Boring but important disclaimers:

- If you are not getting this from the GitHub repository or the associated Canvas page (e.g. CourseHero, Chegg etc.), you are probably getting the substandard version of these slides Don't pay money for those, because you can get the most updated version for free at

  `https://github.com/julianmak/OCES4303_ML_ocean`

  The repository principally contains the compiled products rather than the source for size reasons.
- Associated Python code (as Jupyter notebooks mostly) will be held on the same repository. The source data however might be big, so I am going to be naughty and possibly just refer you to where you might get the data if that is the case (e.g. JRA-55 data). I know I should make properly reproducible binders etc., but I didn't...
- I do not claim the compiled products and/or code are completely mistake free (e.g. I know I don't write Pythonic code). Use the material however you like, but use it at your own risk.
- As said on the repository, I have tried to honestly use content that is self made, open source or explicitly open for fair use, and citations should be there. If however you are the copyright holder and you want the material taken down, please flag up the issue accordingly and I will happily try and swap out the relevant material.

OCES 4303 :

an introduction to data-driven and ML methods in ocean sciences

Session 5: decision trees

Outline

- anatomy of a decision tree
- some concepts of probability
  $\rightarrow$ information entropy and information gain
- usage of decision trees as classifiers or regressors
- demonstration: penguins data

Figure: An example of a tree (a broc-collie).

Oceanic application

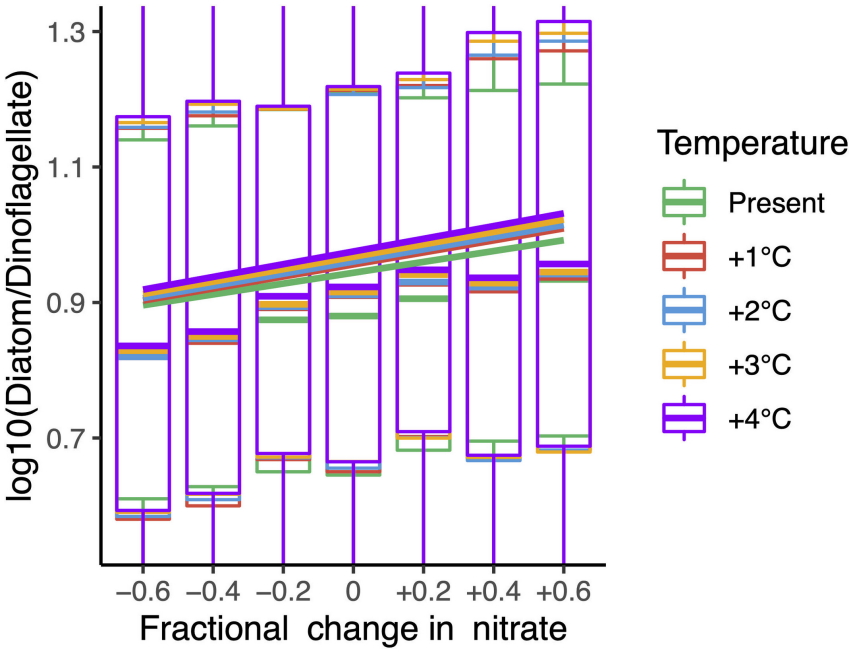

Figure: Predicted log of diatom-dinoflagellate ratio in HK coastal waters using random forests (which is a collection of decision trees) under projected warming scenarios. From Fig. 4 of Cheung *et al.* (2021).

- marine pollution

  → used station data to train up a regression model

  → focused on diatom-dinoflagellate ratio

  → identified key features of importance

  → can do projection (!!!) and find pollution increases under warming

(see also Lee *et al.* (2025) from OCES)

Anatomy

- **root**: head node
- **leaves**: terminal nodes
- **nodes**: the boxes
- **branches**: connectors of nodes with decision

  $\rightarrow$ strictly speaking the 'yes' and 'no' should be branches
- **levels/depth**: how far you are from root

  $\rightarrow$ here it could be tree of depth 1 or 2
- **parent/child**: to talk about nodes, parent is one level closer to root

Figure: Example of a decision tree.

Recap: probability

- main goal: how do we decide on the criterion to split data (i.e. branching)?

  $\rightarrow$ want the data to tell us, rather than us doing it manually

Recap: probability

- main goal: how do we decide on the criterion to split data (i.e. branching)?

