Outline

• Tree representation
• Brief information theory
• Learning decision trees
• Bagging
• Random forests
Decision trees

- Non-linear classifier
- Easy to use
- Easy to interpret
- Susceptible to overfitting but can be avoided.
Anatomy of a decision tree

Each node is a test on one attribute

Possible attribute values of the node

Leafs are the decisions
Anatomy of a decision tree

Each node is a test on one attribute

Possible attribute values of the node

Leafs are the decisions
Anatomy of a decision tree

- Each node is a test on one attribute
- Possible attribute values of the node
- Leafs are the decisions
To ‘play tennis’ or not.

A new test example: (Outlook==rain) and (not Windy==false)

Pass it on the tree -> Decision is yes.
To ‘play tennis’ or not.

(Outlook == overcast) -> yes
(Outlook == rain) and (not Windy == false) -> yes
(Outlook == sunny) and (Humidity == normal) -> yes
Decision trees

Decision trees represent a disjunction of conjunctions of constraints on the attribute values of instances.

(Outlook == overcast)
  OR
(((Outlook == rain) and (not Windy == false))
  OR
((Outlook == sunny) and (Humidity == normal))
=> yes play tennis
Y = ((A and B) or ((not A) and C))
Same concept different representation

\[ Y = ((A \text{ and } B) \text{ or } ((\text{not } A) \text{ and } C)) \]
Which attribute to select for splitting?

the distribution of each class (not attribute)

This is bad splitting…
How do we choose the test?

Which attribute should be used as the test?

Intuitively, you would prefer the one that separates the training examples as much as possible.
Information Gain

Information gain is one criteria to decide on the attribute.
Imagine:
1. Someone is about to tell you your own name
2. You are about to observe the outcome of a dice roll
2. You are about to observe the outcome of a coin flip
3. You are about to observe the outcome of a biased coin flip

Each situation have a different *amount of uncertainty* as to what outcome you will observe.
Information

Information:
reduction in uncertainty (amount of surprise in the outcome)

\[ I(E) = \log_2 \frac{1}{p(x)} = -\log_2 p(x) \]

If the probability of this event happening is small and it happens the information is large.

- Observing the outcome of a coin flip is head \[ I = -\log_2 1/2 = 1 \]
- Observe the outcome of a dice is 6 \[ I = -\log_2 1/6 = 2.58 \]
Entropy

The *expected amount of information* when observing the output of a random variable $X$

$$H(X) = E(I(X)) = \sum_i p(x_i)I(x_i) = -\sum_i p(x_i) \log_2 p(x_i)$$

If there $X$ can have 8 outcomes and all are equally likely

$$H(X) = -\sum_i 1/8 \log_2 1/8 = 3 \text{ bits}$$
Entropy

If there are $k$ possible outcomes

$$H(X) \leq \log_2 k$$

Equality holds when all outcomes are equally likely

The more the probability distribut the deviates from uniformity the lower the entropy
Entropy, purity

Entropy measures the purity

\[ 4 + 4 - \]
\[ 8 + 0 - \]

The distribution is less uniform
Entropy is lower
The node is purer
Conditional entropy

\[ H(X) = -\sum_i p(x_i) \log_2 p(x_i) \]

\[ H(X \mid Y) = -\sum_j p(y_j) H(X \mid Y = y_j) \]

\[ = -\sum_j p(y_j) \sum_i p(x_i \mid y_j) \log_2 p(x_i \mid y_j) \]
Information gain

\[ IG(X,Y) = H(X) - H(X|Y) \]

Reduction in uncertainty by knowing \( Y \)

Information gain:
(information before split) – (information after split)
Information Gain

Information gain:
(information before split) – (information after split)
Example

Attributes | Labels | Count
---|---|---
T | T | + | 2
T | F | + | 2
F | T | - | 5
F | F | + | 1

Which one do we choose X1 or X2?

\[
\text{IG}(X1,Y) = H(Y) - H(Y|X1)
\]

\[
H(Y) = -(5/10 \log(5/10) - 5/10\log(5/10)) = 1
\]

\[
H(Y|X1) = P(X1=T)H(Y|X1=T) + P(X1=F)H(Y|X1=F)
\]

\[
= 4/10 (1\log 1 + 0 \log 0) + 6/10 (5/6\log 5/6 + 1/6\log 1/6)
\]

\[
= 0.39
\]

Information gain \((X1,Y) = 1 - 0.39 = 0.61\)
Which one do we choose?

