UNIVERSITY OF CAMBRIDGE

# Random Forests & Other Random Thoughts

An overview of the fundamentals along with how and when to use these ensemble methods

ICCS Summer School - 10/07/2025 - Robert Edwin Rouse

# Session Outline

## Session One - Fundamentals

- Decision tree overview
- Decision tree explicability
- Random Forest overview
- B(ootstrap) agg(regat)ing
- Random Forest inexplicability
- Classification Exercise

## Session Two - Practical Applications

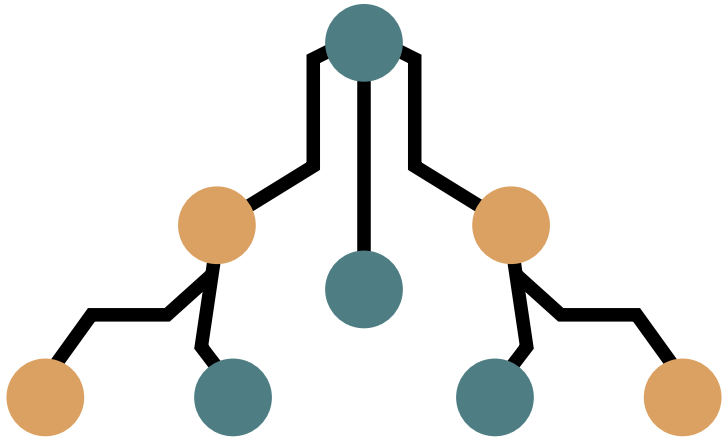
- Regression Exercise
- Interpolation vs Extrapolation
- Feature importance & permutation
- Platt Scaling
- Gradient Boosting
- Other Thoughts

UNIVERSITY OF CAMBRIDGE

# Part I - Fundamentals

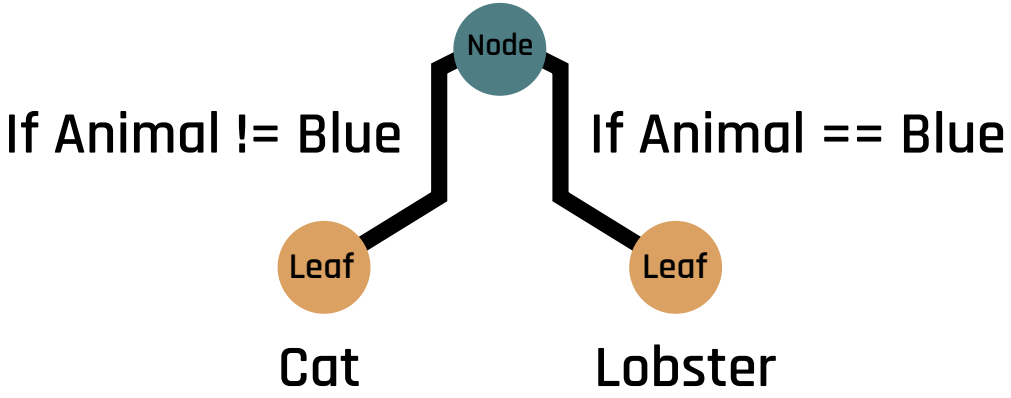
# The Decision Tree

# The Decision Tree



# The Decision Tree

- An algorithm, or representation of one, that is formed of conditional control statements only.
- Features a root (or starting) point, nodes, and leaves.
- Conditional statements cause branching at the root and nodes for information flows to different leaves



# An Example

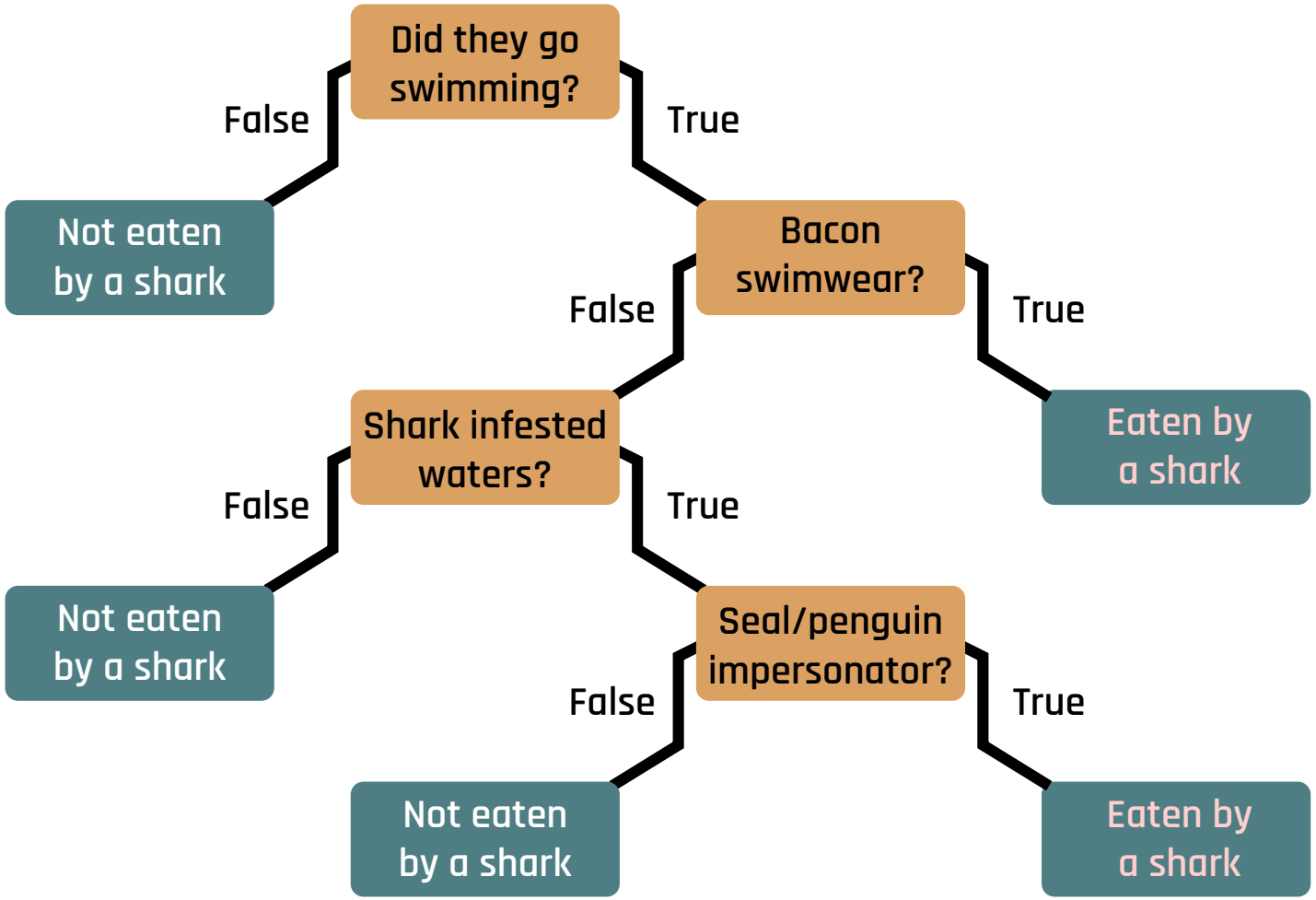
# A Summer Example, More Specifically

# A Scenario That Might Resemble Jaws but Totally Isn't Jaws for Copyright Reasons

# A Summer Example - Attacked by a Shark or Not

| Likelihood of Being Attacked by a Shark at the Beach | | | | |
|---|---|---|---|---|
| Went Swimming | Shark Infested Waters | Impersonated a Seal or Penguin | Swimwear Made of Bacon | Attacked by a Shark |
| Yes | No | Yes | No | No |
| Yes | Yes | Yes | No | Yes |
| Yes | Yes | Yes | No | Yes |
| Yes | No | No | Yes | Yes |
| Yes | Yes | No | No | No |
| No | Yes | No | No | No |
| No | No | Yes | No | No |
| No | Yes | No | Yes | No |

# A Summer Example - Attacked by a Shark or Not



# Impurity, Entropy, & Information Gain, Oh my!

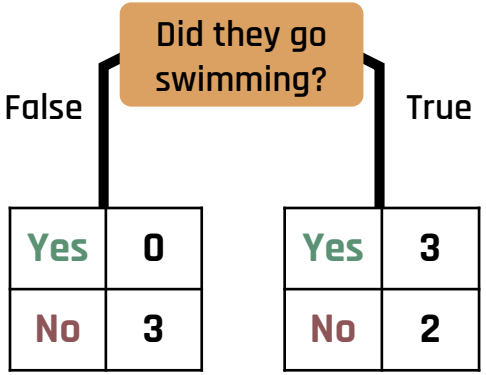
## (In Other Words: How Do We Know Where To Start?)

- Looking to find the conditional that gives the best split
- An 'impure' leaf contains a mixture of class labels
- Need to quantify the impurity
- A few methods available but we'll focus on Gini Impurity

# Impurity, Entropy, & Information Gain, Oh my!

## (In Other Words: How Do We Know Where To Start?)

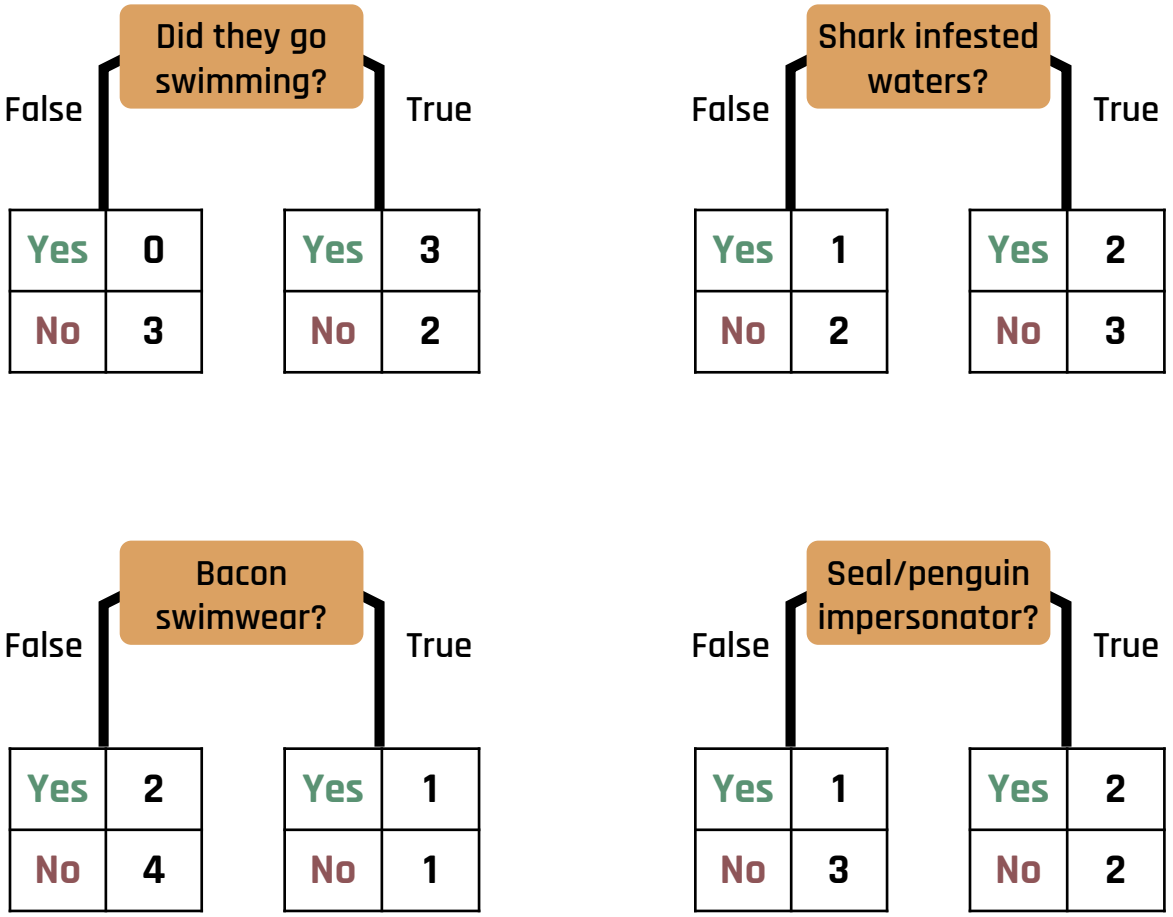
- Looking to find the conditional that gives the best split
- An 'impure' leaf contains a mixture of class labels
- Need to quantify the impurity
- A few methods available but we'll focus on Gini Impurity



# Impurity, Entropy, & Information Gain, Oh my!

## (In Other Words: How Do We Know Where To Start?)

- Looking to find the conditional that gives the best split
- An 'impure' leaf contains a mixture of class labels
- Need to quantify the impurity
- A few methods available but we'll focus on Gini Impurity



# Impurity, Entropy, & Information Gain, Oh my!

## (In Other Words: How Do We Know Where To Start?)

$$G = \sum_{i=1}^{C} P(i) \cdot (1 - P(i))$$

Did they go swimming?

| False | | True | |
|---|---|---|---|
| Yes | 0 | Yes | 3 |
| No | 3 | No | 2 |

Shark infested waters?

| False | | True | |
|---|---|---|---|
| Yes | 1 | Yes | 2 |
| No | 2 | No | 3 |

Bacon swimwear?