  $\rightarrow$ want the data to tell us, rather than us doing it manually

- need some concepts in probability, do this through examples

  $\rightarrow$ e.g. for a **fair** 6-side dice, the possible events are

  $$X = \{1, 2, 3, 4, 5, 6\},$$

  and assigned to these events are probabilities $p_i \in [0, 1]$

  $\rightarrow$ all probabilities should sum to 1, i.e. $\sum_i p_i = 1$

  $\rightarrow$ **fair** $\Rightarrow$ uniform distribution $\Rightarrow p_i = 1/N = 1/6$

Recap: probability

- there is a measure called the information entropy defined as

$$H = \sum_i H_i = -\sum_i p_i \log_a p_i$$

$\rightarrow$ called Shannon entropy/index (from ecology)

$\rightarrow$ base a is usually 2, e or 10; the value itself doesn't actually matter too much (I will take $a = \mathrm{e}$ so $\log = \ln$)

Recap: probability

- there is a measure called the information entropy defined as

$$H = \sum_i H_i = -\sum_i p_i \log_a p_i$$

  $\rightarrow$ called Shannon entropy/index (from ecology)

  $\rightarrow$ base a is usually 2, e or 10; the value itself doesn't actually matter too much (I will take $a = \mathrm{e}$ so $\log = \ln$)

- measure of species diversity (in ecology); think of it as measure of **surprise**, e.g.

  $\rightarrow$ if $X = \{1, 1, 1, 1, 1, 1\}$, then there is no 'surprise' on what you should draw ($H = 0$)

  $\rightarrow$ if $X = \{1, 2, 3, 4, 5, 6\}$, events are maximally different and maximally 'surprising', ($H = \log N = \log 6$ here)

Recap: probability

- from that, the information gain is defined as

$$\mathrm{IG} = H_p - \sum_{i=0}^{N} p_{c,i} H_{c,i}$$

$\rightarrow$ think entropy of parent class minus the weighted averages of entropy of child class

Recap: probability

- from that, the information gain is defined as

$$\mathrm{IG} = H_p - \sum_{i=0}^{N} p_{c,i} H_{c,i}$$

  $\rightarrow$ think entropy of parent class minus the weighted averages of entropy of child class

- e.g., if I have cats and dogs of various colours, hair length...

  $\rightarrow$ my parent class could be 'cats' + 'dogs' (2 classes)

  $\rightarrow$ if I segment on **three** 'colours', then I want
  - probabilities of the resulting **six** classes (probably)
  - entropy of that

Recap: probability

$$\text{IG} = H_p - \sum_{i=0}^{N} p_{c,i} H_{c,i}$$

- positive values mean entropy of data at the next level has **decreased**

  $\rightarrow$ data is getting more 'pure'

  $\rightarrow$ if proceeding to $H = 0$ then we have maximum purity with no uncertainty

  $\rightarrow$ no uncertainty = max information, and information **gain**

- want to split according to maximum information gain

(or maximise entropy decrease)

Recap: probability

- Gini index measure a similar thing to entropy, and is defined as

$$G = \sum_i G_i = \sum_i p_i(1 - p_i)$$

  $\rightarrow$ single species gives $G = 0$, maximally pure

- can do a similar thing to information gain to minimise the Gini index with splitting

  $\rightarrow$ used more in decision trees (computationally faster)

  $\rightarrow$ entropy measures used more in things like neural networks

Demonstration: penguins data

- consider **classification** problem first
  $\rightarrow$ predict species from other features

