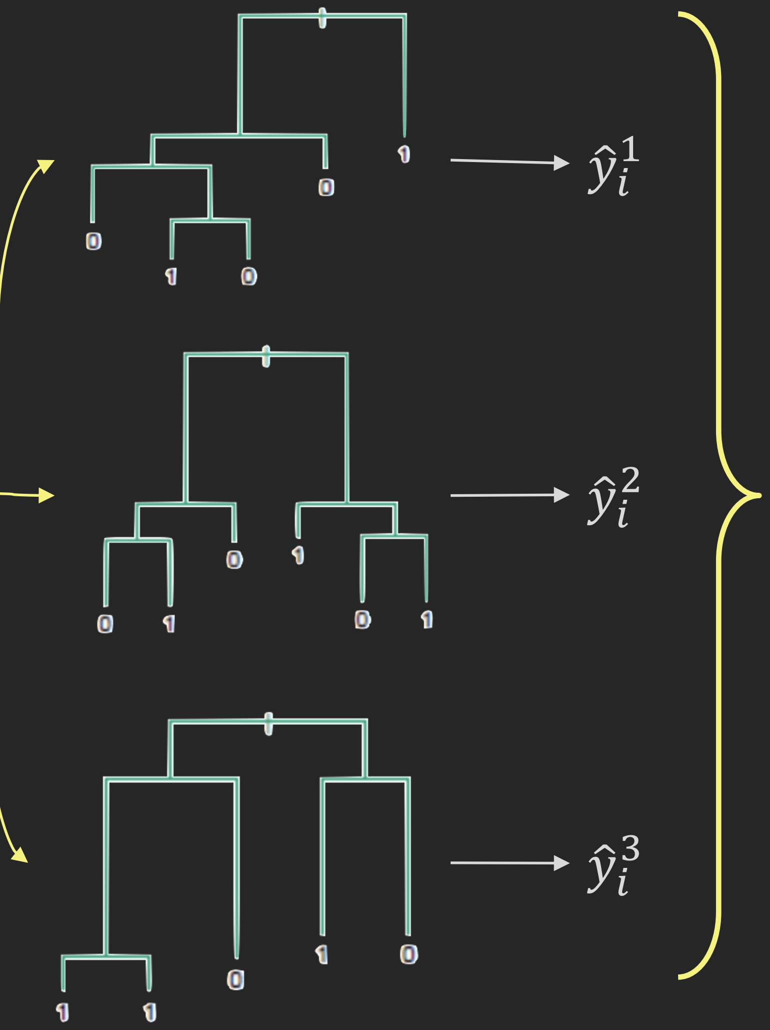
$$\hat{y}_{i,pw} = \frac{1}{B}\sum_{j \in B} \hat{y}_{i,j}$$

$$e_i = (y_i - \hat{y}_{i,pw})^2$$

Point-wise out-of-bag error

X	Y
X_1	y_1
X_2	y_2
X_3	y_3
X_4	y_4
X_5	y_5
.. X_i ..	.. y_i ..
X_n	y_n

B Trees that did not see $\{X_i, y_i\}$



Point-wise prediction

Point-wise out-of-bag error

Classification

$$\hat{y}_{i,pw} = majority(\hat{y}_i^j)$$

$$e_i = \mathbb{I}(\hat{y}_{i,pw} \neq y_i)$$

Regression

$$\hat{y}_{i,pw} = \frac{1}{B}\sum_{j \in B} \hat{y}_{i,j}$$

$$e_i = (y_i - \hat{y}_{i,pw})^2$$

OOB Error

We average the point-wise out-of-bag errors over the full training set.

Classification

$$\boxed{Error_{OOB} = \frac{1}{N}\sum_{i}^{N} e_i = \frac{1}{N}\sum_{i}^{N} \mathbb{I}\left(\hat{y}_{i,pw} \neq y_i\right)}$$

Regression

$$\boxed{Error_{OOB} = \frac{1}{N}\sum_{i}^{N} e_i = \frac{1}{N}\sum_{i}^{N} \left(y_i - \hat{y}_{i,pw}\right)^2}$$

