

Module 10

Regression Trees, Random Forests & Boosting

Shmueli et al., Chapter 9 (and 13)

Regression tree

Random forest

Compare

Boosting

Roumen Vesselinov, PhD **Roumen@jhu.edu**

Explore the data · build the model · ask the AI · verify every result

GDP per capita (Y1) and predictors

Var	Description	Var	Description
Y1	GDP per capita, constant 2015 US\$ (mean ≈ \$14,000)		
X1	Annual GDP growth rate (%)	X11	Total population
X2	Inflation rate (CPI, annual %)	X12	Population aged 65+ (% of total)
X3	Trade (% of GDP)	X13	Urban population (% of total)
X4	FDI net inflows (% of GDP)	X14	Top 10% income share (inequality)
X5	Fraser Economic Freedom Index	X15	Government effectiveness (WGI)
X6	Oil rents (% of GDP): resource channel	X16	Rule of law (WGI)
X7	Human Development Index	X17	Political stability & absence of violence (WGI)
X8	Life expectancy at birth (years)	X18	V-Dem electoral democracy index
X9	Internet users (% of population)	X19	Historical region (10 categories)
X10	Government expenditure on education (% of GDP)	X20	Year (numeric, for time-trend modeling)

X19: historical region

Value	Region	Value	Region
1	Eastern Europe & post-Soviet Union	6	East Asia
2	Latin America	7	South-East Asia
3	North Africa & Middle East	8	South Asia
4	Sub-Saharan Africa	9	The Pacific
5	Western Europe & North America (incl. Australia, NZ)	10	The Caribbean

X19 is categorical: in models it becomes dummy variables (Slide 8).

Regression trees for prediction

Same as classification trees

- Used with a continuous outcome variable.
- Procedure similar to the classification tree.
- Many splits attempted; choose the one that minimizes impurity.

Differences

- Prediction = the average of the numerical target in the rectangle (end node).
- Impurity = sum of squared deviations from the leaf mean.
- Performance measured by RMSE (root mean squared error).

Choose the split that reduces squared error most

$$\text{SSE}(\text{split}) = \Sigma(y - \text{mean of left node})^2 + \Sigma(y - \text{mean of right node})^2$$

Try all splits

Every predictor and cut value; keep the smallest total SSE (largest drop in variance).

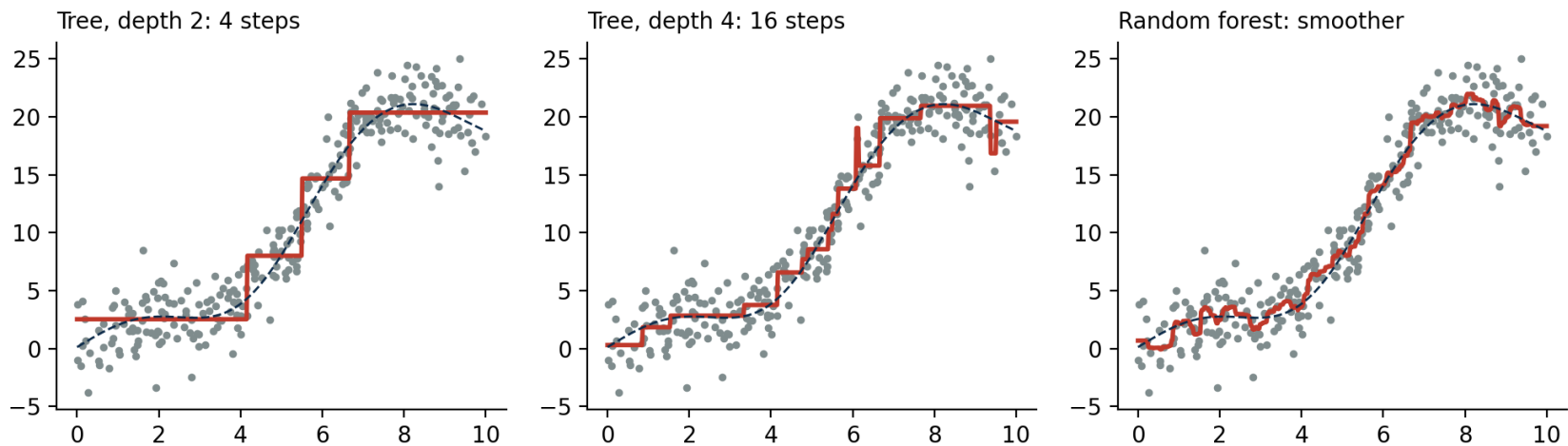
Leaf prediction

The mean Y of the training records that end in the leaf.