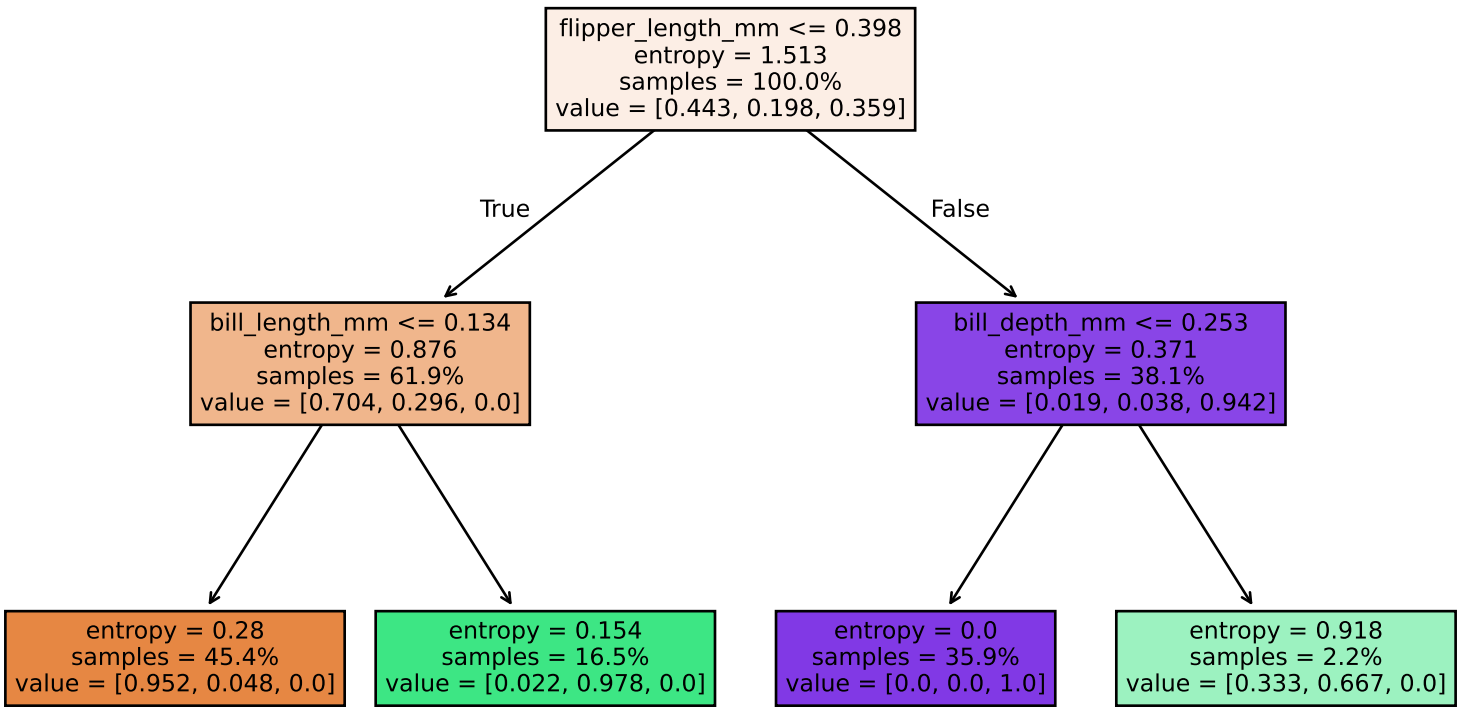

Figure: Tree classifier for penguins data. Maximum depth 2 and uses information entropy criterion.