| False | | True | |
|---|---|---|---|
| Yes | 2 | Yes | 1 |
| No | 4 | No | 1 |

Seal/penguin impersonator?

| False | | True | |
|---|---|---|---|
| Yes | 1 | Yes | 2 |
| No | 3 | No | 2 |

# Impurity, Entropy, & Information Gain, Oh my!

## (In Other Words: How Do We Know Where To Start?)

- $$G = \sum_{i=1}^{C} P(i) \cdot (1 - P(i))$$

- $$G = 1 - \sum_{i=1}^{C} P(i)^2$$

**Did they go swimming?**

| False | | True | |
|---|---|---|---|
| Yes | 0 | Yes | 3 |
| No | 3 | No | 2 |
| G = 0 | | G = 0.48 | |

**Shark infested waters?**

| False | | True | |
|---|---|---|---|
| Yes | 1 | Yes | 2 |
| No | 2 | No | 3 |
| G = 0.44 | | G = 0.48 | |

**Bacon swimwear?**

| False | | True | |
|---|---|---|---|
| Yes | 2 | Yes | 1 |
| No | 4 | No | 1 |
| G = 0.44 | | G = 0.5 | |

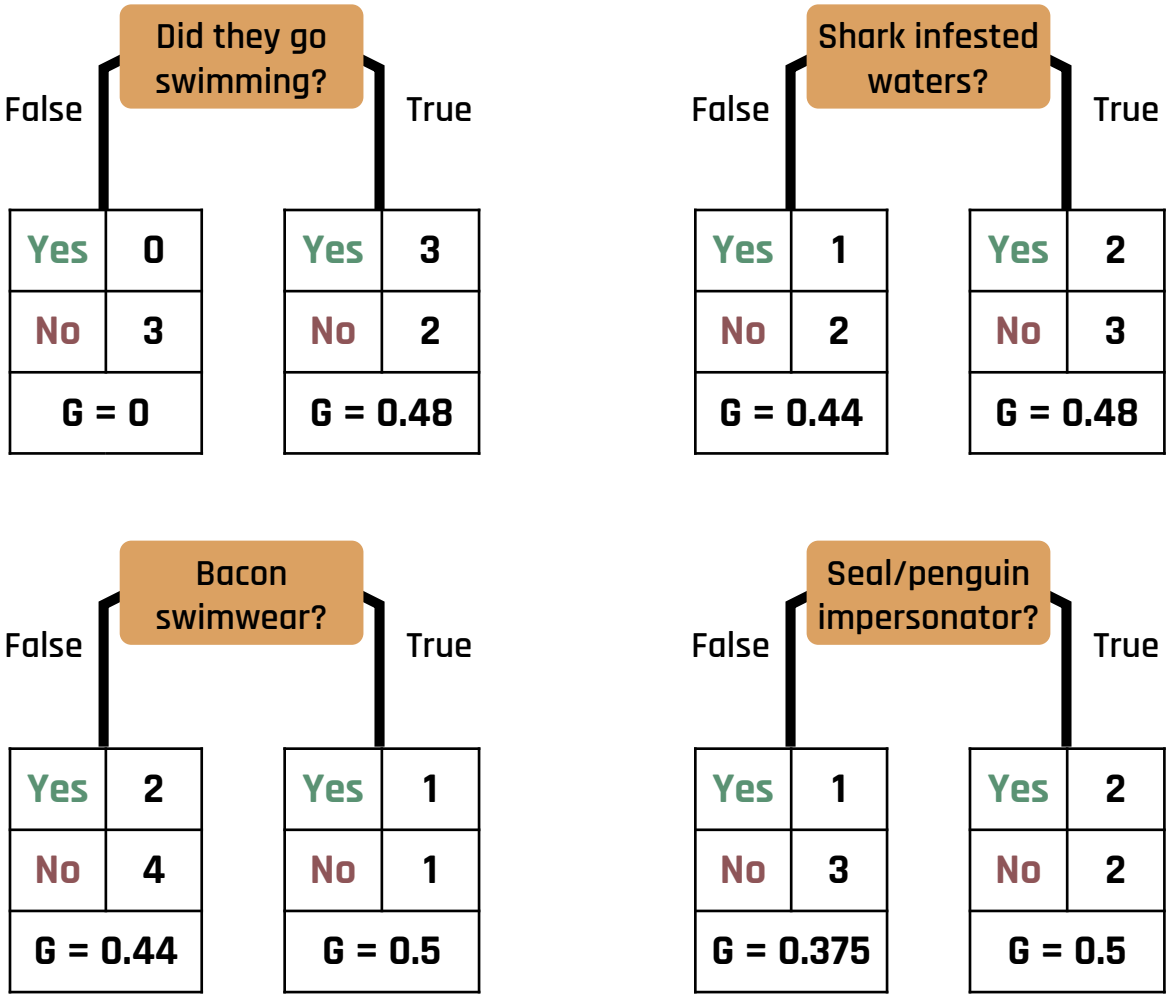
**Seal/penguin impersonator?**

| False | | True | |
|---|---|---|---|
| Yes | 1 | Yes | 2 |
| No | 3 | No | 2 |
| G = 0.375 | | G = 0.5 | |

# Impurity, Entropy, & Information Gain, Oh my!

## (In Other Words: How Do We Know Where To Start?)

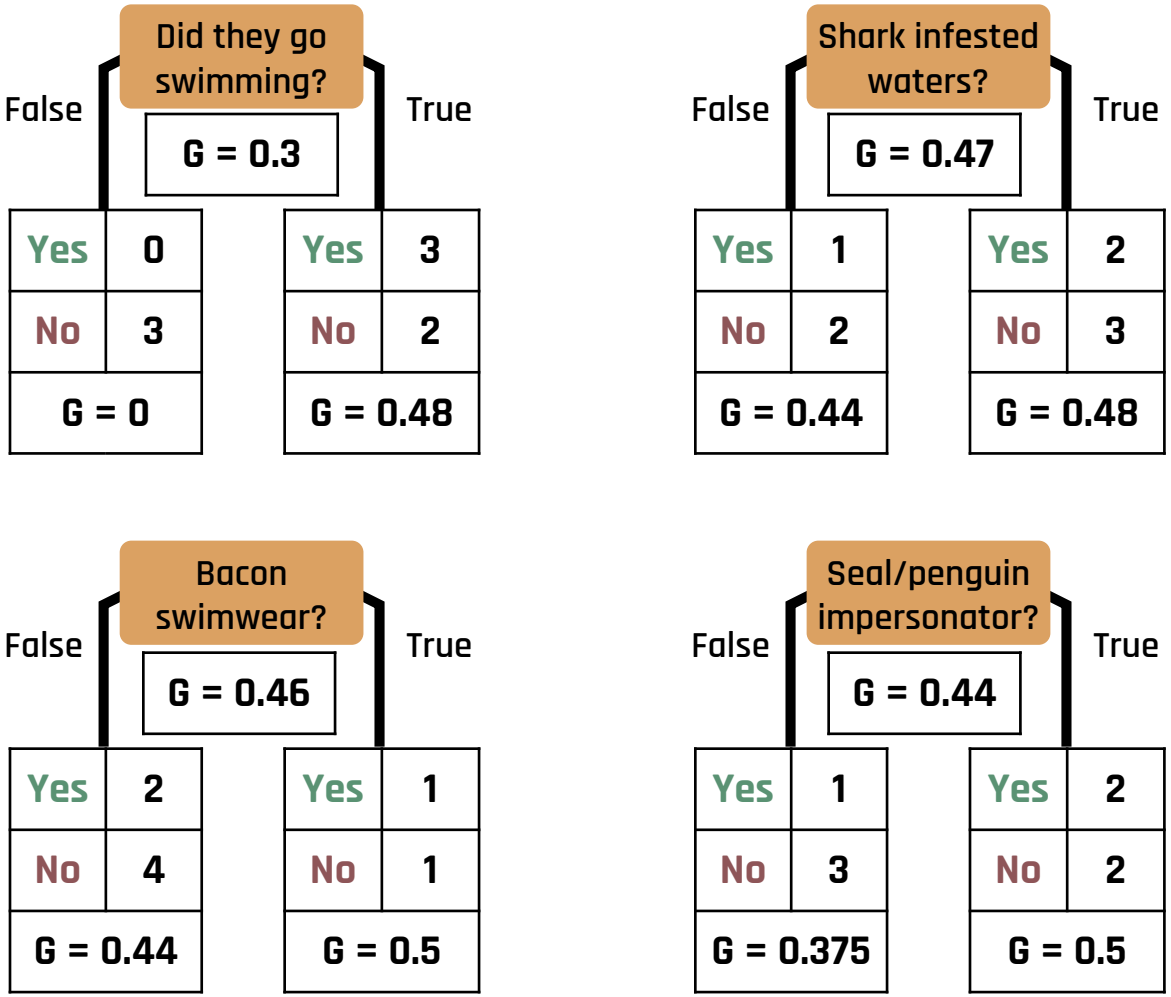
- $$G = \sum_{i=1}^{C} P(i) \cdot (1 - P(i))$$
- $$G = 1 - \sum_{i=1}^{C} P(i)^2$$
- Take weighted averages across branches to get the Gini Impurity for that node
- $$G_\phi = \sum_{b=1}^{\beta} G_b \cdot \phi_b$$



# Impurity, Entropy, & Information Gain, Oh my!

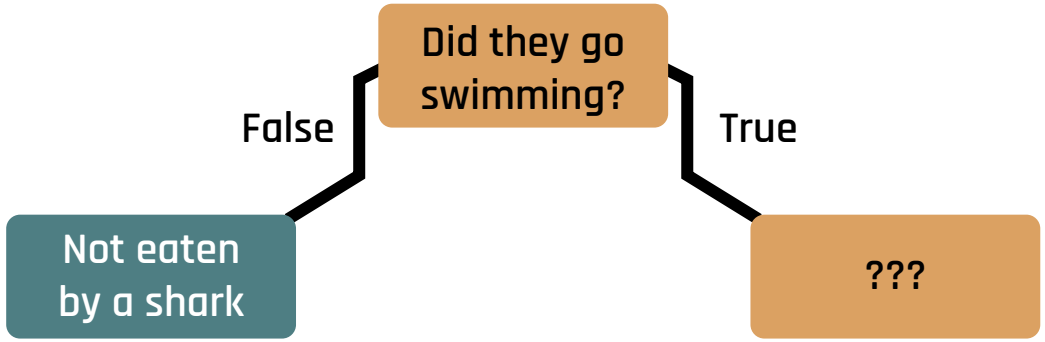
## (In Other Words: How Do We Know Where To Start?)

- $$G = \sum_{i=1}^{C} P(i) \cdot (1 - P(i))$$
- $$G = 1 - \sum_{i=1}^{C} P(i)^2$$
- Take weighted averages across branches to get the Gini Impurity for that node
- $$G_\phi = \sum_{b=1}^{\beta} G_b \cdot \phi_b$$

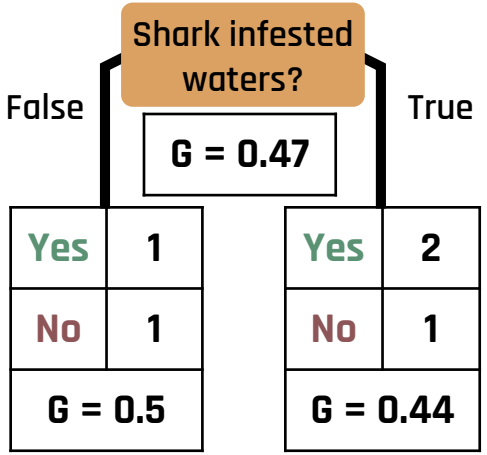


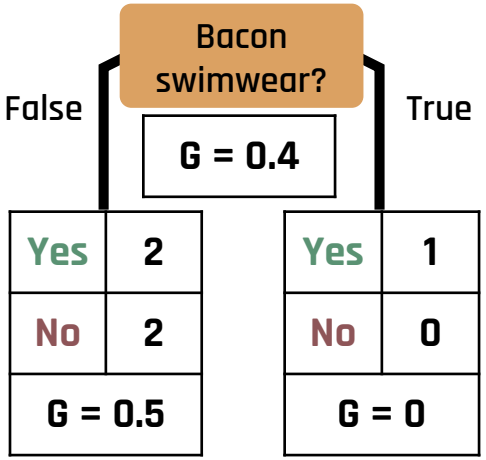
# Impurity, Entropy, & Information Gain, Oh my!

