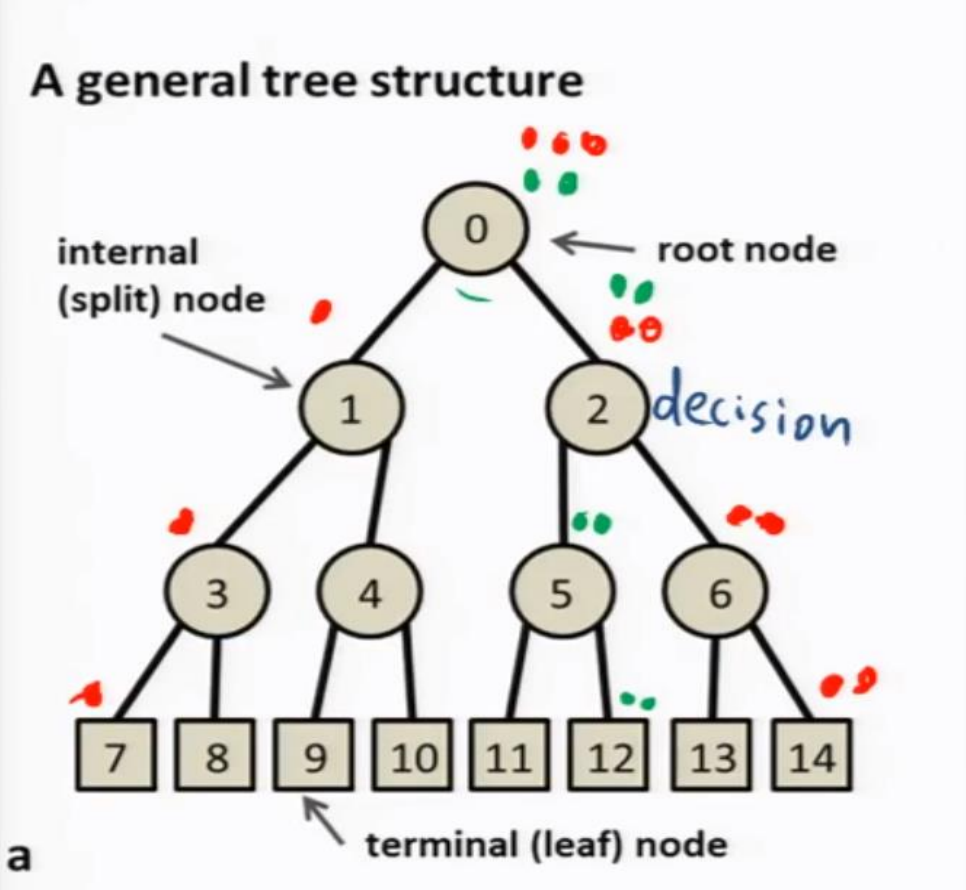
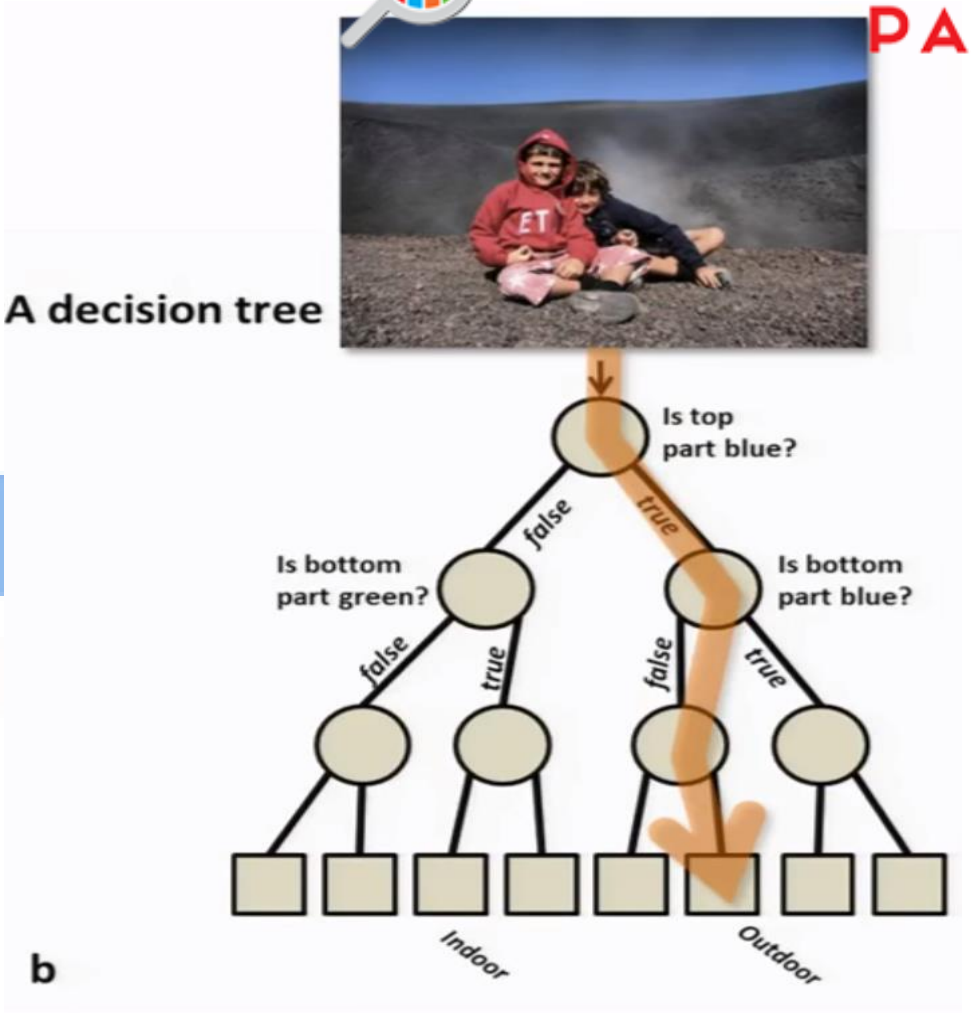
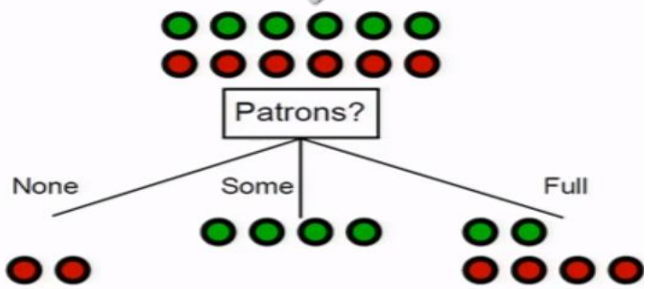
Decision Trees