Demonstration: penguins data

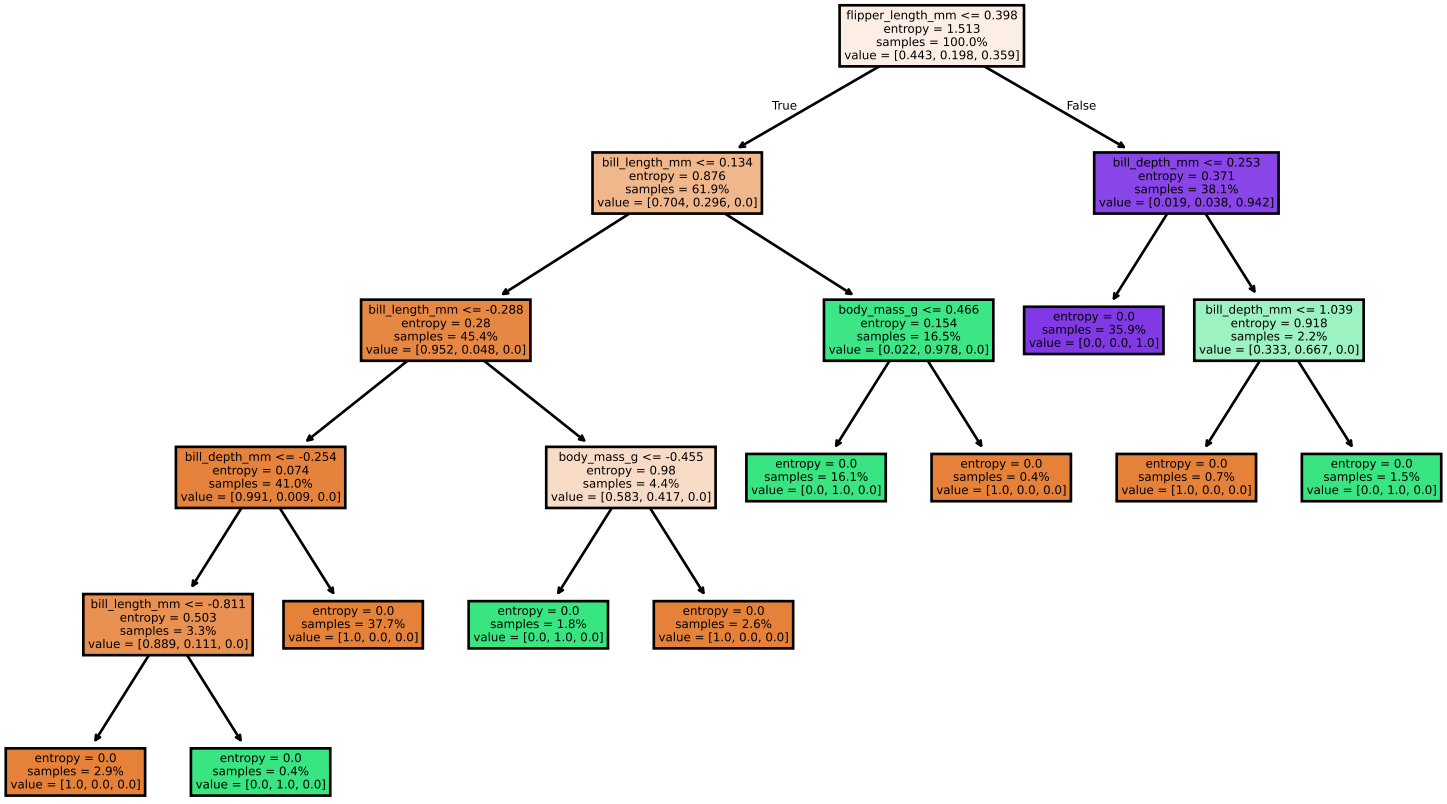

Figure: Tree classifier for penguins data. No maximum depth, and uses information entropy criterion.

Demonstration: penguins data

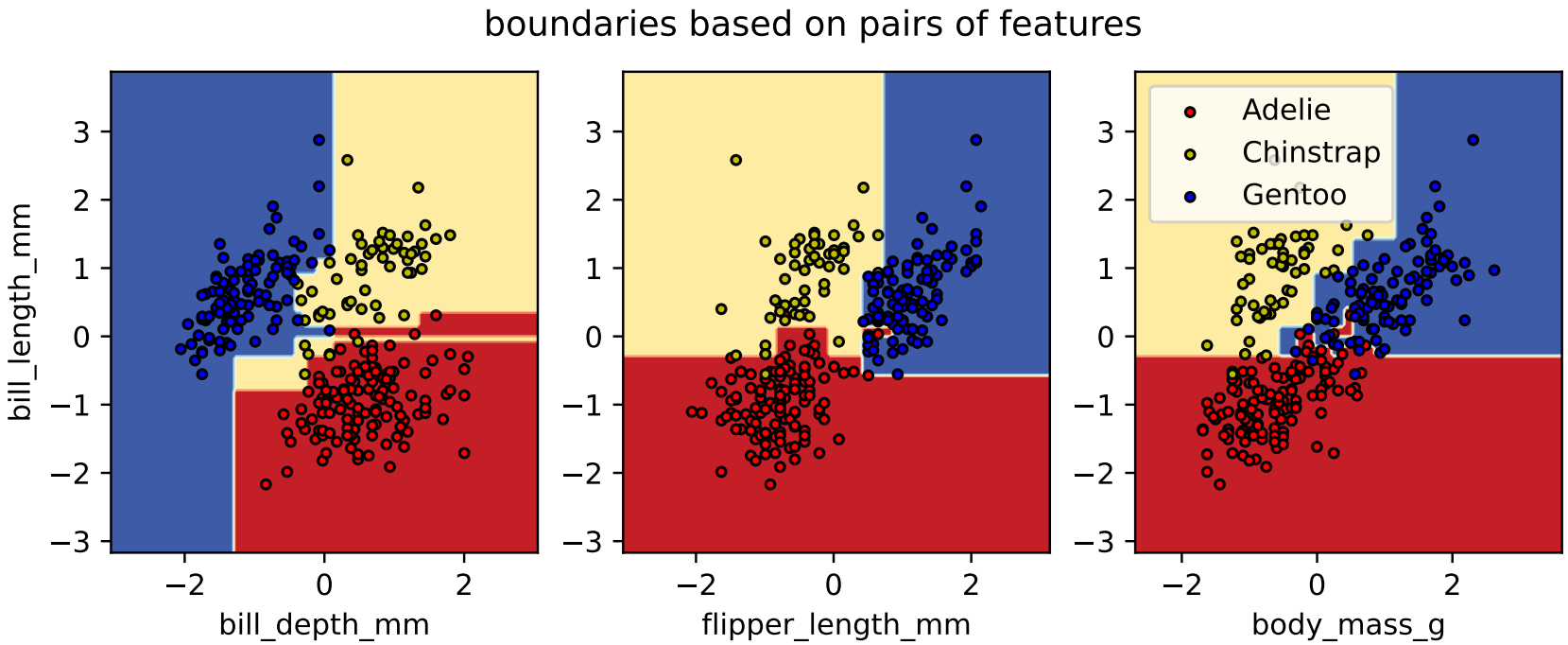

Figure: Tree classifier decision boundaries based on two features for penguins data.

- notice decision trees basically do piecewise constant predictions (see this again when doing regression)

Demonstration: penguins data

- consider now the **regression** problem

  → same problem identified in the second session

  → include species as an input feature?

