DECISION TREES

INTRODUCTION TO DATA SCIENCE

TIM KRASKA


Thomas J. Watson, Sr.
Center for Information Technology

Clicker (Vote 1)

Potential topics for last 1-2 classes:

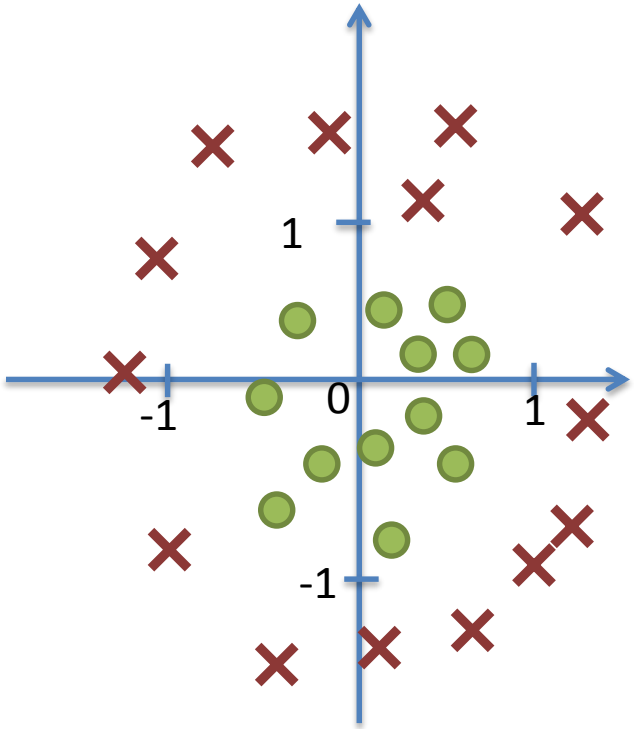
A) Advances in data management tools (NOSQL, graph processing engines, etc.)
B) Crowdsourcing (how to leverage humans at scale for acquiring data, cleaning data, etc).
C) Open question session
D) Lying with statistics
E) More ML (PCA, time series analysis)
F) Research topics

Clicker (Vote 2)

Potential topics for last 1-2 classes:

A) Advances in data management tools (NOSQL, graph processing engines, etc.)
B) Crowdsourcing (how to leverage humans at scale for acquiring data, cleaning data, etc).
C) Open question session
D) Lying with statistics
E) More ML (PCA, time series analysis)
F) Research topics

Clicker Question



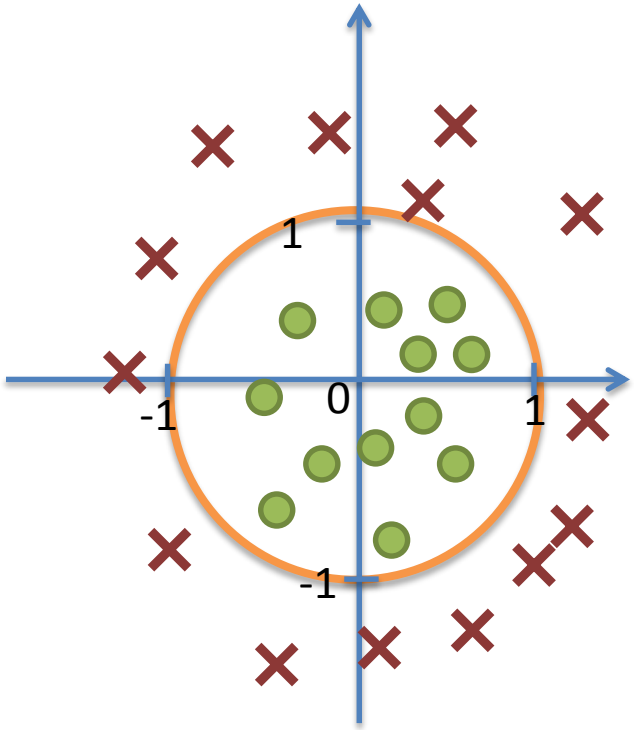
Our hypothesis;

$$h_q(x) = g(q_0 + q_1 x_1 + q_2 x_2 + q_3 x_1^2 + q_4 x_2^2)$$

Which θ would predict all red (y=1) and Green dots (y=0) correctly

a) $q = \begin{pmatrix} -1 \\ 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}$ b) $q = \begin{pmatrix} -1 \\ 1 \\ 1 \\ 0 \\ 0 \end{pmatrix}$ c) $q = \begin{pmatrix} -1 \\ 0 \\ 0 \\ 1 \\ 1 \end{pmatrix}$

Clicker Question



Our hypothesis;

$$h_q(x) = g(q_0 + q_1 x_1 + q_2 x_2 + q_3 x_1^2 + q_4 x_2^2)$$

Which θ would predict all red (y=1) and Green dots (y=0) correctly

$$\begin{pmatrix} -1 \\ 0 \\ 0 \\ 1 \\ 1 \end{pmatrix}$$

Predict "y=1" if

$$-1 + x_1^2 + x_2^2 > 0$$

$$x_1^2 + x_2^2 > 1$$

Stochastic Gradient Descent

Loop {

for i = 1 to m, {

$\theta_j := \theta_j + \alpha \left(y^{(i)} - h_\theta(x^{(i)}) \right) x_j^{(i)}$ (for every j).

}

}

Linear Regression

$$h(x) = \sum_{i=0}^{n} \theta_i x_i = \theta^T x$$

Logistic Regression

$$h_\theta(x) = g(\theta^T x) = \frac{1}{1 + e^{-\theta^T x}}$$

Interpretation of Hypothesis

$$h_q = \text{estimated probability that } y = 1 \text{ on input } x$$

Example:

$$x = \begin{bmatrix} x_0 \\ x_1 \end{bmatrix} = \begin{bmatrix} 1 \\ Hours \end{bmatrix}$$

$$h_q = 0.7$$

Student has a 70% chance of passing

More formally:

$$h_q(x) = p\left(y = 1 | x, q\right)$$

Summary

- (Linear) Regression → Regression technique
- Logistic Regression → Classification technique
- Batch/Mini-Batch/Stochastic – Gradient Descent → Optimization technique
- Important tuning parameters
  - Learning rate → speed and convergence
  - Polynomials → degrees of freedom
  - Regularization

