Decision Trees

Data Science: Jordan Boyd-Graber
University of Maryland
MARCH 11, 2018

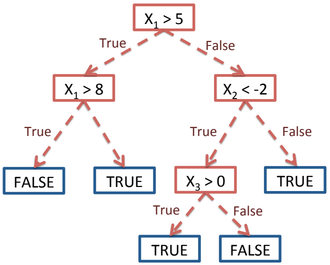
Roadmap

- Classification: machines labeling data for us
- Last time: naïve Bayes
- This time:
  - Decision Trees
    - Simple, nonlinear, interpretable
  - **Discussion:** Which classifier should I use for my problem?

Trees

Suppose that we want to construct a set of rules to represent the data

- can represent data as a series of if-then statements
- here, “if” splits inputs into two categories
- “then” assigns value
- when “if” statements are nested, structure is called a tree

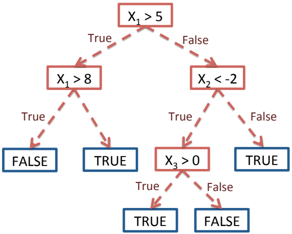


Trees

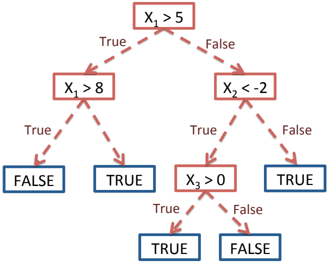
Ex: data (X_1, X_2, X_3, Y) with X_1, X_2, X_3 are real, Y Boolean

First, see if $X_1 > 5$:

- if TRUE, see if $X_1 > 8$
  - if TRUE, return FALSE
  - if FALSE, return TRUE
- if FALSE, see if $X_2 < -2$
  - if TRUE, see if $X_3 > 0$
    - if TRUE, return TRUE
    - if FALSE, return FALSE
  - if FALSE, return TRUE



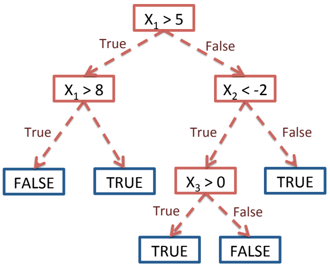
Trees



Example 1: $(X_1, X_2, X_3) = (1, 1, 1)$

Example 2: $(X_1, X_2, X_3) = (10, -3, 0)$

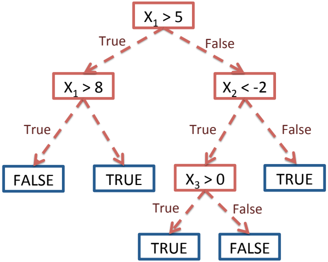
Trees



Example 1: $(X_1, X_2, X_3) = (1, 1, 1) \rightarrow$ TRUE

Example 2: $(X_1, X_2, X_3) = (10, -3, 0)$

Trees



Example 1: $(X_1, X_2, X_3) = (1, 1, 1) \rightarrow$ TRUE

Example 2: $(X_1, X_2, X_3) = (10, -3, 0) \rightarrow$ FALSE

Trees

Terminology:

- branches: one side of a split
- leaves: terminal nodes that return values

Why trees?

- trees can be used for regression or classification
  - regression: returned value is a real number
  - classification: returned value is a class
- unlike linear regression, SVMs, naive Bayes, etc, trees fit *local models*
  - in large spaces, global models may be hard to fit
  - results may be hard to interpret
- fast, interpretable predictions

Example: Predicting Electoral Results

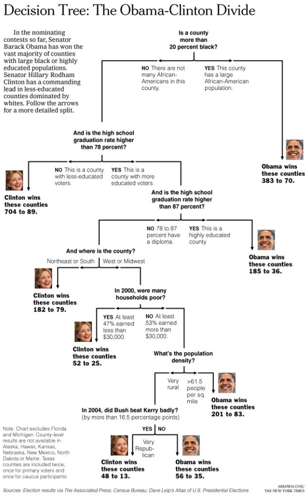
2008 Democratic primary:

- Hillary Clinton
- Barack Obama

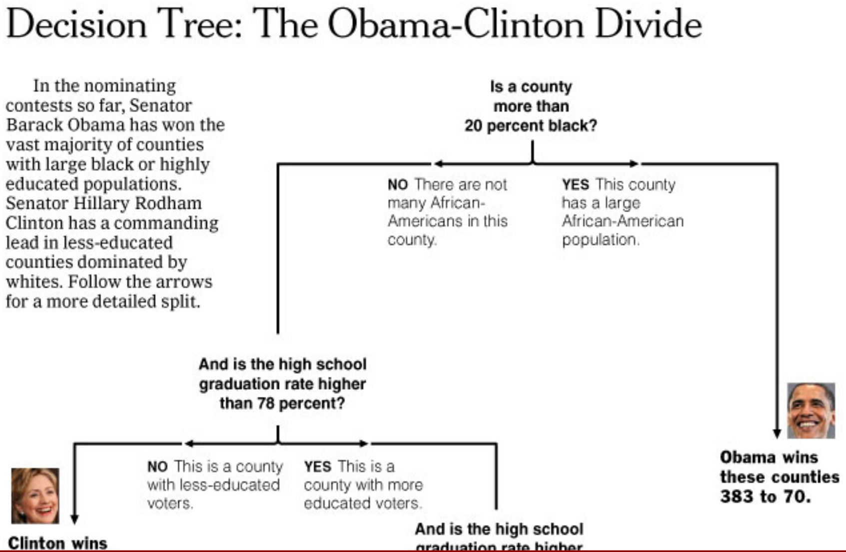
Given historical data, how will a count vote?

- can extrapolate to state level data
- might give regions to focus on increasing voter turnout
- would like to know how variables interact