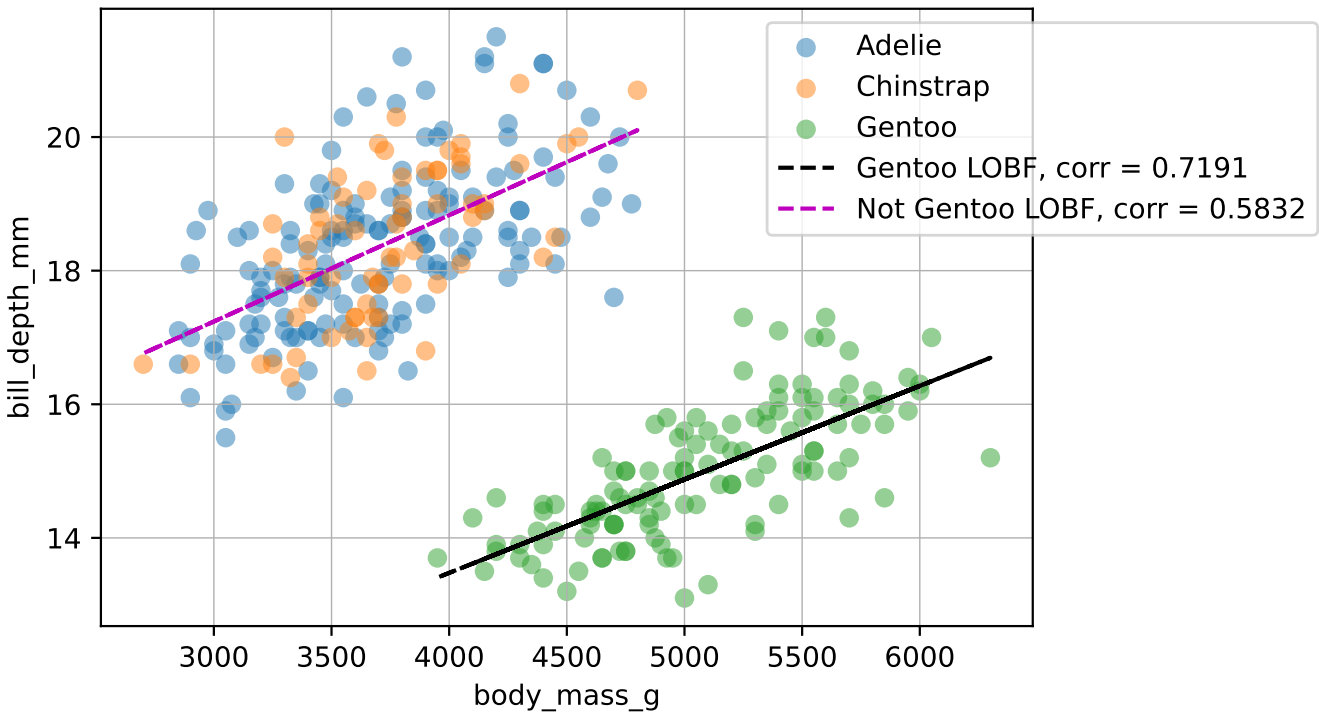

Figure: Instead of models for each species, one model that does different decisions based on species?

