The Mighty Random Forest

Section 3

TDWI Orlando Data Literacy Bootcamp

Slides courtesy of
"Machine Learning Made Easy – No, Really!"

True Data Literacy

Descriptive Analytics

Diagnostic Analytics

Predictive Analytics

Business Analysis for Everyone with Excel

Predictive Analytics with Excel – Linear Regression Made Easy

Virtual Option available for these courses

Advanced Analytics with Excel – Logistic Regression Made Easy

Machine Learning Made Easy – No, Really!

https://bit.ly/OrlandoDataLiteracyBootcamp

Trees gone wild!

Bad Tree! Bad!

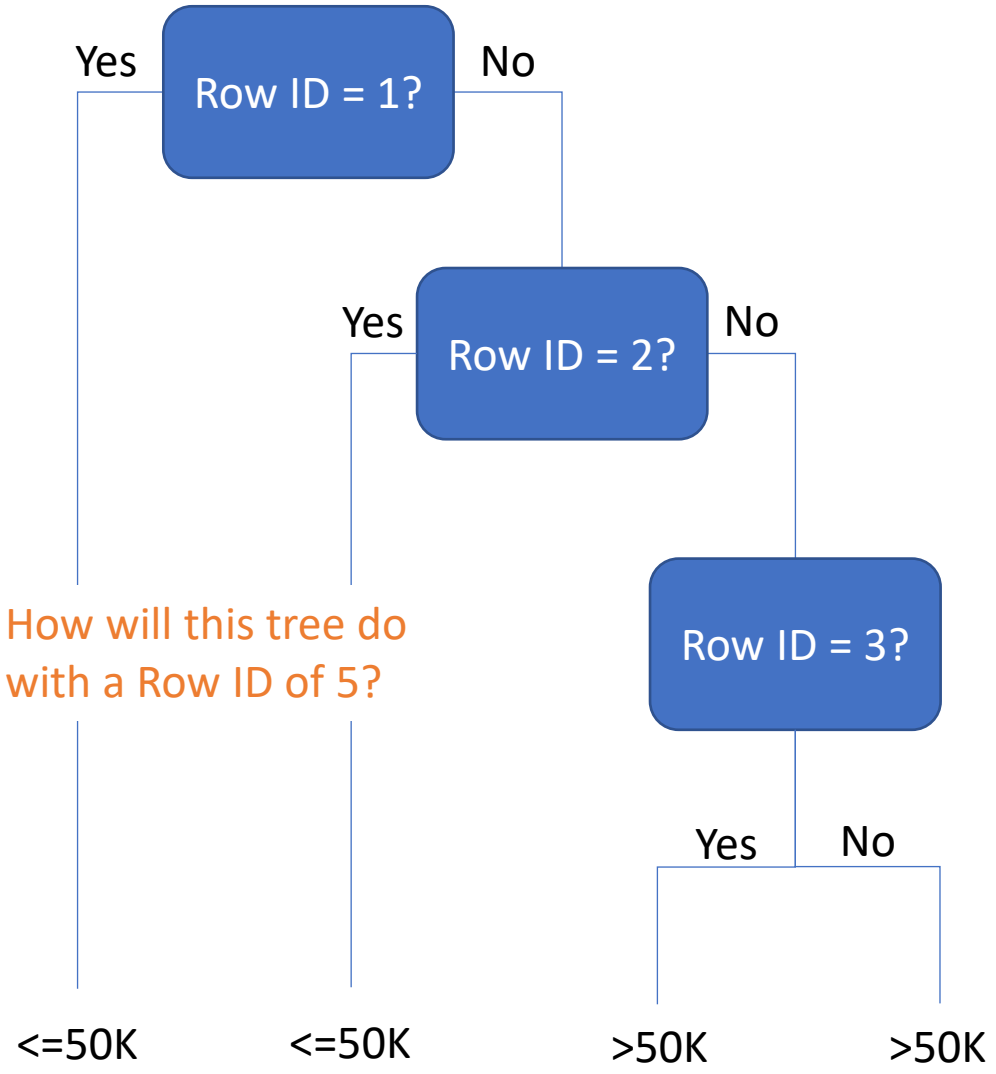
As awesome as decision trees are they have a major problem.

Trees will greedily memorize data, producing models that suffer from *overfitting*.

Models that *overfit* don't handle new data very well.

Data Frame

Row ID	Marital Status	Education Num	Hours Per Week	Age	Label
1	Divorced	11	40	30	<=50K
2	Married-civ-spouse	9	40	46	<=50K
3	Never-married	13	40	26	>50K
4	Divorced	10	58	44	>50K



Trees can radically change their shape in response to data changes!

“As much as we love decision trees, in practice they got problems!”

Wisdom of the Crowd

"The three conditions for a group to be intelligent are diversity, independence, and decentralization." - James Surowiecki

ENSEMBLING:

Compiling predictions from multiple machine learning models in order to make more accurate predictions than any individual model.

However, ensembling works best when the individual models are diverse and ***have low correlation between their predictions.***

The mighty random forest is an ensemble of many decision trees. The algorithm ***manufactures*** independence and diversity across decision trees, making it an effective ensembling method.