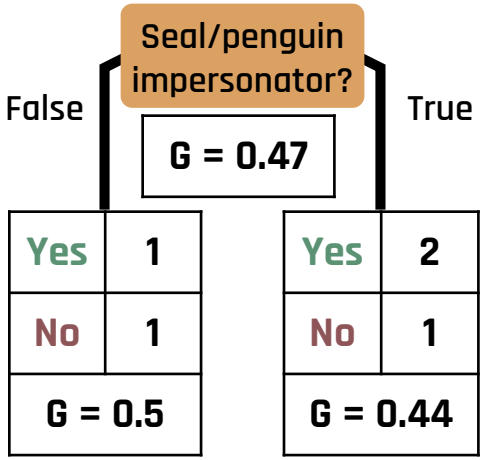
## (In Other Words: How Do We Know Where To Start?)




| Shark Infested Waters | Impersonated a Seal or Penguin | Swimwear Made of Bacon | Attacked by a Shark |
|---|---|---|---|
| No | Yes | No | No |
| Yes | Yes | No | Yes |
| Yes | Yes | No | Yes |
| No | No | Yes | Yes |
| Yes | No | No | No |



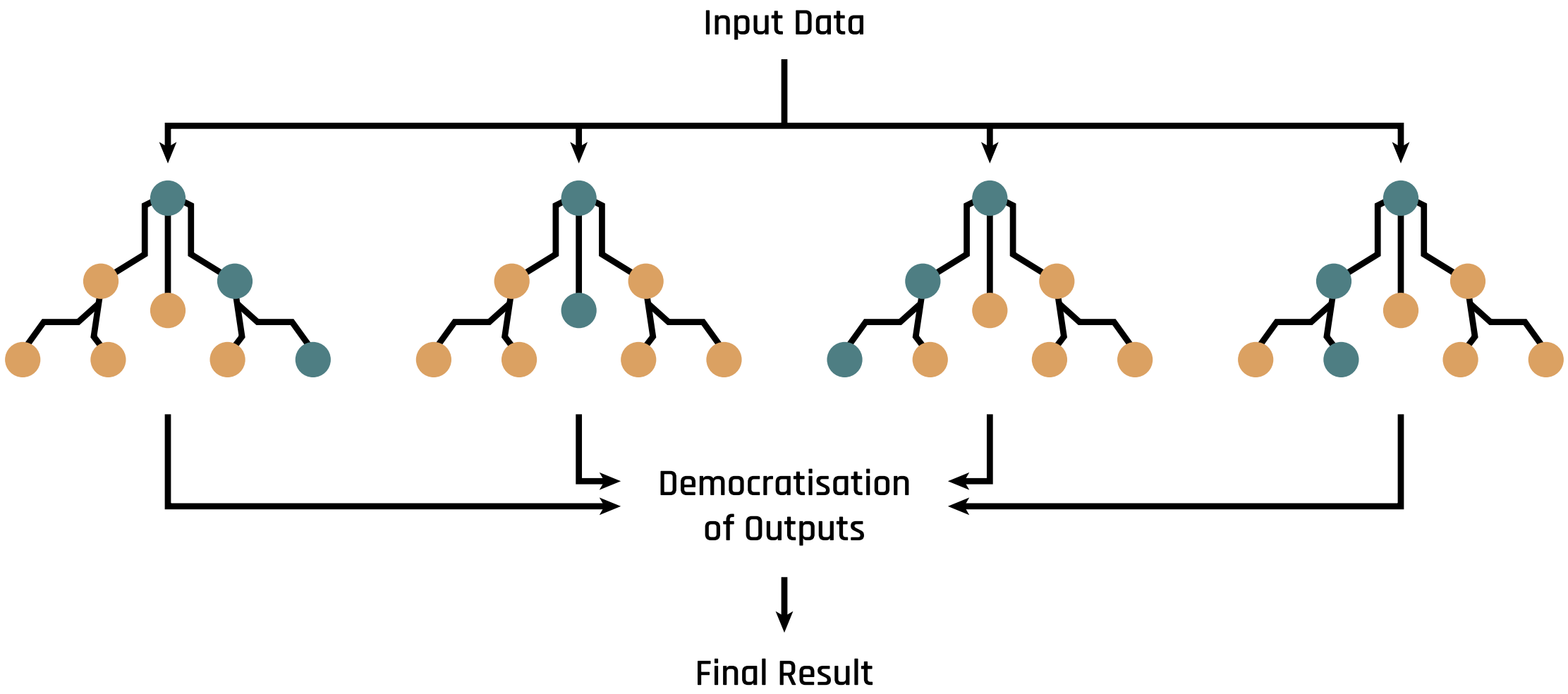



# Explicability

- A single decision tree is a highly interpretable algorithm
- We can just follow the decision statements to determine how the model arrives at its output
- Using our example, then this becomes:
    - If you go swimming and either wear swimwear made out of bacon or impersonate a seal/penguin in shark infested waters, then you will probably be attacked by a shark.
    - (I say probably because I'm assuming my data collection wasn't exhaustive)

# Random Forests

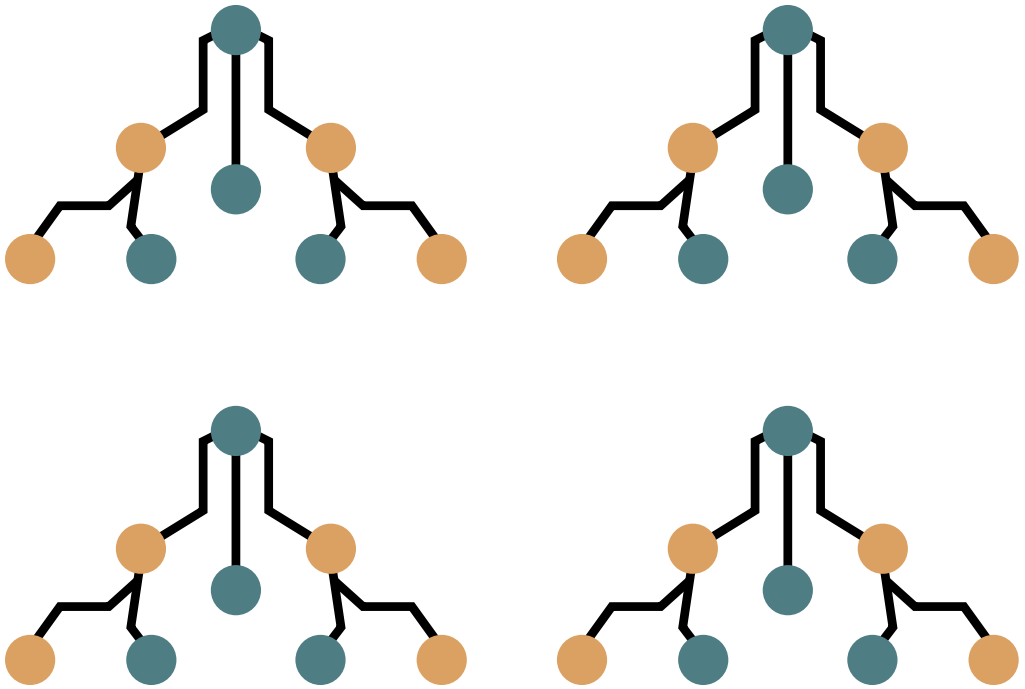


# Random Forests - Tree Uniformity

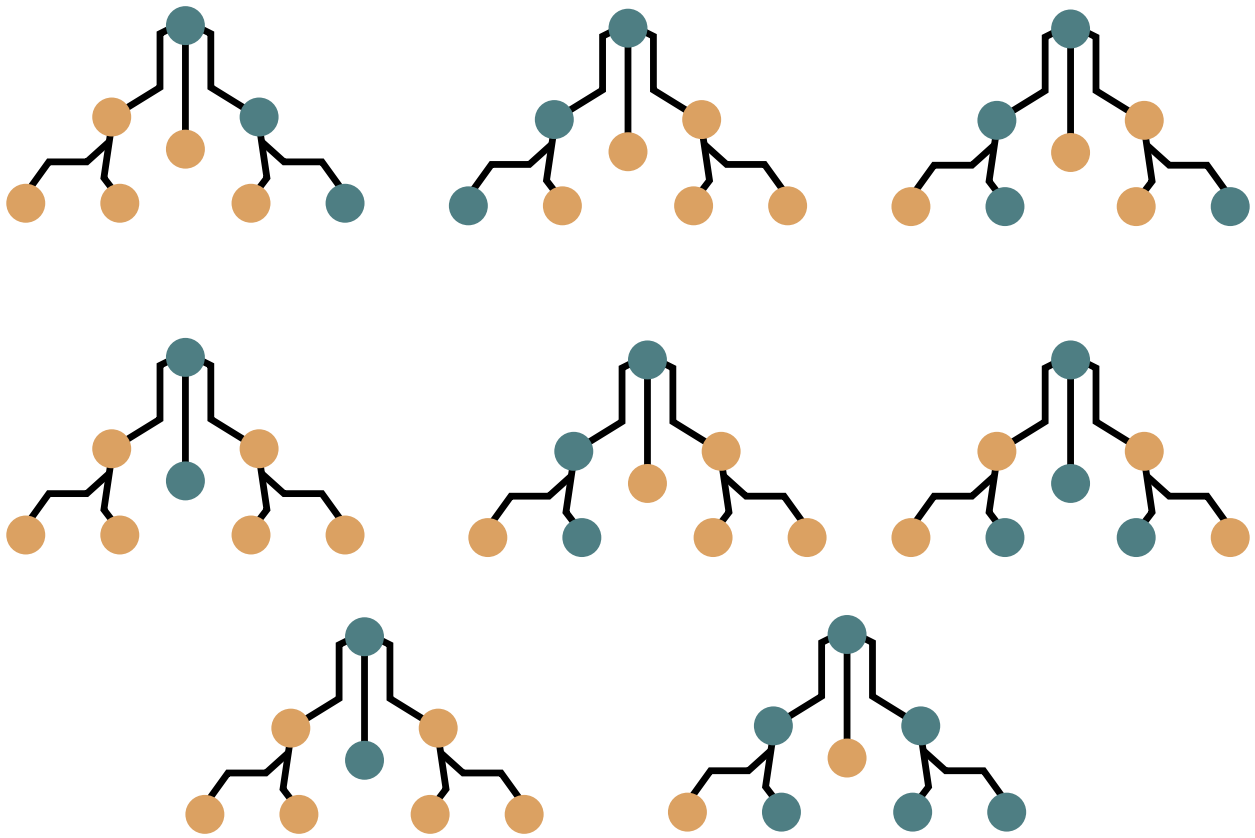
- If we construct a series of decision trees based on the full training data set for each tree, then each tree in the forest will be the same

- If all the trees are the same, then they will have the same output and our aggregated output will be the same as a single decision tree

- Monocultures are bad (see Panama disease and Cavendish bananas)



# Random Forests - Tree Diversity



# Random Forests - B(Ootstrap) Agg(Regat)Ing

# Random Forests - B(Ootstrap) Agg(Regat)Ing
- B() Agg()Ing

**Random Forests - B(Ootstrap) Agg(Regat)Ing**

**- B() Agg()Ing**

**- Bagging!**

# Random Forests - B(Ootstrap) Agg(Regat)Ing
# - B() Agg()Ing
# - Bagging!

- Clearly we need different growing conditions for each of our trees!

- We create these different conditions by sampling from the training dataset with replacement

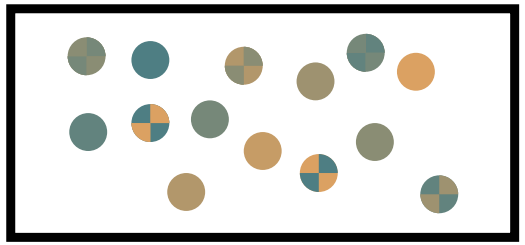
- Then we use each subset to train a different tree

# Random Forests - B(Ootstrap) Agg(Regat)Ing
# - B() Agg()Ing
# - Bagging!

- Clearly we need different growing conditions for each of our trees!

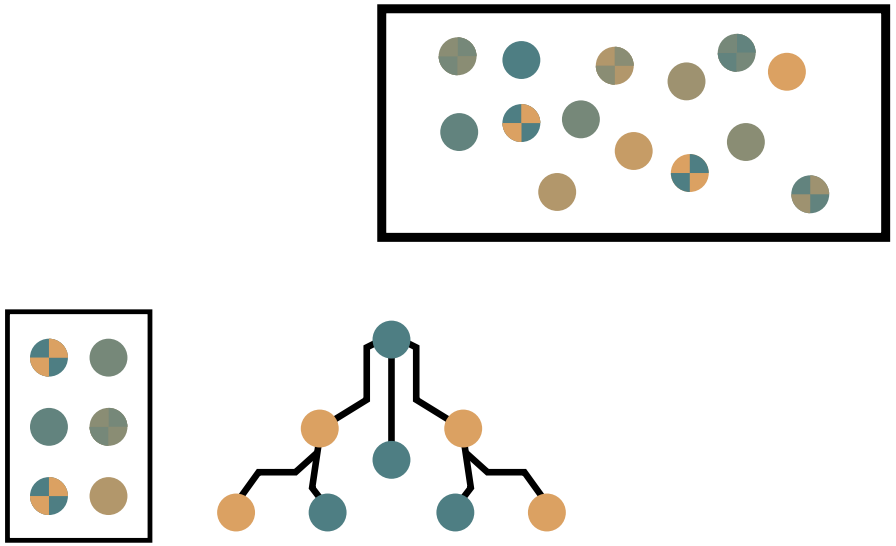
- We create these different conditions by sampling from the training dataset with replacement

- Then we use each subset to train a different tree



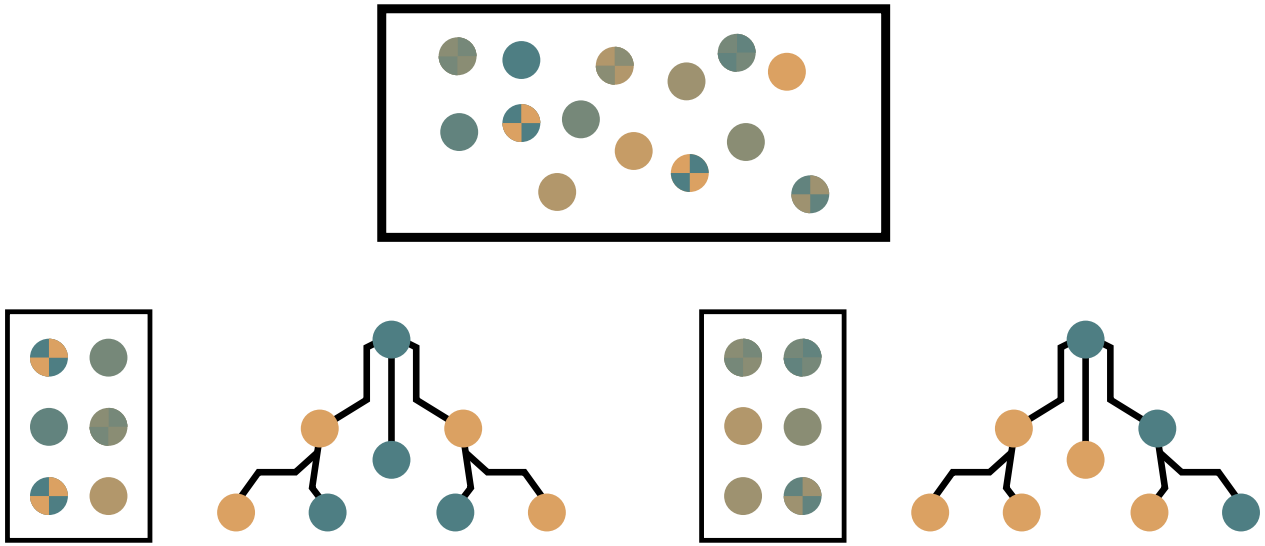
# Random Forests - B(Ootstrap) Agg(Regat)Ing
# - B() Agg()Ing
# - Bagging!