Demonstration: penguins data

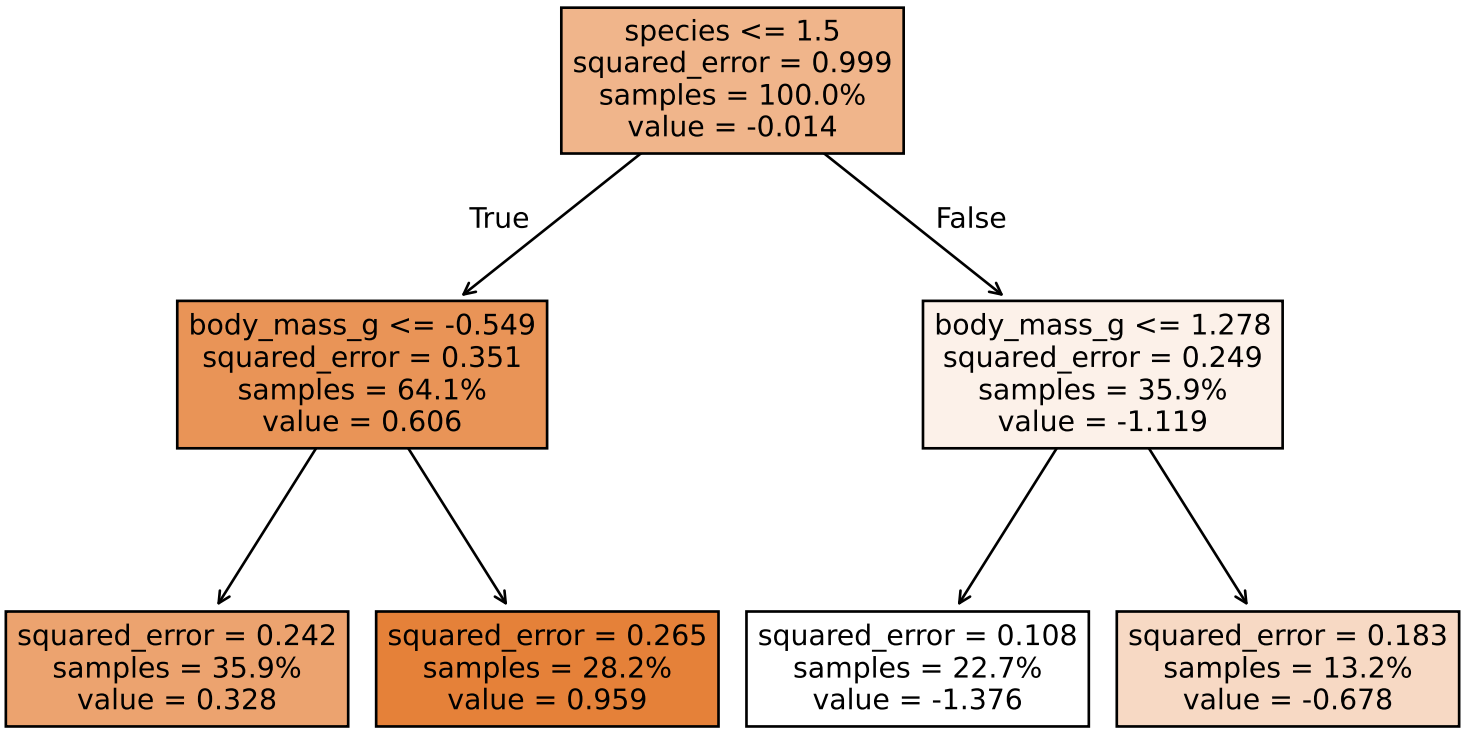

Figure: Tree regressor for penguins data. Maximum depth two, squared loss.