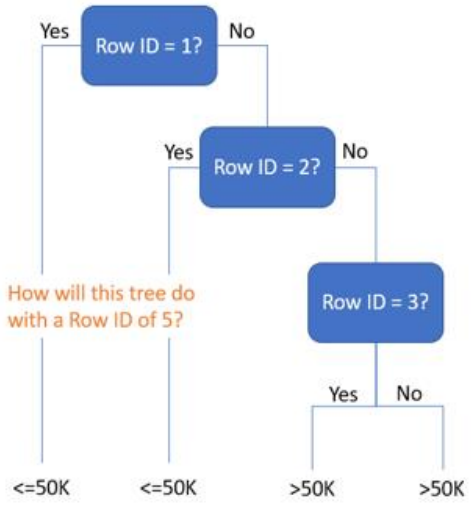
Manufacturing Independence

The mighty random forest manufactures independence via three mechanisms:

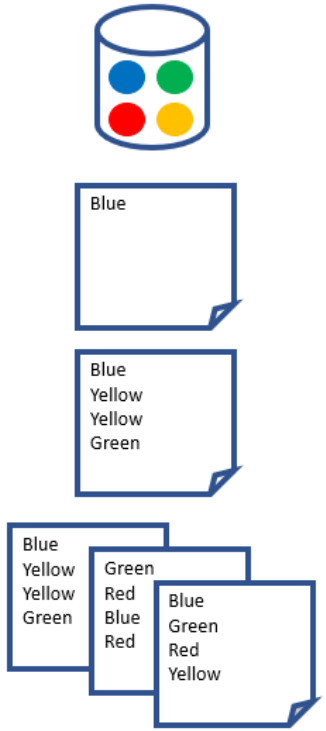
1 – Decision Tree Variance

Data Frame

Row ID	Marital Status	Education Num	Hours Per Week	Age	Label
1	Divorced	11	40	30	<=50K
2	Married-civ-spouse	9	40	46	<=50K
3	Never-married	13	40	26	>50K
4	Divorced	10	58	44	>50K



2 – Bagging



3 – Feature Randomization

Original Data Frame

Marital Status	Education Num	Hours Per Week	Age	Label
Divorced	11	40	30	<=50K
Married-civ-spouse	9	40	46	<=50K
Never-married	13	40	26	>50K
Divorced	10	58	44	>50K

Education Num	Age	Label
11	30	<=50K
9	46	<=50K
13	26	>50K
10	44	>50K

Marital Status	Age	Label
Never-married	26	>50K
Married-civ-spouse	46	<=50K
Never-married	26	>50K
Divorced	44	>50K

Marital Status	Hours Per Week	Label
Divorced	40	<=50K
Married-civ-spouse	40	<=50K
Divorced	40	<=50K
Married-civ-spouse	40	<=50K

Bagging

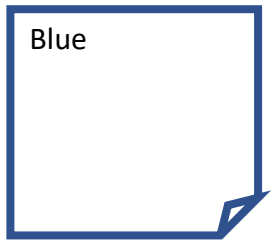
We know that individual decision trees will change their shape depending on the data provided for training (i.e., decision trees have high variance).

Bagging (aka "bootstrap aggregation") is a technique that allows for taking a single dataset and "manufacturing" many different datasets.

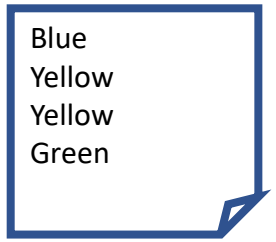
Bagging works via *random sampling with replacement*:

I have a jar with 4 marbles – blue, green red, & yellow.

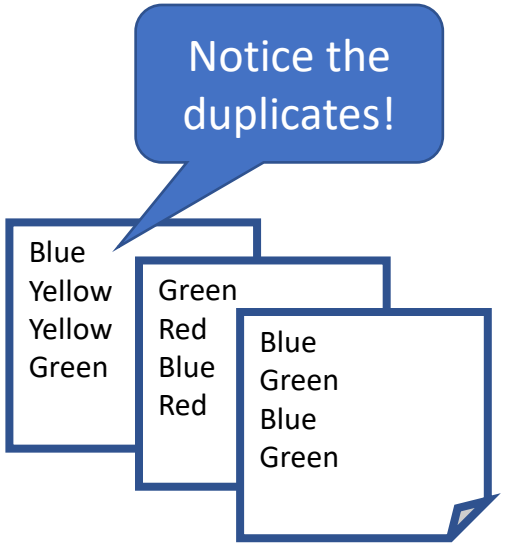
I reach into the jar and draw a marble at random.



I write down the marble I drew and put it back.



I repeat this process, getting a list that's different from the jar.



I can make as many lists as I would like.

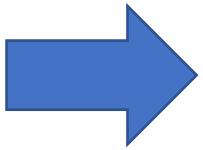
Bagging

The mighty random forest uses bagging to create different data sets to train the trees in the forest. Since the trees have different data, they will be different.

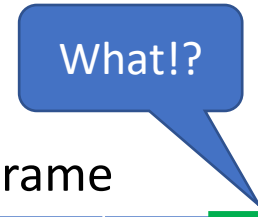
Per the "wisdom of the crowd", we want the trees to be *wildly different* from each other. Bagging helps to achieve that goal:

Original Data Frame

Marital Status	Education Num	Hours Per Week	Age	Label
Divorced	11	40	30	<=50K
Married-civ-spouse	9	40	46	<=50K
Never-married	13	40	26	>50K
Divorced	10	58	44	>50K



"Manufactured" Data Frame



Marital Status	Education Num	Hours Per Week	Age	Label
Divorced	11	40	30	<=50K
Married-civ-spouse	9	40	46	<=50K
Divorced	11	40	30	<=50K
Married-civ-spouse	9	40	46	<=50K