Learning Agents

Chapter 18

Supervised Learning and Decision Trees

Jim Rehg
Georgia Tech

Example of Decision Tree Learning

Example

Learn decision tree from dataset D:

Example	A_1	A_2	A_3	A_4	A_5	Output
1	1	1	0	0	0	1
2	0	0	1	1	0	1
3	1	0	0	1	0	1
4	1	0	1	0	1	1
5	0	1	0	0	0	0
6	1	1	0	1	1	0
7	0	1	1	1	1	0

How does the resulting tree classify:

8	1	1	1	0	1	???

Example	A_1	A_2	A_3	A_4	A_5	Output
1	1	1	0	0	0	1
2	0	0	1	1	0	1
3	1	0	0	1	0	1
4	1	0	1	0	1	1
5	0	1	0	0	0	0
6	1	1	0	1	1	0
7	0	1	1	1	1	0

$$H(P(y \mid D_0)) = B(\tfrac{4}{4+3})$$

$$= 0.985$$

Example	A_1	A_2	A_3	A_4	A_5	Output
1	1	1	0	0	0	1
2	0	0	1	1	0	1
3	1	0	0	1	0	1
4	1	0	1	0	1	1
5	0	1	0	0	0	0
6	1	1	0	1	1	0
7	0	1	1	1	1	0

H(Goal)=0.985

Split on A_1:

$$H(y \mid A_1) = \tfrac{4}{7}\, B\left(\frac{3}{3+1}\right)$$

H(Goal)=0.985

Example	A_1	A_2	A_3	A_4	A_5	Output
1	1	1	0	0	0	1
2	0	0	1	1	0	1
3	1	0	0	1	0	1
4	1	0	1	0	1	1
5	0	1	0	0	0	0
6	1	1	0	1	1	0
7	0	1	1	1	1	0

Split on A_1:

$$H(y \mid A_1) = 4/7\, B\left(\frac{3}{3+1}\right) + 3/7\, B\left(\frac{1}{1+2}\right)$$

Example	A_1	A_2	A_3	A_4	A_5	Output
1	1	1	0	0	0	1
2	0	0	1	1	0	1
3	1	0	0	1	0	1
4	1	0	1	0	1	1
5	0	1	0	0	0	0
6	1	1	0	1	1	0
7	0	1	1	1	1	0

H(Goal)=0.985
$H(y|A_1) = 0.857$

Split on A_1:

$$H(y \mid A_1) = {}^{4}\!/\!_{7}\, B\left(\tfrac{3}{3+1}\right) + {}^{3}\!/\!_{7}\, B\left(\tfrac{1}{1+2}\right)$$

$$= {}^{4}\!/\!_{7}(0.811) + {}^{3}\!/\!_{7}(0.918)$$

$$= 0.857$$

Example	A_1	A_2	A_3	A_4	A_5	Output
1	1	1	0	0	0	1
2	0	0	1	1	0	1
3	1	0	0	1	0	1
4	1	0	1	0	1	1
5	0	1	0	0	0	0
6	1	1	0	1	1	0
7	0	1	1	1	1	0

H(Goal)=0.985

$H(y|A_1)$ =0.857

Split on A_2:

$$H(y \mid A_2) = 4/7\, B(\tfrac{1}{1+3})$$

Example	A_1	A_2	A_3	A_4	A_5	Output
1	1	1	0	0	0	1
2	0	0	1	1	0	1
3	1	0	0	1	0	1
4	1	0	1	0	1	1
5	0	1	0	0	0	0
6	1	1	0	1	1	0
7	0	1	1	1	1	0

H(Goal)=0.985

$H(y|A_1) = 0.857$

Split on A_2:

$$H(y \mid A_2) = \tfrac{4}{7} B\left(\frac{1}{1+3}\right) + \tfrac{3}{7} B\left(\frac{3}{3+0}\right)$$

Example	A_1	A_2	A_3	A_4	A_5	Output
1	1	1	0	0	0	1
2	0	0	1	1	0	1
3	1	0	0	1	0	1
4	1	0	1	0	1	1
5	0	1	0	0	0	0
6	1	1	0	1	1	0
7	0	1	1	1	1	0

H(Goal)=0.985
$H(y|A_1)$ =0.857
$H(y|A_2)$=0.463

Split on A_2:

$$H(y \mid A_2) = \tfrac{4}{7} B\left(\frac{1}{1+3}\right) + \tfrac{3}{7} B\left(\frac{3}{3+0}\right)$$

$$= \tfrac{4}{7}(0.811) + \tfrac{3}{7}(0)$$

$$= 0.463$$

Example	A_1	A_2	A_3	A_4	A_5	Output
1	1	1	0	0	0	1
2	0	0	1	1	0	1
3	1	0	0	1	0	1
4	1	0	1	0	1	1
5	0	1	0	0	0	0
6	1	1	0	1	1	0
7	0	1	1	1	1	0

H(Goal)=0.985
$H(y|A_1)$ =0.857
$H(y|A_2)$=0.463

Split on A_3:

$$\text{Remainder}(A_3) = {}^{3}\!/_{7}\, B\left(\frac{2}{2+1}\right)$$

Example	A_1	A_2	A_3	A_4	A_5	Output
1	1	1	0	0	0	1
2	0	0	1	1	0	1
3	1	0	0	1	0	1
4	1	0	1	0	1	1
5	0	1	0	0	0	0
6	1	1	0	1	1	0
7	0	1	1	1	1	0