Stopping

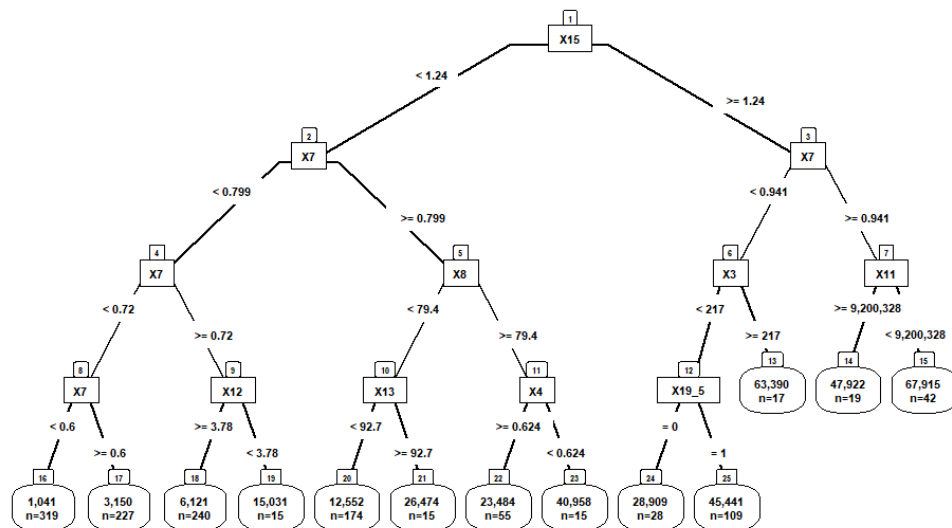
minbucket, maxdepth, or pruning with cross-validated cp, as in Module 9.

A staircase approximation



Simulated data: smooth truth (dashed) vs. tree and forest predictions

The tree for GDP per capita



rpart, minbucket = 15, maxdepth = 4

- Hyperparameters:
- minbucket = 15: the minimum number of observations in any end node.
- maxdepth = 4: the maximum depth of any node of the final tree.

Interpretation of end-node paths

End node #16: \$1,041 (low)

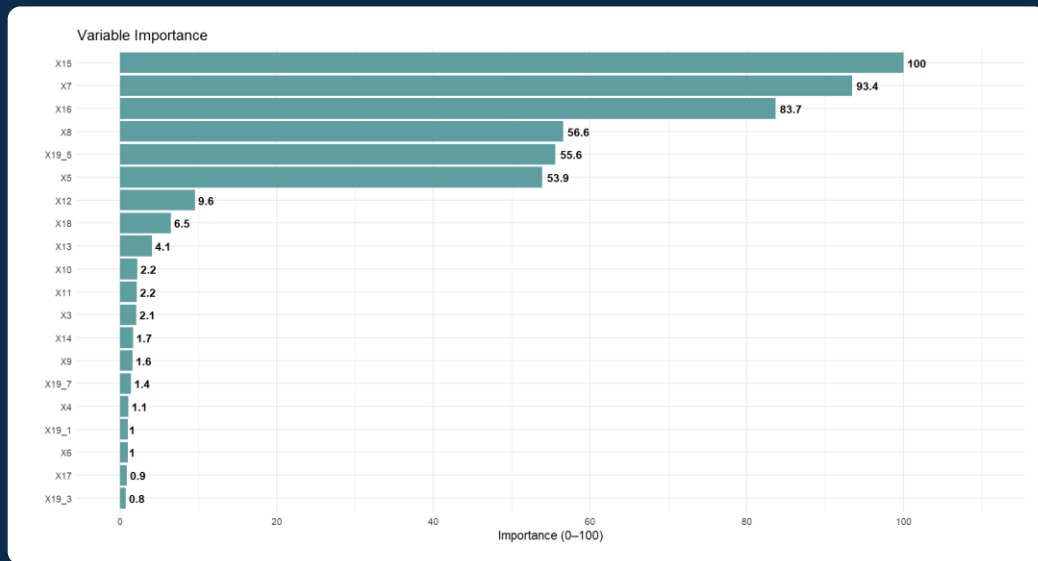
- Countries with government effectiveness (WGI estimate) < 1.24 ,
- and Human Development Index (HDI) < 0.6 .