Plot No. 28, 4th Floor, Suraj Trade Center,**Opp. Cyber Towers,** Hitech City, Hyderabad - 500081, Telangana.
India Tel: 040 - 66828899, Mob:+91 7842828899, Email: info@analyticspath.com

From a spreadsheet to a decision node



Examples described by attribute values (Boolean, discrete, continuous, etc.)
E.g., situations where I will/won't wait for a table:

Example	Attributes										Target
	Alt	*Bar*	*Fri*	*Hun*	*Pat*	*Price*	*Rain*	*Res*	*Type*	*Est*	*WillWait*
X_1	*T*	*F*	*F*	*T*	*Some*	*$$$*	*F*	*T*	*French*	*0–10*	*T*
X_2	*T*	*F*	*F*	*T*	*Full*	*$*	*F*	*F*	*Thai*	*30–60*	*F*
X_3	*F*	*T*	*F*	*F*	*Some*	*$*	*F*	*F*	*Burger*	*0–10*	*T*
X_4	*T*	*F*	*T*	*T*	*Full*	*$*	*F*	*F*	*Thai*	*10–30*	*T*
X_5	*T*	*F*	*T*	*F*	*Full*	*$$$*	*F*	*T*	*French*	*>60*	*F*
X_6	*F*	*T*	*F*	*T*	*Some*	*$$*	*T*	*T*	*Italian*	*0–10*	*T*
X_7	*F*	*T*	*F*	*F*	*None*	*$*	*T*	*F*	*Burger*	*0–10*	*F*
X_8	*F*	*F*	*F*	*T*	*Some*	*$$*	*T*	*T*	*Thai*	*0–10*	*T*
X_9	*F*	*T*	*T*	*F*	*Full*	*$*	*T*	*F*	*Burger*	*>60*	*F*
X_{10}	*T*	*T*	*T*	*T*	*Full*	*$$$*	*F*	*T*	*Italian*	*10–30*	*F*
X_{11}	*F*	*F*	*F*	*F*	*None*	*$*	*F*	*F*	*Thai*	*0–10*	*F*
X_{12}	*T*	*T*	*T*	*T*	*Full*	*$*	*F*	*F*	*Burger*	*30–60*	*T*