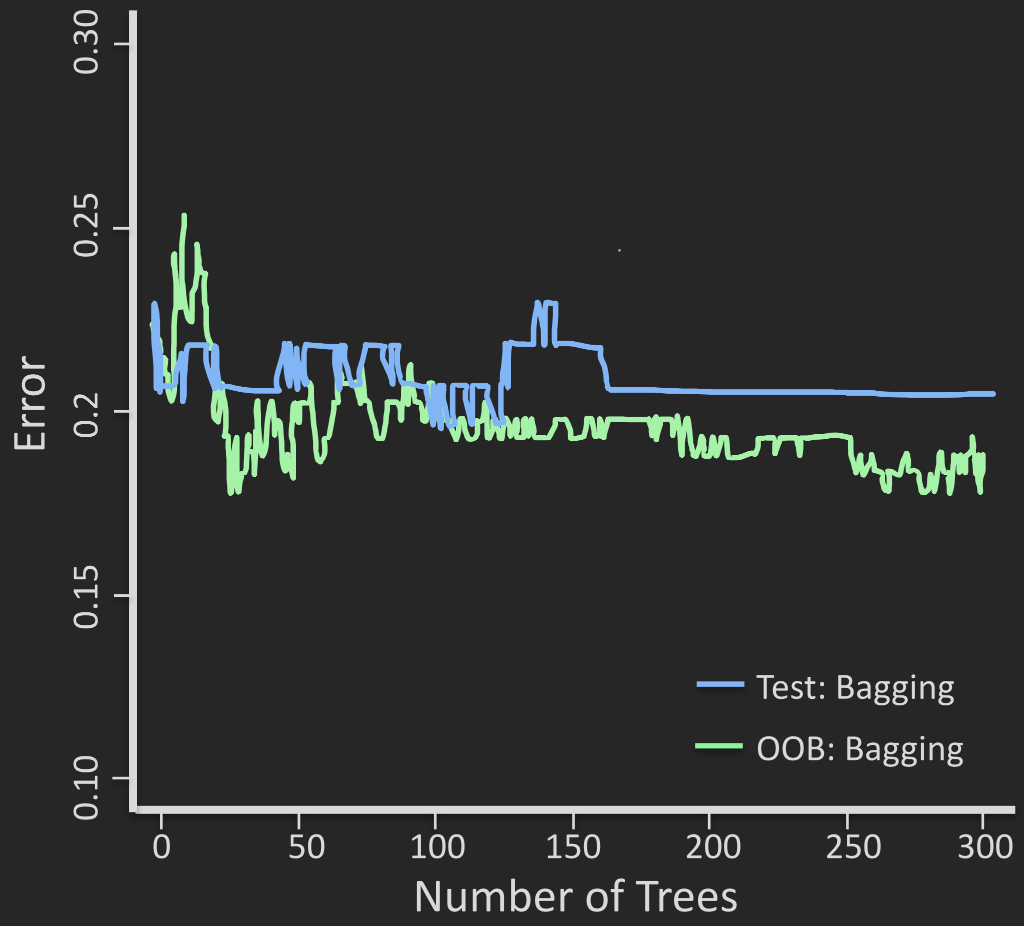
Out-of-Bag Error: Summary

With ensemble methods, we get a new metric for assessing the predictive performance of the model, the ***out-of-bag error***.

Given a training set and an ensemble of models, we compute the ***out-of-bag error*** by

1. For each point x_i in the training set, we average the predicted outputs $\hat{y}_i'$s. To do so we only use the ***B*** trees whose bootstrap training set excludes this point.
2. We compute the error of this averaged prediction. We call this the **point-wise out-of-bag error.**
3. We average the point-wise out-of-bag error over the full training set ***N***.

Why OOB Error? COMPARING OOB AND CROSS VALIDATION



- While using the cross-validation technique, every validation set has already been seen in training by a few decision trees and hence there is a leakage of data.
- OOB Error prevents leakage and yields a better model with lower variance or less overfitting.
- There is also lesser computational cost for OOB as compared to CV for bagging.