End node #15: \$67,915 (high)

- Government effectiveness ≥ 1.24 ,
- HDI ≥ 0.941 ,
- and total population < 9.2 million.

Each end node predicts the average GDP per capita of its training countries.

Variable importance measure (VIM)



CART variable importance

- X15 government effectiveness (WGI estimate).
- X7 Human Development Index (HDI).
- X16 rule of law (WGI estimate).
- X15 and X16 correlate 0.95: surrogate splits share importance.

Accuracy measures

```
> accuracy(predict(rf1,valid1),valid1$Y1)
```

	ME	RMSE	MAE	MPE
Test set	336.7609	3754.396	1521.877	-9.458398

RMSE on validation data

- Y1 is the outcome; RMSE based on validation data.
- $RMSE < SD$: $\$3,754 < \$19,165$.
- The model has good predictive ability ($RMSE/SD = 0.20$).
- Linear regression in Module 4: $RMSE/SD \approx 0.47$.

```
> summary(QoGwork$Y1)
```

Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
333.4	1561.7	5410.3	14241.7	18089.6	110873.2

```
> sd(QoGwork$Y1, na.rm = TRUE)
```

```
[1] 19165.62
```

SD of Y1

PART 2

Random forest

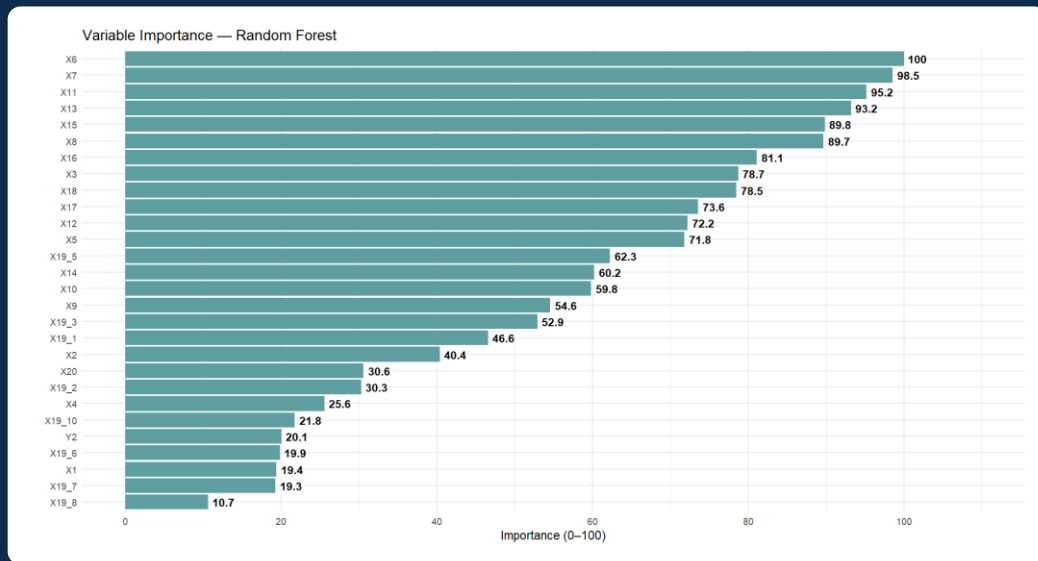
Random forest

- Can be used with binary and continuous outcomes.
- Draw multiple bootstrap resamples of cases from the data.
- For each resample, use a random subset of predictors and produce a tree (default $m_{try} \approx p/3$ for prediction).
- Combine the predictions from all the trees (the "forest").
- Voting for classification; averaging for prediction.

