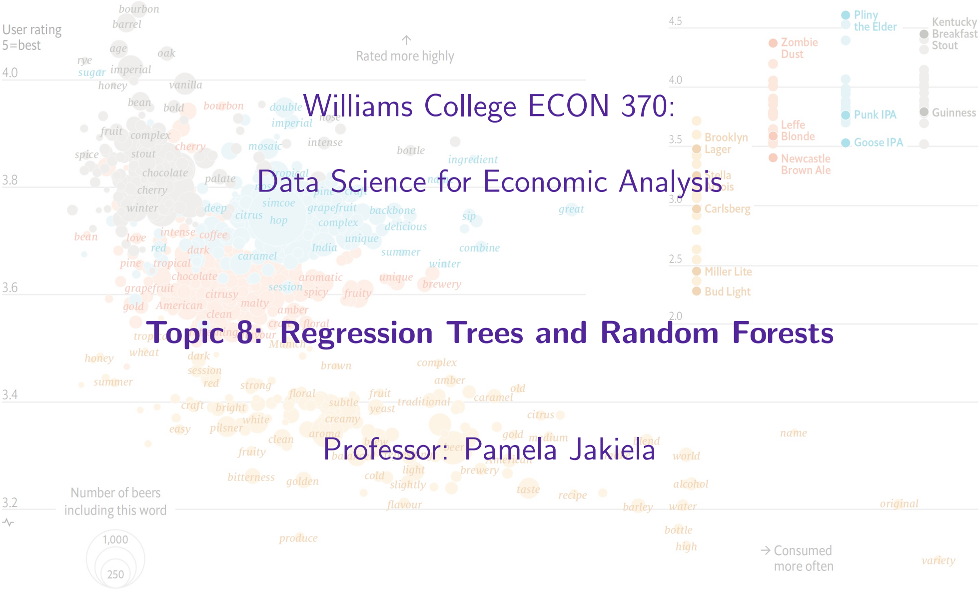
Williams College ECON 370:

Data Science for Economic Analysis

Topic 8: Regression Trees and Random Forests

Professor: Pamela Jakiela



Outline

- Regression trees
- Bagging and random forests
- Lab: using DHS data to predict height-for-age

Thought Experiment

Suppose X_1, X_2 and ε are iid standard normal, and let $Y = X_1 + \varepsilon$

The variance of Y is 2, so if you predicted Y with the mean of Y, $E[MSE] = 2$

Suppose you split the sample based on X_2, into an $X_2 < 0$ subsample and an $X_2 \geq 0$ subsample

What is the expected variance within each subsample?

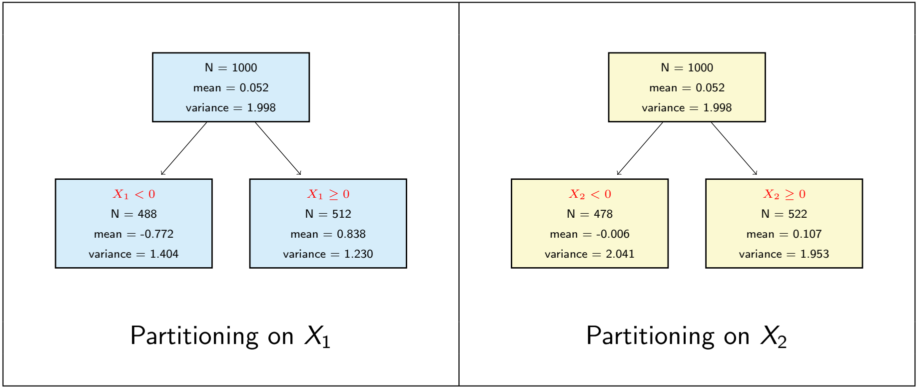
If you predict Y with the mean in each subsample, what is the expected MSE?

What if you split the sample based on X_1, into an $X_1 < 0$ subsample and an $X_1 \geq 0$ subsample

What is the expected variance within each subsample?

If you predict Y with the mean in each subsample, what is the expected MSE?

Partitioning the Sample to Reduce (Subsample) Variance



What Is a Regression Tree?