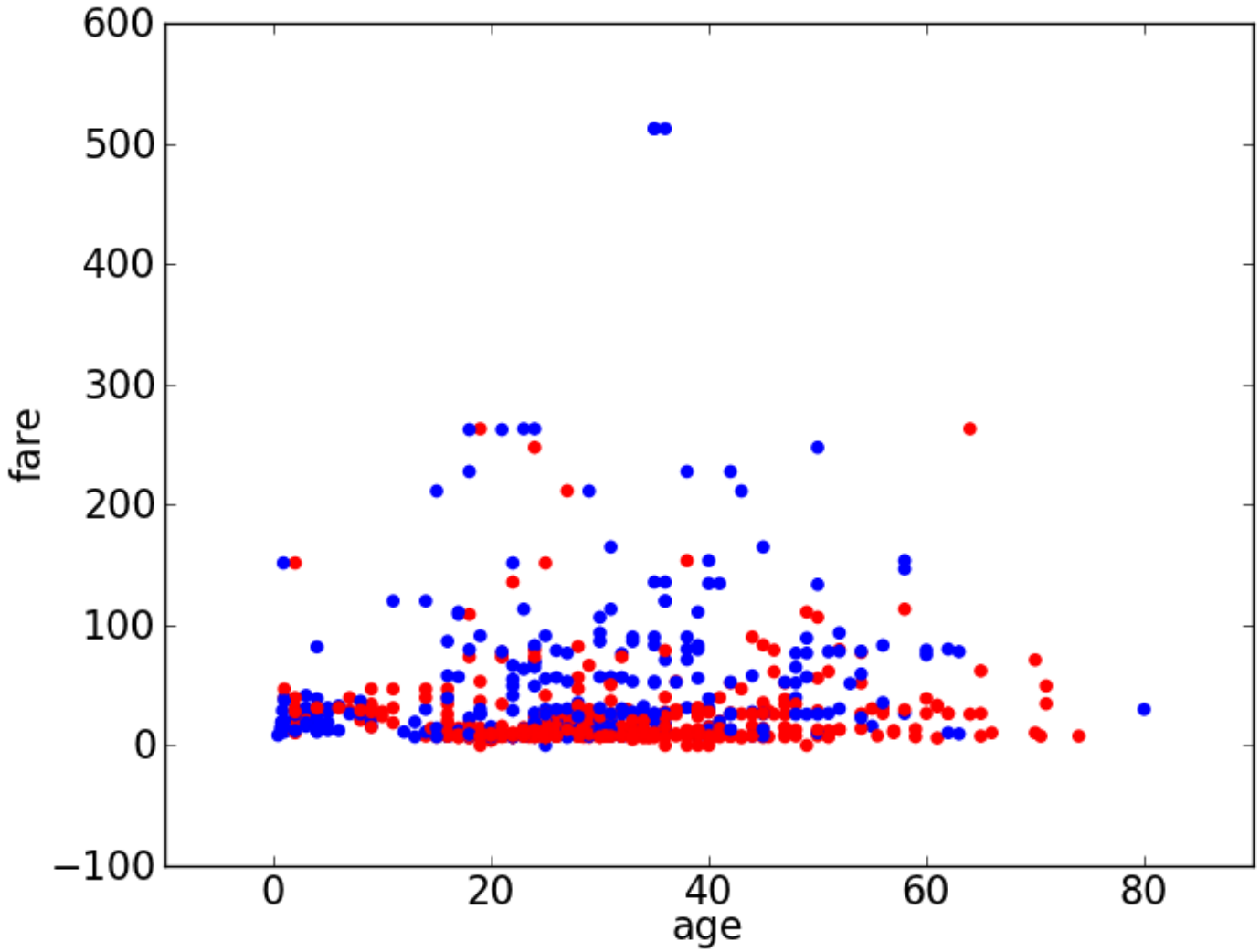
What We Covered So Far

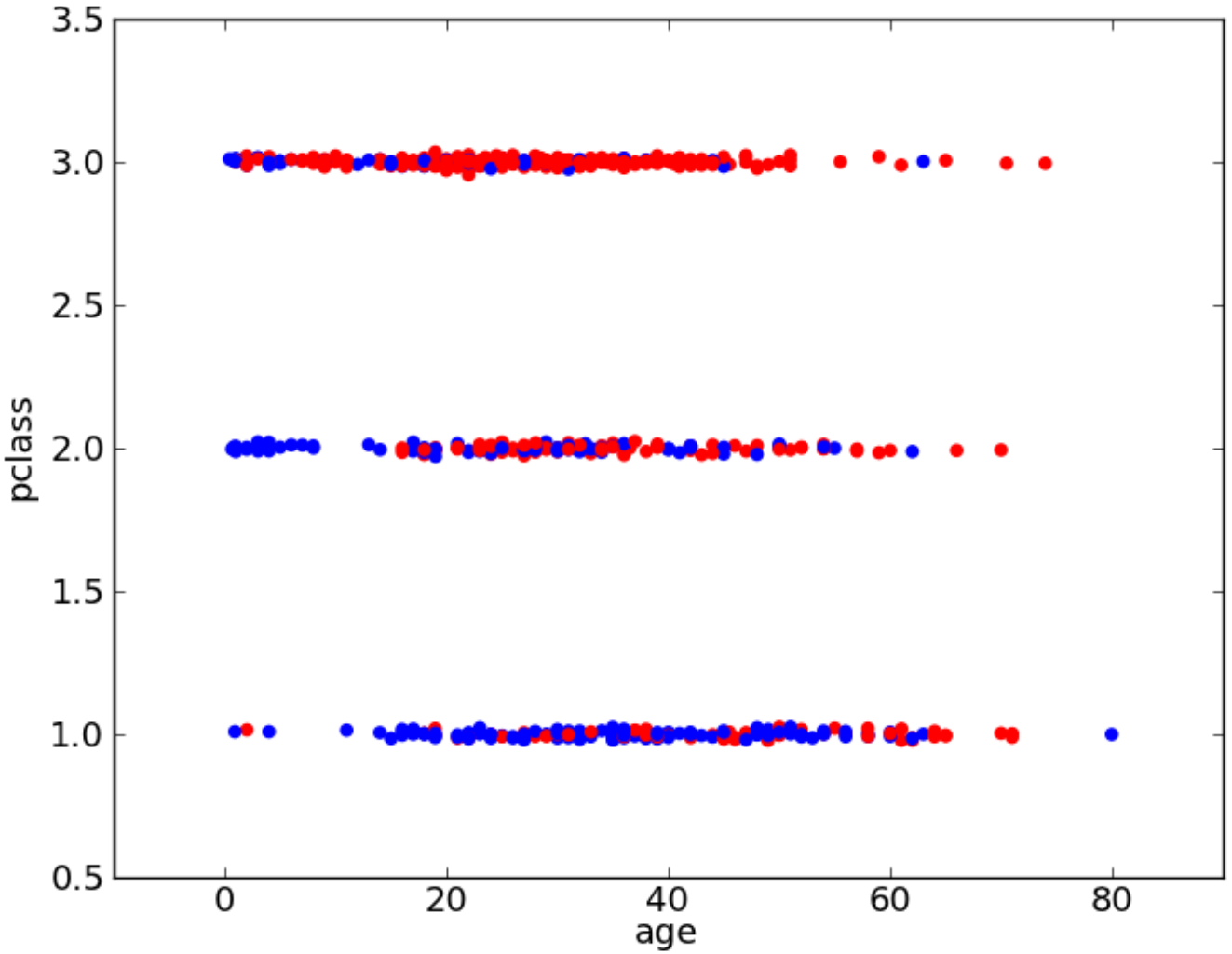
- Machine Learning / Data Mining Techniques
  - Supervised
    - (Linear)-Regression
    - Classification
      - KNN
      - Naïve Bayes
      - SVM
      - Logistic Regression
      - Decision Trees
  - Unsupervised
    - Clustering: K-Means
    - PageRank
    - Association Rule Mining
    - PCA
- Ensemble Algorithms: Boosting
- Optimization Technique: Batch-/MiniBatch/Stochastic Gradient Descent
- Testing: Cross-Validation