Demonstration: penguins data

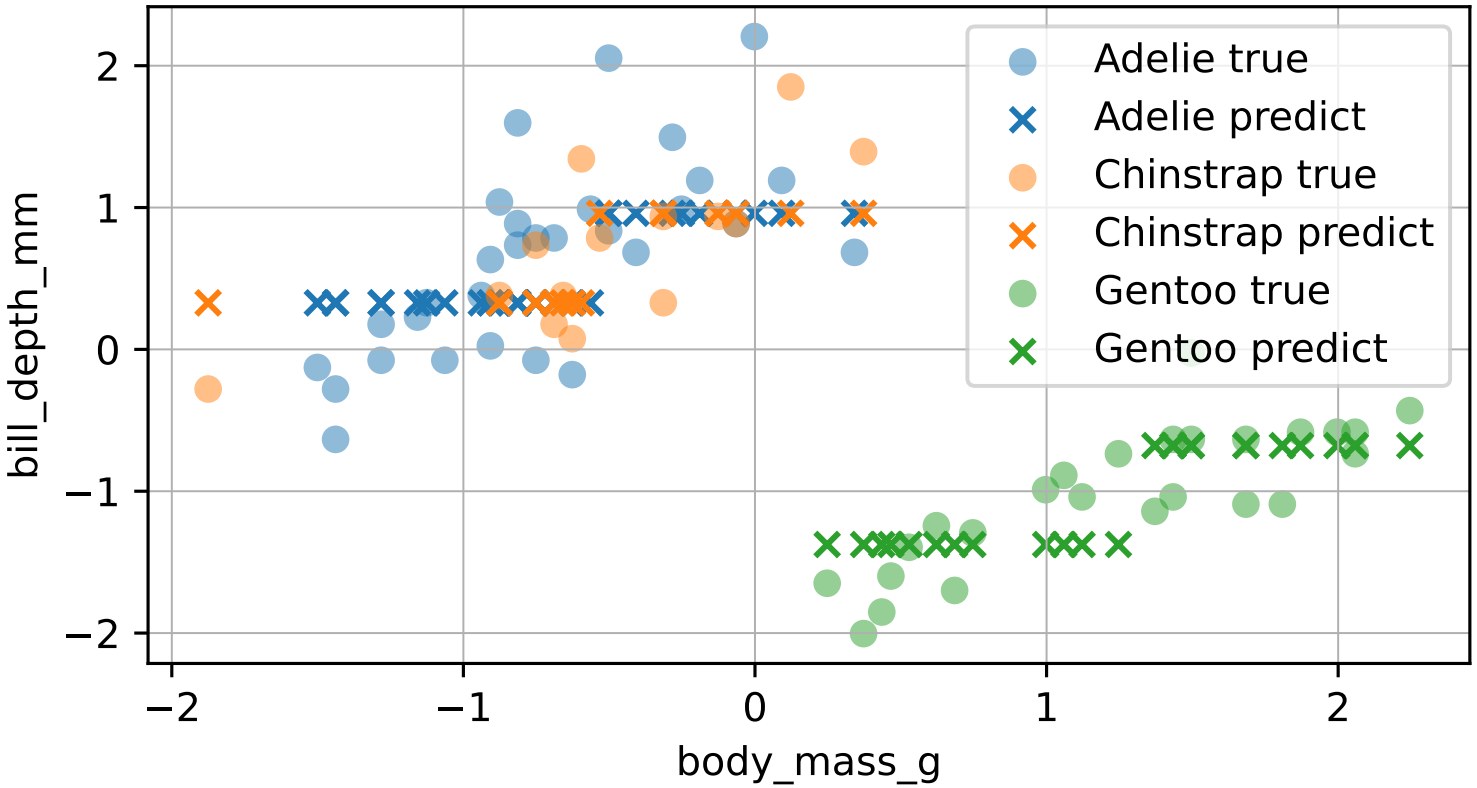

Figure: Tree regressor for penguins data. Maximum depth two, squared loss.

- notice the species predictions are clustered accordingly

  → notice the piecewise constant predictions (model too shallow)

Demonstration: penguins data

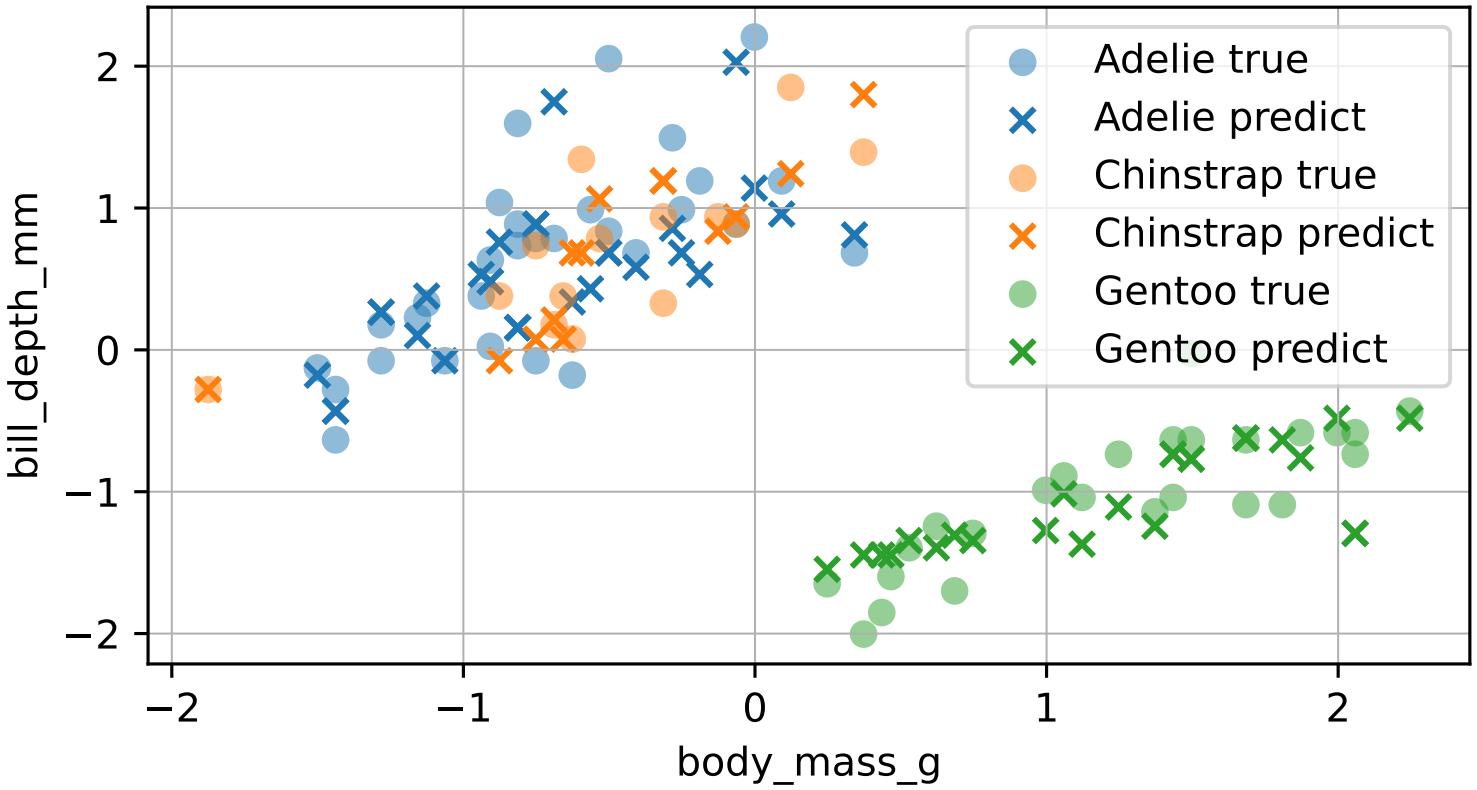

Figure: Tree regressor for penguins data. No maximum depth, squared loss.