Classification of examples is positive (T) or negative (F)

How do we construct the tree ?
i.e., how to pick attribute (nodes)?

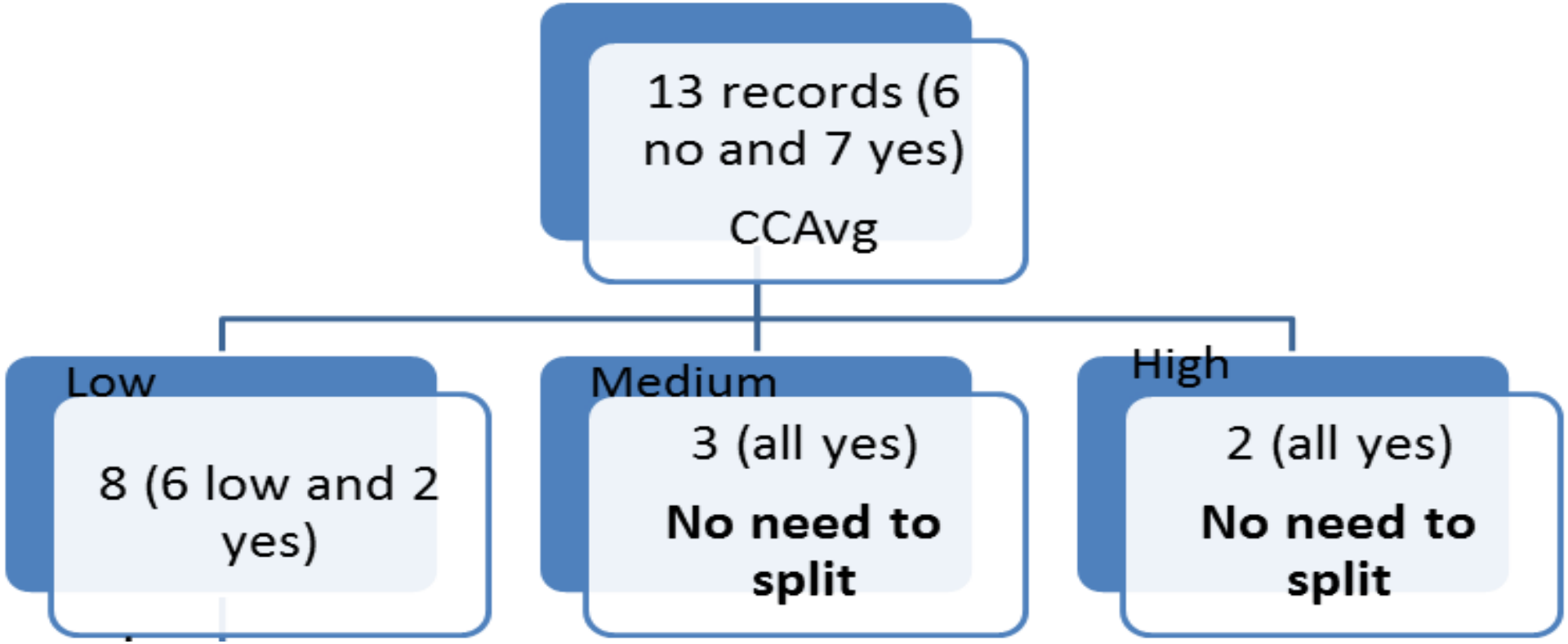
Idea: a good attribute splits the examples into subsets that are (ideally) "all positive" or "all negative"