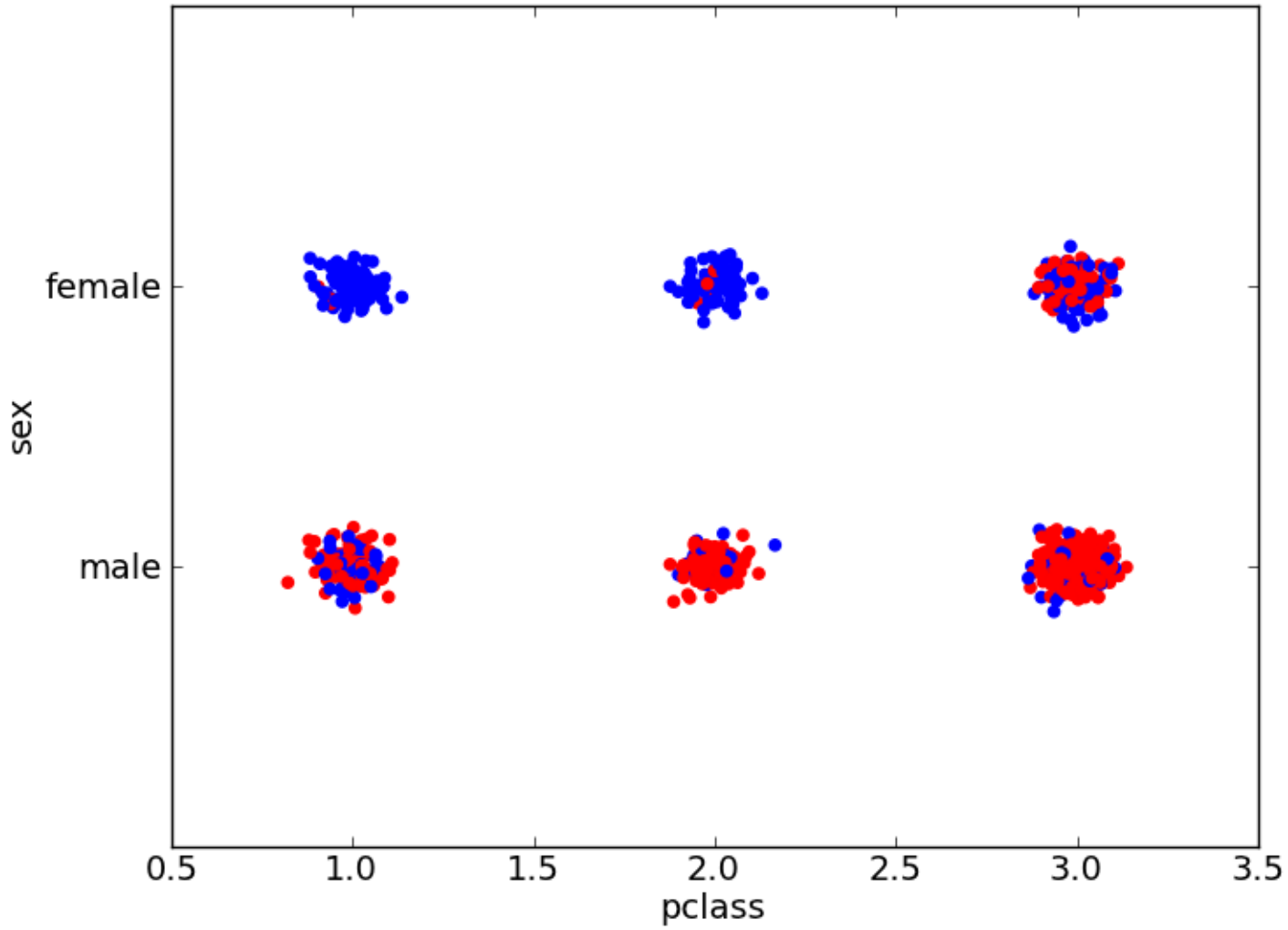
Decision Trees

Titanic Dataset

survived	pclass	sex	age	sibsp	parch	fare	cabin	embarked
0	3	male	22	1	0	7.25		S
1	1	female	38	1	0	71.2833	C85	C
1	3	female	26	0	0	7.925		S
1	1	female	35	1	0	53.1	C123	S
0	3	male	35	0	0	8.05		S
0	3	male		0	0	8.4583		Q
0	1	male	54	0	0	51.8625	E46	S
0	3	male	2	3	1	21.075		S
1	3	female	27	0	2	11.1333		S
1	2	female	14	1	0	30.0708		C
1	3	female	4	1	1	16.7	G6	S
1	1	female	58	0	0	26.55	C103	S
0	3	male	20	0	0	8.05		S


600
500
400
300
200
100
0
−100
fare
0
20
40
60
80
age


3.5
3.0
2.5
2.0
1.5
1.0
0.5
pclass
0
20
40
60
80
age


female
male
sex
0.5
1.0
1.5
2.0
2.5
3.0
3.5
pclass

IF sex=‘female’ THEN survive=yes

ELSE IF sex=‘male’ THEN survive = no

```
confusion matrix

no   yes    <-- classified as
468  109  |   no
 81  233  |  yes
```

(468 + 233) / (468+109+81+233) = 79% correct (and 21% incorrect)

Not bad!

IF pclass=‘1’ THEN survive=yes

ELSE IF pclass=‘2’ THEN survive=yes

ELSE IF pclass=‘3’ THEN survive=no

```
confusion matrix

no   yes    <-- classified as
372  119  |   no
177  223  |  yes
```

(372 + 223) / (372+119+223+177) = 67% correct (and 33% incorrect)

a little worse

1-Rule

For each attribute A:
For each value V of that attribute, create a rule:
1. count how often each class appears
2. find the most frequent class, c
3. make a rule "if A=V then Class=c"
Calculate the error rate of this rule

Pick the attribute whose rules produce the lowest error rate

How far can we go?

IF pclass=‘1’ AND sex=‘female’ THEN survive=yes
IF pclass=‘2’ AND sex=‘female’ THEN survive=yes
IF pclass=‘3’ AND sex=‘female’ AND age < 4 THEN survive=yes
IF pclass=‘3’ AND sex=‘female’ AND age >= 4 THEN survive=no
IF pclass=‘2’ AND sex=‘male’ THEN survive=no
IF pclass=‘3’ AND sex=‘male’ THEN survive=no
IF pclass=‘1’ AND sex=‘male’ AND age < 5 THEN survive=yes
…

Sequential Covering

```
Initialize R to the empty set

for each class C {

    while D is nonempty {

        Construct one rule r that correctly classifies
        some instances in D that belong to class C and
        does not incorrectly classify any non-C instances

        Add rule r to ruleset R

        Remove from D all instances correctly classified by r
    }

}

return R
```


src: Alvarez

Sequential Covering: Finding next rule for class C

```
Initialize A as the set of all attributes over D

while r incorrectly classifies some non-C instances of D {

    write r as ant(r) => C

    for each attribute-value pair (a=v),
    where a belongs to A and v is a value of a,
    compute the accuracy of the rule

        ant(r) and (a=v) => C

    let (a*=v*) be the attribute-value pair of
    maximum accuracy over D; in case of a tie,
    choose the pair that covers the greatest
    number of instances of D

    update r by adding (a*=v*) to its antecedent:
        r = ( ant(r) and (a*=v*) ) => C
    remove the attribute a* from the set A:
        A = A - {a*}
}
```