- Clearly we need different growing conditions for each of our trees!
- We create these different conditions by sampling from the training dataset with replacement
- Then we use each subset to train a different tree



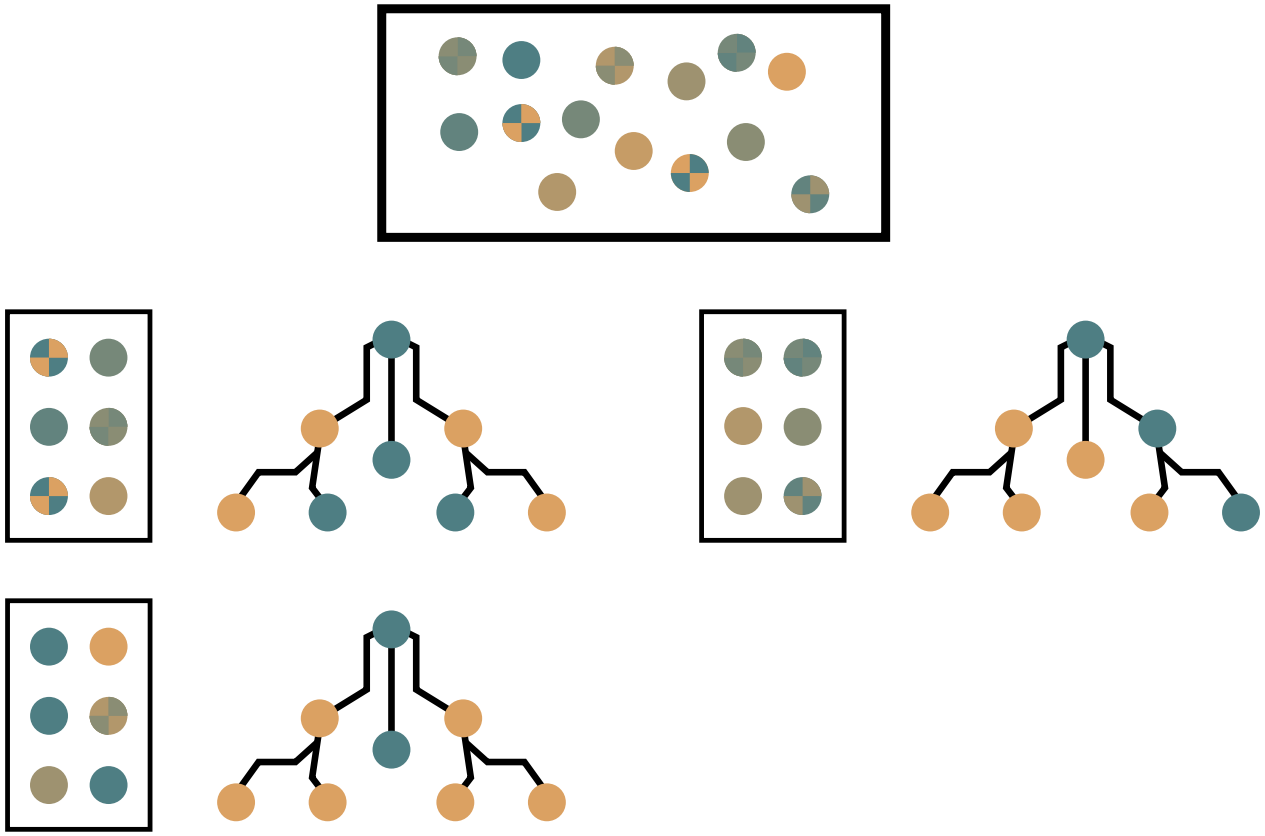
# Random Forests - B(Ootstrap) Agg(Regat)Ing
# - B() Agg()Ing
# - Bagging!

- Clearly we need different growing conditions for each of our trees!
- We create these different conditions by sampling from the training dataset with replacement
- Then we use each subset to train a different tree



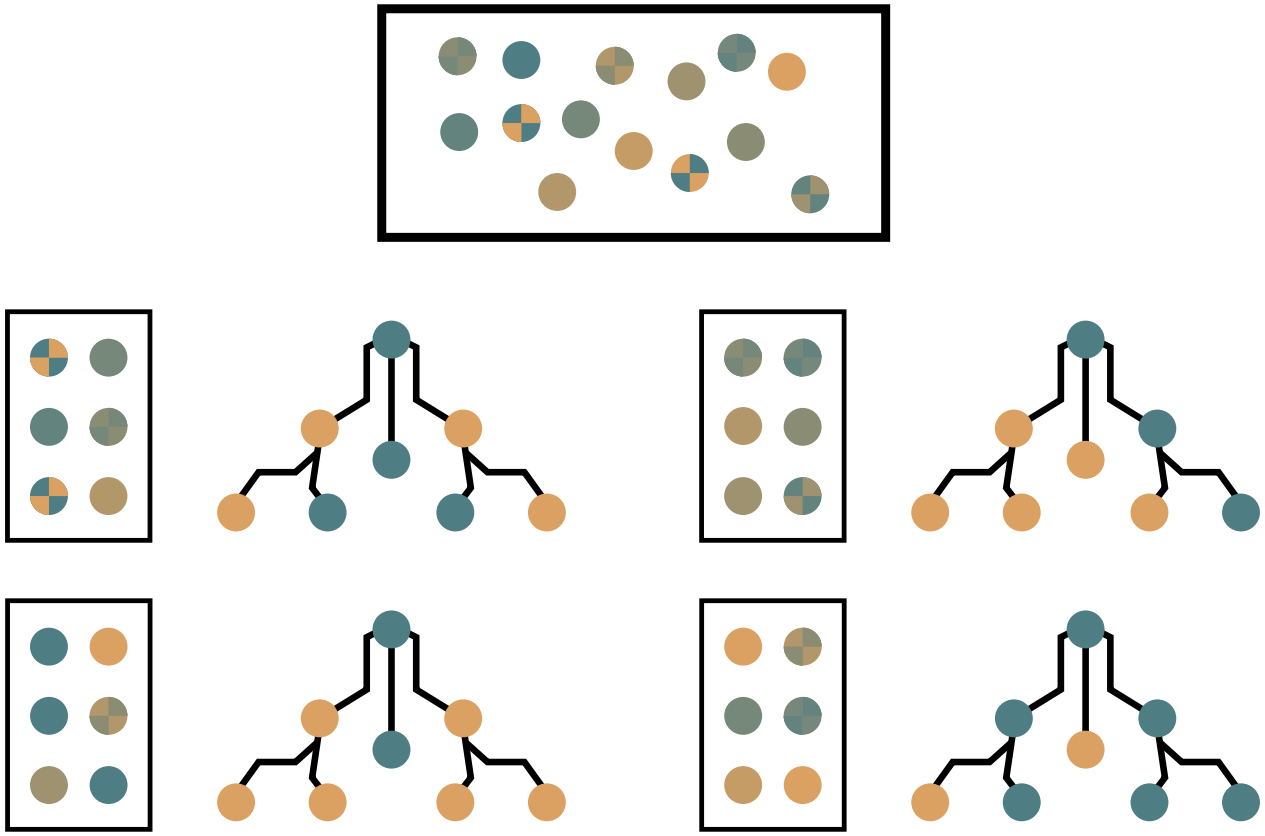
# Random Forests - B(Ootstrap) Agg(Regat)Ing
# - B() Agg()Ing
# - Bagging!

- Clearly we need different growing conditions for each of our trees!
- We create these different conditions by sampling from the training dataset with replacement
- Then we use each subset to train a different tree



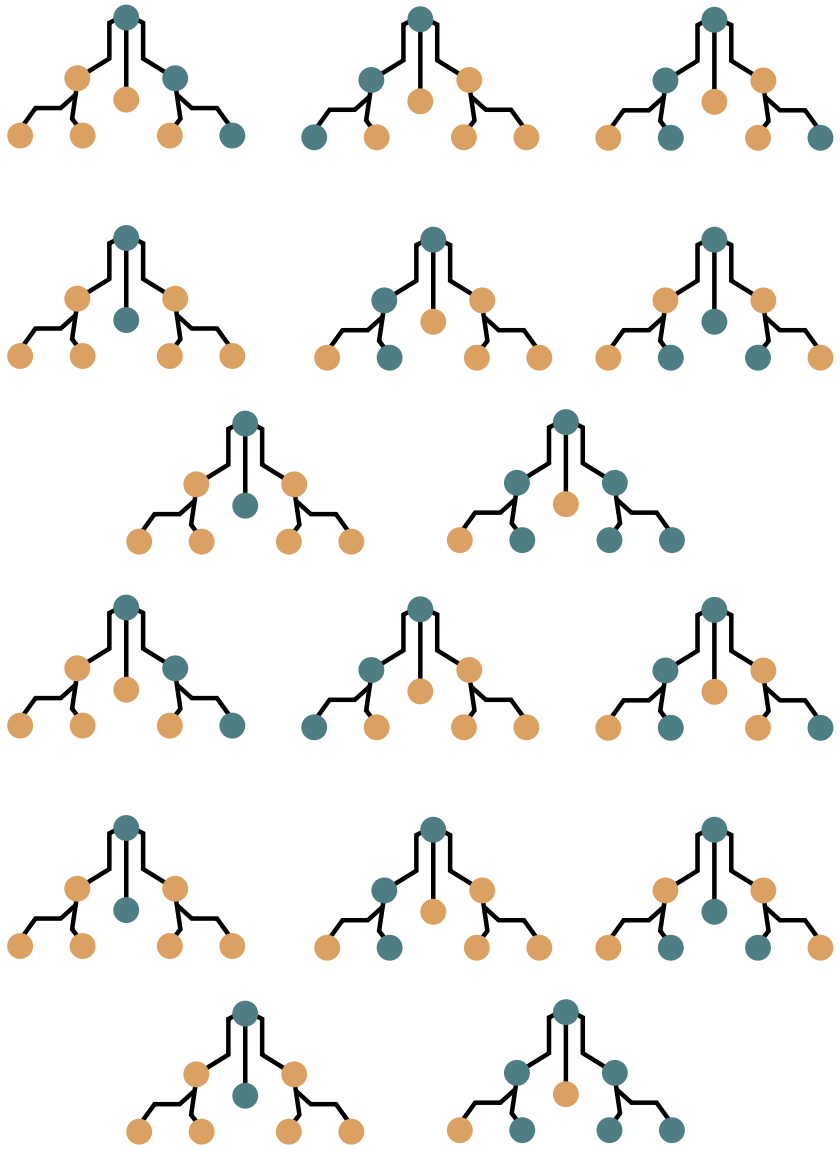
# Random Forests - B(Ootstrap) Agg(Regat)Ing
# - B() Agg()Ing
# - Bagging!

