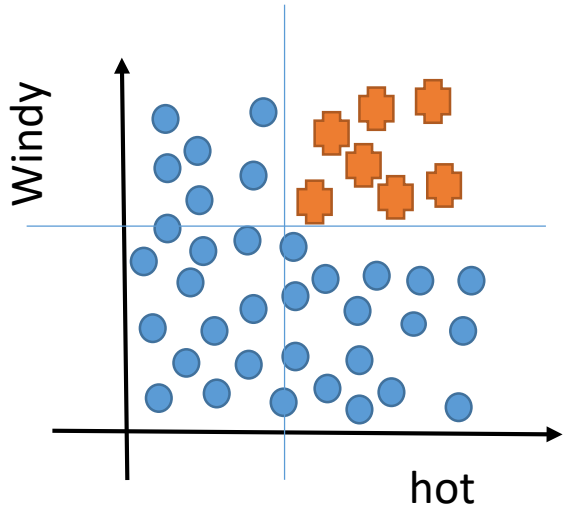
Decision Trees & Random Forests

Decision Tree – Intuition



Is this data linearly separable? – YES OR NO

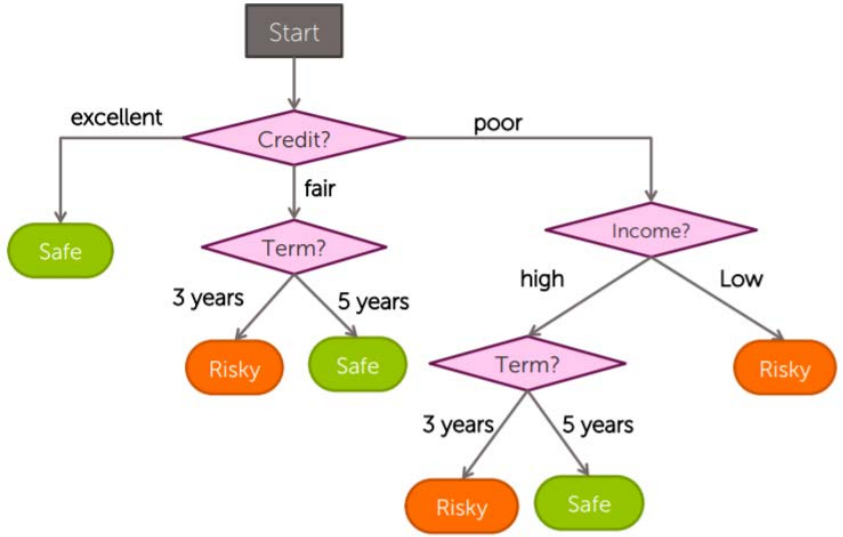
Any threshold value for ‘hot’ which will decide – PLAY OR NOT PLAY

Any threshold value for ‘Windy’ which will decide – PLAY OR NOT PLAY

Decision Tree

- With the given data , I want to predict if I can give the loan or not.

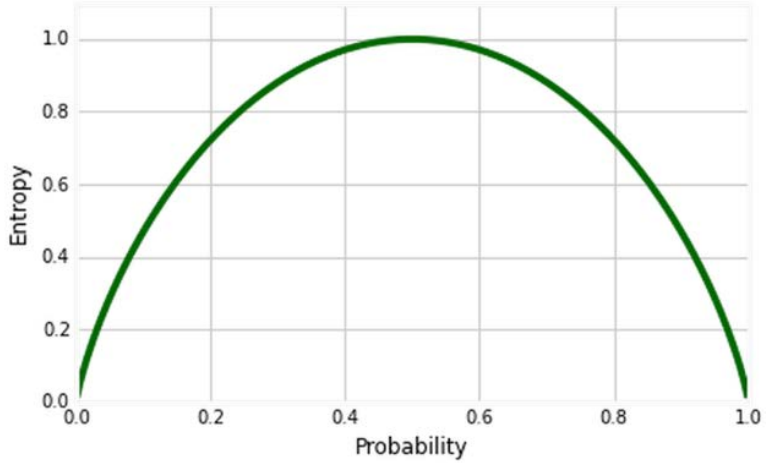
Credit	Term	Income	y
excellent	3 yrs	high	safe
fair	5 yrs	low	risky
fair	3 yrs	high	safe
poor	5 yrs	high	risky
excellent	3 yrs	low	risky
fair	5 yrs	low	safe
poor	3 yrs	high	risky
poor	5 yrs	low	safe
fair	3 yrs	high	safe



Entropy

- DEF : Measure of impurity in a bunch of samples
- Controls how the DT has to split the data
- It has a formula

$$H = -\sum_{i} p_i (\log_2 p_i)$$



Entropy versus Probability of belonging to a class.