Patrons? is a better choice—gives **information** about the classification

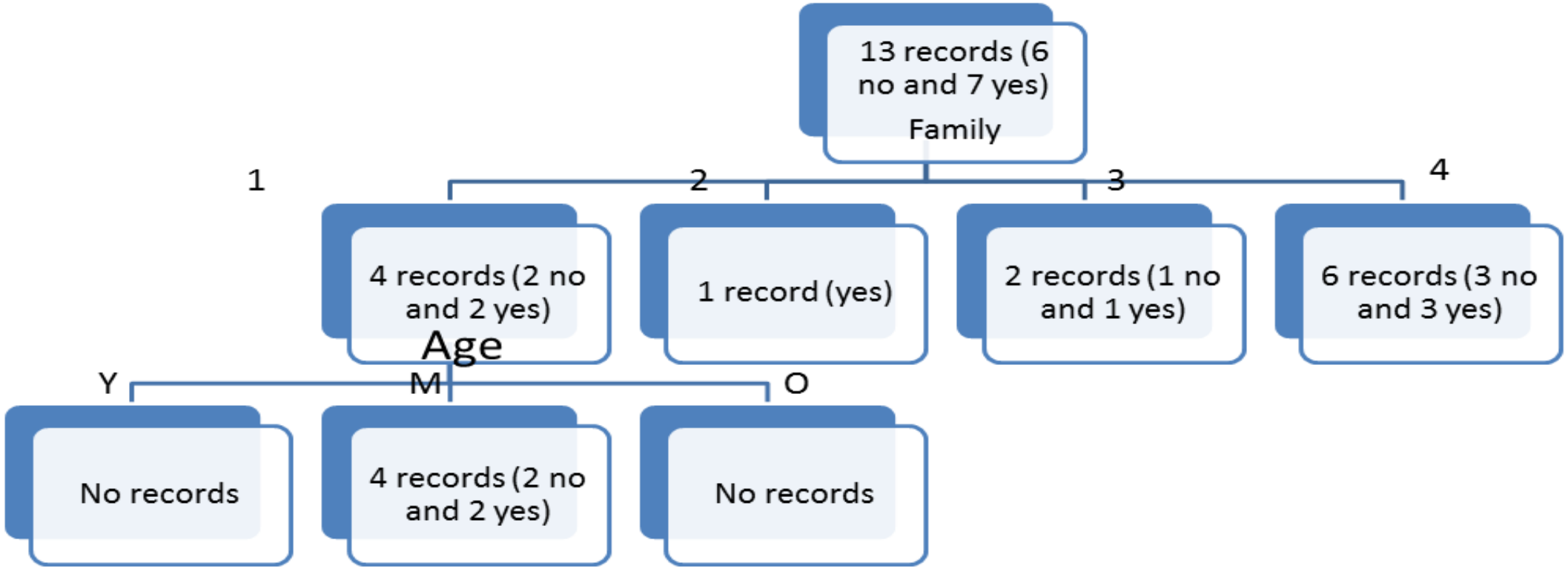
Decision Trees

ID	Age	Income	Family	CCAvg	Personal Loan
1	Young	Low	4	Low	0
2	Old	Low	3	Low	0
3	Middle	Low	1	Low	0
4	Middle	Medium	1	Low	0
5	Middle	Low	4	Low	0
6	Middle	Low	4	Low	0
10	Middle	High	1	High	1
17	Middle	Medium	4	Medium	1
19	Old	High	2	High	1
30	Middle	Medium	1	Medium	1
39	Old	Medium	3	Medium	1
43	Young	Medium	4	Low	1
48	Middle	High	4	Low	1