src: Alvarez

Sequential Covering: Finding next rule for class C

```
Initialize A as the set of all attributes over D

while r incorrectly classifies some non-C instances of D {

    write r as ant(r) => C

    for each attribute-value pair (a=v),
    where a belongs to A and v is a value of a,
    compute the accuracy of the rule

        ant(r) and (a=v) => C

    let (a*=v*) be the attribute-value pair of
    maximum accuracy over D; in case of a tie,
    choose the pair that covers the greatest
    number of instances of D

    update r by adding (a*=v*) to its antecedent:
        r = ( ant(r) and (a*=v*) ) => C
    remove the attribute a* from the set A:
        A = A - {a*}
}
```

r := {} =>S
A:={pclass, age, sex,…}

ant(r) := {}

{} and (pclass=1) => S : 200/323
{} and (pclass=2) => S : 119/277
….
{} and (sex=f) => S : 339 / 466

r := {} and (sex=f) => S
A = {pclass, age, sex,…} – sex

src: Alvarez

Sequential Covering: Finding next rule for class C

```
Initialize A as the set of all attributes over D

while r incorrectly classifies some non-C instances of D {

    write r as ant(r) => C

    for each attribute-value pair (a=v),
    where a belongs to A and v is a value of a,
    compute the accuracy of the rule

        ant(r) and (a=v) => C

    let (a*=v*) be the attribute-value pair of
    maximum accuracy over D; in case of a tie,
    choose the pair that covers the greatest
    number of instances of D

    update r by adding (a*=v*) to its antecedent:
        r = ( ant(r) and (a*=v*) ) => C
    remove the attribute a* from the set A:
        A = A - {a*}
}
```

r := {sex=f} =>S
A:={pclass, age,…}

ant(r) := {sex=f}

{sex=f} and (pclass=1) => S : 139/144
{sex=f} and (pclass=2) => S : 94/106
….

r := (sex=f) and (pclass=1) => S
A = {pclass, age,…} – pclass

src: Alvarez

Strategies for Learning Each Rule

- General-to-Specific
  - Start with an empty rule
  - Add constraints to eliminate negative examples
  - Stop when only positives are covered
- Specific-to-General (not shown)
  - Start with a rule that identifies a single random instance
  - Remove constraints in order to cover more positives
  - Stop when further generalization results in covering negatives

Conflicts

- If more than one rule is triggered
  - Choose the “most specific” rule
  - Use domain knowledge to order rules by priority

Recap

- Representation
  - A set of rules: IF…THEN conditions
- Evaluation
  - accuracy = # of correct predictions / coverage
  - coverage: # of data points that satisfy conditions
- Optimization
  - Build rules by finding conditions that maximize accuracy

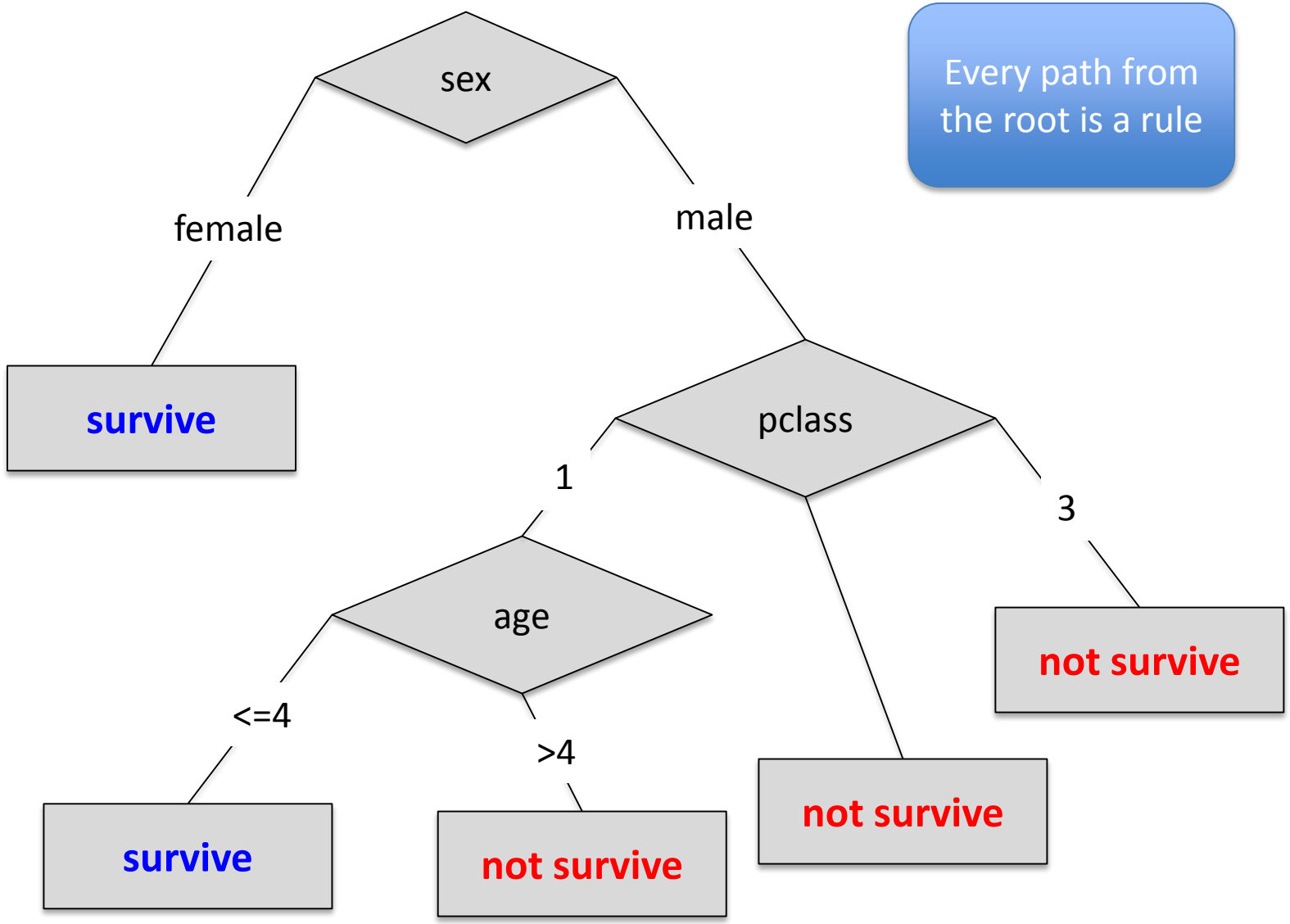
One rule is easy to interpret, but a complex set of rules probably isn't

How far can we go?