- Pi – Fraction of examples in class i
- All examples are **same class** -> entropy = 0
- All examples are **evenly split** between classes -> entropy = 1.0

Calculate Entropy

Credit	Term	Income	y
excellent	3 yrs	high	safe
fair	5 yrs	low	risky
fair	3 yrs	high	safe
poor	5 yrs	high	risky
excellent	3 yrs	low	risky
fair	5 yrs	low	safe
poor	3 yrs	high	risky
poor	5 yrs	low	safe
fair	3 yrs	high	safe

In python:

```
import math
-0.5*math.log(0.5,2) - -0.5*math.log(0.5,2)
```

Lets calculate the entropy of y!!

S R S R R S R S S

How many safe = 5 P s = 5/9
How many risky = 4 P r = 4/9
Total = 9

Entropy = 5/9(log 5/9) + 4/9(log 4/9) = 0.99

Information Gain

- Which feature we need to choose to split on??
- To see entropy helps us in creating trees
- Decision Trees decides to split on feature with maximum Information Gain
- Information Gain = Entropy (Parent) – [Weighted Average] Entropy(Children)

Credit	Term	Income	y
excellent	3 yrs	high	safe
fair	5 yrs	low	risky
fair	3 yrs	high	safe
poor	5 yrs	high	risky
excellent	3 yrs	low	risky
fair	5 yrs	low	safe
poor	3 yrs	high	risky
poor	5 yrs	low	safe
fair	3 yrs	high	safe

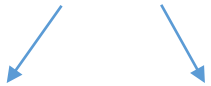
Entropy(Parent) = 5/9(log 5/9) + 4/9(log 4/9) = 0.99

We need to choose between Credit , Term , Income…

Calculate Information Gain

We need to choose between Credit , Term , Income…

Start with Income..



High	Low
SSRRS	RRSS
P safe = 3/5	P safe = 2/4
P risky = 2/5	P risky = 2/4
Entropy = 0.808	Entropy = 1

[Weighted Average] Entropy(Income)

= 5/9 * 0.808 + 4/9 * 1

= 0.89

Credit	Term	Income	y
excellent	3 yrs	high	safe
fair	5 yrs	low	risky
fair	3 yrs	high	safe
poor	5 yrs	high	risky
excellent	3 yrs	low	risky
fair	5 yrs	low	safe
poor	3 yrs	high	risky
poor	5 yrs	low	safe
fair	3 yrs	high	safe

Information Gain (Income) = 0.99 + 0.89 = 1.88

Calculate Information Gain

grade	bumpiness	speed limit?	speed
steep	bumpy	yes	slow
steep	smooth	yes	slow
flat	bumpy	no	fast
steep	smooth	no	fast

Information Gain (Grade) = 1 –(¾(0.9184) + ¼(0)) = 0.3112

Information Gain (Bumpiness) = 1 – (½(1) + ½(1)) = 0

Information Gain (Bumpiness) = 1 – (½(0) + ½(0)) = 1

Pruning

- Pruning helps us to avoid overfitting
- Generally it is preferred to have a simple model, it avoids overfitting issue
- Any additional split that does not add significant value is not worth while.
- We can avoid overfitting by changing the parameters like
  - max_leaf_nodes
  - min_samples_leaf
  - max_depth

Pruning Parameters

- max_leaf_nodes
  - Reduce the number of leaf nodes
- min_samples_leaf
  - Restrict the size of sample leaf
  - Minimum sample size in terminal nodes can be fixed to 30, 100, 300 or 5% of total
- max_depth
  - Reduce the depth of the tree to build a generalized tree
  - Set the depth of the tree to 3, 5, 10 depending after verification on test data