- Clearly we need different growing conditions for each of our trees!
- We create these different conditions by sampling from the training dataset with replacement
- Then we use each subset to train a different tree



# (In)Explicability

- A single decision tree is a highly interpretable algorithm
- But when we have dozens, if not hundreds, if not thousands (probably an exaggeration) of trees each with their own conditional logic, then this interpretability becomes lost…
- Our white box tree has turned into a black box forest
- Turns out we can't see the trees for the forest

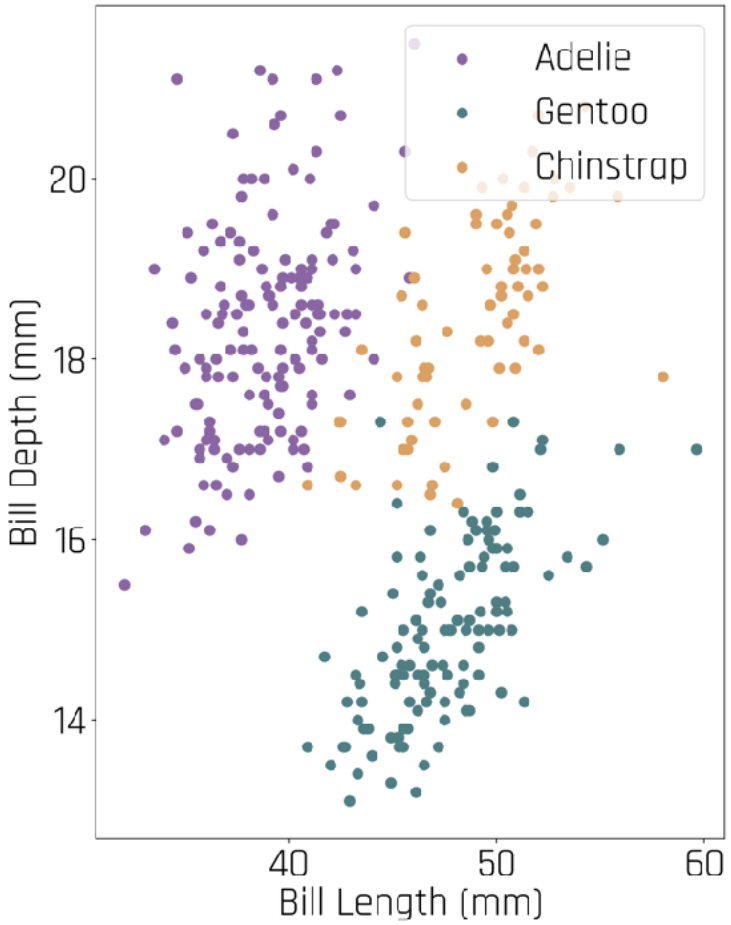


# Classification Exercise

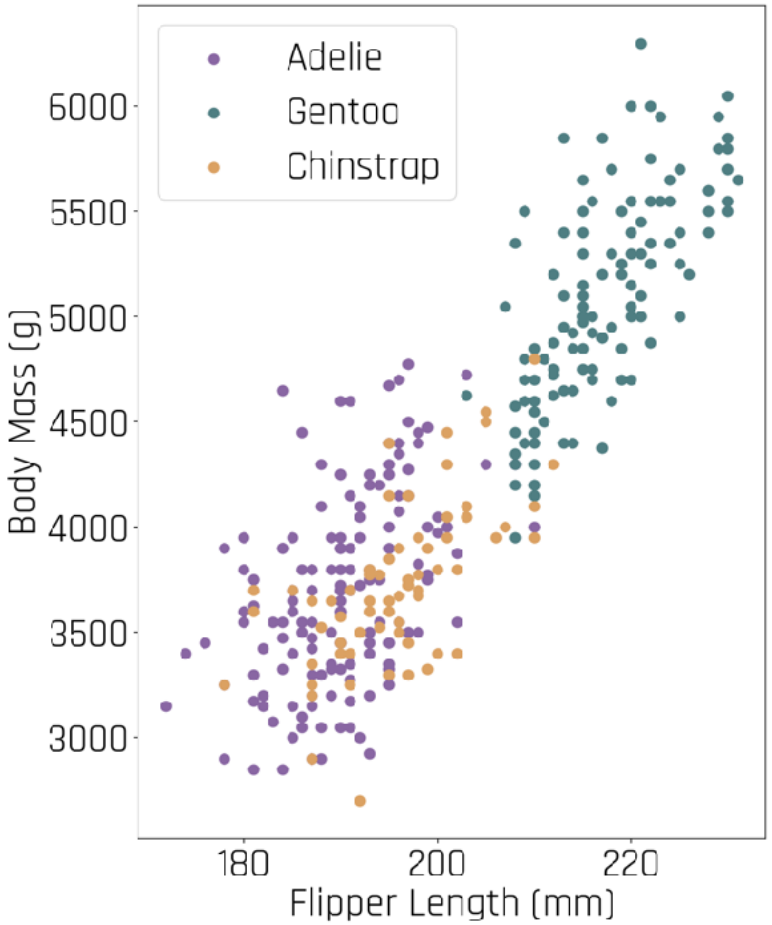
1. Import the Palmer Penguin dataset
2. Take body mass, bill length, bill depth, and flipper length as input features and penguin species as output
3. Create scatter plots for each pairwise combination of the variables with labels of penguin species
4. Fit a Random Forest Classifier to the data using pairs of variables as inputs with 6 trees and a maximum depth of 2
5. Calculate the classification accuracy
6. How well does the Random Forest do and does this match to your expectations with the data?
7. Now fit a Random Forest using all 4 input variables using the same number of trees and maximum depth
8. Vary the number of trees and maximum depth and repeat the training process
9. What does this do to the decision boundary and the accuracy?

# Classification Exercise

**Accuracy: 0.963**

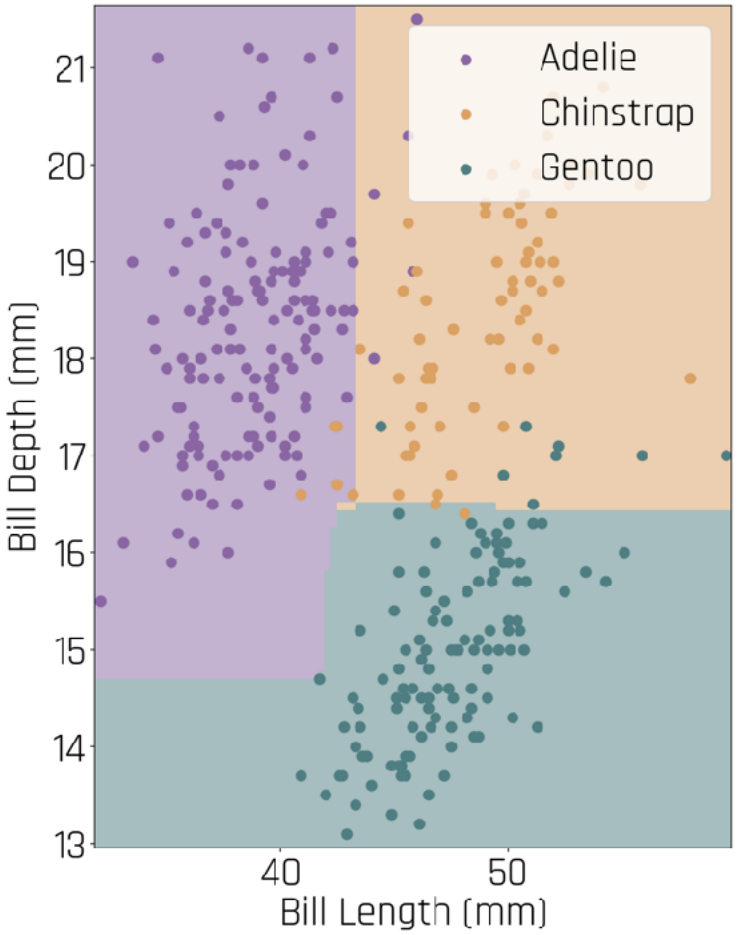



**Accuracy: 0.799**

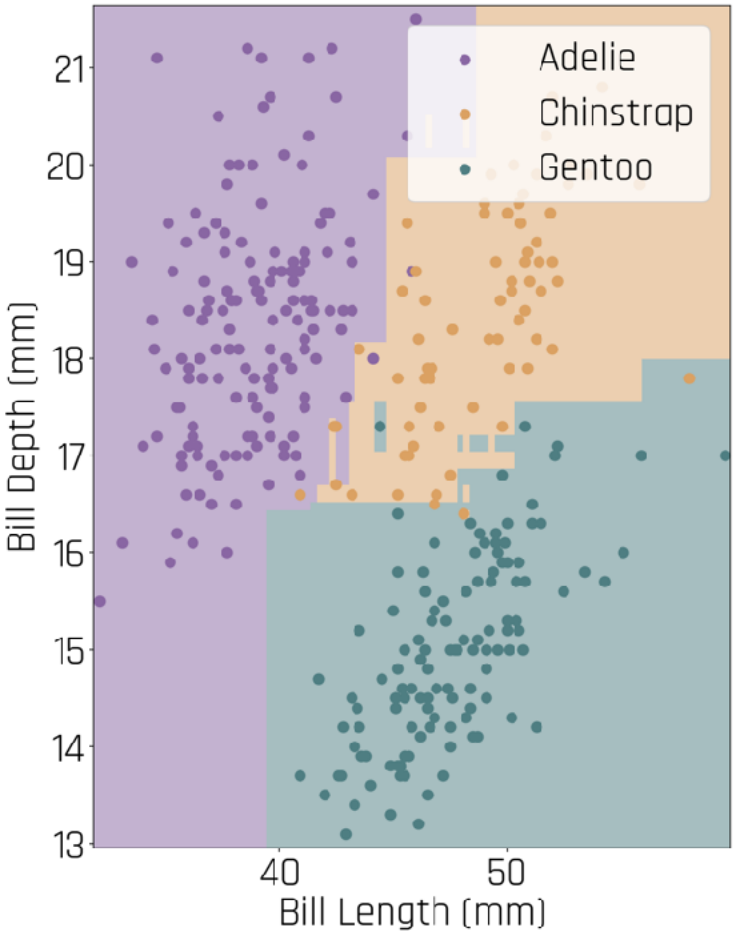


# Classification Exercise

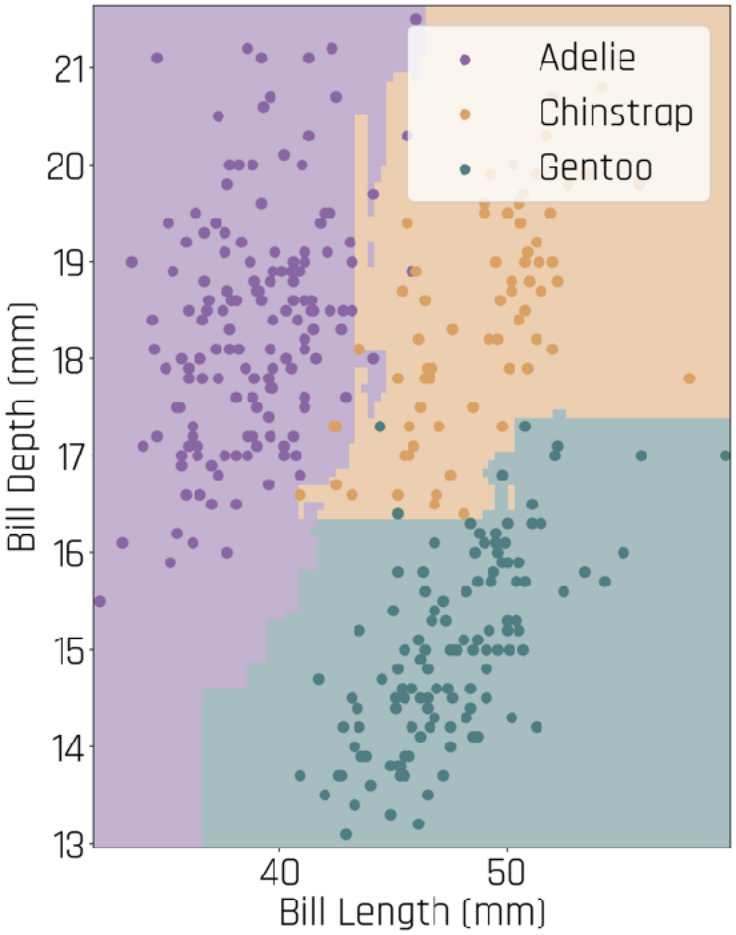
**Accuracy: 0.933**




**Accuracy: 0.955**




**Accuracy: 0.925**



UNIVERSITY OF CAMBRIDGE

# Part II - Practical Applications

# Regression Exercise

1. Create a set of x values with:

$$x \in [-3.2, +3.2]$$

2. Create y observations with:

$$y = 1.28x^3 - 6.4x + 1$$

3. Fit a RandomForestRegressor with 2 trees and a maximum depth of 2

4. Create predictions using your x values and plot them!

5. Calculate the $R^2$

6. Vary the number of trees and the maximum depth and investigate the effect on $R^2$

7. Create a set of new x values with:

$$x \in [-4.8, +4.8]$$

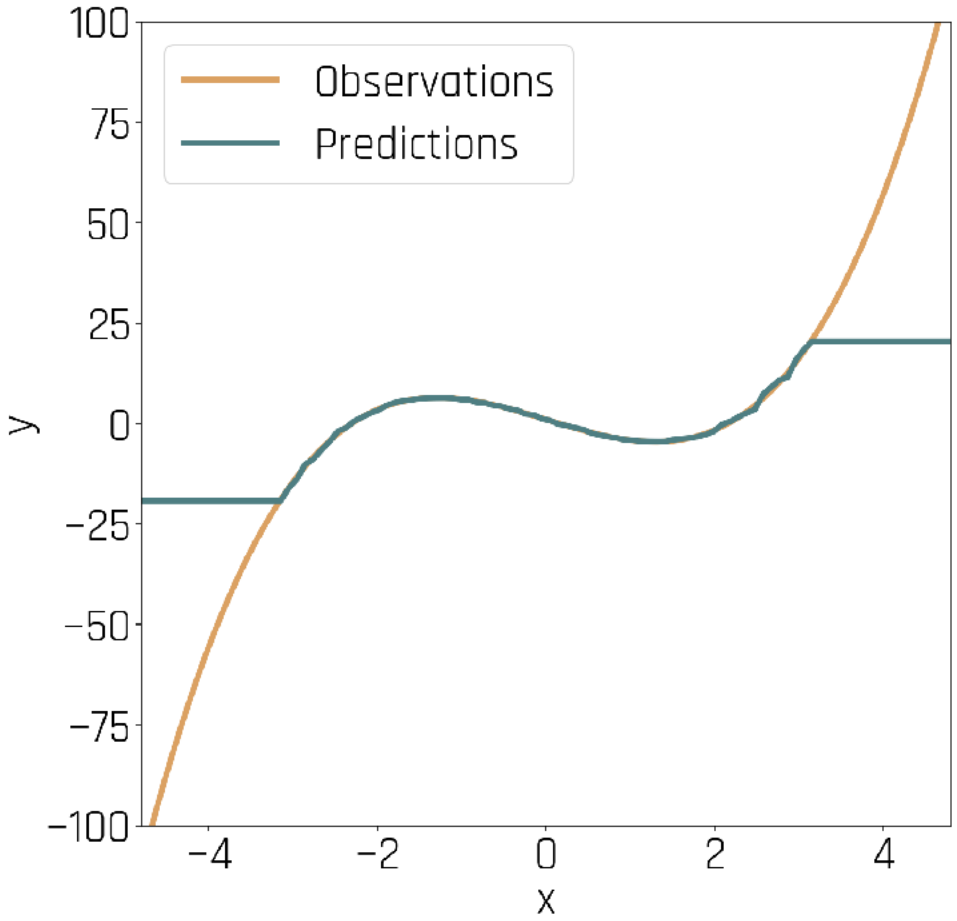
8. Create predictions using your new x values along with a set of new y observations.