H(Goal)=0.985
$H(y|A_1)$ =0.857
$H(y|A_2)$=0.463
$H(y|A_3)$=0.965

Split on A_3:

$$\text{Remainder}(A_3) = 3/7\, B\left(\frac{2}{2+1}\right) + 4/7\, B\left(\frac{2}{2+2}\right)$$

$$= 0.965$$

Example	A_1	A_2	A_3	A_4	A_5	Output
1	1	1	0	0	0	1
2	0	0	1	1	0	1
3	1	0	0	1	0	1
4	1	0	1	0	1	1
5	0	1	0	0	0	0
6	1	1	0	1	1	0
7	0	1	1	1	1	0

H(Goal)=0.985
$H(y|A_1)$ =0.857
$H(y|A_2)$=0.463
$H(y|A_3)$=0.965

Split on A_4:

$$\text{Remainder}(A_4) = 4/7\, B\left(\frac{2}{2+2}\right)$$

Example	A_1	A_2	A_3	A_4	A_5	Output
1	1	1	0	0	0	1
2	0	0	1	1	0	1
3	1	0	0	1	0	1
4	1	0	1	0	1	1
5	0	1	0	0	0	0
6	1	1	0	1	1	0
7	0	1	1	1	1	0

H(Goal)=0.985
$H(y|A_1)$ =0.857
$H(y|A_2)$=0.463
$H(y|A_3)$=0.965
$H(y|A_4)$=0.965

Split on A_4:

$$\text{Remainder}(A_4) = \tfrac{4}{7}\, B\left(\frac{2}{2+2}\right) + \tfrac{3}{7}\, B\left(\frac{2}{2+1}\right)$$

$$= 0.965$$

Example	A_1	A_2	A_3	A_4	A_5	Output
1	1	1	0	0	0	1
2	0	0	1	1	0	1
3	1	0	0	1	0	1
4	1	0	1	0	1	1
5	0	1	0	0	0	0
6	1	1	0	1	1	0
7	0	1	1	1	1	0

H(Goal)=0.985
$H(y|A_1)$ =0.857
$H(y|A_2)$=0.463
$H(y|A_3)$=0.965
$H(y|A_4)$=0.965

Split on A_5:

$$H(y \mid A_5) = 3/7\, B\left(\frac{1}{1+2}\right)$$

Example	A_1	A_2	A_3	A_4	A_5	Output
1	1	1	0	0	0	1
2	0	0	1	1	0	1
3	1	0	0	1	0	1
4	1	0	1	0	1	1
5	0	1	0	0	0	0
6	1	1	0	1	1	0
7	0	1	1	1	1	0

H(Goal)=0.985
$H(y|A_1)$ =0.857
$H(y|A_2)$=0.463
$H(y|A_3)$=0.965
$H(y|A_4)$=0.965
$H(y|A_5)$=0.857

Split on A_5:

$$H(y \mid A_5) = 3/7\, B(\tfrac{1}{1+2}) + 4/7\, B(\tfrac{3}{3+1})$$
$$= 0.857$$

Example	A_1	A_2	A_3	A_4	A_5	Output
1	1	1	0	0	0	1
2	0	0	1	1	0	1
3	1	0	0	1	0	1
4	1	0	1	0	1	1
5	0	1	0	0	0	0
6	1	1	0	1	1	0
7	0	1	1	1	1	0

H(Goal)=0.985
$H(y|A_1)$ =0.857
$H(y|A_2)$=0.463
$H(y|A_3)$=0.965
$H(y|A_4)$=0.965
$H(y|A_5)$=0.857

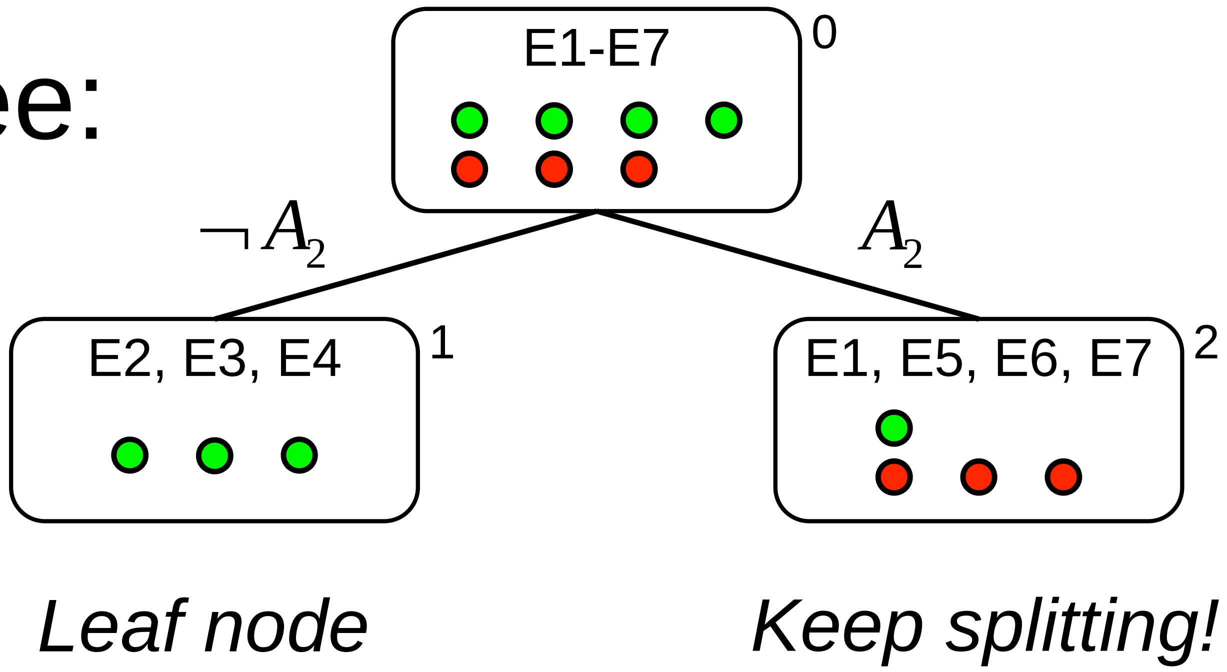
Now identify the best attribute:

$\text{Gain}(A_1) = 0.985 - 0.857 = 0.128$
$\boxed{\text{Gain}(A_2) = 0.985 - 0.463 = 0.522}$
$\text{Gain}(A_3) = 0.985 - 0.965 = 0.020$
$\text{Gain}(A_4) = 0.985 - 0.965 = 0.020$
$\text{Gain}(A_5) = 0.985 - 0.857 = 0.128$

Attribute 2 is the best!

