Supervised learning methods
Lecture 5

David Sontag
New York University

Slides adapted from Vibhav Gogate, Carlos Guestrin, Luke Zettlemoyer, and Andrew Moore
Plan for next few weeks

Spring break: no class or lab!

Midterm exam: in class

Guest lecture: Yann LeCun on deep learning

Lab: meet & chat with project advisers (registered students only)

Project proposals due

Today: nearest neighbor methods, decision trees, random forests

PS4 released tomorrow
Nearest Neighbor Algorithm

• Learning Algorithm:
  – Store training examples

• Prediction Algorithm:
  – To classify a new example \( \mathbf{x} \) by finding the training example \((\mathbf{x}^i,y^i)\) that is nearest to \( \mathbf{x} \)
  – Guess the class \( y = y^i \)
K-Nearest Neighbor Methods

• To classify a new input vector $x$, examine the $k$-closest training data points to $x$ and assign the object to the most frequently occurring class.

common values for $k$: 3, 5
The nearest neighbor algorithm does not explicitly compute decision boundaries. However, the decision boundaries form a subset of the Voronoi diagram for the training data.

The more examples that are stored, the more complex the decision boundaries can become.
Example results for k-NN

[Figures from Hastie and Tibshirani, Chapter 13]
Nearest Neighbor

**When to Consider**
- Instance maps to points in $R^n$
- Less than 20 attributes per instance
- Lots of training data

**Advantages**
- Training is very fast
- Learn complex target functions
- Do not lose information

**Disadvantages**
- Slow at query time
- Easily fooled by irrelevant attributes
Issues

• Distance measure
  – Most common: Euclidean

• Choosing k
  – Increasing k reduces variance, increases bias

• For high-dimensional space, problem that the nearest neighbor may not be very close at all!

• Memory-based technique. Must make a pass through the data for each classification. This can be prohibitive for large data sets.
$k$-NN and irrelevant features
$k$-NN and irrelevant features
Weighted $k$-NN

- Consider the following generalization of the $k$-NN algorithm (specialized to binary classification):

$$
\hat{y}(x) \leftarrow \text{sign} \left( \sum_{i=1}^{k} y_i s(x_i, x) \right) \text{ with } s(x_i, x) = \frac{1}{\|x_i - x\|^2_2} \text{ or } s(x_i, x) = \exp \left( -\frac{\|x_i - x\|^2_2}{2\sigma^2} \right)
$$

- Weighs the $i$th training point’s label by how far $x_i$ is from $x$
**k-NN is similar to SVM with Gaussian kernel!**

- Consider the following generalization of the $k$-NN algorithm (specialized to binary classification):

  $\hat{y}(\bar{x}) \leftarrow \text{sign} \left( \sum_{i=1}^{N} y_i s(\bar{x}_i, \bar{x}) \right)$ with $s(\bar{x}_i, \bar{x}) = \frac{1}{\|\bar{x}_i - \bar{x}\|^2}$ or...

  $s(\bar{x}_i, \bar{x}) = \exp \left( -\frac{\|\bar{x}_i - \bar{x}\|^2}{2\sigma^2} \right)$

- Looks at *all* training points (i.e., $k=N$), but weighs the $i$’th training point’s label by how far $x_i$ is from $x$

- Now compare this to classification with SVM and a Gaussian kernel:

  $\hat{y}(\bar{x}) \leftarrow \text{sign} \left( \sum_{i=1}^{N} \alpha_i y_i K(\bar{x}_i, \bar{x}) \right)$

  $K(\bar{u}, \bar{v}) = \exp \left( -\frac{||\bar{u} - \bar{v}||^2}{2\sigma^2} \right)$

- The discriminant functions are nearly identical! The SVM has parameters $\alpha_i$ that can be learned
KNN Advantages

- Easy to program
- No optimization or training required
- Classification accuracy can be very good; can outperform more complex models
Decision Trees
Machine Learning in the ER

Triage Information
(blood pressure, heart rate, temperature, …)

MD comments
(free text)

Specialist consults

Physician documentation

Lab results
(continuous valued)

Repeated vital signs
(continuous values)
Measured every 30 s

Disposition

T=0

30 min

2 hrs
Can we predict infection?

Many crucial decisions about a patient’s care are made here!