A **regression tree**:

- Partitions data into homogeneous **leaves** (subsamples) through recursive binary splitting
- Each observation is mapped to a single leaf (i.e. a terminal node at the bottom of the tree)
- The predicted outcome Y for observation i in leaf j is the mean in Y in leaf j

If you split at random, then every leaf would be representative of the original sample

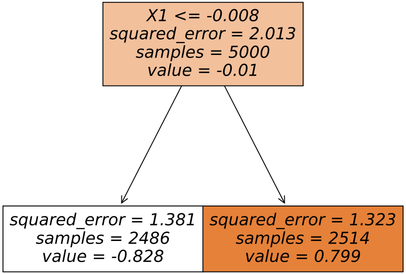
- Instead, we split the sample with the intention of reducing within-leave variance in Y

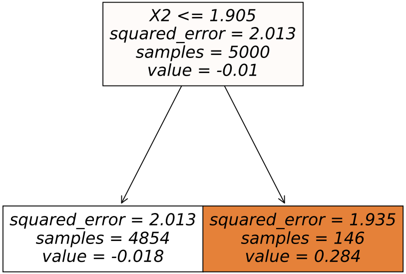
Building a Regression Tree

- To make the first partition of the sample, we identify the binary split of the data that leads to the largest reduction in (training) MSE by searching over all the...
  - dummy variables in the data, including dummies for values of categorical variables
    - When we encode categoricals as dummies, we do not leave one out (**one hot encoding**)
  - possible cutoff values s for all the continuous/numeric variables (that partition the sample)
- To make all subsequent splits, we repeat this process for each subsample to identify the best next split (to reduce MSE) until we reach a maximum depth, minimum leaf size, etc.

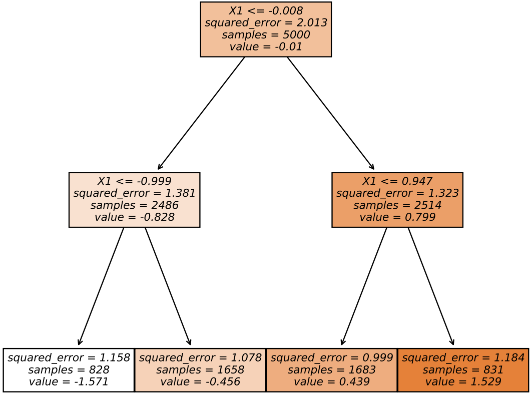
Building a Regression Tree: Example

Example: X_1, X_2 and ε are iid standard normal, and $Y = X_1 + \varepsilon$ (so only X_1 predicts Y)





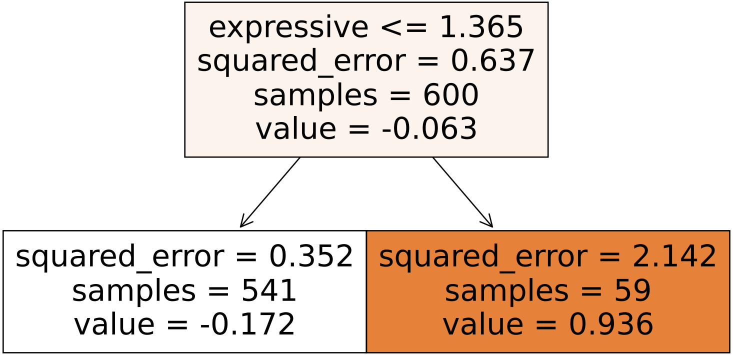
Building a Regression Tree: Example



Building a Regression Tree with the EMERGE Data

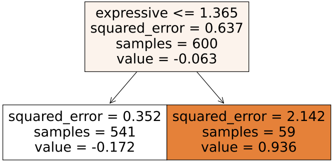
- Outcome: literacy
- Variables: household size, mother's education, (child is) male, height-for-age z-score, child age in months, receptive vocabulary, expressive vocabulary, fine motor skills
  - Most important predictors of literacy if we run OLS or lasso or ridge regression: mother's education, male dummy, height-for-age z-score, expressive vocabulary
- Sample: $N = 1{,}000$ children ages 3 to 6 from rural western Kenya

Building a Regression Tree with the EMERGE Data: `max_depth` = 1



data source: EMERGE, $N = 1000$, Y = literacy, X variables (child age, expressive vocab, fine motor skills, HAZ, HH size, male, mother's education, receptive vocab)

Evaluating a Regression Tree

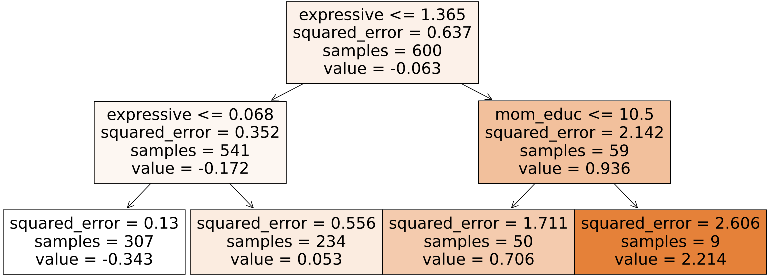


data source: EMERGE

Does partitioning the sample succeed, generating more accurate predictions of Y and reducing test MSE?