Let’s build a Quick Decision tree

Story – DT and RF

- Dan is at a library – Decides to read a book
- Sam – A friend helps him to find one

- Sam – Asks few question, Dan ans' Yes or No
- Who is the author? What genre? Etc

- Dan – Decision Tree here ☺

Story – DT and RF

- Now only Dan is deciding the book – Overfitting !
- So we ask some other friends too for help




- They each gave a vote on the book , you decide on majority option
- ENSEMBLE Classifiers ☺

Story – DT and RF

- Now if we have similar circle of friends ,you want avoid them having same answer.
- So , you will give them a different sample from your list of books.
- By cutting the list and placing in a bag , and randomly draw from the bag, tell your friend whether or not you enjoyed that book.

- Place the sample back in the bag

Bootstrapped Aggregated Forest

- You'll be randomly drawing a sub sample from your original list with replacement (bootstrapping your original data).
- This gives some books more emphasis, if you drew a particular book several times for one friend, and some books less, possibly never drawn from the bag.

- Then each individual will give a unique recommendation on your book preferences.

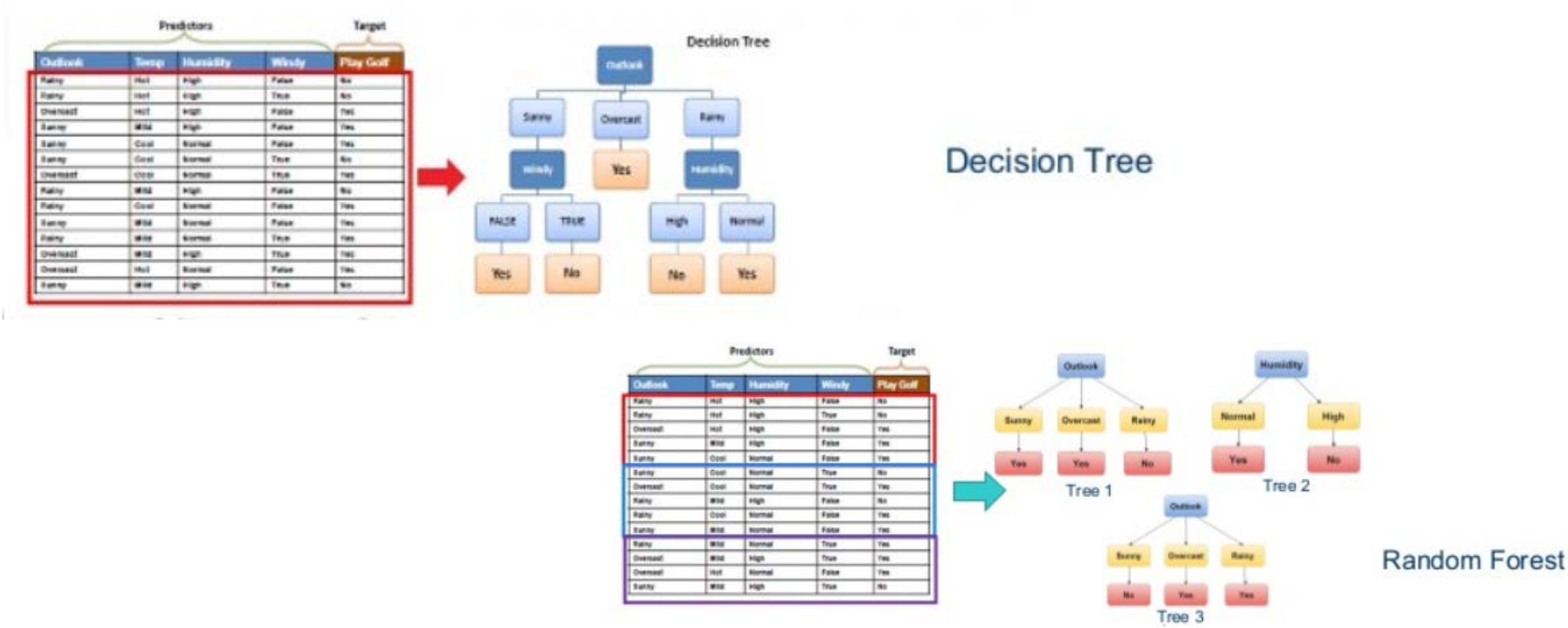
Random Forest – BLACK BOX MODEL

- Last Issue , suppose you enjoyed Harry Potter 1 and Harry Potter 2
- Same author, genre etc
- Dan, your friend, came to a conclusion that you like J.K.Rowling
- But maybe you liked the Genre