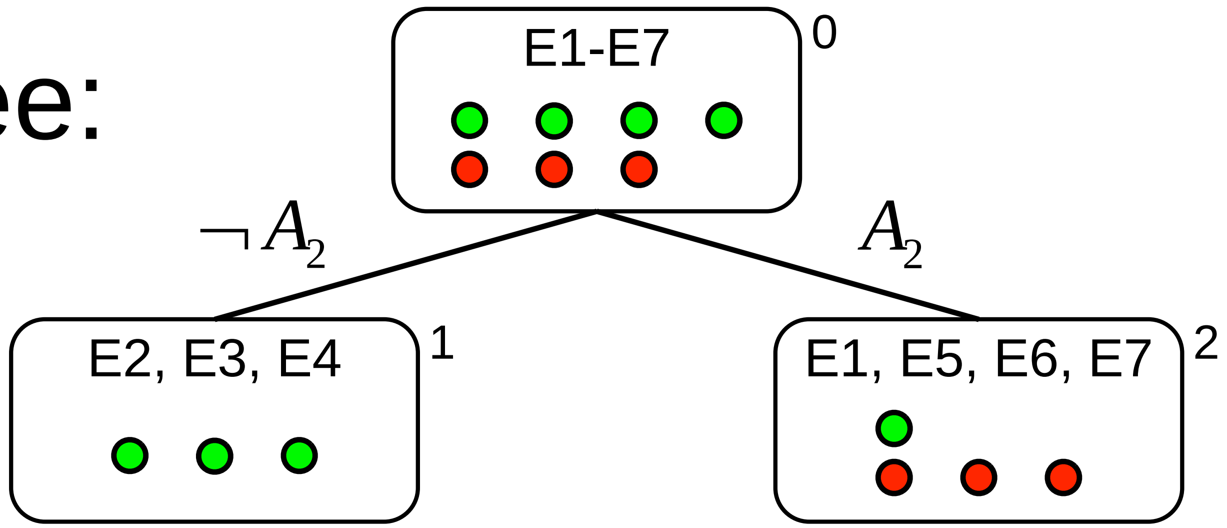
Example	A_1	A_2	A_3	A_4	A_5	Output
1	1	1	0	0	0	1
2	0	0	1	1	0	1
3	1	0	0	1	0	1
4	1	0	1	0	1	1
5	0	1	0	0	0	0
6	1	1	0	1	1	0
7	0	1	1	1	1	0

Decision tree:



Example	A_1	A_2	A_3	A_4	A_5	Output
1	1	1	0	0	0	1
2	0	0	1	1	0	1
3	1	0	0	1	0	1
4	1	0	1	0	1	1
5	0	1	0	0	0	0
6	1	1	0	1	1	0
7	0	1	1	1	1	0

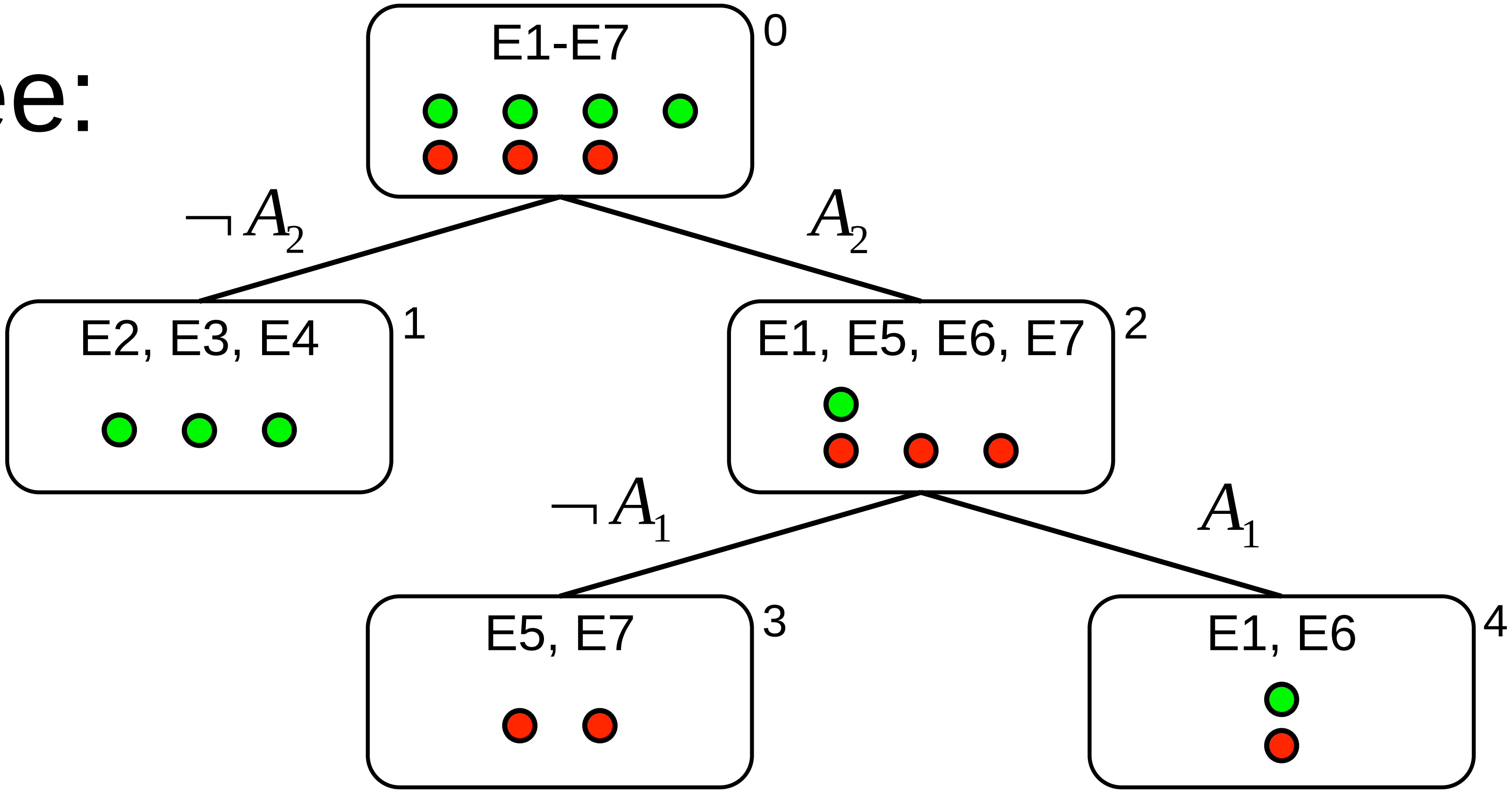
Decision tree:



Attributes A1, A4, A5 equally good.
Produce homogeneous set of 2 0's.
Choose by attribute no. to break tie.

Example	A_1	A_2	A_3	A_4	A_5	Output
1	1	1	0	0	0	1
2	0	0	1	1	0	1
3	1	0	0	1	0	1
4	1	0	1	0	1	1
5	0	1	0	0	0	0
6	1	1	0	1	1	0
7	0	1	1	1	1	0

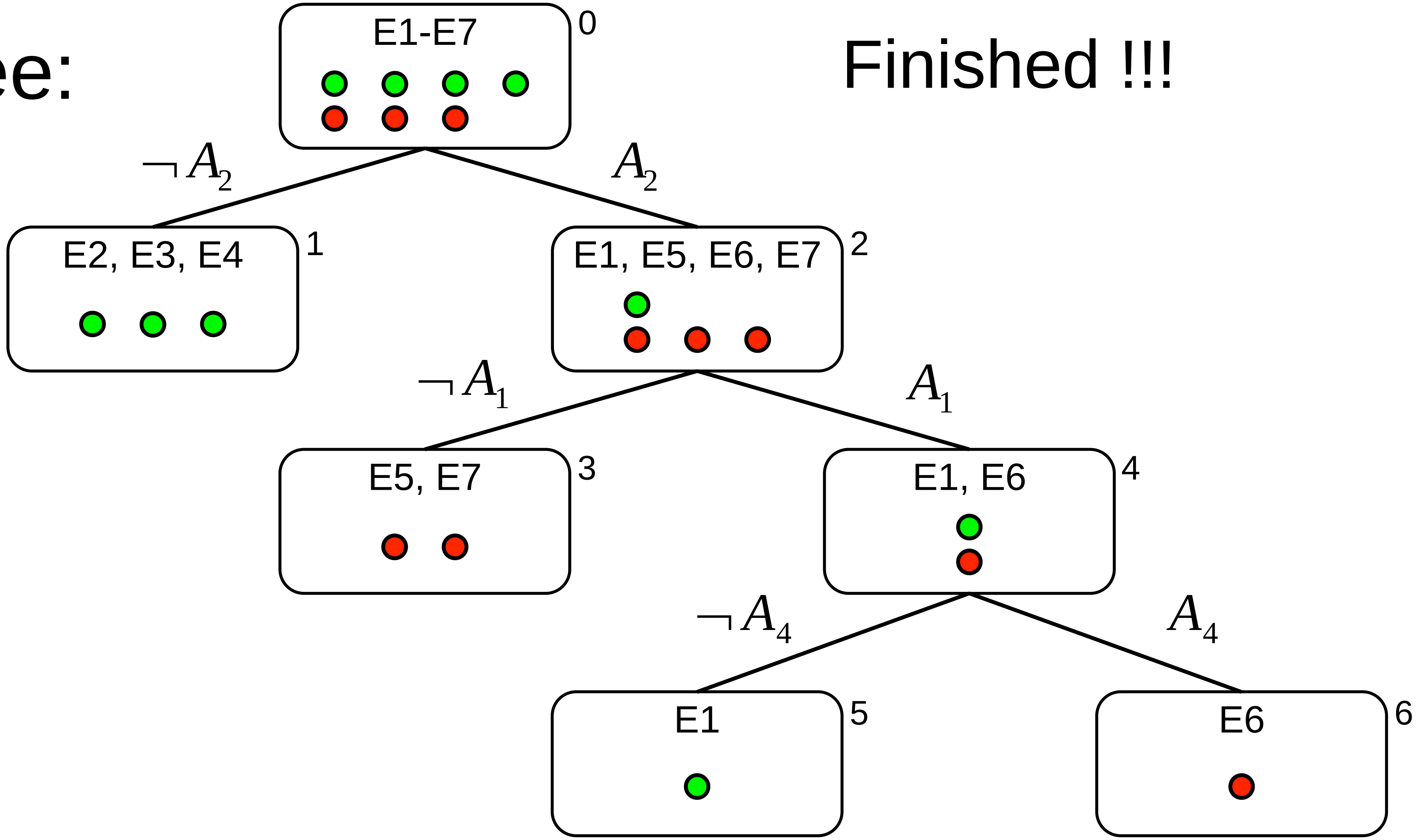
Decision tree:



A4 and A5 are equally good

Example	A_1	A_2	A_3	A_4	A_5	Output
1	1	1	0	0	0	1
2	0	0	1	1	0	1
3	1	0	0	1	0	1
4	1	0	1	0	1	1
5	0	1	0	0	0	0
6	1	1	0	1	1	0
7	0	1	1	1	1	0

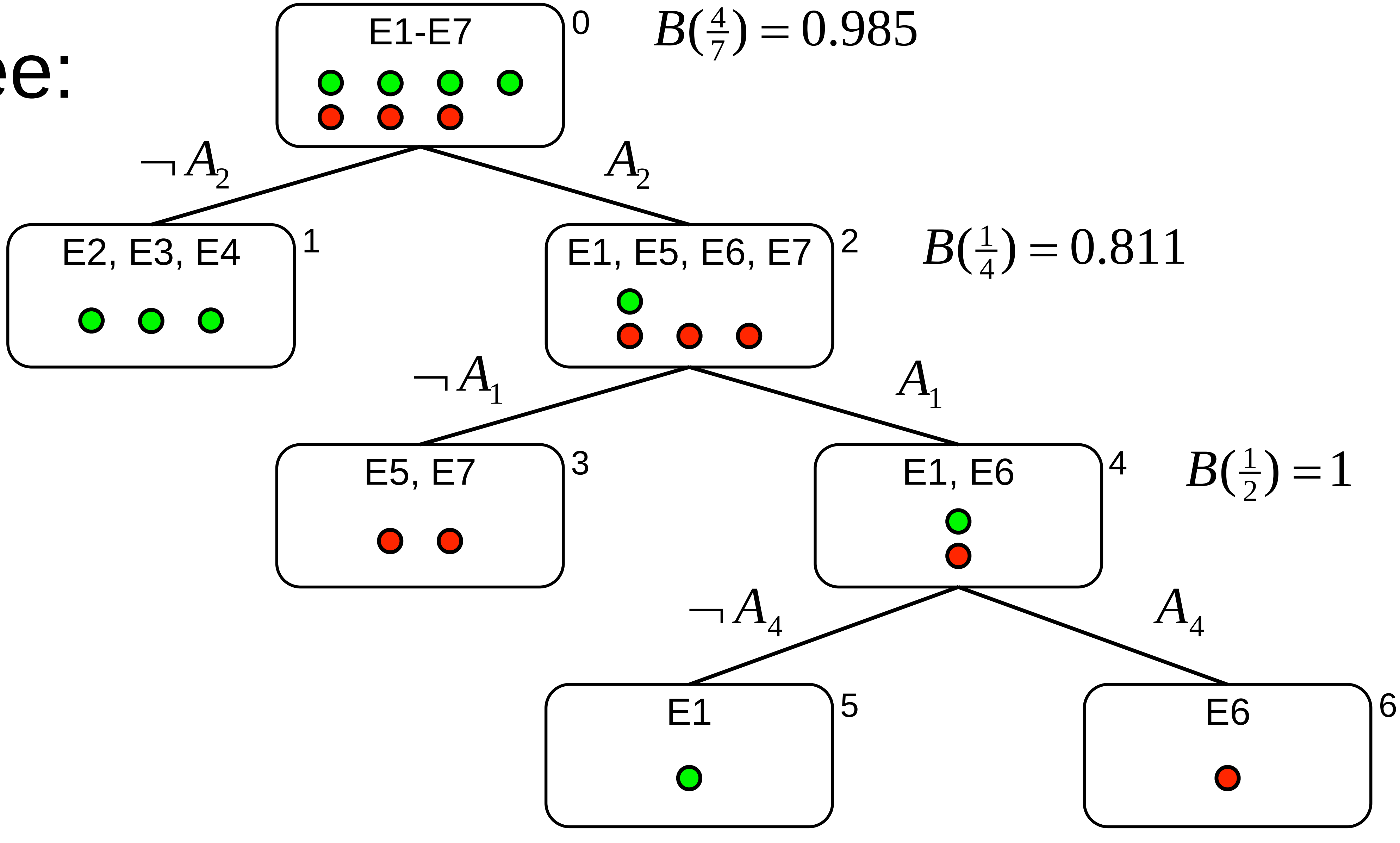
Decision tree:



Finished !!!