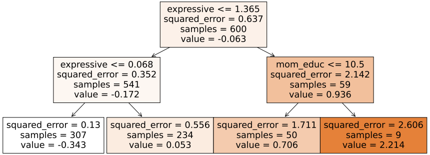
- Split the data into test/train to assess this
- Benchmark: using mean $Y \rightarrow$ test MSE of 0.749
- A tree with one split $\rightarrow$ test MSE of 0.712
- This obviously depends on your variables
  - `expressive` is a strong predictor of `literacy`

Building a Regression Tree with the EMERGE Data: `max_depth` = 2

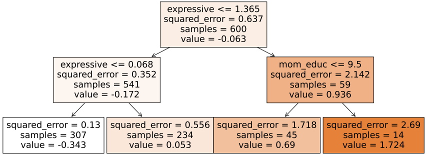


data source: EMERGE, $N = 1000$, Y = literacy, X variables (child age, expressive vocab, fine motor skills, HAZ, HH size, male, mother's education, receptive vocab)

Constraining the Minimum Leaf Size

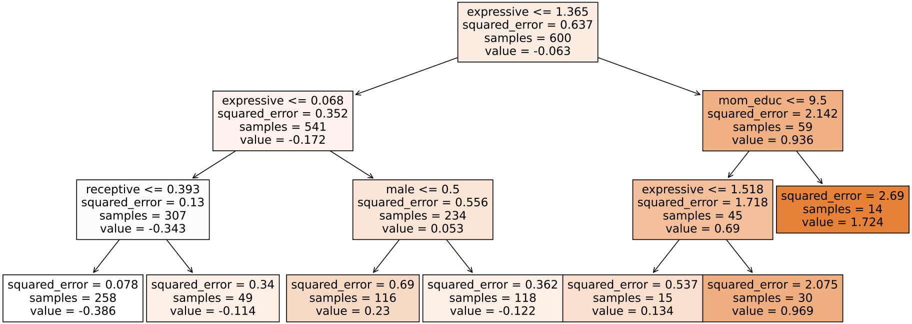


No minimum leaf size: test MSE = 0.660



Minimum leaf size 10: test MSE = 0.637

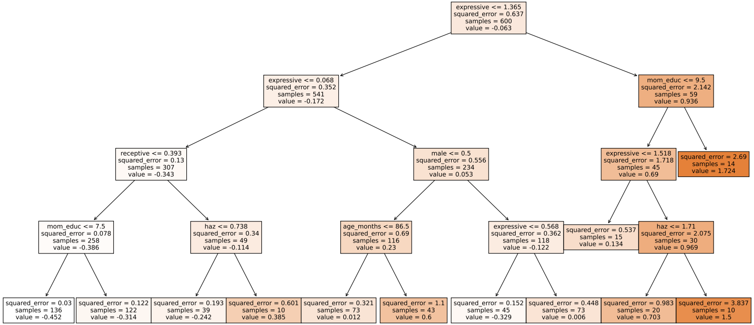
Building a Regression Tree with the EMERGE Data: max_depth = 3



data source: EMERGE, $N = 1000$, Y = literacy, X variables (child age, expressive vocab, fine motor skills, HAZ, HH size, male, mother's education, receptive vocab)

Test MSE = 0.615

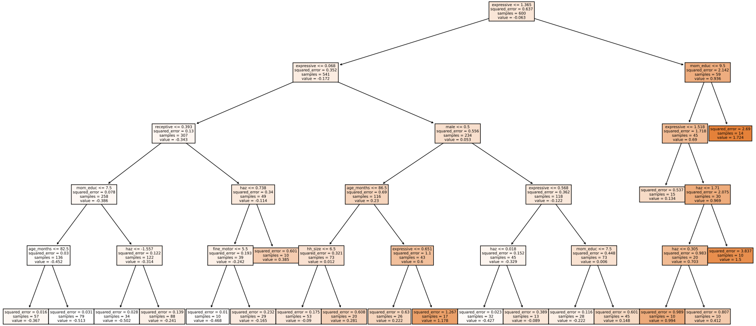
Building a Regression Tree with the EMERGE Data: max_depth = 4



data source: EMERGE, $N = 1000$, Y = literacy, X variables (child age, expressive vocab, fine motor skills, HAZ, HH size, male, mother's education, receptive vocab)