- To fix this – We need randomness in the questions

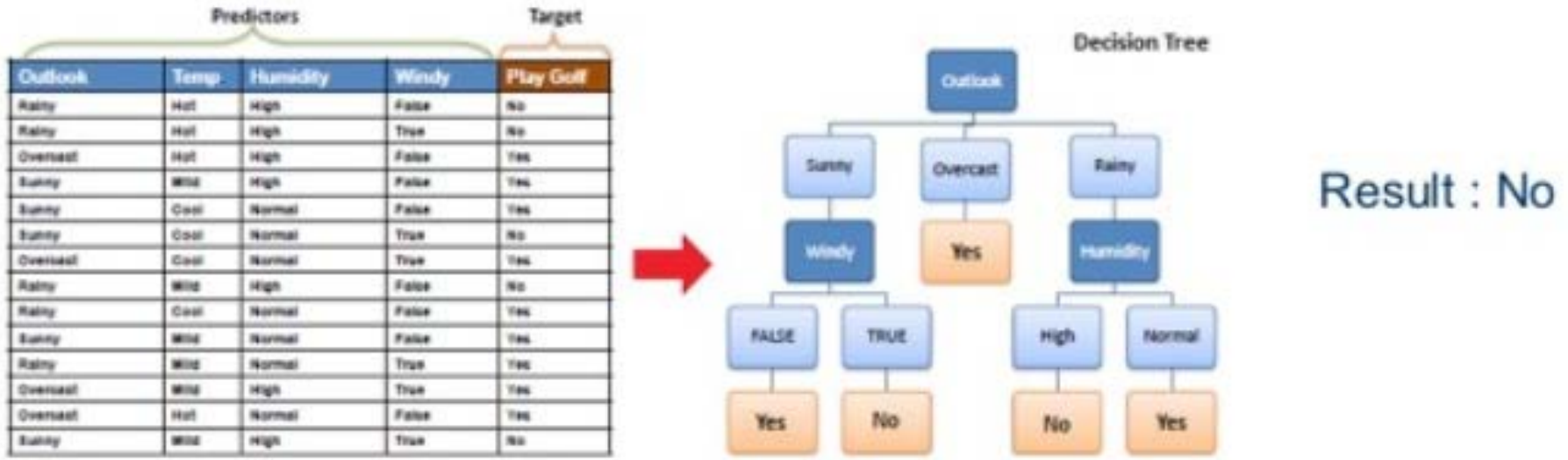
- Analogy ---- How your RANDOM FOREST works!