9. Plot both your new predictions and your new observations against x

10. What do you notice?

# Interpolation & Extrapolation

- Provided you have a sufficient number of trees and tree depth, random forest regressors interpolate really well
- Basic random forests do not, however, extrapolate at all
- This is due to the their structure - with no information on out-of-sample data, no branching can be learned
- Model will default to the prediction at the limits of the training set for out-of-sample predictions



# Platt Scaling

- Machine learning models can be overly certain of outcomes in a way that does not reflect the underlying probability of events
- Platt scaling is probability, or uncertainty, calibration of a classification model to correct model uncertainty to better match observations
- It appends a logistic regression layer to a classification model that maps the prior model output to the observations
- Originally developed for use with Support Vector Machines, which used distance from the decision boundary as proxy for uncertainty

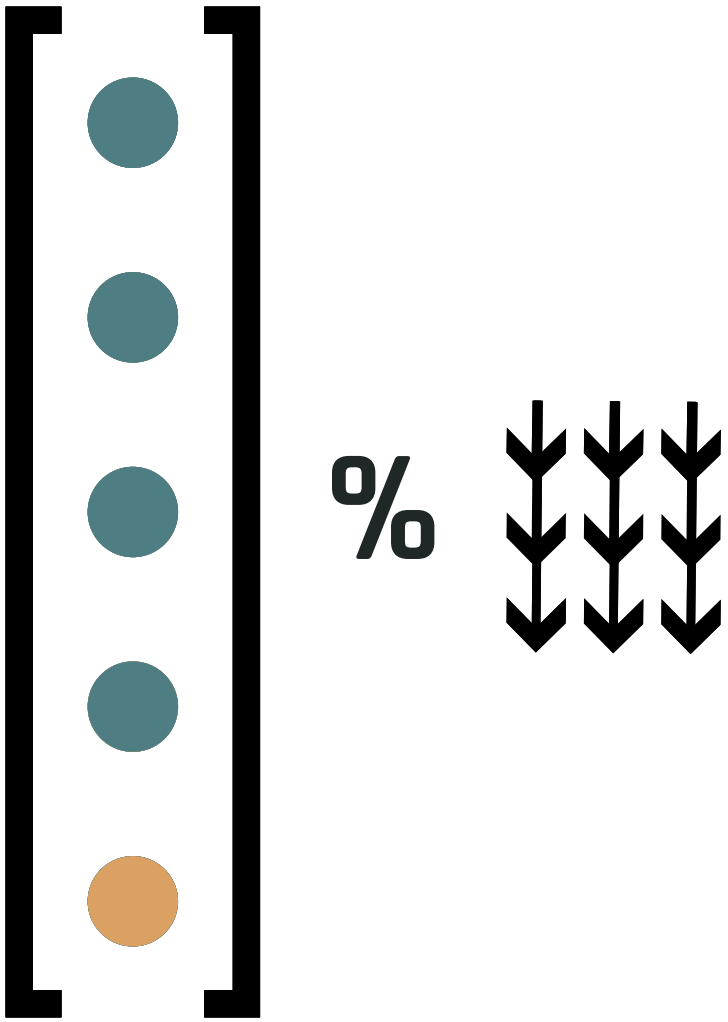
# Platt Scaling

$$P(y = 1 \mid \mathbf{x}) = \frac{1}{1 + e^{(\alpha \cdot y' + \beta)}}$$

- Where P(y=1 | **x**) is the probability of predicting the positive class, given the inputs **x**, y' is the model prediction, and α and β are parameters to be optimised using Maximum Likelihood Estimation

# Interpretability - Feature Importance, Permutation, & Shapley Values

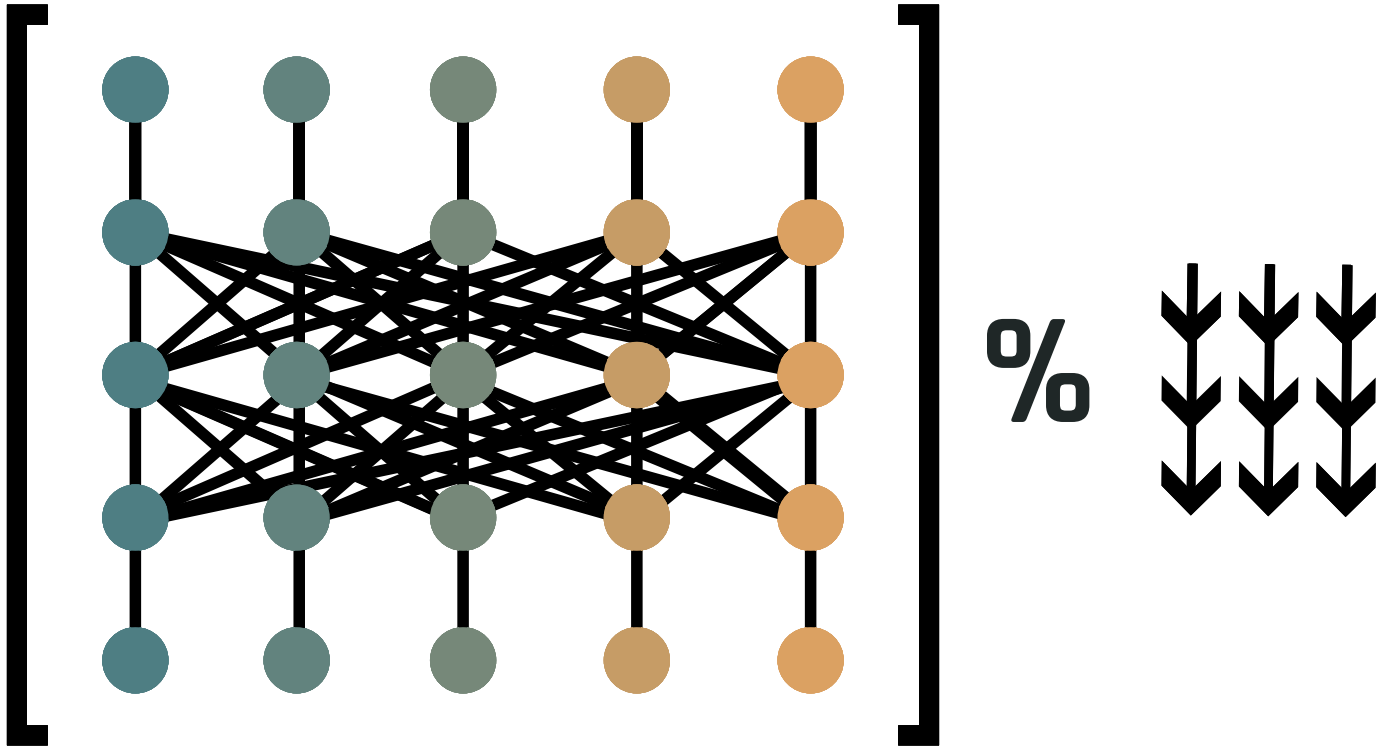
- An easy to implement feature sensitivity study is an ablation study
- This involves removing a variable one at a time and determining the drop in performance without that variable



# Interpretability - Feature Importance, Permutation, & Shapley Values

- Feature permutation takes one variable at a time and shuffles the values corresponding to that variable within the data
- Rinse and repeat

$$i_j = s - \frac{1}{N}\sum_{n=1}^{N} s_{n,j}$$



# Interpretability - Feature Importance, Permutation, & Shapley Values

- A more rigorous way to explore feature importance is by looking at all possible combinations of variables.

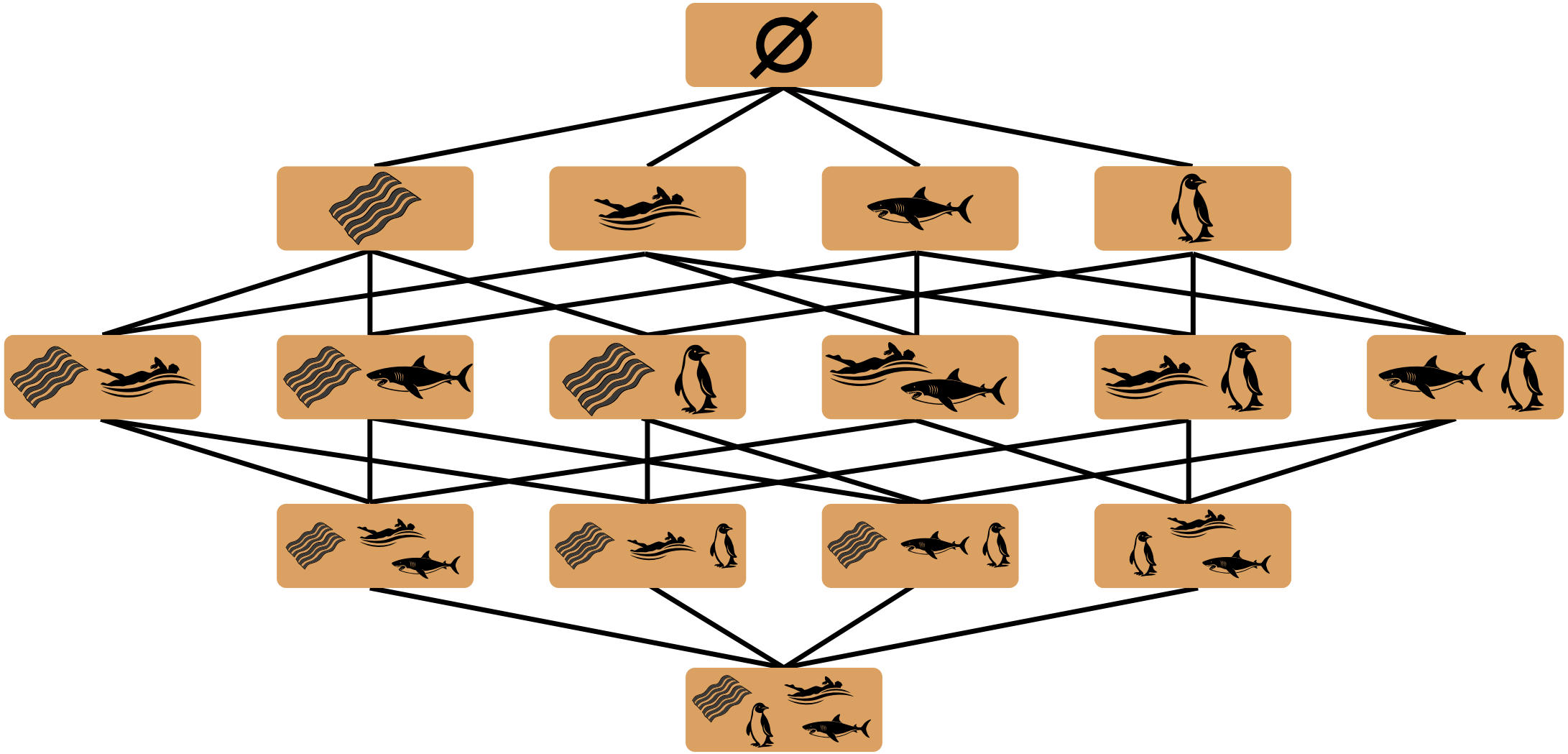
- This would ordinarily require exploring the power set of input variables, for which:

$$|P(S)| = 2^n$$

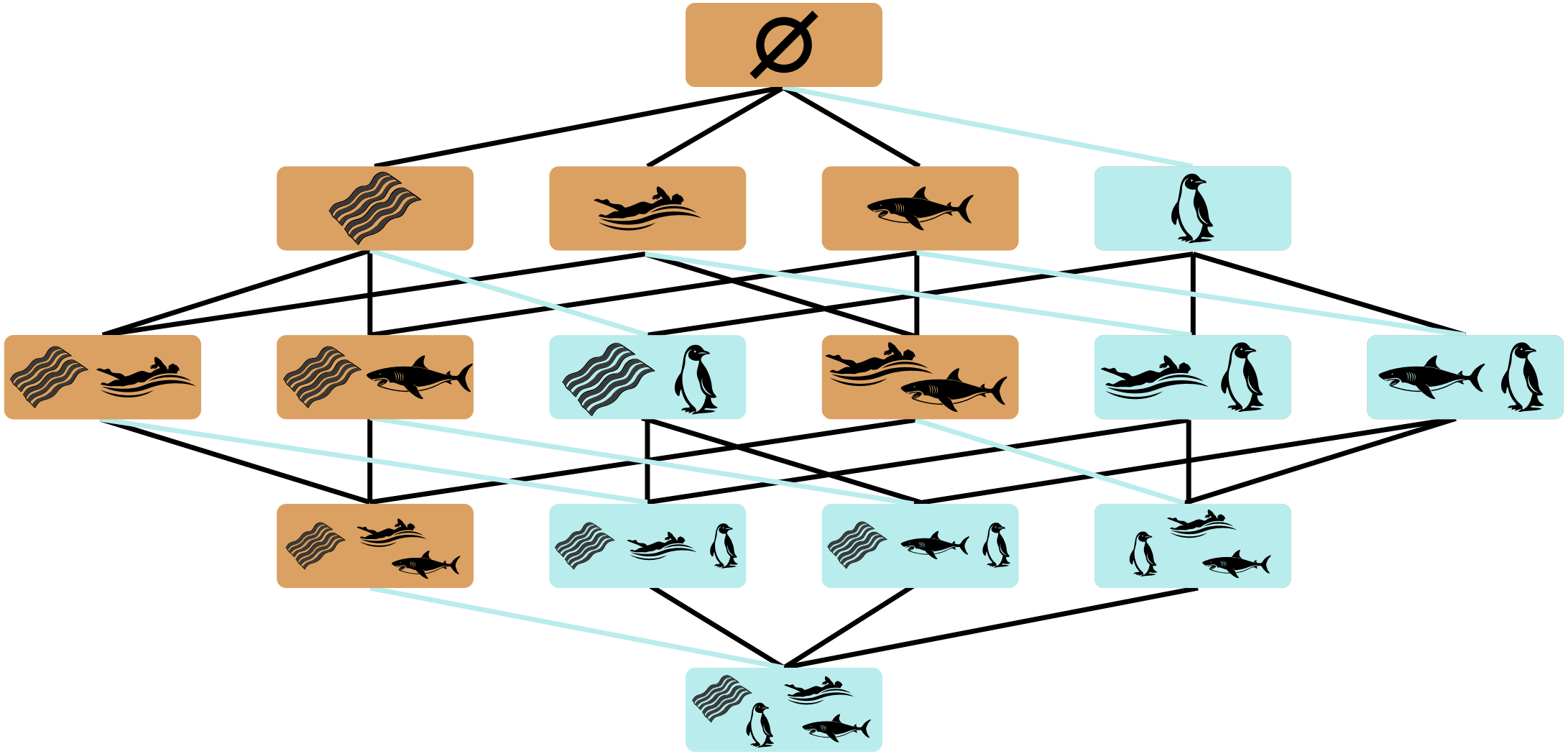
- We can then calculate the marginal contribution made by an individual variable when adding it to a models without that variable

$$MC_{x_i,\{x\}} = \sum_{n=1}^{N} f_{x_i}(\mathbf{x}_n) - f_{\emptyset}(\mathbf{x}_n)$$

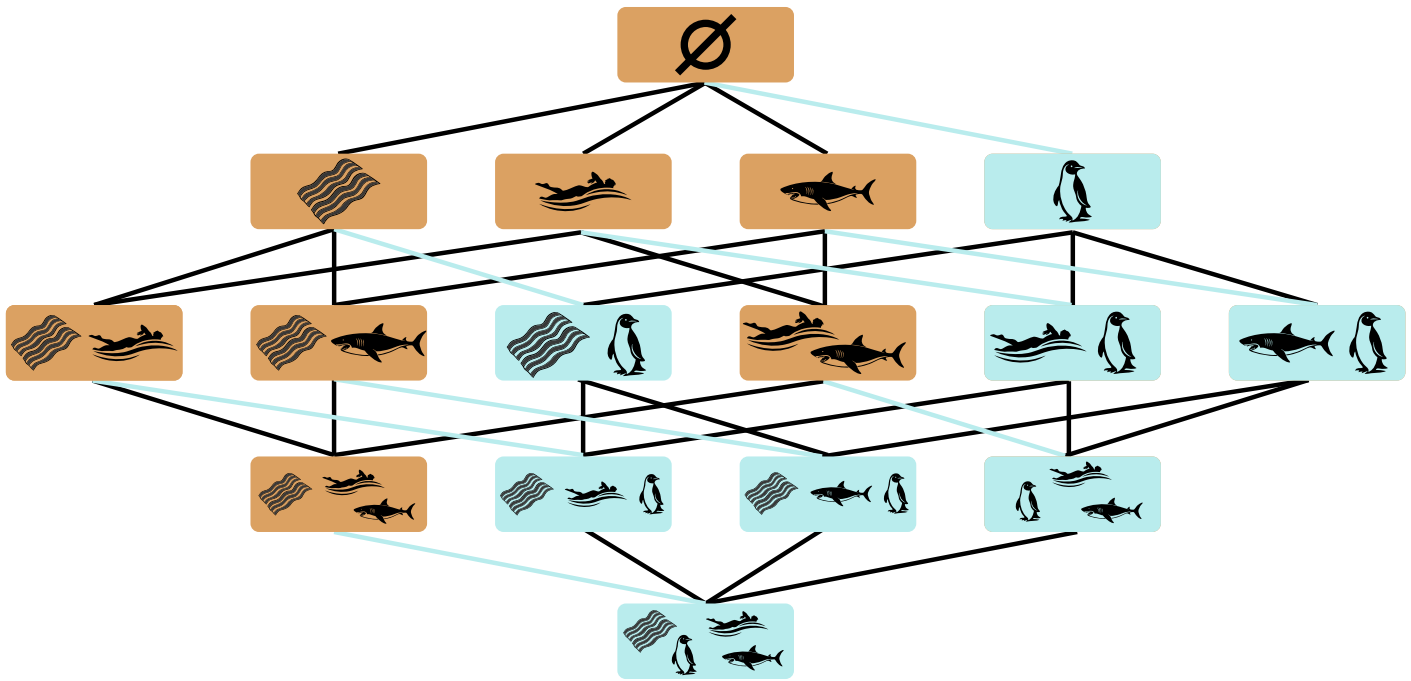
# Interpretability - Feature Importance, Permutation, & Shapley Values



# Interpretability - Feature Importance, Permutation, & Shapley Values



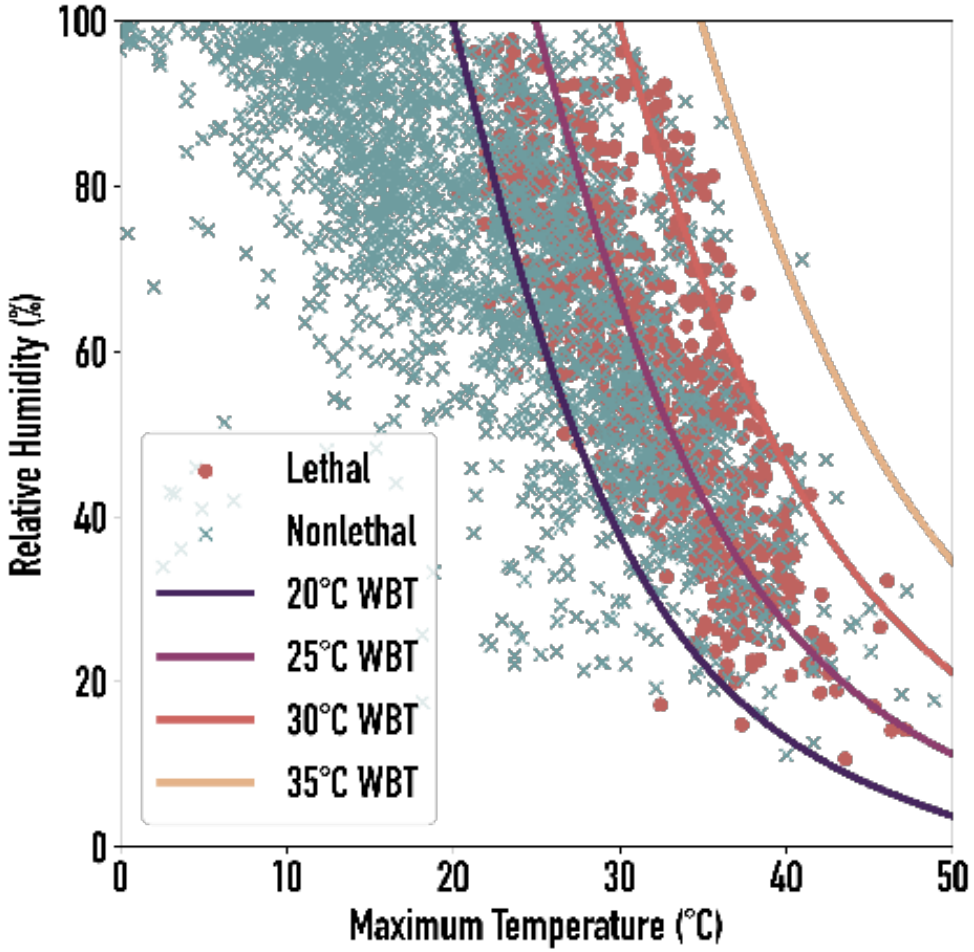
# Interpretability - Feature Importance, Permutation, & Shapley Values



| Edges | Weights |
|---|---|
| 4 | 1/4 |
| 12 | 1/12 |
| 12 | 1/12 |
| 4 | 1/4 |

$$\sum_{n=1}^{N} w_i = 1$$

# Other Thoughts - An Example (Reclassifying Lethal Heat)



# Other Thoughts - An Example (Reclassifying Lethal Heat)

- Forensic pathology/post mortem diagnosis reveal most common cause of death is cardiovascular.
- Increases beyond adapted temperatures result in blood vessel dilation, followed by a drop in blood pressure.
- This leads to a sudden increase in heart rate, creating a system shock for the vulnerable that often proves fatal.

- Needed to encode vulnerability, acclimatisation, and adaptability at the population and regional levels
    - Population age metrics
    - Obesity and other public health markers
    - Rates of temperature change
    - Decadal historical trends in temperature

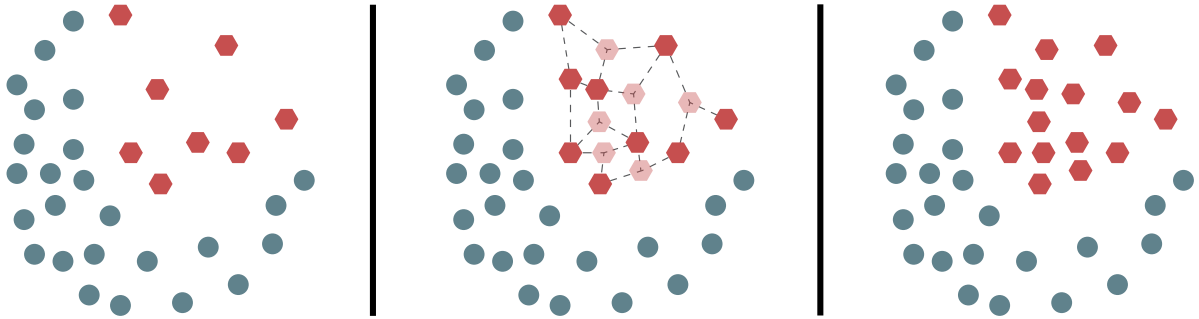
# Other Thoughts - An Example (Reclassifying Lethal Heat)

## SYNTHETIC MINORITY OVERSAMPLING

125,411 events

979 Lethal (0.78%)

124,432 Nonlethal (99.2%)