- recover variability by having more complexity

Demonstration: penguins data

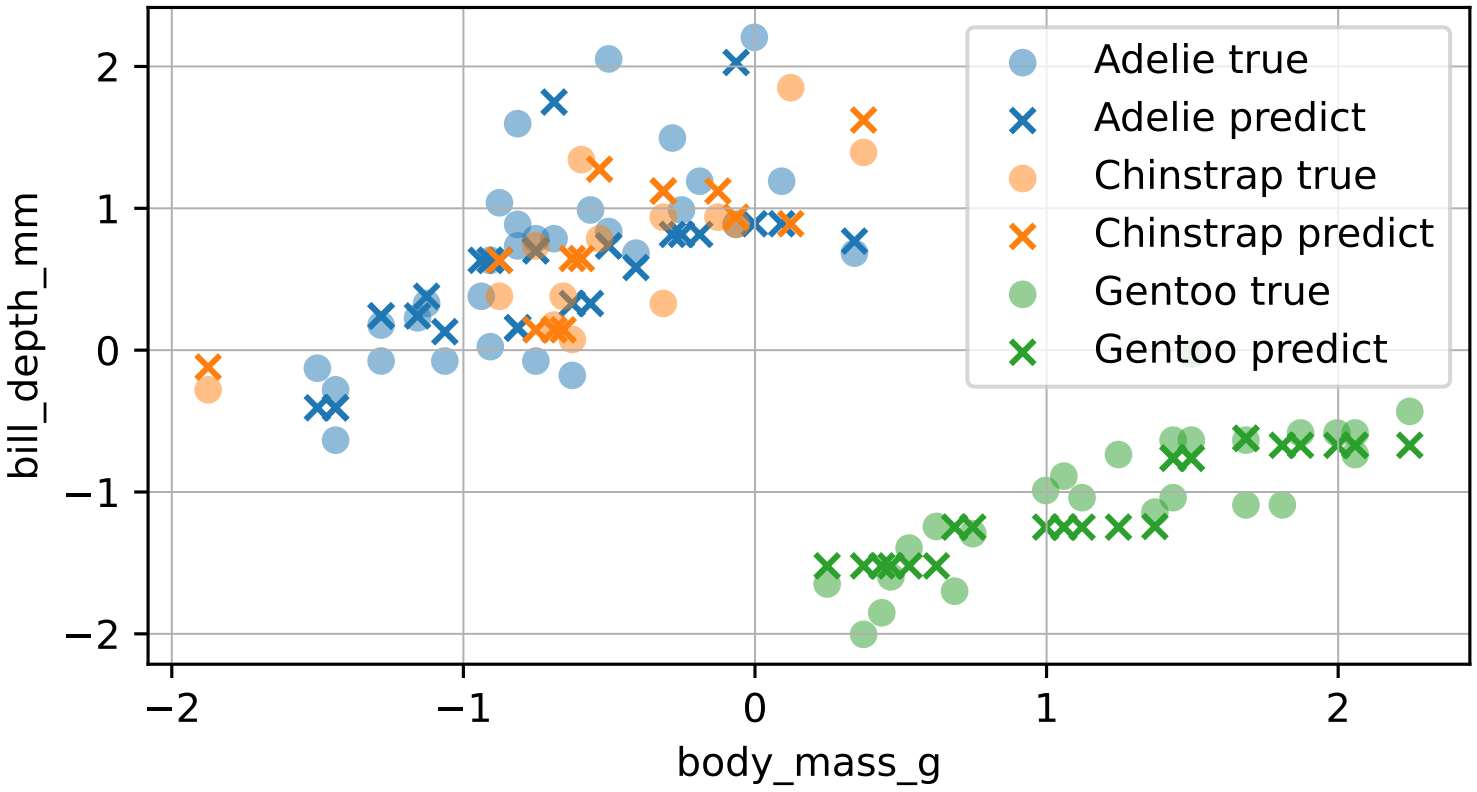

Figure: Tree regressor for penguins data with post-pruning. No maximum depth, squared loss.

- pruning to avoid over-fitting + promote robustness
  → remove nodes that add extra complexity but not that much skill (cf. A/BIC)

Demonstration

- went into some detail in how decision trees are formed

  $\rightarrow$ concepts in information entropy

  $\rightarrow$ used also with model selection and neural networks
- trees as classifiers and/or regressors

  $\rightarrow$ need for cross-validation / hyper-parameter tuning

  $\rightarrow$ ensembles? (see next lec)

  Moving to a Jupyter notebook $\rightarrow$

Figure: An example of a tree (a broc-collie).