Lab results (Continuous valued)

Repeated vital signs (continuous values) Measured every 30 s

Physician documentation

Specialist consults

MD comments (free text)

Repeated vital signs (continuous values) Measured every 30 s

Physician documentation

Specialist consults

MD comments (free text)

Lab results (Continuous valued)

Many crucial decisions about a patient’s care are made here!

Lab results (Continuous valued)

Repeated vital signs (continuous values) Measured every 30 s

Physician documentation

Specialist consults

MD comments (free text)

Repeated vital signs (continuous values) Measured every 30 s

Physician documentation

Specialist consults

MD comments (free text)

Lab results (Continuous valued)

Many crucial decisions about a patient’s care are made here!

Lab results (Continuous valued)

Repeated vital signs (continuous values) Measured every 30 s

Physician documentation

Specialist consults

MD comments (free text)

Repeated vital signs (continuous values) Measured every 30 s

Physician documentation

Specialist consults

MD comments (free text)

Lab results (Continuous valued)

Many crucial decisions about a patient’s care are made here!

Lab results (Continuous valued)

Repeated vital signs (continuous values) Measured every 30 s

Physician documentation

Specialist consults

MD comments (free text)

Repeated vital signs (continuous values) Measured every 30 s

Physician documentation

Specialist consults

MD comments (free text)

Lab results (Continuous valued)

Many crucial decisions about a patient’s care are made here!

Lab results (Continuous valued)

Repeated vital signs (continuous values) Measured every 30 s

Physician documentation

Specialist consults

MD comments (free text)

Repeated vital signs (continuous values) Measured every 30 s

Physician documentation

Specialist consults

MD comments (free text)

Lab results (Continuous valued)

Many crucial decisions about a patient’s care are made here!

Lab results (Continuous valued)

Repeated vital signs (continuous values) Measured every 30 s

Physician documentation

Specialist consults

MD comments (free text)

Repeated vital signs (continuous values) Measured every 30 s

Physician documentation

Specialist consults

MD comments (free text)

Lab results (Continuous valued)

Many crucial decisions about a patient’s care are made here!

Lab results (Continuous valued)

Repeated vital signs (continuous values) Measured every 30 s

Physician documentation

Specialist consults

MD comments (free text)

Repeated vital signs (continuous values) Measured every 30 s

Physician documentation

Specialist consults

MD comments (free text)

Lab results (Continuous valued)

Many crucial decisions about a patient’s care are made here!

Lab results (Continuous valued)

Repeated vital signs (continuous values) Measured every 30 s

Physician documentation

Specialist consults

MD comments (free text)

Repeated vital signs (continuous values) Measured every 30 s

Physician documentation

Specialist consults

MD comments (free text)

Lab results (Continuous valued)

Many crucial decisions about a patient’s care are made here!

Lab results (Continuous valued)

Repeated vital signs (continuous values) Measured every 30 s

Physician documentation

Specialist consults

MD comments (free text)

Repeated vital signs (continuous values) Measured every 30 s

Physician documentation

Specialist consults

MD comments (free text)

Lab results (Continuous valued)

Many crucial decisions about a patient’s care are made here!

Lab results (Continuous valued)

Repeated vital signs (continuous values) Measured every 30 s

Physician documentation

Specialist consults

MD comments (free text)

Repeated vital signs (continuous values) Measured every 30 s

Physician documentation

Specialist consults

MD comments (free text)

Lab results (Continuous valued)

Many crucial decisions about a patient’s care are made here!

Lab results (Continuous valued)

Repeated vital signs (continuous values) Measured every 30 s

Physician documentation

Specialist consults

MD comments (free text)

Repeated vital signs (continuous values) Measured every 30 s

Physician documentation

Specialist consults

MD comments (free text)

Lab results (Continuous valued)

Many crucial decisions about a patient’s care are made here!

Lab results (Continuous valued)

Repeated vital signs (continuous values) Measured every 30 s

Physician documentation

Specialist consults

MD comments (free text)

Repeated vital signs (continuous values) Measured every 30 s

Physician documentation

Specialist consults

MD comments (free text)

Lab results (Continuous valued)

Many crucial decisions about a patient’s care are made here!

Lab results (Continuous valued)

Repeated vital signs (continuous values) Measured every 30 s

Physician documentation

Specialist consults

MD comments (free text)

Repeated vital signs (continuous values) Measured every 30 s

Physician documentation

Specialist consults

MD comments (free text)

Lab results (Continuous valued)

Many crucial decisions about a patient’s care are made here!