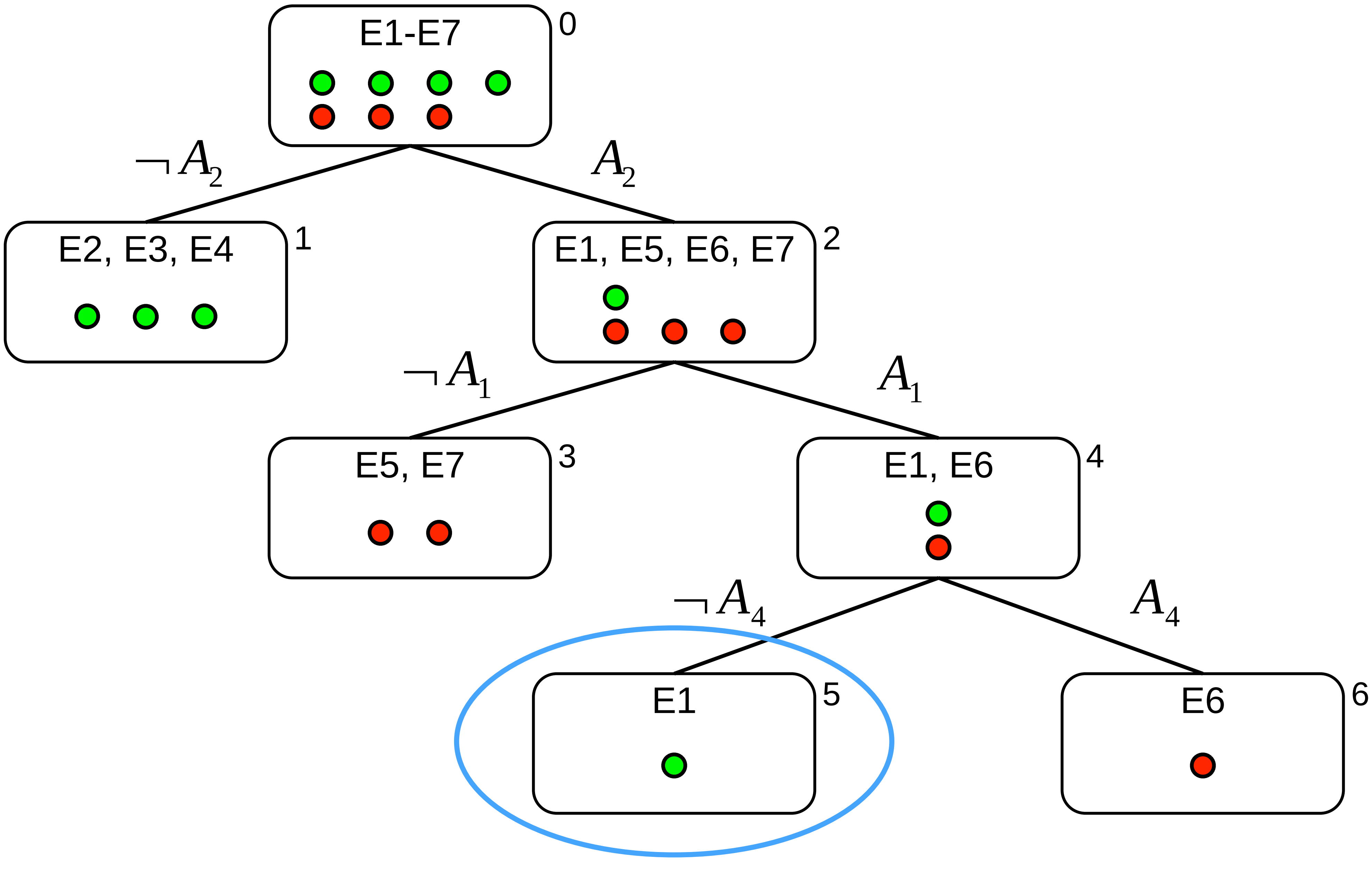
Example	A_1	A_2	A_3	A_4	A_5	Output
1	1	1	0	0	0	1
2	0	0	1	1	0	1
3	1	0	0	1	0	1
4	1	0	1	0	1	1
5	0	1	0	0	0	0
6	1	1	0	1	1	0
7	0	1	1	1	1	0

Decision tree:



How does the resulting tree classify:

Example	A_1	A_2	A_3	A_4	A_5	Output
8	1	1	1	0	1	???



$$h(\{1,1,1,0,1\}) = 1$$

Decision Trees: Overfitting and Pruning

Decision Tree Overfitting

A "fully trained" Decision Tree with homogeneous leaves is unlikely to generalize well

This is because the last few tests before the leaves are based on a very small number of examples (e.g. 2-3)

These tests are fitting the noise in the training dataset, not real patterns in $f(x)$

Combatting Overfitting

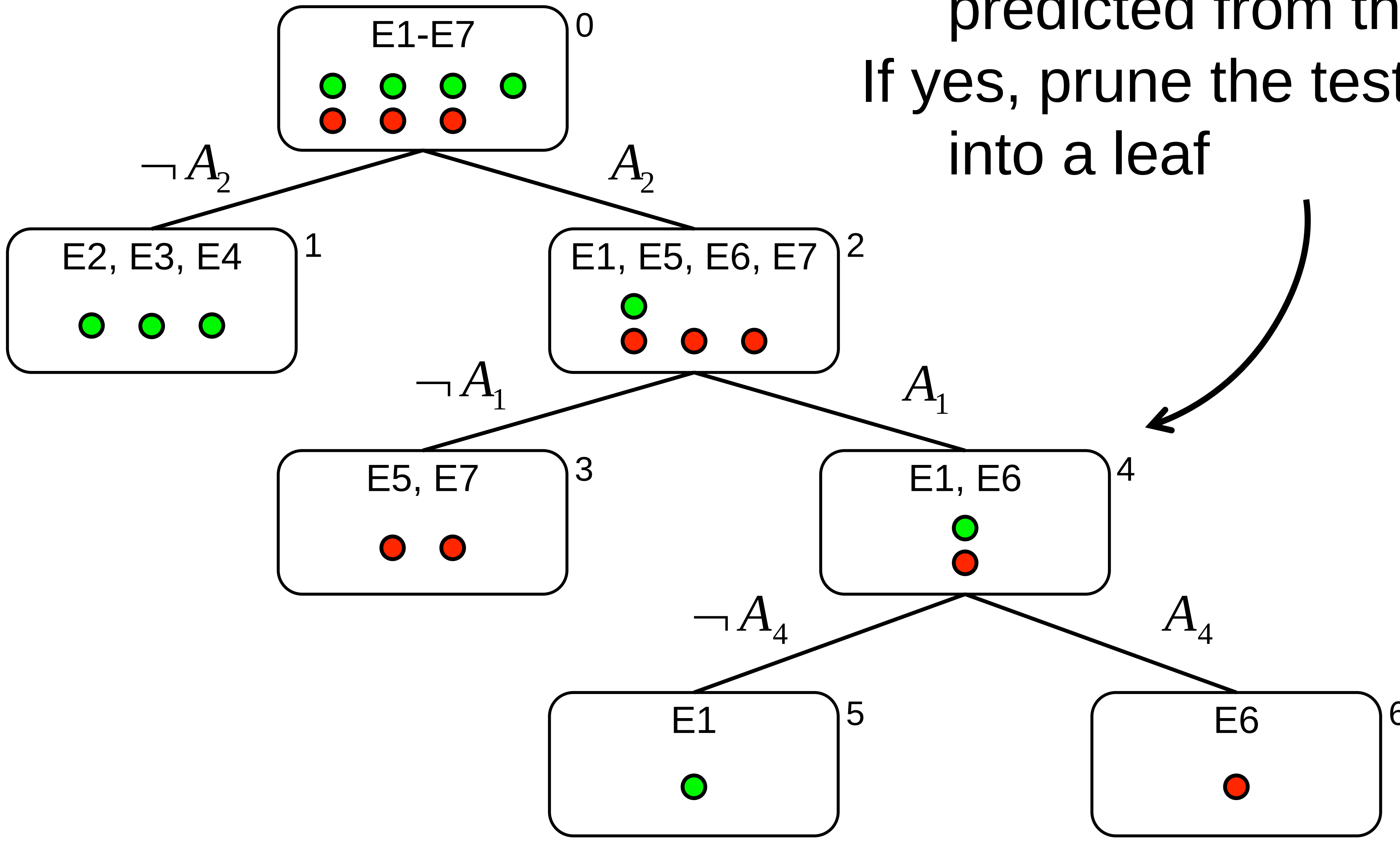
There are two standard solutions:

- *Early Stopping:* Stop splitting before you reach the point of splitting on noise
- *Tree Pruning:* Once the tree is fully-trained, go back and remove nodes which are not relevant (i.e. due to noise)

We will focus on Decision Tree Pruning

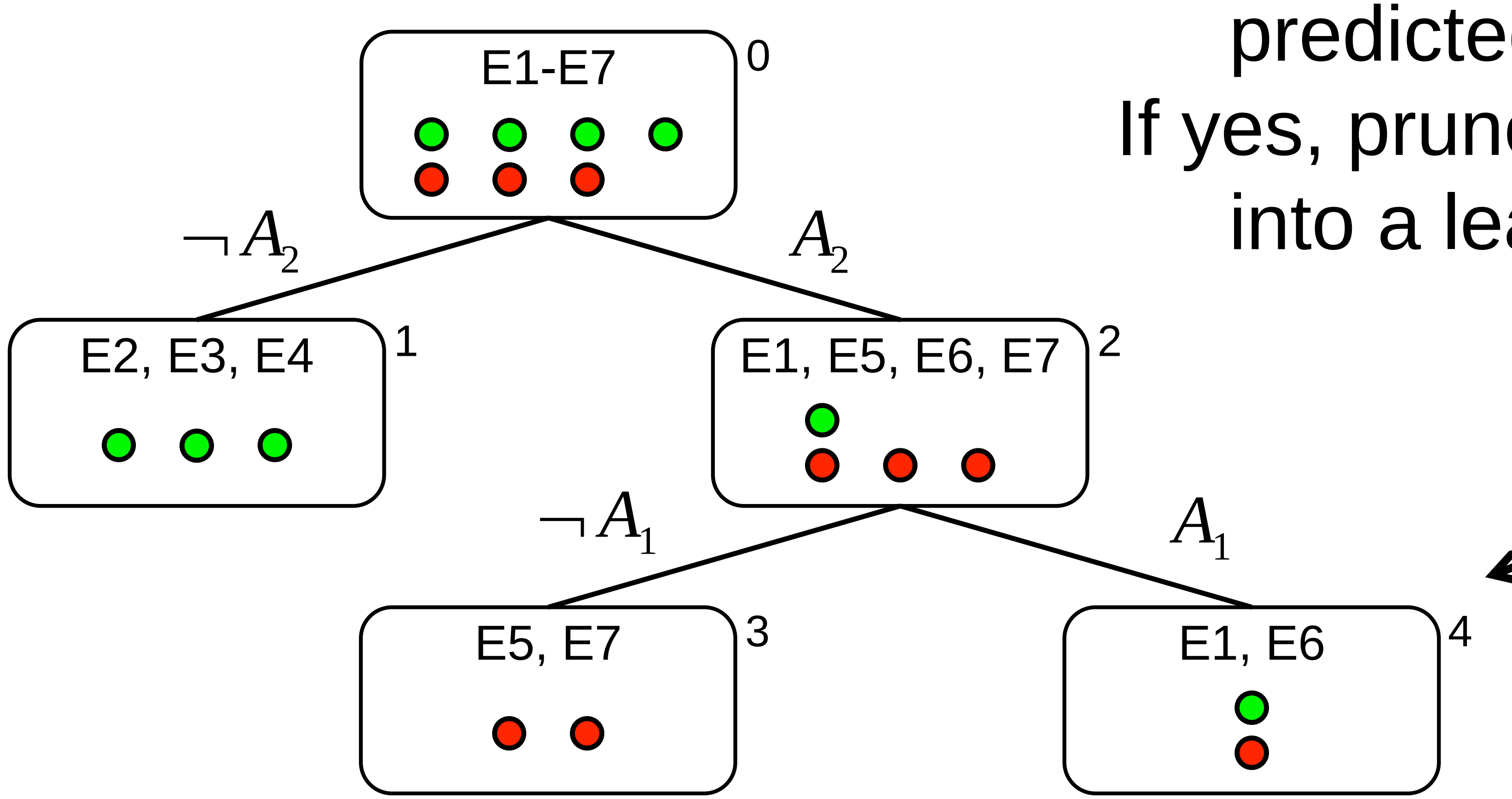
Pruning

Pick a split whose children are leaves
Test whether the attribute is irrelevant:
    Can the leaf distributions be
        predicted from the parent?
    If yes, prune the test and make it
        into a leaf



Pruning

Pick a split whose children are leaves
Test whether the attribute is irrelevant:
- Can the leaf distributions be predicted from the parent?
- If yes, prune the test and make it into a leaf



After pruning, the leaves are gone and the base node becomes the new leaf.