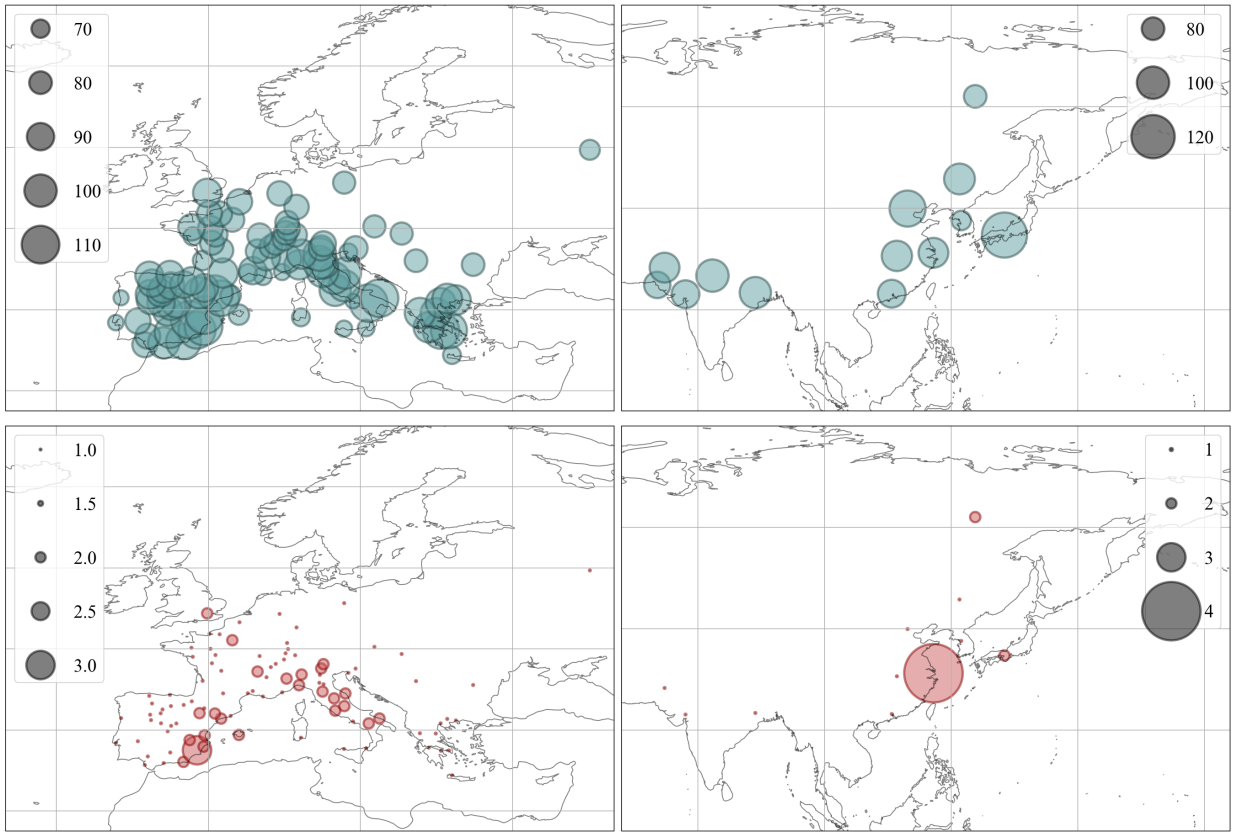


# Other Thoughts - An Example (Reclassifying Lethal Heat)

## GROUND TRUTH

Observed Negatives

Observed Positives

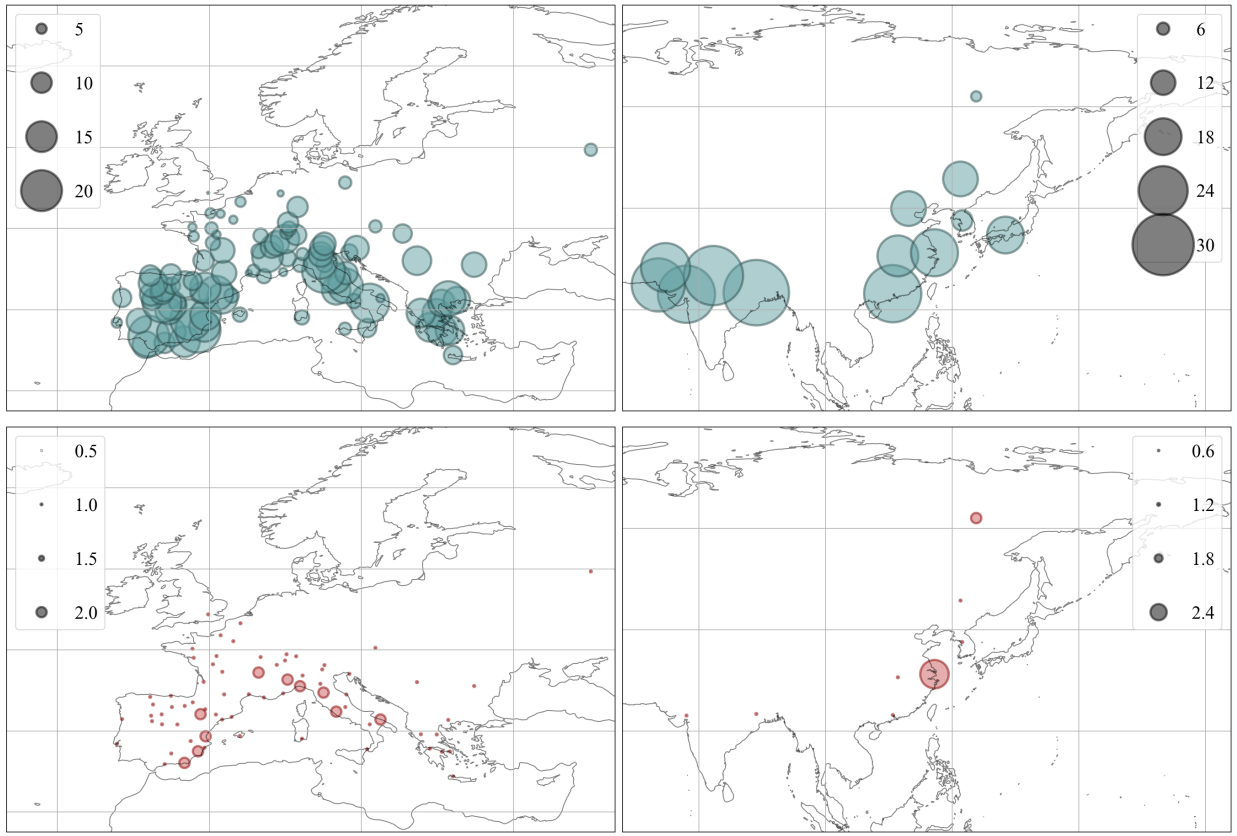


# Other Thoughts - An Example (Reclassifying Lethal Heat)

## WET BULB TEMPERATURE MODEL ERROR

False Positives

False Negatives

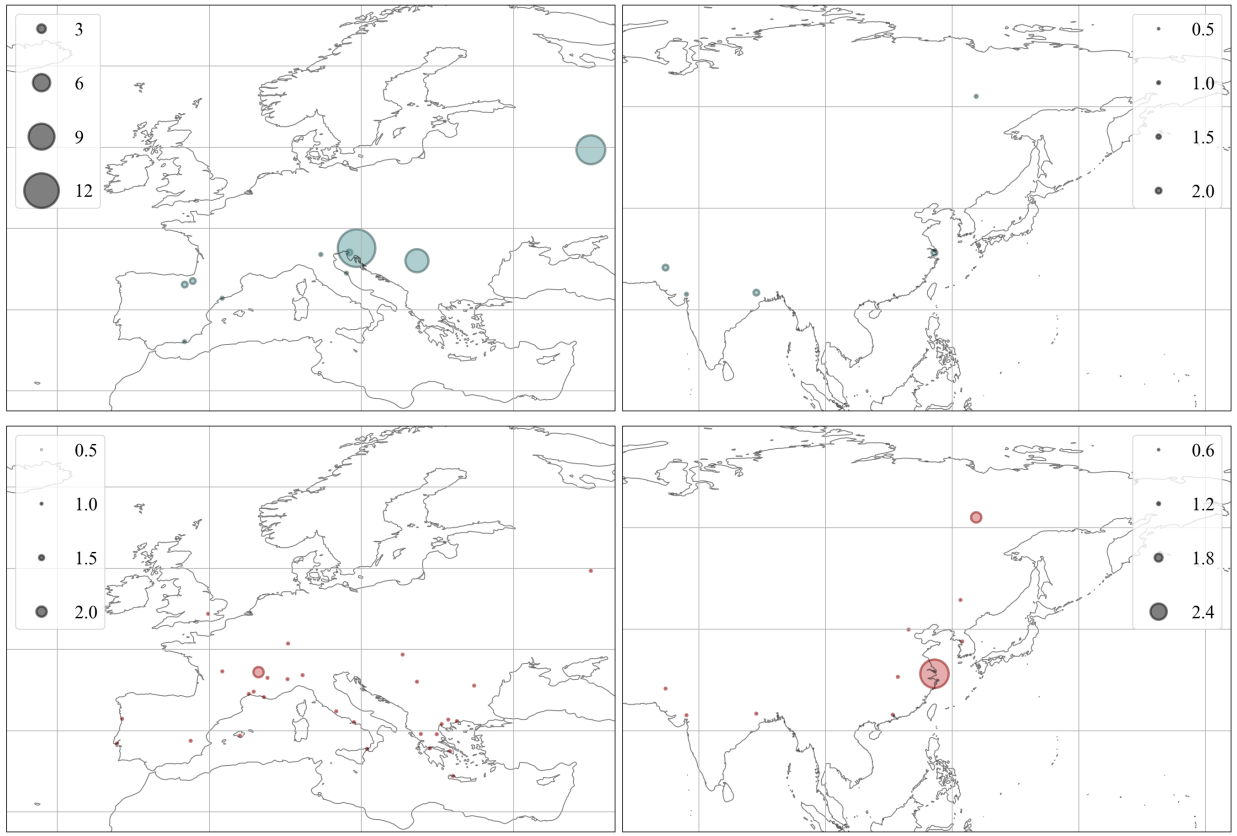


# Other Thoughts - An Example (Reclassifying Lethal Heat)

## VULNERABILITY MODEL ERROR

False Positives

False Negatives



# Other Thoughts - An Example (Reclassifying Lethal Heat)

| Input Variables Used | Input # | Metric | | | |
|---|---|---|---|---|---|
| | | Accuracy | Precision | Recall | F1 |
| Max Temperature | 1 | 0.786 | 0.037 | 0.580 | 0.070 |
| ↪ & Mean BMI | 2 | 0.956 | 0.189 | 0.653 | 0.293 |
| ↪ & Adaptive Humidity | 3 | 0.982 | 0.403 | 0.625 | 0.490 |
| ↪ & Mean SDI | 4 | 0.984 | 0.458 | 0.676 | 0.546 |
| ↪ & $\Delta T_{180}$ | 5 | 0.988 | 0.562 | 0.619 | 0.589 |
| ↪ & Age Gradient | 6 | 0.988 | 0.559 | 0.648 | 0.600 |
| ↪ & $\Delta H_{90}$ | 7 | 0.990 | 0.625 | 0.653 | 0.639 |
| ↪ & $\Delta T_{30}$ | 8 | 0.990 | 0.640 | 0.676 | 0.657 |
| ↪ & Mean Age | 9 | 0.990 | 0.623 | 0.676 | 0.649 |
| ↪ & Mean Humidity | 10 | 0.990 | 0.647 | 0.676 | 0.661 |
| ↪ & $\Delta H_{30}$ | 11 | 0.991 | 0.682 | 0.659 | 0.671 |
| ↪ & $\Delta T_{90}$ | 12 | 0.991 | 0.700 | 0.676 | 0.688 |
| ↪ & Mean Windspeed | 13 | 0.992 | 0.740 | 0.648 | 0.690 |
| ↪ & Adaptive Temperature | 14 | 0.992 | 0.720 | 0.660 | 0.689 |
| ↪ & $\Delta H_{180}$ | 15 | 0.991 | 0.690 | 0.660 | 0.674 |
| Max Temperature & Mean Humidity Only | 2 | 0.867 | 0.033 | 0.307 | 0.060 |
| Without Max Temperature & Mean Humidity | 13 | 0.990 | 0.644 | 0.647 | 0.646 |

# Other Thoughts

- Random Forests are an easy to deploy (low code with scikit learn), flexible ensemble machine learning model
- They can be tuned to prevent overfitting, by controlling the number of trees and maximum depth of those trees
- The are less interpretable than decision trees
- But we can use a range of techniques to explore their decision making
- Remember, explicability =/= causality
- Ordinary random forests do not extrapolate; use with caution
- They do very well with imbalanced data problems
- Moral of the story: **don't wear swimwear made out of bacon**
- No Penguins were eaten by sharks or otherwise harmed in the making of this presentation

# References


[1] Breiman, L., 2001. Random Forests. Machine Learning 45, 5–32. https://doi.org/10.1023/A:1010933404324

[2] Horst, A.M., Hill, A.P., Gorman, K.B., 2020. allisonhorst/palmerpenguins: v0.1.0. https://doi.org/10.5281/ZENODO.3960218

[3] Platt, J., others, 1999. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. Advances in large margin classifiers 10, 61–74.

[4] Mora, C., Dousset, B., Caldwell, I.R., Powell, F.E., Geronimo, R.C., Bielecki, C.R., Counsell, C.W.W., Dietrich, B.S., Johnston, E.T., Louis, L.V., Lucas, M.P., McKenzie, M.M., Shea, A.G., Tseng, H., Giambelluca, T.W., Leon, L.R., Hawkins, E., Trauernicht, C., 2017. Global risk of deadly heat. Nature Clim Change 7, 501–506. https://doi.org/10.1038/nclimate3322

[5] Rouse, R., Debnath, R., Rouse, D., Tebbutt, W., Dousset, B., Hosking, J.S., McRobie, A., Mora, C., Shuckburgh, E., 2024. Reclassifying Lethal Heat. https://doi.org/10.21203/rs.3.rs-4736541/v1

[6] Khoshgoftaar, T.M., Golawala, M., Hulse, J.V., 2007. An Empirical Study of Learning from Imbalanced Data Using Random Forest, in: 19th IEEE International Conference on Tools with Artificial Intelligence(ICTAI 2007). pp. 310–317. https://doi.org/10.1109/ICTAI.2007.46


Schmidt Sciences

UNIVERSITY OF CAMBRIDGE

UNIVERSITY OF CAMBRIDGE

# Random Forests & Other Random Thoughts

An overview of the fundamentals along with how and when to use these ensemble methods

ICCS Summer School - 10/07/2025 - Robert Edwin Rouse