Lab results (Continuous valued)

Repeated vital signs (continuous values) Measured every 30 s

Physician documentation

Specialist consults

MD comments (free text)

Repeated vital signs (continuous values) Measured every 30 s

Physician documentation

Specialist consults

MD comments (free text)

Lab results (Continuous valued)

Many crucial decisions about a patient’s care are made here!

Lab results (Continuous valued)

Repeated vital signs (continuous values) Measured every 30 s

Physician documentation

Specialist consults

MD comments (free text)

Repeated vital signs (continuous values) Measured every 30 s

Physician documentation

Specialist consults

MD comments (free text)

Lab results (Continuous valued)

Many crucial decisions about a patient’s care are made here!

Lab results (Continuous valued)

Repeated vital signs (continuous values) Measured every 30 s

Physician documentation

Specialist consults

MD comments (free text)

Repeated vital signs (continuous values) Measured every 30 s

Physician documentation

Specialist consults

MD comments (free text)

Lab results (Continuous valued)

Many crucial decisions about a patient’s care are made here!

Lab results (Continuous valued)

Repeated vital signs (continuous values) Measured every 30 s

Physician documentation

Specialist consults

MD comments (free text)

Repeated vital signs (continuous values) Measured every 30 s

Physician documentation

Specialist consults

MD comments (free text)

Lab results (Continuous valued)

Many crucial decisions about a patient’s care are made here!

Lab results (Continuous valued)

Repeated vital signs (continuous values) Measured every 30 s

Physician documentation

Specialist consults

MD comments (free text)

Repeated vital signs (continuous values) Measured every 30 s

Physician documentation

Specialist consults

MD comments (free text)

Lab results (Continuous valued)

Many crucial decisions about a patient’s care are made here!

Lab results (Continuous valued)

Repeated vital signs (continuous values) Measured every 30 s

Physician documentation

Specialist consults

MD comments (free text)

Repeated vital signs (continuous values) Measured every 30 s

Physician documentation

Specialist consults

MD comments (free text)

Lab results (Continuous valued)

Many crucial decisions about a patient’s care are made here!

Lab results (Continuous valued)

Repeated vital signs (continuous values) Measured every 30 s

Physician documentation

Specialist consults

MD comments (free text)

Repeated vital signs (continuous values) Measured every 30 s

Physician documentation

Specialist consults

MD comments (free text)

Lab results (Continuous valued)

Many crucial decisions about a patient’s care are made here!

Lab results (Continuous valued)

Repeated vital signs (continuous values) Measured every 30 s

Physician documentation

Specialist consults

MD comments (free text)

Repeated vital signs (continuous values) Measured every 30 s

Physician documentation

Specialist consults

MD comments (free text)

Lab results (Continuous valued)

Many crucial decisions about a patient’s care are made here!

Lab results (Continuous valued)

Repeated vital signs (continuous values) Measured every 30 s

Physician documentation

Specialist consults

MD comments (free text)

Repeated vital signs (continuous values) Measured every 30 s

Physician documentation

Specialist consults

MD comments (free text)

Lab results (Continuous valued)

Many crucial decisions about a patient’s care are made here!

Lab results (Continuous valued)

Repeated vital signs (continuous values) Measured every 30 s

Physician documentation

Specialist consults

MD comments (free text)

Repeated vital signs (continuous values) Measured every 30 s

Physician documentation

Specialist consults

MD comments (free text)

Lab results (Continuous valued)

Many crucial decisions about a patient’s care are made here!

Lab results (Continuous valued)

Repeated vital signs (continuous values) Measured every 30 s

Physician documentation

Specialist consults

MD comments (free text)

Repeated vital signs (continuous values) Measured every 30 s

Physician documentation

Specialist consults

MD comments (free text)

Lab results (Continuous valued)

Many crucial decisions about a patient’s care are made here!

Lab results (Continuous valued)

Repeated vital signs (continuous values) Measured every 30 s

Physician documentation

Specialist consults

MD comments (free text)

Repeated vital signs (continuous values) Measured every 30 s

Physician documentation

Specialist consults

MD comments (free text)

Lab results (Continuous valued)

Many crucial decisions about a patient’s care are made here!