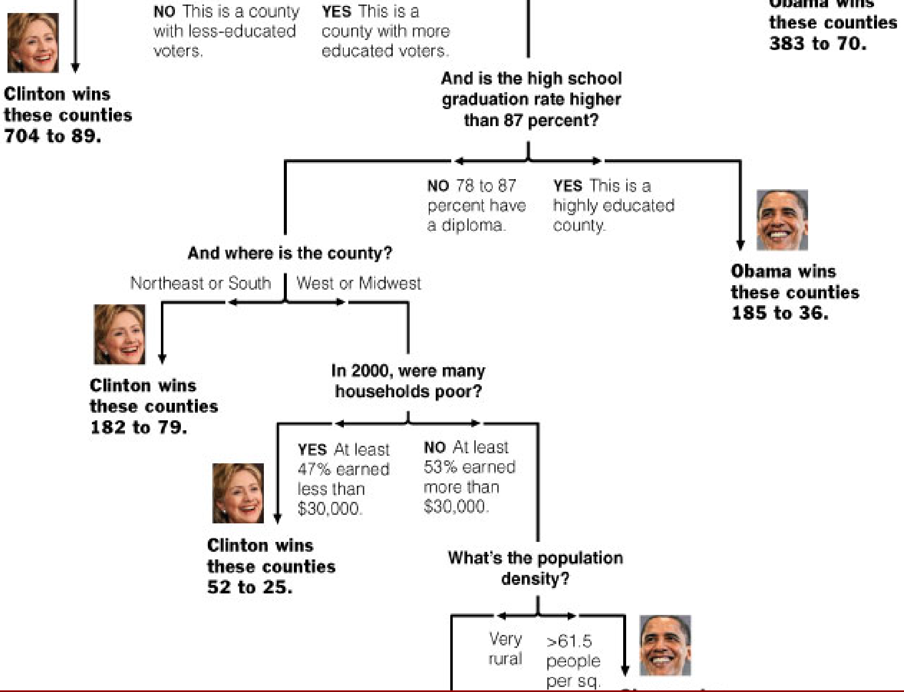
Example: Predicting Electoral Results



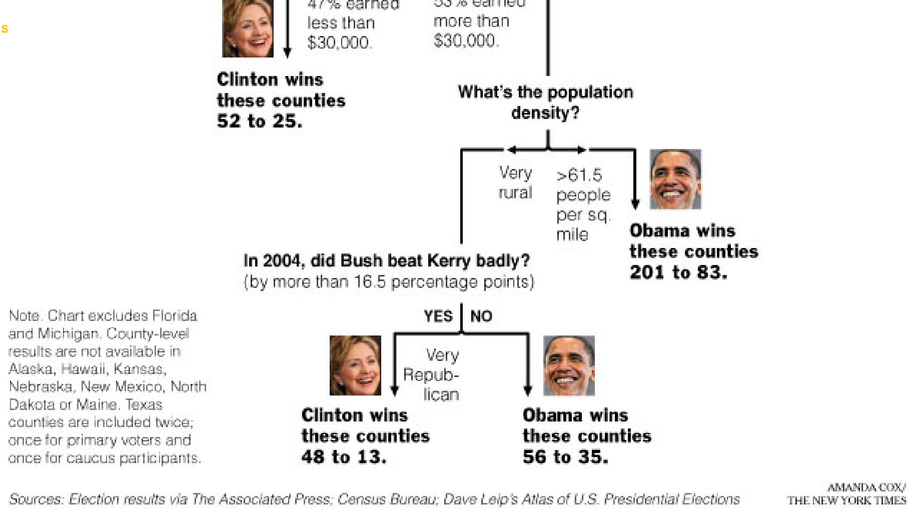
Example: Predicting Electoral Results



Decis



NO This is a county with less-educated voters.
YES This is a county with more educated voters.
these counties 383 to 70.
Clinton wins these counties 704 to 89.
And is the high school graduation rate higher than 87 percent?
NO 78 to 87 percent have a diploma.
YES This is a highly educated county.
Obama wins these counties 185 to 36.
And where is the county?
Northeast or South
West or Midwest
Clinton wins these counties 182 to 79.
In 2000, were many households poor?
YES At least 47% earned less than $30,000.
NO At least 53% earned more than $30,000.
Clinton wins these counties 52 to 25.
What's the population density?
Very rural
>61.5 people per sq.



47% earned less than $30,000.
53% earned more than $30,000.
Clinton wins these counties 52 to 25.
What's the population density?
Very rural
>61.5 people per sq. mile
Obama wins these counties 201 to 83.
In 2004, did Bush beat Kerry badly? (by more than 16.5 percentage points)
YES
NO
Very Repub-lican
Clinton wins these counties 48 to 13.
Obama wins these counties 56 to 35.
Note. Chart excludes Florida and Michigan. County-level results are not available in Alaska, Hawaii, Kansas, Nebraska, New Mexico, North Dakota or Maine. Texas counties are included twice; once for primary voters and once for caucus participants.
Sources: Election results via The Associated Press; Census Bureau; Dave Leip's Atlas of U.S. Presidential Elections
AMANDA COX/ THE NEW YORK TIMES

Decision Trees

Decision tree representation:

- Each internal node tests an attribute
- Each branch corresponds to attribute value
- Each leaf node assigns a classification

How would we represent as a function of X, Y:

- X AND Y (both must be true)
- X OR Y (either can be true)
- X XOR Y (one and only one is true)

When to Consider Decision Trees

- Instances describable by attribute-value pairs
- Target function is discrete valued
- Disjunctive hypothesis may be required
- Possibly noisy training data

Examples:

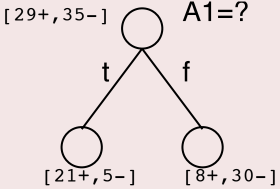
- Equipment or medical diagnosis
- Credit risk analysis
- Modeling calendar scheduling preferences

Top-Down Induction of Decision Trees

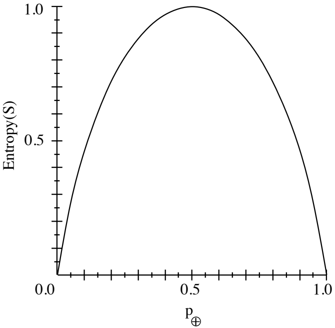
Main loop:

1. $A \leftarrow$ the “best” decision attribute for next *node*
2. Assign A as decision attribute for *node*
3. For each value of A, create new descendant of *node*
4. Sort training examples to leaf nodes
5. If training examples perfectly classified, Then STOP, Else iterate over new leaf nodes

Which attribute is best?