We might consider grouping redundant conditions

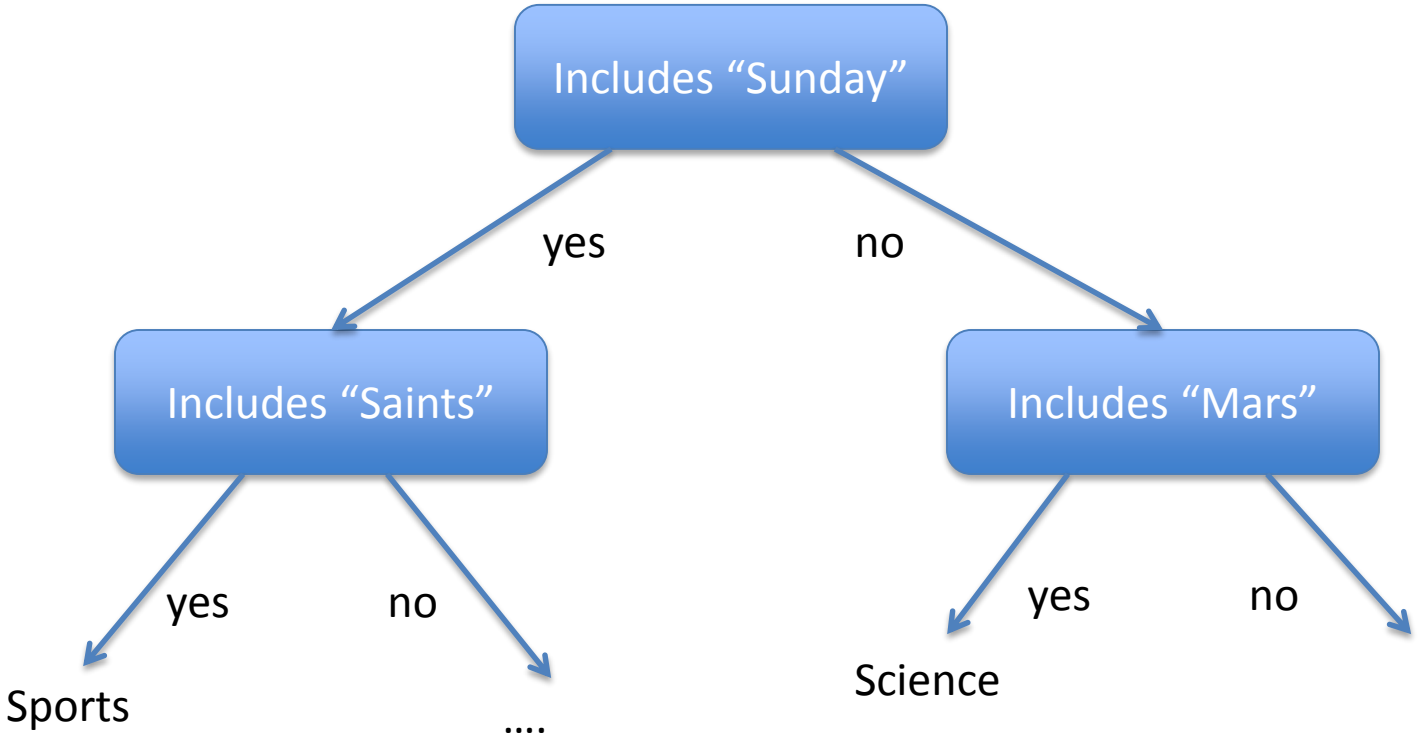
```
IF pclass=‘1’ THEN
        IF sex=‘female’ THEN survive=yes
        IF sex=‘male’ AND age < 5 THEN survive=yes
IF pclass=‘2’
        IF sex=‘female’ THEN survive=yes
        IF sex=‘male’ THEN survive=no
IF pclass=‘3’
        IF sex=‘male’ THEN survive=no
        IF sex=‘female’
                IF age < 4 THEN survive=yes
                IF age >= 4 THEN survive=no
```

A decision tree



What is the problem of using the previous technique for creating a tree?
Does it produce a good tree?

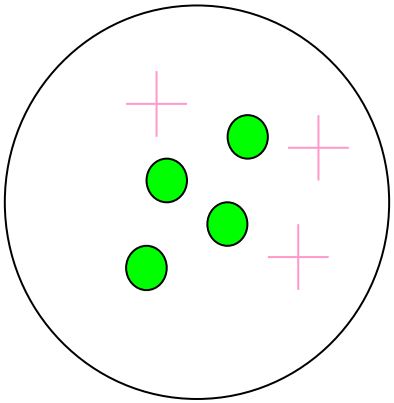
Document Classification Example

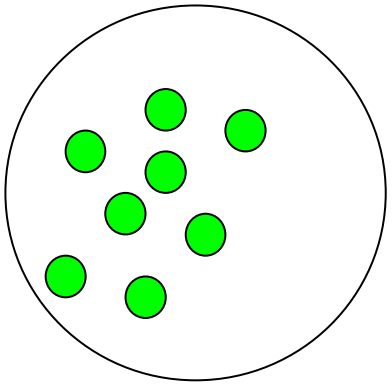


Aside on Entropy

Impurity/Entropy (informal)

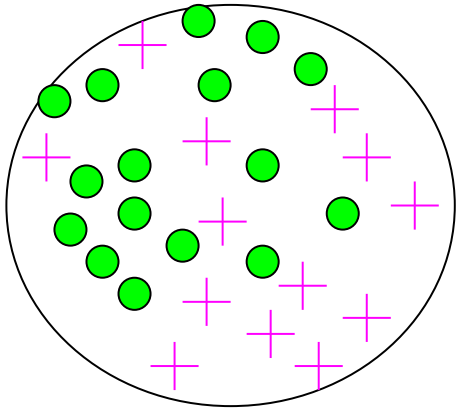
- Measures the level of **impurity** in a group of examples



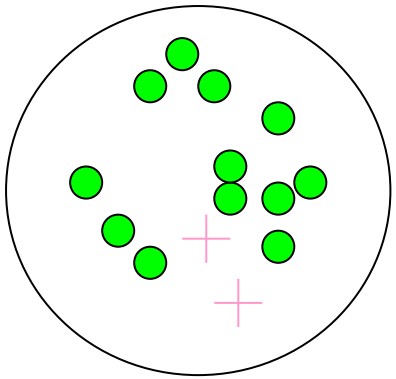


Impurity

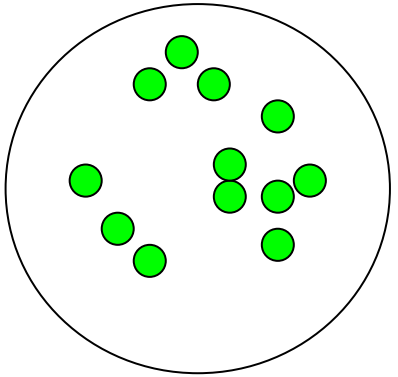
Very impure group



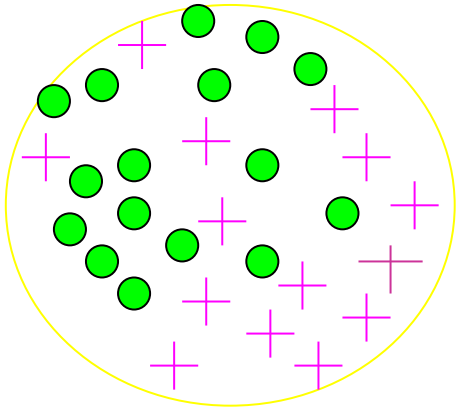
Less impure



Minimum impurity



Entropy



- Entropy = $\sum_i p_i \log_2 p_i$

  p_i is the probability of class i

  Compute it as the proportion of class i in the set.

  16/30 are green circles; 14/30 are pink crosses
  $\log_2(16/30) = -.9$; $\log_2(14/30) = -1.1$
  Entropy = -(16/30)(-.9) –(14/30)(-1.1) = .99

- Entropy comes from information theory. The higher the entropy the more the information content.

  What does that mean for learning from examples?