Random Forest



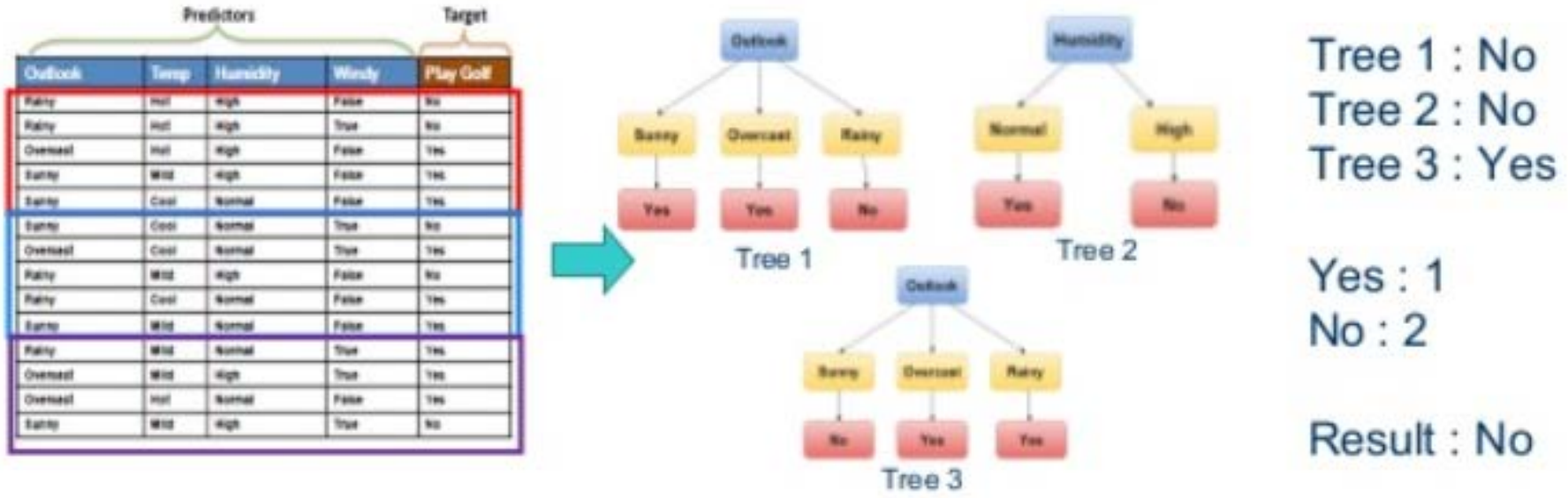
DT vs RF

Outlook	Temp.	Humidity	Windy	Play Golf
Rainy	Mild	High	False	?

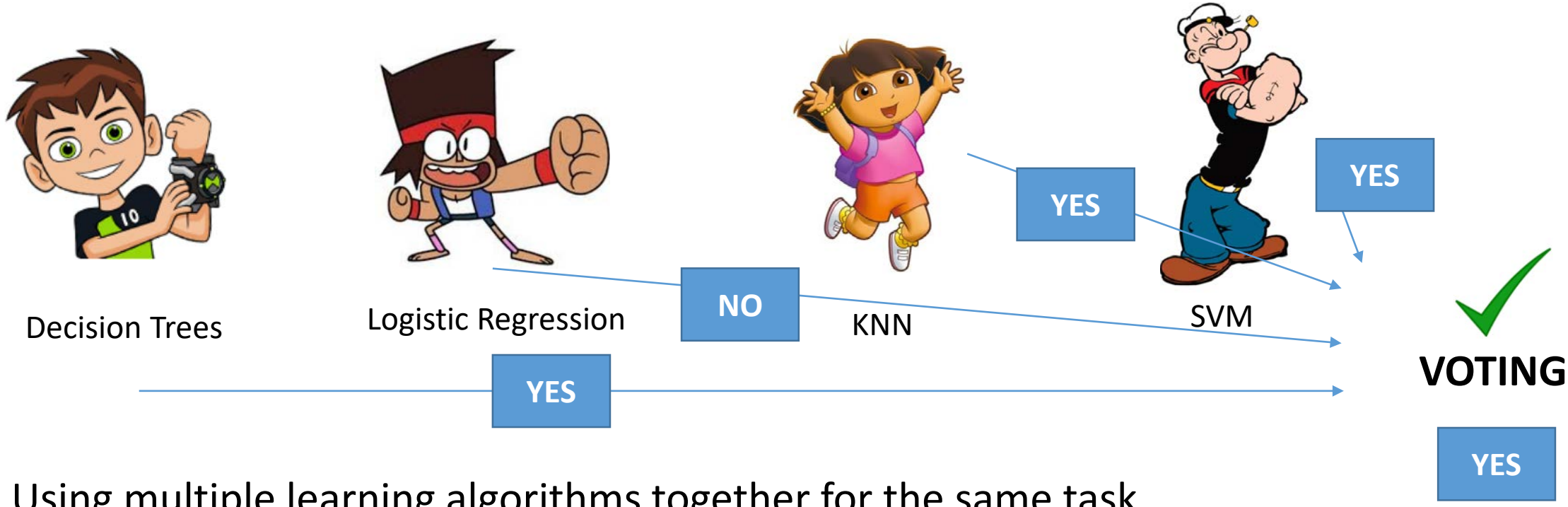


DT vs RF

Outlook	Temp.	Humidity	Windy	Play Golf
Rainy	Mild	High	False	?



Ensemble Learners - Terminology



Using multiple learning algorithms together for the same task .

Better predictions than individual learning models

Higher consistency (Avoids Overfitting)