Entropy: Reminder



- S is a sample of training examples
- $p_\oplus$ is the proportion of positive examples in S
- $p_\ominus$ is the proportion of negative examples in S
- Entropy measures the impurity of S

$$Entropy(S) \equiv -p_\oplus \log_2 p_\oplus - p_\ominus \log_2 p_\ominus$$

Entropy

How spread out is the distribution of S:

$$p_{\oplus}(-\log_2 p_{\oplus}) + p_{\ominus}(-\log_2 p_{\ominus})$$

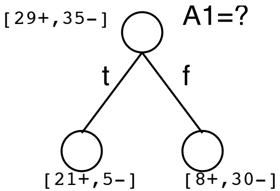
$$Entropy(S) \equiv -p_{\oplus}\log_2 p_{\oplus} - p_{\ominus}\log_2 p_{\ominus}$$

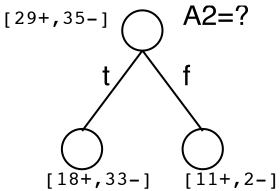
Information Gain

Which feature A would be a more useful rule in our decision tree?

$Gain(S, A)$ = expected reduction in entropy due to sorting on A

$$Gain(S, A) \equiv Entropy(S) - \sum_{v \in Values(A)} \frac{|S_v|}{|S|} Entropy(S_v)$$





$$
\begin{aligned}
H(S) &= -\frac{29}{54}\lg\left(\frac{29}{54}\right) - \frac{35}{64}\lg\left(\frac{35}{64}\right) \\
&=
\end{aligned}
$$

$$H(S) = -\frac{29}{54} \lg\left(\frac{29}{54}\right) - \frac{35}{64} \lg\left(\frac{35}{64}\right)$$
$$= 0.96$$

$$H(S) = -\frac{29}{54}\lg\left(\frac{29}{54}\right) - \frac{35}{64}\lg\left(\frac{35}{64}\right)$$
$$= 0.96$$

$$\begin{aligned}\text{Gain}(S, A_1) = 0.96 &- \frac{26}{64}\left[-\frac{5}{26}\lg\left(\frac{5}{26}\right) - \frac{21}{26}\lg\left(\frac{21}{26}\right)\right] \\ &- \frac{38}{64}\left[-\frac{8}{38}\lg\left(\frac{8}{38}\right) - \frac{30}{38}\lg\left(\frac{30}{38}\right)\right] \\ &=\end{aligned}$$

$$
\begin{aligned}
H(S) &= -\frac{29}{54}\lg\left(\frac{29}{54}\right) - \frac{35}{64}\lg\left(\frac{35}{64}\right) \\
&= 0.96
\end{aligned}
$$

$$
\begin{aligned}
\text{Gain}(S, A_1) &= 0.96 - \frac{26}{64}\left[-\frac{5}{26}\lg\left(\frac{5}{26}\right) - \frac{21}{26}\lg\left(\frac{21}{26}\right)\right] \\
&- \frac{38}{64}\left[-\frac{8}{38}\lg\left(\frac{8}{38}\right) - \frac{30}{38}\lg\left(\frac{30}{38}\right)\right] \\
&= 0.96 - 0.28 - 0.44 = 0.24
\end{aligned}
$$

$$
\begin{aligned}
H(S) &= -\frac{29}{54}\lg\left(\frac{29}{54}\right) - \frac{35}{64}\lg\left(\frac{35}{64}\right)\\
&= 0.96
\end{aligned}
$$

$$
\begin{aligned}
\mathrm{Gain}(S, A_1) = 0.96 &- \frac{26}{64}\left[-\frac{5}{26}\lg\left(\frac{5}{26}\right) - \frac{21}{26}\lg\left(\frac{21}{26}\right)\right]\\
&- \frac{38}{64}\left[-\frac{8}{38}\lg\left(\frac{8}{38}\right) - \frac{30}{38}\lg\left(\frac{30}{38}\right)\right]\\
&= 0.96 - 0.28 - 0.44 = 0.24
\end{aligned}
$$

$$
\begin{aligned}
\mathrm{Gain}(S, A_2) = 0.96 &- \frac{51}{64}\left[-\frac{18}{51}\lg\left(\frac{18}{51}\right) - \frac{33}{51}\lg\left(\frac{33}{51}\right)\right]\\
&- \frac{13}{64}\left[-\frac{11}{13}\lg\left(\frac{11}{13}\right) - \frac{2}{13}\lg\left(\frac{2}{13}\right)\right]\\
&=
\end{aligned}
$$

$$H(S) = -\frac{29}{54}\lg\left(\frac{29}{54}\right) - \frac{35}{64}\lg\left(\frac{35}{64}\right)$$
$$= 0.96$$