Lab results (Continuous valued)

Repeated vital signs (continuous values) Measured every 30 s

Physician documentation

Specialist consults

MD comments (free text)

Repeated vital signs (continuous values) Measured every 30 s

Physician documentation

Specialist consults

MD comments (free text)

Lab results (Continuous valued)
Can we predict infection?

• Previous automatic approaches based on simple criteria:
  – Temperature < 96.8 °F or > 100.4 °F
  – Heart rate > 90 beats/min
  – Respiratory rate > 20 breaths/min

• Too simplified... e.g., heart rate depends on age!
Can we predict infection?

• These are the attributes we have for each patient:
  – Temperature
  – Heart rate (HR)
  – Respiratory rate (RR)
  – Age
  – Acuity and pain level
  – Diastolic and systolic blood pressure (DBP, SBP)
  – Oxygen Saturation (SaO2)

• We have these attributes + label (infection) for 200,000 patients!

• Let’s learn to classify infection
Predicting infection using decision trees
Hypotheses: decision trees $f : X \rightarrow Y$

- Each internal node tests an attribute $x_i$
- One branch for each possible attribute value $x_i = v$
- Each leaf assigns a class $y$
- To classify input $x$: traverse the tree from root to leaf, output the labeled $y$

Human interpretable!
Hypothesis space

- How many possible hypotheses?
- What functions can be represented?

<table>
<thead>
<tr>
<th>mpg</th>
<th>cylinders</th>
<th>displacement</th>
<th>horsepower</th>
<th>weight</th>
<th>acceleration</th>
<th>modelyear</th>
<th>maker</th>
</tr>
</thead>
<tbody>
<tr>
<td>good</td>
<td>4 low</td>
<td>low</td>
<td>low high</td>
<td>high</td>
<td>75 to 78</td>
<td>asia</td>
<td></td>
</tr>
<tr>
<td>bad</td>
<td>6 medium</td>
<td>medium</td>
<td>medium</td>
<td>medium</td>
<td>70 to 74</td>
<td>america</td>
<td></td>
</tr>
<tr>
<td>bad</td>
<td>4 medium</td>
<td>medium</td>
<td>medium low</td>
<td>75 to 78</td>
<td>europe</td>
<td>europe</td>
<td></td>
</tr>
<tr>
<td>bad</td>
<td>8 high</td>
<td>high</td>
<td>high low</td>
<td>70 to 74</td>
<td>america</td>
<td>america</td>
<td></td>
</tr>
<tr>
<td>bad</td>
<td>6 medium</td>
<td>medium</td>
<td>medium</td>
<td>70 to 74</td>
<td>asia</td>
<td>america</td>
<td></td>
</tr>
<tr>
<td>bad</td>
<td>4 low</td>
<td>low</td>
<td>medium</td>
<td>70 to 74</td>
<td>asia</td>
<td>asia</td>
<td></td>
</tr>
<tr>
<td>bad</td>
<td>4 low</td>
<td>low</td>
<td>low</td>
<td>70 to 74</td>
<td>asia</td>
<td>asia</td>
<td></td>
</tr>
<tr>
<td>bad</td>
<td>8 high</td>
<td>high</td>
<td>high low</td>
<td>75 to 78</td>
<td>america</td>
<td>america</td>
<td></td>
</tr>
<tr>
<td>good</td>
<td>8 high</td>
<td>high</td>
<td>high high</td>
<td>low</td>
<td>70 to 74</td>
<td>america</td>
<td></td>
</tr>
<tr>
<td>good</td>
<td>8 high</td>
<td>medium</td>
<td>high high</td>
<td>low</td>
<td>79 to 83</td>
<td>america</td>
<td></td>
</tr>
<tr>
<td>bad</td>
<td>8 high</td>
<td>high</td>
<td>high low</td>
<td>low</td>
<td>79 to 83</td>
<td>america</td>
<td></td>
</tr>
<tr>
<td>good</td>
<td>4 low</td>
<td>low</td>
<td>low</td>
<td>low</td>
<td>79 to 83</td>
<td>america</td>
<td></td>
</tr>
<tr>
<td>good</td>
<td>4 medium</td>
<td>low</td>
<td>low</td>
<td>low</td>
<td>79 to 83</td>
<td>america</td>
<td></td>
</tr>
<tr>
<td>bad</td>
<td>8 high</td>
<td>high</td>
<td>high low</td>
<td>low</td>
<td>79 to 83</td>
<td>america</td>
<td></td>
</tr>
<tr>
<td>good</td>
<td>4 low</td>
<td>medium</td>
<td>medium</td>
<td>75 to 78</td>
<td>europe</td>
<td>europe</td>
<td></td>
</tr>
<tr>
<td>bad</td>
<td>5 medium</td>
<td>medium</td>
<td>medium</td>
<td>75 to 78</td>
<td>europe</td>
<td>europe</td>
<td></td>
</tr>
</tbody>
</table>
What functions can be represented?