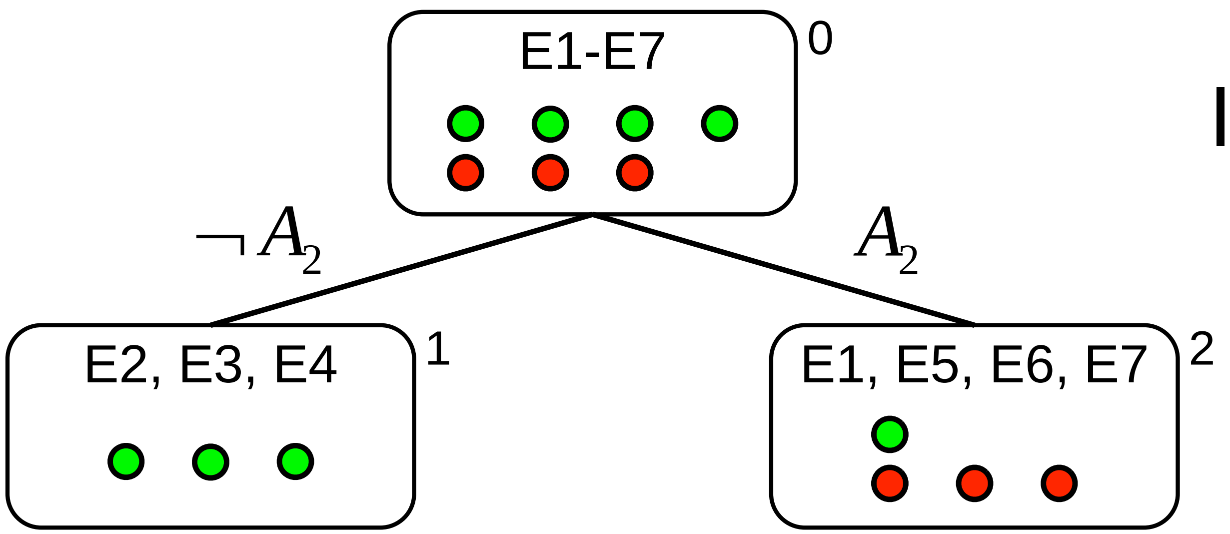
Pruning

Pick a split whose children are leaves

Test whether the attribute is irrelevant:

Can the leaf distributions be predicted from the parent?

If yes, prune the test and make it into a leaf



Pruning can be continued iteratively, as new leaves are created

$$h_2(x) = \arg\max\{\tfrac{3}{3+1}, \tfrac{1}{1+3}\} = 0$$

Test for Irrelevant Attributes

Test whether the data distribution in the leaves can be predicted from the parent

Use Chi-Squared test

- Predict the distribution at leaf k

$$\hat{p}_k = p_0 \frac{p_k + n_k}{p_0 + n_0} \quad \hat{n}_k = n_0 \frac{p_k + n_k}{p_0 + n_0}$$

Test for Irrelevant Attributes

Test whether the data distribution in the leaves can be predicted from the parent

Use Chi-Squared test

– Predict the distribution at leaf k

$$\hat{p}_k = p_0 \boxed{\frac{p_k + n_k}{p_0 + n_0}} \qquad \hat{n}_k = n_0 \frac{p_k + n_k}{p_0 + n_0}$$

Proportion of examples in parent node which follow branch k

Test for Irrelevant Attributes

Test whether the data distribution in the leaves can be predicted from the parent

Use Chi-Squared test

– Predict the distribution at leaf k

$$\hat{p}_k = p_0 \boxed{\frac{p_k + n_k}{p_0 + n_0}} \qquad \hat{n}_k = n_0 \frac{p_k + n_k}{p_0 + n_0}$$

Proportion of examples in parent node which follow branch k

Number of positive examples in parent node

Test for Irrelevant Attributes

Test whether the data distribution in the leaves can be predicted from the parent

Use Chi-Squared test

– Predict the distribution at leaf k

$$\hat{p}_k = p_0 \boxed{\frac{p_k + n_k}{p_0 + n_0}} \qquad \hat{n}_k = n_0 \frac{p_k + n_k}{p_0 + n_0}$$

Predicted number of positive examples in leaf k

Number of positive examples in parent node

Proportion of examples in parent node which follow branch k

Test for Irrelevant Attributes

Test whether the data distribution in the leaves can be predicted from the parent

Use Chi-Squared test

– Predict the distribution at leaf k

$$\hat{p}_k = p_0 \frac{p_k + n_k}{p_0 + n_0} \quad \hat{n}_k = n_0 \frac{p_k + n_k}{p_0 + n_0}$$

– Calculate error statistic

$$\Delta = \sum_{k=1}^{d_i} \frac{(p_k - \hat{p}_k)^2}{\hat{p}_k} + \frac{(n_k - \hat{n}_k)^2}{\hat{n}_k}$$

– Accept or reject the null hypothesis at a desired significance level (e.g. 5%)

Chi-Squared Test

Use a Chi-Square distribution with d_i-1 degrees of freedom (one less than # of attribute values)

Based on desired significance level (e.g. 5%) look-up threshold T on the statistic

Test for pruning:

$$\text{If } \Delta \leq T, \text{accept null hypothesis and prune the test}$$