Decision Trees

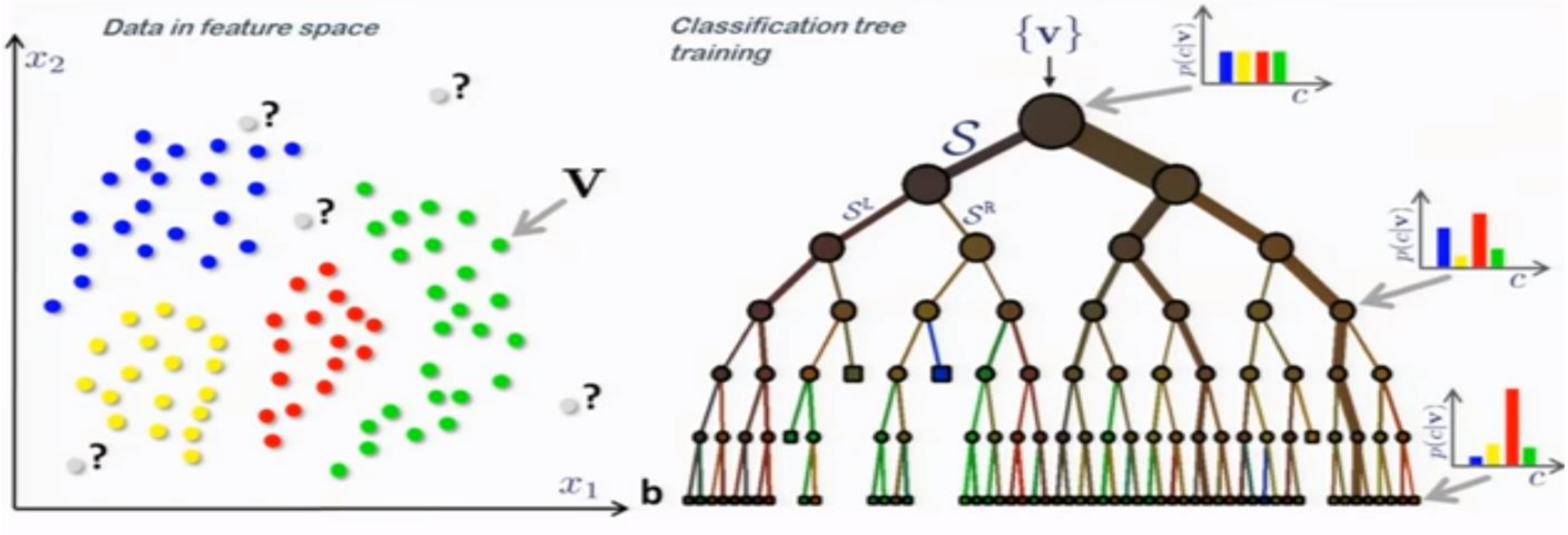


Decision Trees



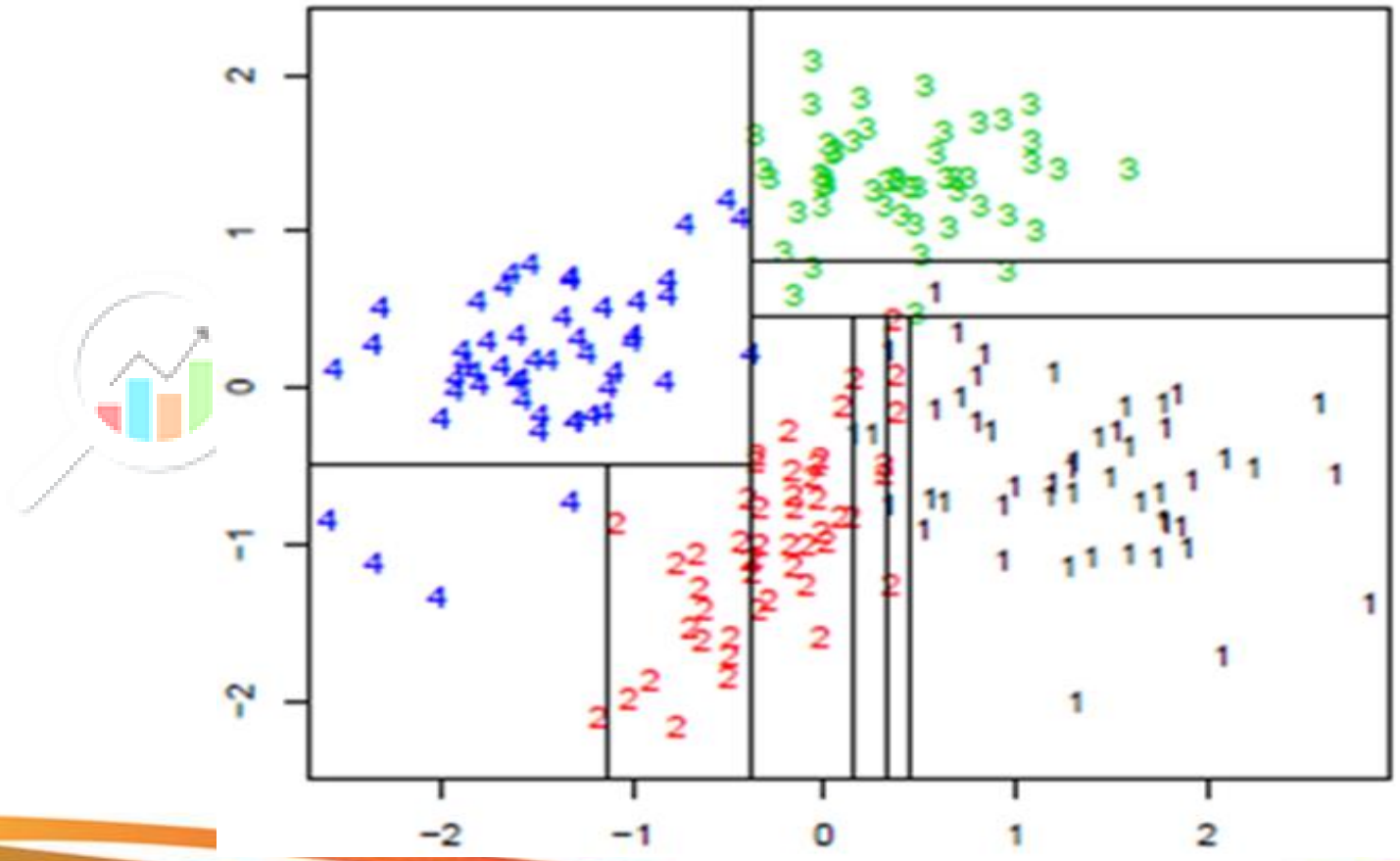
Plot No. 28, 4th Floor, Suraj Trade Center,**Opp. Cyber Towers,** Hitech City, Hyderabad - 500081, Telangana.
India Tel: 040 - 66828899, Mob:+91 7842828899, Email: info@analyticspath.com