Averaging helps

Many staircases with different steps average into a smooth, stable prediction.

Variable importance: ntree = 500, nodesize = 5



Random forest variable importance

- X6 oil rents (% of GDP): resource channel.
- X7 Human Development Index.
- X11 total population.
- Different from CART: forests spread credit across correlated predictors and reward interactions.

Accuracy measures

```
> accuracy(predict(rf1,valid1),valid1$Y1)
```

	ME	RMSE	MAE	MPE
Test set	329.5462	3797.327	1541.35	-9.850553

RMSE on validation data

- Y1 is the outcome; RMSE based on validation data.
- $RMSE < SD$: \$3,797 < \$19,165.
- The model has good predictive ability.

```
> summary(QoGwork$Y1)
```

Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
333.4	1561.7	5410.3	14241.7	18089.6	110873.2

```
> sd(QoGwork$Y1, na.rm = TRUE)
```

```
[1] 19165.62
```

SD of Y1

CART vs. random forest

\$3,754

RMSE CART

\$3,797

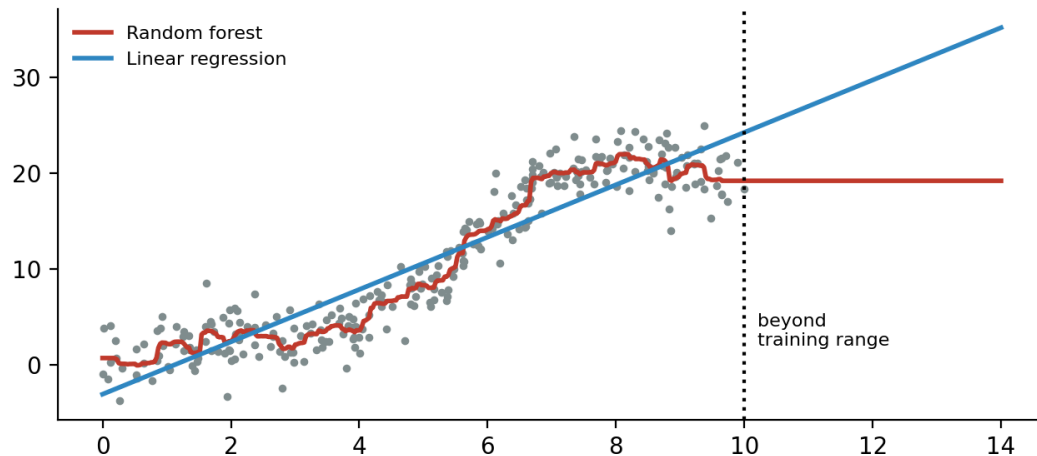
RMSE random forest

CART

Slightly better (1%)

- Use RMSE: $\text{RMSE CART} < \text{RMSE RF}$, so the CART model is better here.
- A 1% difference from one validation split is within random variation: cross-validation would give a firmer comparison.

Trees cannot extrapolate



Beyond the training range a forest predicts a constant; a linear model continues the trend

- Predictions are averages of training outcomes.
- A tree or forest never predicts above the richest (or below the poorest) training country.
- Keep this in mind for countries or years outside the training data.

Advantages of CART models

Easy

Easy to use and understand.

Rules

Produce rules that are easy to interpret and implement.

Automatic

Variable selection and reduction is automatic.

Few assumptions

Do not require the assumptions of statistical models; can work without extensive handling of missing data.

Disadvantage of single trees: instability.

Ensemble CART methods

- Random forests (RF), bagging, and boosting.
- Predictions from many trees are combined.
- Very good predictive performance, better than single trees: often the top choice for predictive modeling.
- Disadvantage: a "black box", with no simple rules.