Clicker

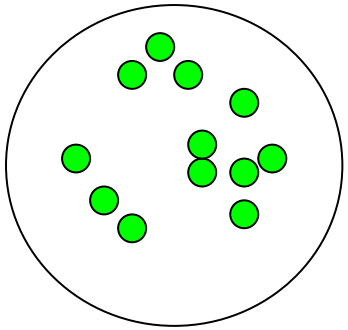
What is the entropy if all examples belong to the same class?

a) 0
b) 1
c) Infinite

2 Class Example

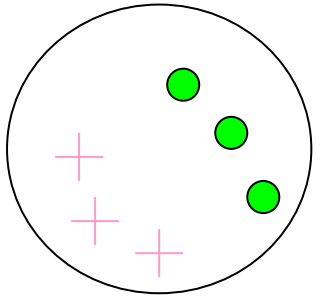
- What is the entropy of a group in which all examples belong to the same class?
  - entropy = $- 1 \log_2 1 = 0$

Minimum impurity



- What is the entropy of a group with 50% in either class?
  - entropy = $-0.5 \log_2 0.5 - 0.5 \log_2 0.5 = 1$

Maximum impurity



Example: Rolling a die

$$p_1 = \frac{1}{6},\ p_2 = \frac{1}{6},\ p_3 = \frac{1}{6},\ \ldots$$

$$\text{Entropy} = -\sum_i p_i \log_2 p_i$$

$$= -6 \times \left(\frac{1}{6} \log_2 \frac{1}{6} \right)$$

$$\approx 2.58$$

Clicker

Has an unfair/weighted die a higher or lower entropy?

A) Higher

B) Lower

Example: Rolling a weighted die

$$p_1 = 0.1,\ p_2 = 0.1,\ p_3 = 0.1,\ \ldots\ p_6 = 0.5$$

$$\text{Entropy} = -\sum_i p_x \log_2 p_x$$

$$= -5 \times (0.1 \log_2 0.1) - 0.5 \log_2 0.5$$

$$= 2.16$$

The weighted die is has less uncertainty than a fair die

How uncertain is your data?

- 342/891 survivors in titanic training set

$$-\left(\frac{342}{891}\log_2\frac{342}{891}+\frac{549}{891}\log_2\frac{549}{891}\right)=0.96$$

- Say there were only 50 survivors

$$-\left(\frac{50}{891}\log_2\frac{50}{891}+\frac{841}{891}\log_2\frac{841}{891}\right)=0.31$$

In Class Task

How can you use Entropy to build a decision tree.

Discuss with your neighbor(s)

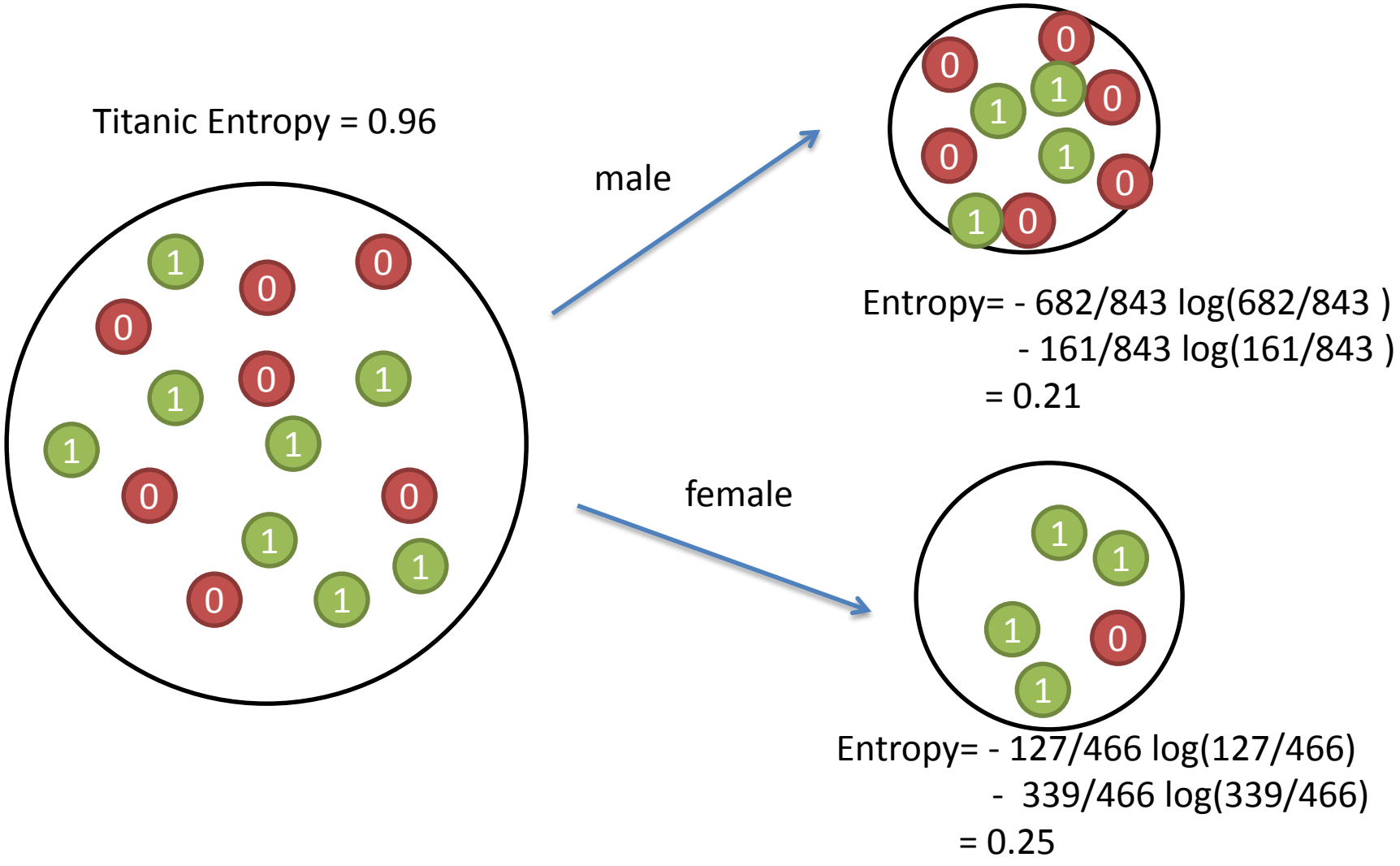
Discuss the following ideas

- Select the feature based on the highest entropy
- Select the feature based on the lowest entropy
- Stop splitting if the entropy is 0
- Select the feature based on the entropy after the split
- What if one group is under-/over represented

Back to decision trees

- Which attribute do we choose at each level?
- The one with the highest information gain
  - The one that reduces the uncertainty/impurity the most
- We want to determine which attribute in a given set of training feature vectors is most useful for discriminating between the classes to be learned.
- Information gain tells us how important a given attribute of the feature vectors is.
- We will use it to decide the ordering of attributes in the nodes of a decision tree.