Classification tree



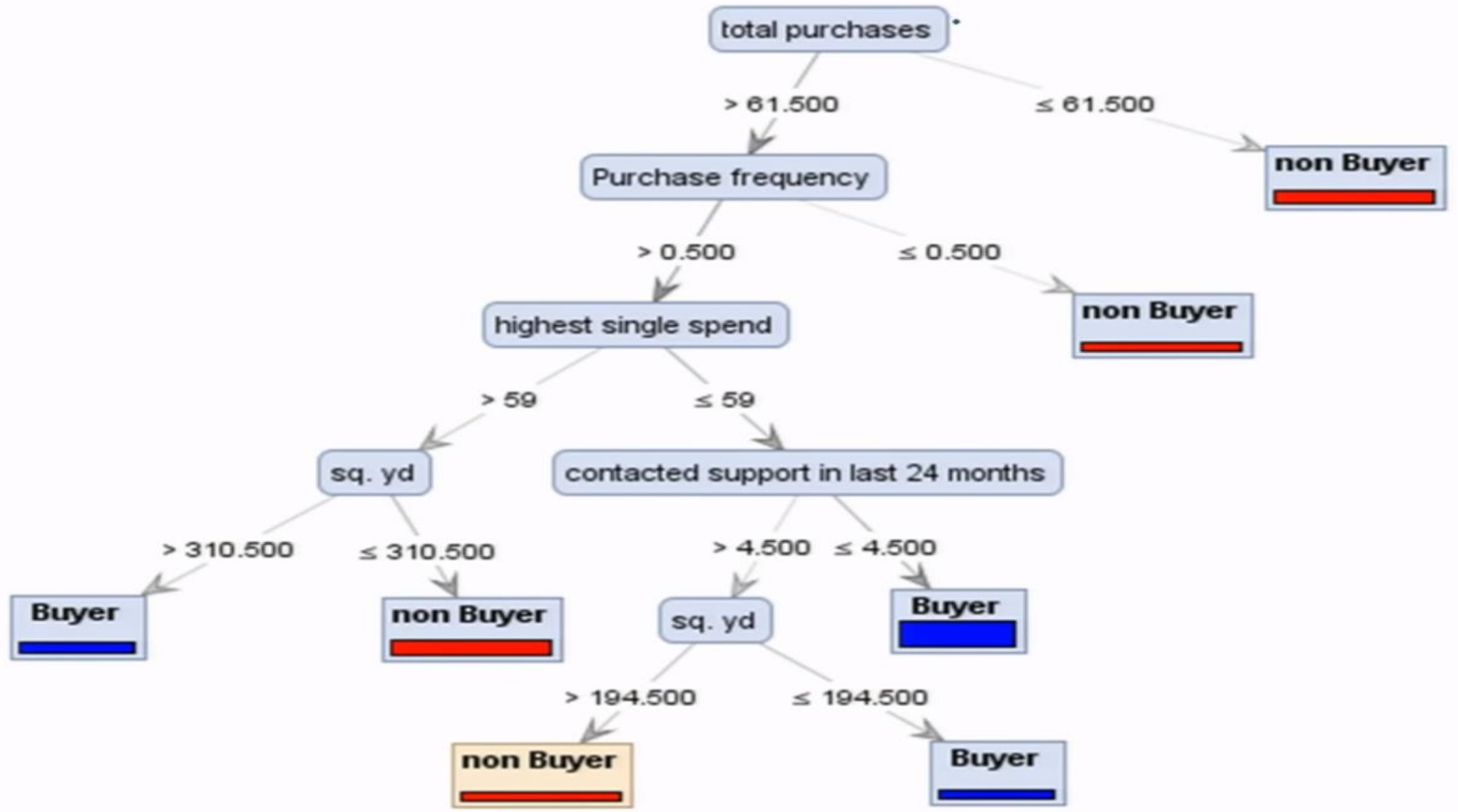
Plot No. 28, 4th Floor, Suraj Trade Center,**Opp. Cyber Towers,** Hitech City, Hyderabad - 500081, Telangana.
India Tel: 040 - 66828899, Mob:+91 7842828899, Email: info@analyticspath.com

Axis Aligned Splits



Plot No. 28, 4th Floor, Suraj Trade Center,**Opp. Cyber Towers,** Hitech City, Hyderabad - 500081, Telangana.
India Tel: 040 - 66828899, Mob:+91 7842828899, Email: info@analyticspath.com