- Decision trees can represent any function of the input attributes!
- For Boolean functions, path to leaf gives truth table row
- But, could require exponentially many nodes...

\[ \text{cyl}=3 \lor (\text{cyl}=4 \land (\text{maker}=\text{asia} \lor \text{maker}=\text{europe})) \lor \ldots \]
Are all decision trees equal?

• Many trees can represent the same concept
• But, not all trees will have the same size!
  — e.g., \( \phi = (A \land B) \lor (\neg A \land C) \) -- ((A and B) or (not A and C))

\[
\begin{array}{c}
\text{A} \\
\text{t} \quad \text{f} \\
\text{B} \\
\text{t} \quad \text{f} \\
\text{t} \quad \text{f} \\
\text{C} \\
\text{t} \quad \text{f} \\
\text{t} \quad \text{f} \\
\text{t} \quad \text{f} \\
\text{t} \quad \text{f} \\
\end{array}
\]

• Which tree do we prefer?
Learning decision trees is hard!!!

• Learning the simplest (smallest) decision tree is an NP-complete problem [Hyafil & Rivest ’76]
• Resort to a greedy heuristic:
  – Start from empty decision tree
  – Split on **next best attribute** (feature)
  – Recurse
Key idea: Greedily learn trees using recursion

Take the Original Dataset...

And partition it according to the value of the attribute we split on
Recursive Step

mpg values: bad good

root
22 18
pchance = 0.001

cylinders = 3
cylinders = 4
cylinders = 5
cylinders = 6
cylinders = 8
0 0
4 17
1 0
8 0
9 1
Predict bad
Predict good
Predict bad
Predict bad
Predict bad

Build tree from These records..
Build tree from These records..
Build tree from These records..
Build tree from These records..

Records in which cylinders = 4
Records in which cylinders = 5
Records in which cylinders = 6
Records in which cylinders = 8
Recursively build a tree from the seven records in which there are four cylinders and the maker was based in Asia.

(Similar recursion in the other cases)
**Splitting: choosing a good attribute**

Would we prefer to split on $X_1$ or $X_2$?

![Decision tree diagram](image)

**Idea:** use counts at leaves to define probability distributions, so we can measure uncertainty!
Measuring uncertainty

• Good split if we are more certain about classification after split
  – Deterministic good (all true or all false)
  – Uniform distribution bad
  – What about distributions in between?

\[
\begin{array}{cccc}
P(Y=A) &=& \frac{1}{2} & P(Y=B) = \frac{1}{4} \\
P(Y=C) &=& \frac{1}{8} & P(Y=D) = \frac{1}{8} \\
\end{array}
\]

\[
\begin{array}{cccc}
P(Y=A) &=& \frac{1}{4} & P(Y=B) = \frac{1}{4} \\
P(Y=C) &=& \frac{1}{4} & P(Y=D) = \frac{1}{4} \\
\end{array}
\]
Entropy

Entropy $H(Y)$ of a random variable $Y$

\[ H(Y) = - \sum_{i=1}^{k} P(Y = y_i) \log_2 P(Y = y_i) \]

**More uncertainty, more entropy!**

*Information Theory interpretation:*

$H(Y)$ is the expected number of bits needed to encode a randomly drawn value of $Y$ (under most efficient code)
High, Low Entropy

• “High Entropy”
  – Y is from a uniform like distribution
  – Flat histogram
  – Values sampled from it are less predictable

• “Low Entropy”
  – Y is from a varied (peaks and valleys) distribution
  – Histogram has many lows and highs
  – Values sampled from it are more predictable