Drawbacks of Bagging (revisited)

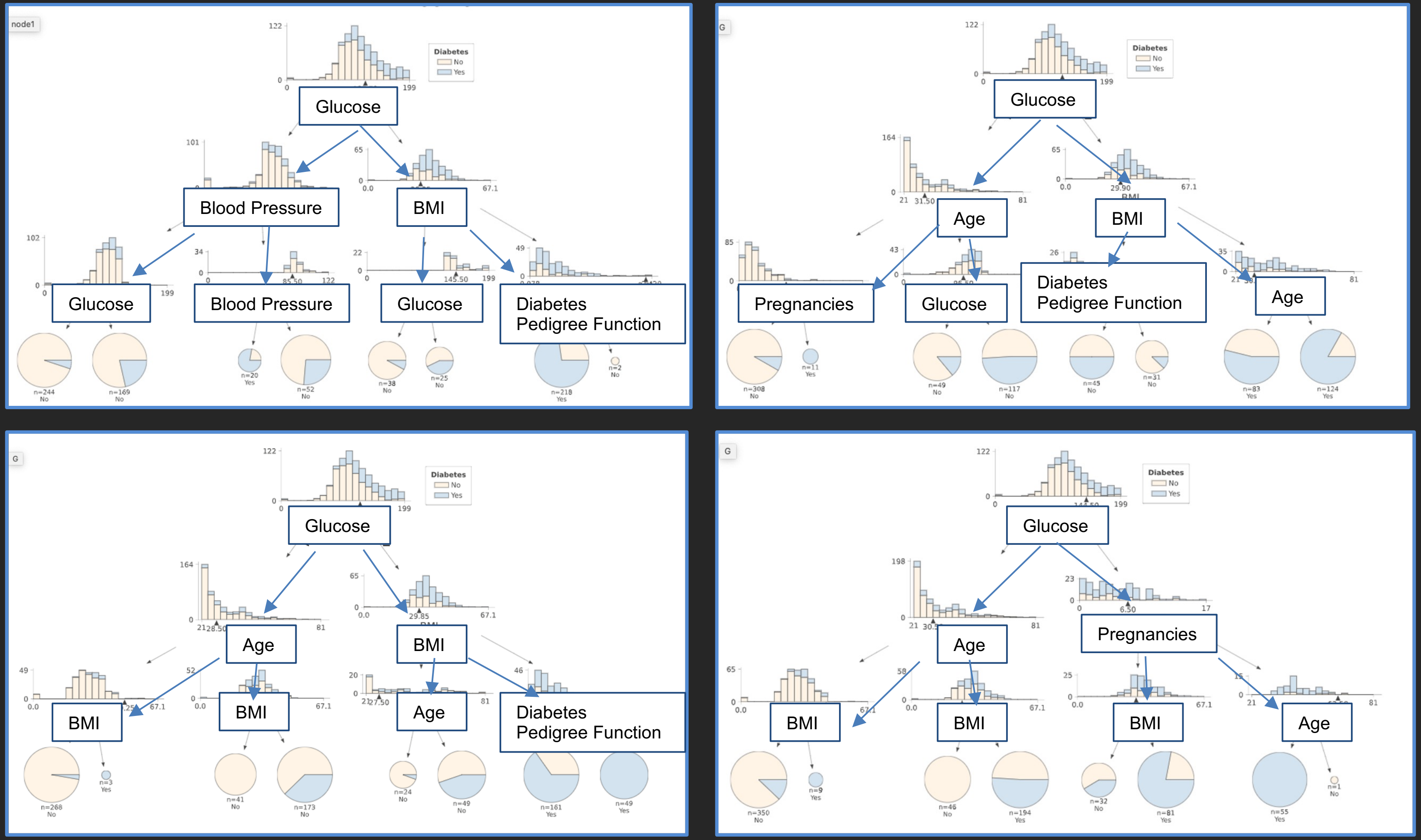
Interpretability:

Still an issue and we will address this later.

If the individual trees are too shallow, the ensembled model can still underfit. Even if we combine many underfitting trees we will still underfit.

If the individual trees are too large, the ensembled model could still overfit.

Drawbacks of Bagging



For each bootstrap, we build a decision tree.

Created by: Dr. Rahul Dave

Improving on Bagging

In practice, the trees in Bagging tend to be **highly correlated**.

- Suppose we have an **extremely strong predictor**, x_j, in the training set amongst **moderate predictors**. Then the greedy learning algorithm ensures that most of the models in the ensemble will choose to split on x_j in early iterations.

- However, we assumed (or hope) that each tree in the ensemble is **independently and identically distributed**.

Next Wednesday, on 'Tree Mysteries Unveiled': Can trees ever truly be independent? 🌳 The secrets unraveled! Tune in and unlock the enigma... Only at the Wednesday Lecture!"

📺🍿

Thank you