Information Gain



Weighted Entropy: 466/1309 * 0.25 + 843 / 1309 * 0.21 = 0.22

Information Gain for split: 0.96 – 0.22 = 0.74

outlook	temperature	humidity	windy	play
overcast	cool	normal	TRUE	yes
overcast	hot	high	FALSE	yes
overcast	hot	normal	FALSE	yes
overcast	mild	high	TRUE	yes
rainy	cool	normal	TRUE	no
rainy	mild	high	TRUE	no
rainy	cool	normal	FALSE	yes
rainy	mild	high	FALSE	yes
rainy	mild	normal	FALSE	yes
sunny	hot	high	FALSE	no
sunny	hot	high	TRUE	no
sunny	mild	high	FALSE	no
sunny	cool	normal	FALSE	yes
sunny	mild	normal	TRUE	yes

Before: 14 records, 9 are “yes”

$$-\left(\frac{9}{14}\log_2\frac{9}{14}+\frac{5}{14}\log_2\frac{5}{14}\right)=0.94$$

If we choose **outlook**:

overcast : 4 records, 4 are “yes”

$$-\left(\frac{4}{4}\log_2\frac{4}{4}\right)=0$$

rainy : 5 records, 3 are “yes”

$$-\left(\frac{3}{5}\log_2\frac{3}{5}+\frac{2}{5}\log_2\frac{2}{5}\right)=0.97$$

sunny : 5 records, 2 are “yes”

$$-\left(\frac{2}{5}\log_2\frac{2}{5}+\frac{3}{5}\log_2\frac{3}{5}\right)=0.97$$

Expected new entropy:

$$\frac{4}{14}\times 0.0+\frac{5}{14}\times 0.97+\frac{5}{14}\times 0.97$$

= 0.69

outlook	temperature	humidity	windy	play
overcast	cool	normal	TRUE	yes
overcast	hot	high	FALSE	yes
overcast	hot	normal	FALSE	yes
overcast	mild	high	TRUE	yes
rainy	cool	normal	TRUE	no
rainy	mild	high	TRUE	no
rainy	cool	normal	FALSE	yes
rainy	mild	high	FALSE	yes
rainy	mild	normal	FALSE	yes
sunny	hot	high	FALSE	no
sunny	hot	high	TRUE	no
sunny	mild	high	FALSE	no
sunny	cool	normal	FALSE	yes
sunny	mild	normal	TRUE	yes

Before: 14 records, 9 are "yes"

$$-\left(\frac{9}{14}\log_2\frac{9}{14}+\frac{5}{14}\log_2\frac{5}{14}\right)=0.94$$

Clicker:

If we choose windy, what is the expected entropy?

a) 0.81

= - (6/8 log(6/8) + 2/8 log(2/8))

b) 0.89

= 6/14 * 1 +
(-8/14* (6/8 log(6/8) + 2/8 log(2/8)))

c) 1

= - (0.5 * log(0.5)+0.5 * log(0.5))

outlook	temperature	humidity	windy	play
overcast	cool	normal	TRUE	yes
overcast	hot	high	FALSE	yes
overcast	hot	normal	FALSE	yes
overcast	mild	high	TRUE	yes
rainy	cool	normal	TRUE	no
rainy	mild	high	TRUE	no
rainy	cool	normal	FALSE	yes
rainy	mild	high	FALSE	yes
rainy	mild	normal	FALSE	yes
sunny	hot	high	FALSE	no
sunny	hot	high	TRUE	no
sunny	mild	high	FALSE	no
sunny	cool	normal	FALSE	yes
sunny	mild	normal	TRUE	yes

Before: 14 records, 9 are “yes”

$$-\left(\frac{9}{14}\log_2\frac{9}{14}+\frac{5}{14}\log_2\frac{5}{14}\right)=0.94$$

If we choose **windy**:

FALSE: 8 records, 6 are “yes”

0.81 = -(6/8*log(6/8)+2/8*log(2/8))

TRUE: 6 records, 3 are “yes”

1

Expected new entropy:

0.81(8/14) + 1 (6/14)

= 0.89

outlook	temperature	humidity	windy	play
overcast	cool	normal	TRUE	yes
overcast	hot	high	FALSE	yes
overcast	hot	normal	FALSE	yes
overcast	mild	high	TRUE	yes
rainy	cool	normal	TRUE	no
rainy	mild	high	TRUE	no
rainy	cool	normal	FALSE	yes
rainy	mild	high	FALSE	yes
rainy	mild	normal	FALSE	yes
sunny	hot	high	FALSE	no
sunny	hot	high	TRUE	no
sunny	mild	high	FALSE	no
sunny	cool	normal	FALSE	yes
sunny	mild	normal	TRUE	yes

Before: 14 records, 9 are “yes”

$$-\left(\frac{9}{14}\log_2\frac{9}{14}+\frac{5}{14}\log_2\frac{5}{14}\right)=0.94$$

If we choose **temperature**:

cool : 4 records, 3 are “yes”

0.81

rainy : 4 records, 2 are “yes”

1.0

sunny : 6 records, 4 are “yes”

0.92

Expected new entropy:

0.81(4/14) + 1.0(4/14) + 0.92(6/14)

= 0.91

outlook	temperature	humidity	windy	play
overcast	cool	normal	TRUE	yes
overcast	hot	high	FALSE	yes
overcast	hot	normal	FALSE	yes
overcast	mild	high	TRUE	yes
rainy	cool	normal	TRUE	no
rainy	mild	high	TRUE	no
rainy	cool	normal	FALSE	yes
rainy	mild	high	FALSE	yes
rainy	mild	normal	FALSE	yes
sunny	hot	high	FALSE	no
sunny	hot	high	TRUE	no
sunny	mild	high	FALSE	no
sunny	cool	normal	FALSE	yes
sunny	mild	normal	TRUE	yes

Before: 14 records, 9 are “yes”

$$-\left(\frac{9}{14}\log_2\frac{9}{14}+\frac{5}{14}\log_2\frac{5}{14}\right)=0.94$$

If we choose **humidity**:

normal : 7 records, 6 are “yes”

0.59

high : 7 records, 2 are “yes”

0.86

Expected new entropy:

0.59(7/14) + 0.86(7/14)

= 0.725

outlook	temperature	humidity	windy	play
overcast	cool	normal	TRUE	yes
overcast	hot	high	FALSE	yes
overcast	hot	normal	FALSE	yes
overcast	mild	high	TRUE	yes
rainy	cool	normal	TRUE	no
rainy	mild	high	TRUE	no
rainy	cool	normal	FALSE	yes
rainy	mild	high	FALSE	yes
rainy	mild	normal	FALSE	yes
sunny	hot	high	FALSE	no
sunny	hot	high	TRUE	no
sunny	mild	high	FALSE	no
sunny	cool	normal	FALSE	yes
sunny	mild	normal	TRUE	yes