(Slide from Vibhav Gogate)
Entropy Example

\[ H(Y) = - \sum_{i=1}^{k} P(Y = y_i) \log_2 P(Y = y_i) \]

\[ P(Y=t) = \frac{5}{6} \]

\[ P(Y=f) = \frac{1}{6} \]

\[ H(Y) = - \frac{5}{6} \log_2 \frac{5}{6} - \frac{1}{6} \log_2 \frac{1}{6} \]

\[ = 0.65 \]
Conditional Entropy

Conditional Entropy $H(Y|X)$ of a random variable $Y$ conditioned on a random variable $X$

$$H(Y | X) = - \sum_{j=1}^{v} P(X = x_j) \sum_{i=1}^{k} P(Y = y_i | X = x_j) \log_2 P(Y = y_i | X = x_j)$$

Example:

<table>
<thead>
<tr>
<th>$X_1$</th>
<th>$X_2$</th>
<th>$Y$</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>F</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>F</td>
<td>T</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>F</td>
<td>F</td>
<td>F</td>
</tr>
</tbody>
</table>

$P(X_1=t) = 4/6$
$P(X_1=f) = 2/6$

$H(Y|X_1) = - 4/6 \left(1 \log_2 1 + 0 \log_2 0\right)$
$\quad \quad - 2/6 \left(1/2 \log_2 1/2 + 1/2 \log_2 1/2\right)$
$\quad \quad = 2/6$
Information gain

- Decrease in entropy (uncertainty) after splitting

\[ IG(X) = H(Y) - H(Y \mid X) \]

In our running example:

\[ IG(X_1) = H(Y) - H(Y \mid X_1) \]

\[ = 0.65 - 0.33 \]

\[ IG(X_1) > 0 \rightarrow \text{we prefer the split!} \]
Learning decision trees

• Start from empty decision tree

• Split on next best attribute (feature)
  – Use, for example, information gain to select attribute:

\[
\arg \max_i IG(X_i) = \arg \max_i H(Y) - H(Y \mid X_i)
\]

• Recurse
First split looks good! But, when do we stop?
Don’t split a node if all matching records have the same output value.
Don’t split a node if data points are identical on remaining attributes.
Base Cases: An idea

- **Base Case One**: If all records in current data subset have the same output then *don’t recurse*
- **Base Case Two**: If all records have exactly the same set of input attributes then *don’t recurse*

**Proposed Base Case 3**: If all attributes have small information gain then *don’t recurse*

*This is not a good idea*
The problem with proposed case 3

\[ y = a \text{ XOR } b \]

<table>
<thead>
<tr>
<th></th>
<th></th>
<th>y</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

The information gains:

<table>
<thead>
<tr>
<th>Input</th>
<th>Value</th>
<th>Distribution</th>
<th>Info Gain</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>0</td>
<td></td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>b</td>
<td>0</td>
<td></td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
If we omit proposed case 3:

\[ y = a \text{ XOR } b \]

The resulting decision tree:

Instead, perform **pruning** after building a tree
Decision trees will overfit
Decision trees will overfit

- **Standard decision trees have no learning bias**
  - Training set error is always zero!
    - (If there is no label noise)
  - Lots of variance
  - Must introduce some bias towards simpler trees

- **Many strategies for picking simpler trees**
  - Fixed depth
  - Fixed number of leaves

- **Random forests**
Real-Valued inputs

What should we do if some of the inputs are real-valued?