Test MSE = 0.612

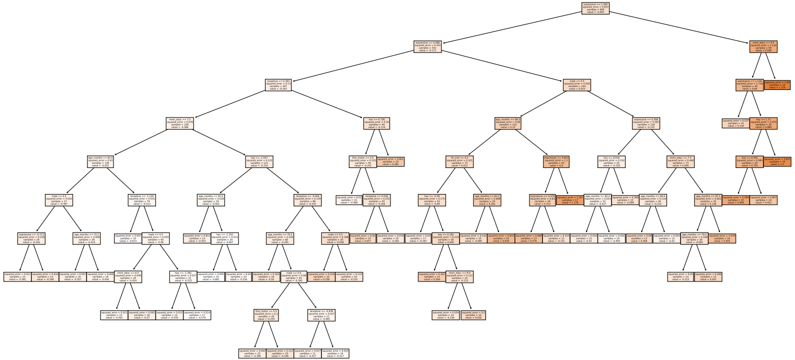
Building a Regression Tree with the EMERGE Data: `max_depth` = 5



data source: EMERGE, $N = 1000$, Y = literacy, X variables (child age, expressive vocab, fine motor skills, HAZ, HH size, male, mother's education, receptive vocab)

Test MSE = 0.609

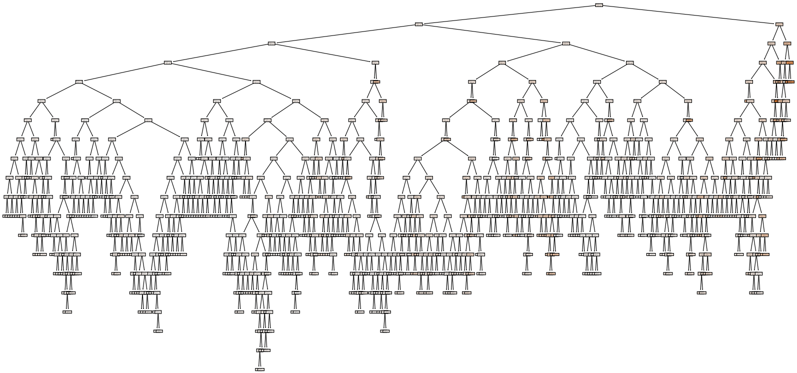
No Constraints on Depth Can Lead to Over-Fitting



data source: EMERGE, $N = 1000$, $Y =$ literacy, X variables (child age, expressive vocab, fine motor skills, HAZ, HH size, male, mother's education, receptive vocab)

With no constraints on tree depth, test MSE = 0.645

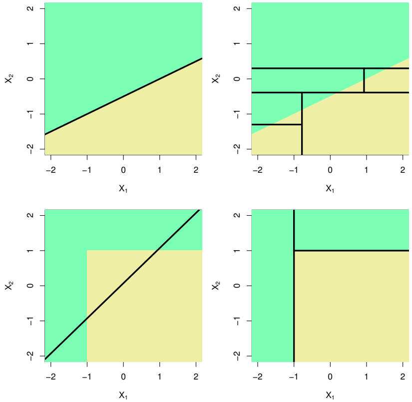
With No Constraints on Depth or Minimum Leaf Size, Test MSE = 1.123



data source: EMERGE, $N = 1000$, Y = literacy, X variables (child age, expressive vocab, fine motor skills, HAZ, HH size, male, mother's education, receptive vocab)

Strengths of Regression Trees

- Tend to identify important predictors of Y
  - Continuous predictors may be used multiple times
  - Expressive vocabulary, mother's education, height-for-age used repeatedly in a single tree
- Leverage Xs in a parsimonious and intuitive way
  - Distinguishes between important, unimportant Xs
- Good at identifying interactions between covariates
  - Is the linear model a reasonable approximation of the true underlying relationship between Xs and Y?



source: James el at. (2021)

Weaknesses of Regression Trees

- Can be prone to over-fitting, particularly when the number of variables large relative to N
  - Techniques for pruning and constraining tree depth or leaf size are somewhat ad hoc
- Not particularly robust (e.g. to small changes in the analysis sample)
- May not compete with other approaches in terms of predictive accuracy