Before: 14 records, 9 are “yes”

$$-\left(\frac{9}{14}\log_2\frac{9}{14}+\frac{5}{14}\log_2\frac{5}{14}\right)=0.94$$

outlook

0.94 – 0.69 = 0.25 highest gain

temperature

0.94 – 0.91 = 0.03

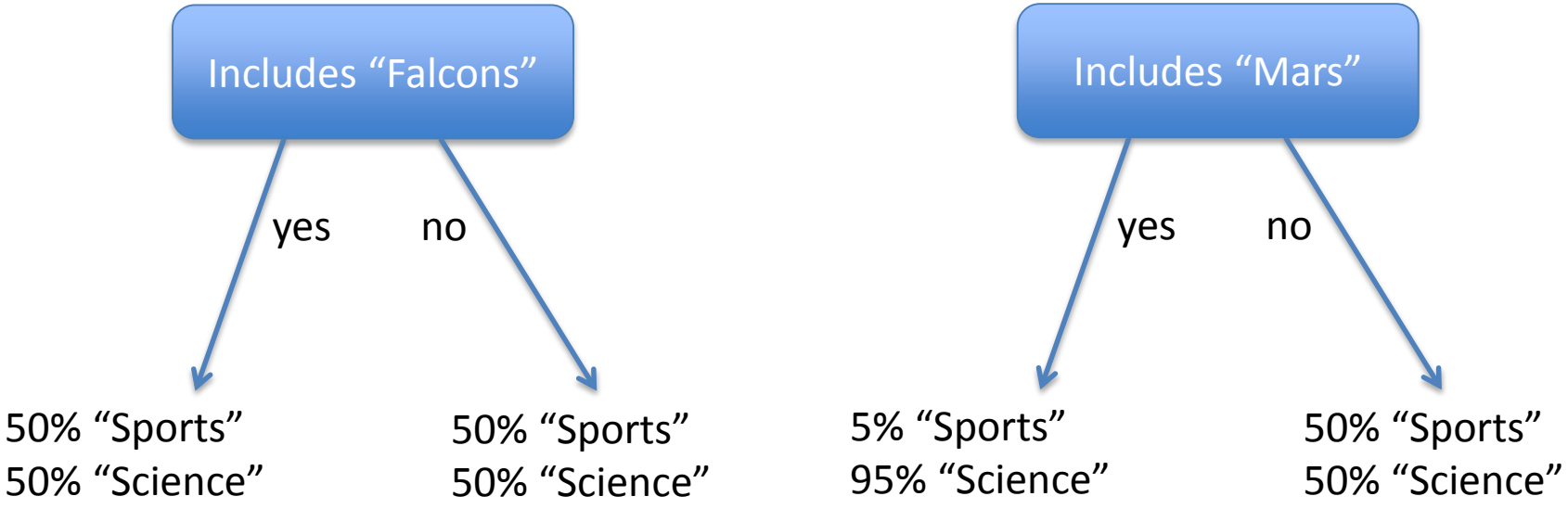
humidity

0.94 – 0.725 = 0.215

windy

0.94 – 0.87 = 0.07

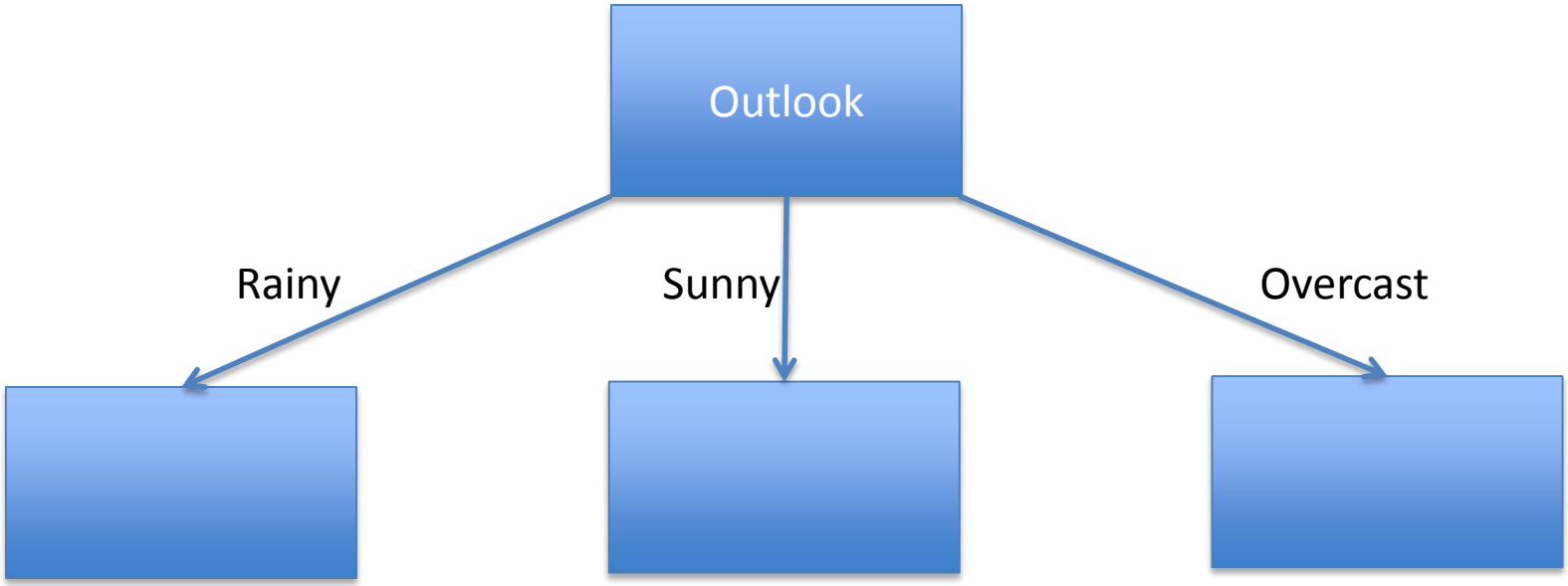
Document Classification



Clicker Question (assuming equal size):
a) Falcon’s Information Gain is higher
b) Mars’ Information Gain is higher

Building a Decision Tree (ID3 Algorithm)

- Assume attributes are discrete
  - Discretize continuous attributes
- Choose the attribute with the highest Information Gain
- Create branches for each value of attribute
- Examples partitioned based on selected attributes
- Repeat with remaining attributes
- Stopping conditions
  - All examples assigned the same label
  - No examples left


Outlook
Rainy
Sunny
Overcast

Problems

- Expensive to train
- Prone to overfitting
  - Drive to perfection on training data, bad on test data
  - Pruning can help: remove or aggregate subtrees that provide little discriminatory power (C45)

C4.5 Extensions

- Continuous Attributes

outlook	temperature	humidity	windy	play
overcast	cool	60	TRUE	yes
overcast	hot	80	FALSE	yes
overcast	hot	63	FALSE	yes
overcast	mild	81	TRUE	yes
rainy	cool	58	TRUE	no
rainy	mild	90	TRUE	no
rainy	cool	54	FALSE	yes
rainy	mild	92	FALSE	yes
rainy	mild	59	FALSE	yes
sunny	hot	90	FALSE	no
sunny	hot	89	TRUE	no
sunny	mild	90	FALSE	no
sunny	cool	60	FALSE	yes
sunny	mild	62	TRUE	yes

Consider every possible binary partition; choose the partition with the highest gain

outlook	temperature	humidity	windy	play
rainy	mild	54	FALSE	yes
overcast	hot	58	FALSE	yes
overcast	cool	59	TRUE	yes
rainy	cool	60	FALSE	yes
overcast	mild	60	TRUE	yes
overcast	hot	62	FALSE	yes
rainy	mild	63	TRUE	no
sunny	cool	80	FALSE	yes
rainy	mild	81	FALSE	yes
sunny	mild	89	TRUE	yes
sunny	hot	90	FALSE	no
rainy	cool	90	TRUE	no
sunny	hot	90	TRUE	no
sunny	mild	92	FALSE	no

E(6/6)

= 0.0

E(3/8) + E(5/8)

= 0.95

E(9/10)+ E(1/10)

= 0.47

E(4/4)

= 0.0

Expect = 8/14*0.95 + 6/14*0

= 0.54

Expect = 10/14*0.47 + 4/14*0

= 0.33