<table>
<thead>
<tr>
<th>mpg</th>
<th>cylinders</th>
<th>displacement</th>
<th>horsepower</th>
<th>weight</th>
<th>acceleration</th>
<th>modelyear</th>
<th>maker</th>
</tr>
</thead>
<tbody>
<tr>
<td>good</td>
<td>4</td>
<td>97</td>
<td>75</td>
<td>2265</td>
<td>18.2</td>
<td>77</td>
<td>asia</td>
</tr>
<tr>
<td>bad</td>
<td>6</td>
<td>199</td>
<td>90</td>
<td>2648</td>
<td>15</td>
<td>70</td>
<td>america</td>
</tr>
<tr>
<td>bad</td>
<td>4</td>
<td>121</td>
<td>110</td>
<td>2600</td>
<td>12.8</td>
<td>77</td>
<td>europe</td>
</tr>
<tr>
<td>bad</td>
<td>8</td>
<td>350</td>
<td>175</td>
<td>4100</td>
<td>13</td>
<td>73</td>
<td>america</td>
</tr>
<tr>
<td>bad</td>
<td>6</td>
<td>198</td>
<td>95</td>
<td>3102</td>
<td>16.5</td>
<td>74</td>
<td>america</td>
</tr>
<tr>
<td>bad</td>
<td>4</td>
<td>108</td>
<td>94</td>
<td>2379</td>
<td>16.5</td>
<td>73</td>
<td>asia</td>
</tr>
<tr>
<td>bad</td>
<td>4</td>
<td>113</td>
<td>95</td>
<td>2228</td>
<td>14</td>
<td>71</td>
<td>asia</td>
</tr>
<tr>
<td>bad</td>
<td>8</td>
<td>302</td>
<td>139</td>
<td>3570</td>
<td>12.8</td>
<td>78</td>
<td>america</td>
</tr>
<tr>
<td>good</td>
<td>4</td>
<td>120</td>
<td>79</td>
<td>2625</td>
<td>18.6</td>
<td>82</td>
<td>america</td>
</tr>
<tr>
<td>bad</td>
<td>8</td>
<td>455</td>
<td>225</td>
<td>4425</td>
<td>10</td>
<td>70</td>
<td>america</td>
</tr>
<tr>
<td>good</td>
<td>4</td>
<td>107</td>
<td>86</td>
<td>2464</td>
<td>15.5</td>
<td>76</td>
<td>europe</td>
</tr>
<tr>
<td>bad</td>
<td>5</td>
<td>131</td>
<td>103</td>
<td>2830</td>
<td>15.9</td>
<td>78</td>
<td>europe</td>
</tr>
</tbody>
</table>
“One branch for each numeric value” idea:

Hopeless: hypothesis with such a high branching factor will shatter any dataset and overfit
Threshold splits

• **Binary tree:** split on attribute $X$ at value $t$
  – One branch: $X < t$
  – Other branch: $X \geq t$

• **Requires small change**
  • Allow repeated splits on same variable *along a path*
The set of possible thresholds

• Binary tree, split on attribute $X$
  – One branch: $X < t$
  – Other branch: $X \geq t$

• Search through possible values of $t$
  – Seeks hard!!!

• But only a finite number of $t$’s are important:
  – Sort data according to $X$ into $\{x_1, \ldots, x_m\}$
  – Consider split points of the form $x_i + (x_{i+1} - x_i)/2$
  – Moreover, only splits between examples of different classes matter!

(Figures from Stuart Russell)
Picking the best threshold

• Suppose $X$ is real valued with threshold $t$
• Want $\text{IG}(Y \mid X:t)$, the information gain for $Y$ when testing if $X$ is greater than or less than $t$
• Define:
  - $H(Y|X:t) = p(X < t) H(Y|X < t) + p(X \geq t) H(Y|X \geq t)$
  - $\text{IG}(Y|X:t) = H(Y) - H(Y|X:t)$
  - $\text{IG}^*(Y|X) = \max_t \text{IG}(Y|X:t)$

• **Use:** $\text{IG}^*(Y|X)$ for continuous variables
Example with MPG

<table>
<thead>
<tr>
<th>Input</th>
<th>Value</th>
<th>Distribution</th>
<th>Info Gain</th>
</tr>
</thead>
<tbody>
<tr>
<td>cylinders</td>
<td>&lt; 5</td>
<td></td>
<td>0.48268</td>
</tr>
<tr>
<td></td>
<td>&gt;= 5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>displacement</td>
<td>&lt; 198</td>
<td></td>
<td>0.428205</td>
</tr>
<tr>
<td></td>
<td>&gt;= 198</td>
<td></td>
<td></td>
</tr>
<tr>
<td>horsepower</td>
<td>&lt; 94</td>
<td></td>
<td>0.48268</td>
</tr>
<tr>
<td></td>
<td>&gt;= 94</td>
<td></td>
<td></td>
</tr>
<tr>
<td>weight</td>
<td>&lt; 2789</td>
<td></td>
<td>0.379471</td>
</tr>
<tr>
<td></td>
<td>&gt;= 2789</td>
<td></td>
<td></td>
</tr>
<tr>
<td>acceleration</td>
<td>&lt; 18.2</td>
<td></td>
<td>0.159982</td>
</tr>
<tr>
<td></td>
<td>&gt;= 18.2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>modelyear</td>
<td>&lt; 81</td>
<td></td>
<td>0.319193</td>
</tr>
<tr>
<td></td>
<td>&gt;= 81</td>
<td></td>
<td></td>
</tr>
<tr>
<td>maker</td>
<td>america</td>
<td></td>
<td>0.0437265</td>
</tr>
<tr>
<td></td>
<td>asia</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>europe</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Example tree for our continuous dataset
What you need to know about decision trees