<table>
<thead>
<tr>
<th>X1</th>
<th>X2</th>
<th>Y</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>T</td>
<td>+</td>
<td>2</td>
</tr>
<tr>
<td>T</td>
<td>F</td>
<td>+</td>
<td>2</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
<td>-</td>
<td>5</td>
</tr>
<tr>
<td>F</td>
<td>F</td>
<td>+</td>
<td>1</td>
</tr>
</tbody>
</table>

Information gain (X1,Y)= 0.61
Information gain (X2,Y)= 0.12

Pick the variable which provides the most information gain about Y  
Pick X1
Recurse on branches

<table>
<thead>
<tr>
<th>X1</th>
<th>X2</th>
<th>Y</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>T</td>
<td>+</td>
<td>2</td>
</tr>
<tr>
<td>T</td>
<td>F</td>
<td>+</td>
<td>2</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
<td>-</td>
<td>5</td>
</tr>
<tr>
<td>F</td>
<td>F</td>
<td>+</td>
<td>1</td>
</tr>
</tbody>
</table>

One branch

The other branch
Caveats

- The number of possible values influences the information gain.

- The more possible values, the higher the gain (the more likely it is to form small, but pure partitions)
Purity (diversity) measures

Purity (Diversity) Measures:

- Gini (population diversity)
- Information Gain
- Chi-square Test
Overfitting

• You can perfectly fit to any training data
• Zero bias, high variance

Two approaches:
1. Stop growing the tree when further splitting the data does not yield an improvement
2. Grow a full tree, then prune the tree, by eliminating nodes.
Bagging

• Bagging or *bootstrap aggregation* a technique for reducing the variance of an estimated prediction function.

• For classification, a *committee* of trees each cast a vote for the predicted class.
Bootstrap

The basic idea:

randomly draw datasets \textit{with replacement} from the training data, each sample \textit{the same size as the original training set}
Bagging

Create bootstrap samples from the training data

N examples

M features
Random Forest Classifier

Construct a decision tree

N examples → M features → Decision Tree

Location Similarity → Gene Expression → Domain-motif

Gene Expression → Neighbor Function Similarity

Degree → Interact

Interact → Not interact

Interact → Not interact

Interact → Not interact

Interact → Not interact
Random Forest Classifier

- N examples
- M features

Take the majority vote
Bagging

\[ Z = \{(x_1, y_1), (x_2, y_2), \ldots, (x_N, y_N)\} \]

\[ Z^b \text{ where } b = 1, \ldots, B. \]

The prediction at input \( x \) when bootstrap sample \( b \) is used for training

\[ \hat{f}_{bag}(x) = \frac{1}{B} \sum_{b=1}^{B} \hat{f}^*_b(x). \]

http://www-stat.stanford.edu/~hastie/Papers/ESLII.pdf (Chapter 8.7)
Bagging: an simulated example

Generated a sample of size $N = 30$, with two classes and $p = 5$ features, each having a standard Gaussian distribution with pairwise Correlation 0.95.

The response $Y$ was generated according to

$\Pr(Y = 1/x_1 \leq 0.5) = 0.2,$

$\Pr(Y = 0/x_1 > 0.5) = 0.8.$
Bagging

Notice the bootstrap trees are different than the original tree.
Bagging

Treat the voting Proportions as probabilities

FIGURE 8.10. Error curves for the bagging example of Figure 8.9. Shown is the test error of the original tree and bagged trees as a function of the number of bootstrap samples. The orange points correspond to the consensus vote, while the green points average the probabilities.

Hastie

http://www-stat.stanford.edu/~hastie/Papers/ESLII.pdf  Example 8.7.1
Random forest classifier

Random forest classifier, an extension to bagging which uses *de-correlated* trees.
Random Forest Classifier

Training Data

N examples

M features
Random Forest Classifier

Create bootstrap samples from the training data

N examples

M features
Random Forest Classifier

Construct a decision tree

N examples

M features

Location Similarity

Gene Expression

Domain-motif

Neighbor Function Similarity

Interact

Not interact

Degree

Gene Expression

Interact

Not interact

Interact

Not interact

Interact

Not interact
Random Forest Classifier

At each node in choosing the split feature choose only among $m<M$ features
Random Forest Classifier

Create decision tree from each bootstrap sample

N examples -> M features

Don't understand the last part of the text.
Random Forest Classifier

- N examples
- M features

Take the majority vote
FIGURE 15.9. Correlations between pairs of trees drawn by a random-forest regression algorithm, as a function of $m$. The boxplots represent the correlations at 600 randomly chosen prediction points $x$. 
Random forest

Available package:
http://www.stat.berkeley.edu/~breiman/RandomForests/cc_home.htm

To read more:
http://www-stat.stanford.edu/~hastie/Papers/ESLII.pdf