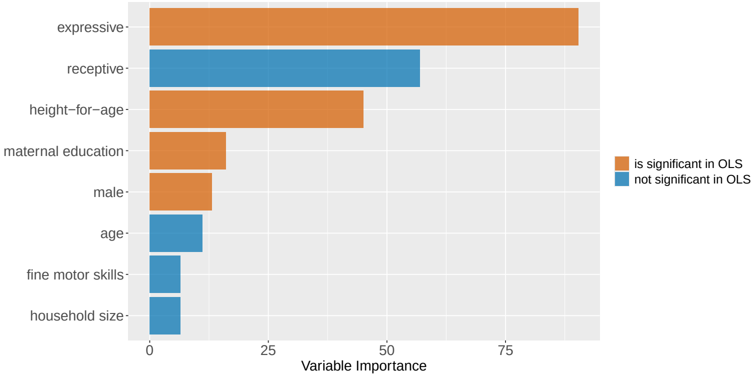
Bagging

- Bootstrap sampling: sampling from a data set with replacement
  - Provides a measure of how much estimates might change because of sampling variation
- Bootstrap aggregation or **bagging** involves estimating multiple regression trees on bootstrapped samples of a data set, an then averaging predictions across trees
  - Bootstrapped data sets are slightly different
  - Tends to lower test MSE relative to a single tree
- Not all observations are included in a bootstrap sample (or bag)
  - Out-of-bag (OOB) predictions, MSE are a natural analog to cross-validation MSE

Random Forests

- **Random forests** extend bagging by considering only a random subset of Xs at each split
  - Limits over-reliance on a small number of key predictors
- Bagging can be seen as a special case of a random forest where all variables are considered
- In practice, no one uses a single regression tree except as an example
- We cannot visualize a random forest the way we can visualize a regression tree
  - Measures of **variable importance** indicate how often a variable is chosen across trees

Example: Variable Importance in EMERGE Data



data source: EMERGE, $N = 1000$, $Y =$ literacy, X variables (child age, expressive vocab, fine motor skills, HAZ, HH size, male, mother's education, receptive vocab)

Gradient-Boosted Trees

- In a random forest, each regression tree is independent of all other trees
  - A **gradient-boosted tree** is part of a (non-random) forest with interdependent trees
- The basic idea is simple:
  - Step 1: fit a regression tree that is not very deep (one or two spilts)
  - Step 2a: residualize Y by subtracting initial predictions $\times$ some very small positive number
  - Step 2b: fit a regression tree that is not very deep on the residualized Ys
  - Step 3: repeat steps 2a and 2b a hundred or so more times

Summary: Regression Trees and Random Forests

- Regression trees are an elegant prediction technique based on repeated binary splitting
  - The technique usually doesn't work that well in practice
- Random forests and gradient-boosted trees are ensemble methods that average the predictions of large numbers of regression trees to generate more robust predictions
  - In random forests, we estimate many independent regression trees, bootstrapping the sample for each tree and within each tree choosing a subset of predictors to consider at each split
  - In gradient-boosted trees, we update Y values before estimating each new tree, subtracting off a scaled-down version of the predicted values for the previous iteration

Lab #8

Objective: compare the predictive power of tree-based machine learning techniques using DHS data on the height-for-age z-scores of young children in Kenya (using the 2014 births recode)

Overview of the DHS

- Standard DHS surveys in multiple countries, conducted every 5–10 years in many
  - Representative of women aged 15–49 (i.e. of childbearing age)
  - Women asked about all pregnancies and births, detailed info on births in last 5 years
  - Children under 5 years old are weighed and measured (height-for-age z-score)
  - Random sample of women's husbands are also interviewed (in some countries/rounds)
- DHS **births recode** survey contains information on all births by surveyed women
  - Includes (most) data from survey of mother and most information about the household
  - Data set includes deceased children, those born more than 5 years ago (for whom there is no information on either anthropometrics of birth outcomes), children of "visitors" to household
  - Information on variables is contained in separate text files

Lab #7: Steps

Objective: build a random forest with the lowest test mean squared error

- R and Python templates read in births recode, restrict sample, estimate regression trees
  - Your main task is to refine the preprocessing pipeline to to improve fit
- The templates implement a simple regression tree and a random forest
  - Choose how to preprocess categorical variables (and which should be categorical)
  - Choose number of variables to consider at each decision node
  - Choose number of trees in the random forest