Trade-off

*Accuracy vs.
interpretability.*

PART 3

Boosted trees: the idea

Boosted trees

1 Fit a single tree

Start with one tree on the training data.

2 Reweighted bootstrap

Draw a bootstrap sample with higher selection probability for misclassified (poorly predicted) records.

3 Fit a new tree

On the new bootstrap sample.

4 Repeat

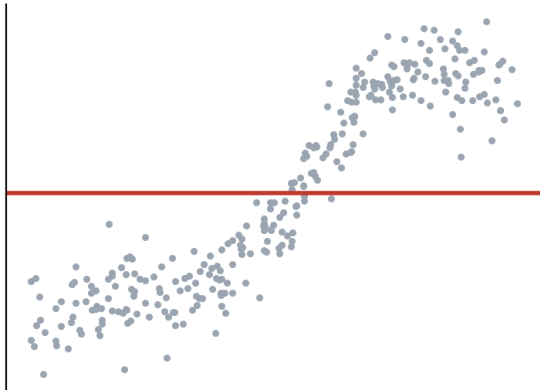
Steps 2 and 3, many times.

5 Weighted vote

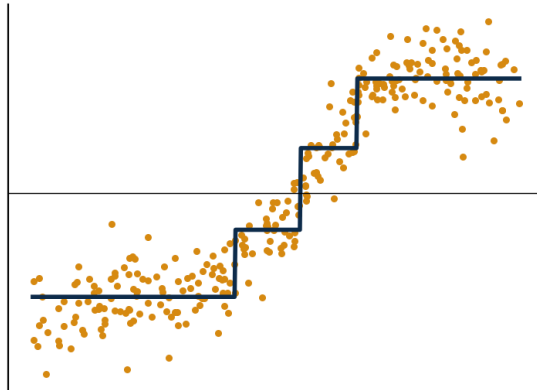
Classification: weighted voting;
prediction: weighted averaging, with heavier weights for better trees.

Each new tree fixes what is left

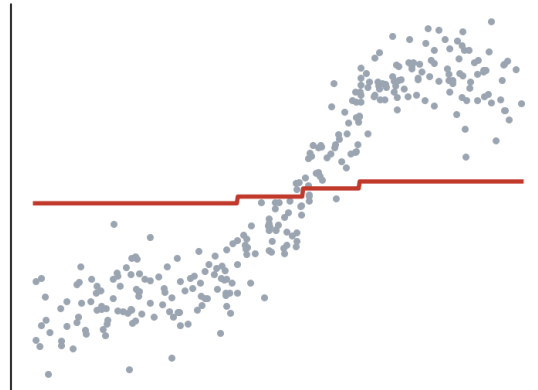
1. Start: predict the mean



2. Fit a small tree to the residuals



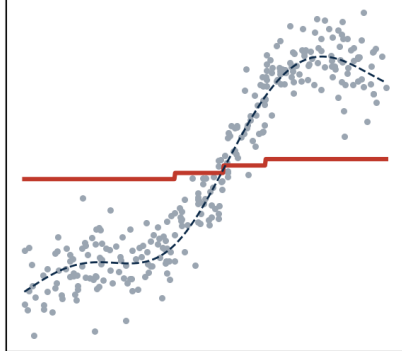
3. Add it $\times$ learning rate (0.1); repeat