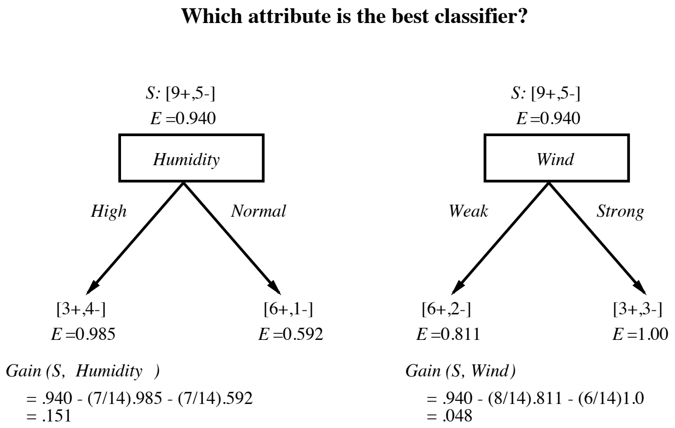
$$\text{Gain}(S, A_1) = 0.96 - \frac{26}{64}\left[-\frac{5}{26}\lg\left(\frac{5}{26}\right) - \frac{21}{26}\lg\left(\frac{21}{26}\right)\right]$$
$$- \frac{38}{64}\left[-\frac{8}{38}\lg\left(\frac{8}{38}\right) - \frac{30}{38}\lg\left(\frac{30}{38}\right)\right]$$
$$= 0.96 - 0.28 - 0.44 = 0.24$$

$$\text{Gain}(S, A_2) = 0.96 - \frac{51}{64}\left[-\frac{18}{51}\lg\left(\frac{18}{51}\right) - \frac{33}{51}\lg\left(\frac{33}{51}\right)\right]$$
$$- \frac{13}{64}\left[-\frac{11}{13}\lg\left(\frac{11}{13}\right) - \frac{2}{13}\lg\left(\frac{2}{13}\right)\right]$$
$$= 0.96 - 0.75 - 0.13 = 0.08$$

Training Examples

Day	Outlook	Temperature	Humidity	Wind	PlayTennis
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rain	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No

Selecting the Next Attribute



ID3 Algorithm

- Start at root, look for best attribute
- Repeat for subtrees at each attribute outcome
- Stop when information gain is below a threshold
- Bias: prefers shorter trees (Occam's Razor)
  - $\longrightarrow$ a short hyp that fits data unlikely to be coincidence
  - $\longrightarrow$ a long hyp that fits data might be coincidence
  - Prevents overfitting (more later)

Text classification

- Many commercial applications
- There are many applications of text classification for corporate Intranets, government departments, and Internet publishers.
- Often greater performance gains from exploiting domain-specific text features than from changing from one machine learning method to another.
- Representing features is often a big challenge (e.g., zero mean, standard variance)

Choosing what kind of classifier to use

When building a text classifier, first question: **how much training data is there currently available?**

- None?
- Very little?
- A fair amount?
- A huge amount

Choosing what kind of classifier to use

When building a text classifier, first question: **how much training data is there currently available?**

- None? **Hand write rules or use active learning**
- Very little?
- A fair amount?
- A huge amount

Choosing what kind of classifier to use

When building a text classifier, first question: **how much training data is there currently available?**

- None? **Hand write rules or use active learning**
- Very little? **Naïve Bayes**
- A fair amount?
- A huge amount

Choosing what kind of classifier to use

When building a text classifier, first question: **how much training data is there currently available?**

- None? **Hand write rules or use active learning**
- Very little? **Naïve Bayes**
- A fair amount? **SVM** (later)
- A huge amount

Choosing what kind of classifier to use

When building a text classifier, first question: **how much training data is there currently available?**

- None? **Hand write rules or use active learning**
- Very little? **Naïve Bayes**
- A fair amount? **SVM** (later)
- A huge amount **Doesn't matter, use whatever works**

Recap

- Is there a learning method that is optimal for all text classification problems?
- No, because there is a tradeoff between bias and variance.
- Factors to take into account:
  - How much training data is available?
  - How simple/complex is the problem? (linear vs. nonlinear decision boundary)
  - How noisy is the problem?
  - How stable is the problem over time?
    - For an unstable problem, it's better to use a simple and robust classifier.