Another commerce example

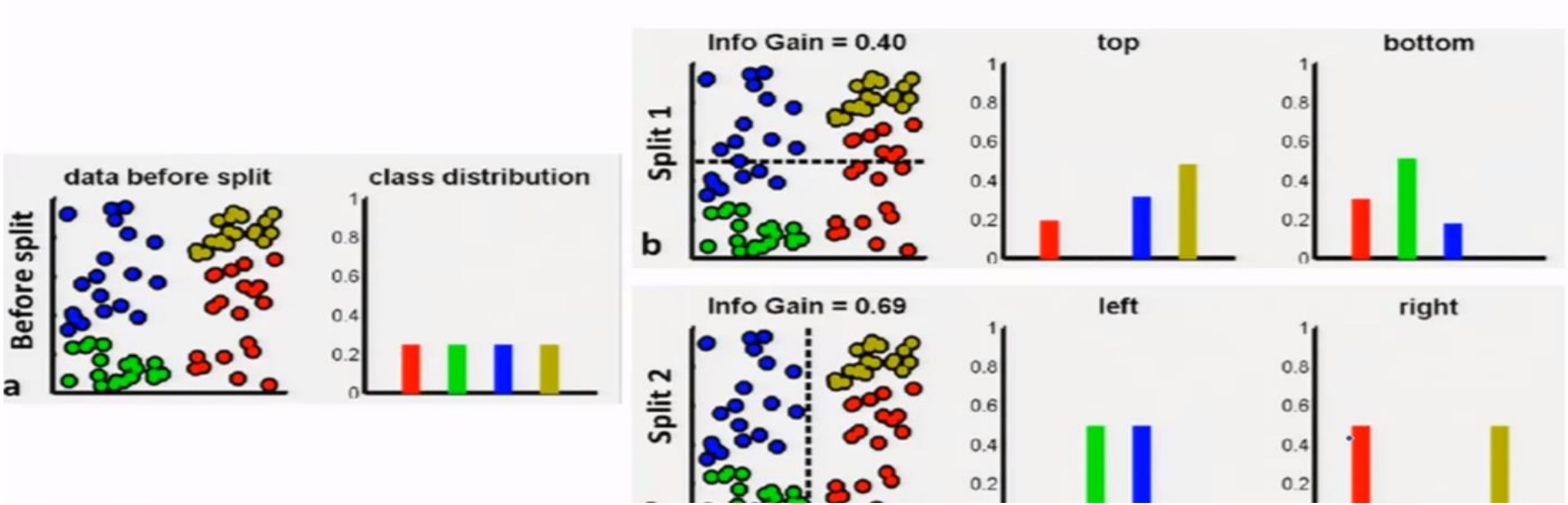


Entropy and Information gain

$$H = -\sum_i p_i \log_2 p_i$$

- Information gain = Entropy of the system before split – Entropy of the system after split

Using Information gain to Split



```
     ends-vowel
      [9m,5f]          <--- the [..,..] notation represents the class
     /       \              distribution of instances that reached a node
   =1         =0
 -------    -------
 [3m,4f]    [6m,1f]
```

ANALYTICS PATH

As you can see, before the split we had 9 males and 5 females, i.e. `P(m)=9/14` and `P(f)=5/14`. According to the definition of entropy:

```
Entropy_before = - (5/14)*log2(5/14) - (9/14)*log2(9/14) = 0.9403
```

Next we compare it with the entropy computed after considering the split by looking at two child branches. In the left branch of `ends-vowel=1`, we have:

```
Entropy_left = - (3/7)*log2(3/7) - (4/7)*log2(4/7) = 0.9852
```

and the right branch of `ends-vowel=0`, we have:

```
Entropy_right = - (6/7)*log2(6/7) - (1/7)*log2(1/7) = 0.5917
```

We combine the left/right entropies using the number of instances down each branch as weight factor (7 instances went left, and 7 instances went right), and get the final entropy after the split:

```
Entropy_after = 7/14*Entropy_left + 7/14*Entropy_right = 0.7885
```

Now by comparing the entropy before and after the split, we obtain a measure of **information gain**, or how much information we gained by doing the split using that particular feature:

```
Information_Gain = Entropy_before - Entropy_after = 0.1518
```

Gini Index

$$1 - \sum_{1}^{m} p_i^2$$

- It is computed on binary splits only.

- So, if we take ccAvg (low, medium and high), it considers all binary options
- {Low}, {medium, high} or {medium}, {low, high} etc.

Advantages

- Explicability
- They are fast
- Robust
- Requires very little experimentation
- You may also build some intuitions about your customer base. E.g. “Are customers with different family sizes truly different?

Can we use a decision tree only for classification or can we use them for predicting a numeric attribute?

Regression Trees

- It turns out that, we are collecting very similar records at each leaf. So, we can use median or mean of the records at a leaf as the predictor value for all the new records that obey similar conditions.
- Such trees are called regression trees.

Two most popular decision tree algorithms

- CART (Classification and Regression Trees)
  - Binary split
  - Gini index
- C5.0
  - Multi split
  - Info gain

Overfitting in Decision Trees

How do we understand and over come the demon of Overfitting in Decision Trees

Plot No. 28, 4th Floor, Suraj Trade Center,**Opp. Cyber Towers,** Hitech City, Hyderabad - 500081, Telangana.
India Tel: 040 - 66828899, Mob:+91 7842828899, Email: info@analyticspath.com