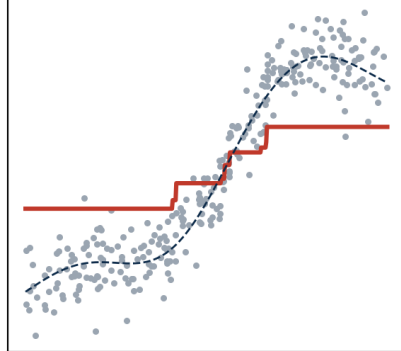
Simulated data

From one tree to two hundred

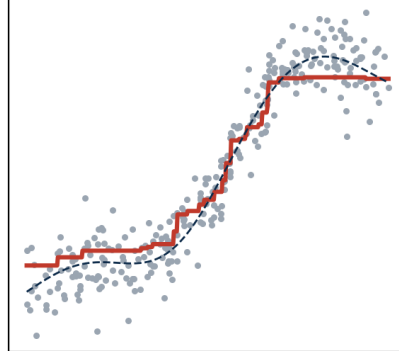
After 1 tree



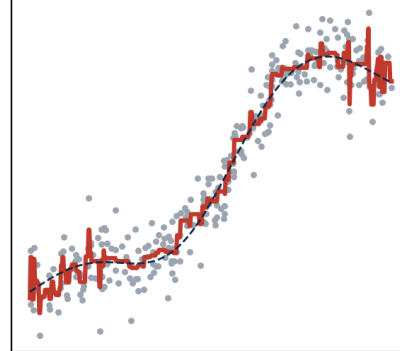
After 5 trees



After 20 trees

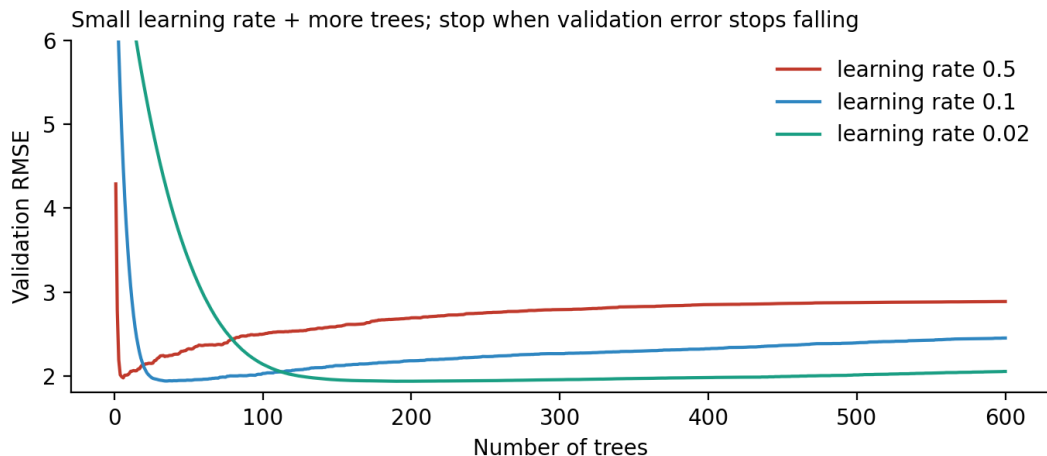


After 200 trees



Prediction (red) vs. truth (dashed) after 1, 5, 20, and 200 shallow trees

Learning rate and number of trees



Validation error as trees are added (simulated)

- Large learning rate: fast, but can overfit.
- Small learning rate: slow, needs more trees, usually better.
- Stop adding trees when validation error stops improving (early stopping).

CART vs. random forest vs. boosted trees

Aspect	CART	Random forest	Boosted trees
How trees are built	A single tree	Many trees, in parallel	Many trees, in sequence
Each tree learns from	All data	Bootstrap sample + random predictors	The errors of previous trees
Tree depth	Often deep, then pruned	Deep, unpruned	Shallow (a few levels)
Prediction	One tree's output	Average / majority vote	Sum of all trees $\times$ learning rate
Main risk	Overfits easily	Less interpretable	Overfits if not tuned

XGBoost, the most popular boosting implementation, and how to interpret it with SHAP plots: Module 12, Part 4.

Summary

- 1 Classification and regression trees are an easily understandable and transparent method for predicting or classifying new records.
- 2 A single tree is a graphical representation of a set of rules.
- 3 Tree growth must be stopped to avoid overfitting; cross-validation helps pick the right cp level.
- 4 Ensembles (random forests, bagging, and boosting) improve predictive performance, but you lose interpretability and the rules of a single tree.
- 5 For Y1: CART RMSE \$3,754 and RF \$3,797, both far below the SD of \$19,165.

MODULE 10

Regression Trees, Random Forests & Boosting

Roumen Vesselinov · JHBigData.com