- Decision trees are one of the most popular ML tools
  - Easy to understand, implement, and use
  - Computationally cheap (to solve heuristically)
- Information gain to select attributes (ID3, C4.5,...)
- Presented for classification, can be used for regression and density estimation too
- Decision trees will overfit!!!
  - Must use tricks to find “simple trees”, e.g.,
    - Fixed depth/Early stopping
    - Pruning
  - Or, use ensembles of different trees (random forests)
Ensemble learning

Slides adapted from Navneet Goyal, Tan, Steinbach, Kumar, Vibhav Gogate
Ensemble methods

Machine learning competition with a $1 million prize
Bias/Variance Tradeoff

Hastie, Tibshirani, Friedman “Elements of Statistical Learning” 2001
Reduce Variance Without Increasing Bias

• Averaging reduces variance:

$$\overline{Var(X)} = \frac{Var(X)}{N}$$

(when predictions are independent)

Average models to reduce model variance

One problem:
  only one training set
  where do multiple models come from?
Bagging: Bootstrap Aggregation

• Leo Breiman (1994)
• Take repeated **bootstrap samples** from training set $D$
• *Bootstrap sampling*: Given set $D$ containing $N$ training examples, create $D'$ by drawing $N$ examples at random **with replacement** from $D$.

• Bagging:
  – Create $k$ bootstrap samples $D_1 \ldots D_k$.
  – Train distinct classifier on each $D_i$.
  – Classify new instance by majority vote / average.
General Idea

Step 1: Create Multiple Data Sets

Step 2: Build Multiple Classifiers

Step 3: Combine Classifiers
Example of Bagging

• Sampling with replacement

<table>
<thead>
<tr>
<th>Data ID</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Original Data</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bagging (Round 1)</td>
<td>7</td>
<td>8</td>
<td>10</td>
<td>8</td>
<td>2</td>
<td>5</td>
<td>10</td>
<td>10</td>
<td>5</td>
<td>9</td>
</tr>
<tr>
<td>Bagging (Round 2)</td>
<td>1</td>
<td>4</td>
<td>9</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>7</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>Bagging (Round 3)</td>
<td>1</td>
<td>8</td>
<td>5</td>
<td>10</td>
<td>5</td>
<td>5</td>
<td>9</td>
<td>6</td>
<td>3</td>
<td>7</td>
</tr>
</tbody>
</table>

• Build classifier on each bootstrap sample
• Each data point has probability $(1 – 1/n)^n$ of being selected as test data
• Training data = $1 - (1 – 1/n)^n$ of the original data
Bagging Example
decision tree learning algorithm; very similar to ID3

CART decision boundary
100 bagged trees

shades of blue/red indicate strength of vote for particular classification
Random Forests

- Ensemble method specifically designed for decision tree classifiers

- Introduce two sources of randomness: “Bagging” and “Random input vectors”
  - Bagging method: each tree is grown using a bootstrap sample of training data
  - Random vector method: At each node, best split is chosen from a random sample of $m$ attributes instead of all attributes
Random Forests

Original Training data

Step 1: Create random vectors

Step 2: Use random vector to build multiple decision trees

Step 3: Combine decision trees

Figure 5.40. Random forests.
Random Forests Algorithm

1. For $b = 1$ to $B$:

   (a) Draw a bootstrap sample $Z^*$ of size $N$ from the training data.

   (b) Grow a random-forest tree $T_b$ to the bootstrapped data, by recursively repeating the following steps for each terminal node of the tree, until the minimum node size $n_{\text{min}}$ is reached.

      i. Select $m$ variables at random from the $p$ variables.

      ii. Pick the best variable/split-point among the $m$.

      iii. Split the node into two daughter nodes.

2. Output the ensemble of trees $\{T_b\}_1^B$.

To make a prediction at a new point $x$:

Regression: $\hat{f}_{\text{rf}}^B(x) = \frac{1}{B} \sum_{b=1}^{B} T_b(x)$.

Classification: Let $\hat{C}_b(x)$ be the class prediction of the $b$th random-forest tree. Then $\hat{C}_{\text{rf}}^B(x) = \text{majority vote} \{\hat{C}_b(x)\}_1^B$. 