Admin

Assignment 1 available and is due on Friday (printed out at the beginning of class)

Door code for MBH632

Keep up with the reading

Videos

Quick refresher from last time...

Representing examples

What is an example?
How is it represented?
Features

Examples

\[ f_1, f_2, f_3, \ldots, f_n \]

Features are the questions we can ask about the examples

How our algorithms actually “view” the data

Classification revisited

Examples

\[ \text{red, round, leaf, } 3\text{oz}, \ldots \]
\[ \text{green, round, no leaf, } 4\text{oz}, \ldots \]
\[ \text{yellow, curved, no leaf, } 4\text{oz}, \ldots \]
\[ \text{green, curved, no leaf, } 5\text{oz}, \ldots \]

Label

\[ \text{apple} \]
\[ \text{apple} \]
\[ \text{banana} \]
\[ \text{banana} \]

During learning/training/induction, learn a model of what distinguishes apples and bananas based on the features

Model/classifier

Learn

Predict

Apple or banana?

The model can then classify a new example based on the features
Classification revisited

The model can then classify a new example based on the features.

Why?

Classification revisited

Training data
- red, round, no leaf, 4oz, ...
- green, round, no leaf, 4oz, ...
- yellow, curved, no leaf, 4oz, ...
- green, curved, no leaf, 5oz, ...

Test set
- red, round, leaf, 3oz, ...
- green, round, no leaf, 4oz, ...
- yellow, curved, no leaf, 4oz, ...
- green, curved, no leaf, 5oz, ...

Classification revisited

Training data
- red, round, leaf, 3oz, ...
- green, round, no leaf, 4oz, ...
- yellow, curved, no leaf, 4oz, ...
- green, curved, no leaf, 5oz, ...

Test set
- red, round, no leaf, 4oz, ...
- red, round, no leaf, 4oz, ...
- red, round, no leaf, 4oz, ...
- red, round, no leaf, 4oz, ...

Past predicts future

Training data
- apple
- apple
- banana
- banana

Test set
- red, round, no leaf, 4oz, ...
- red, round, no leaf, 4oz, ...
- red, round, no leaf, 4oz, ...
- red, round, no leaf, 4oz, ...

Learning is about generalizing from the training data.

What does this assume about the training and test set?
Past predicts future

Training data

Test set

Past predicts future

Training data

Test set

Not always the case, but we’ll often assume it is!

Not always the case, but we’ll often assume it is!

More technically...

We are going to use the probabilistic model of learning.

There is some probability distribution over example/label pairs called the data generating distribution.

Both the training data and the test set are generated based on this distribution.

What is a probability distribution?

Probability distribution

Describes how likely (i.e. probable) certain events are.
Probability distribution

Training data

High probability
- round apples
- curved bananas
- apples with leaves
- ...

Low probability
- curved apples
- red bananas
- yellow apples
- ...

data generating distribution

Training data

Test set

data generating distribution

data generating distribution

Training data

Test set

data generating distribution

data generating distribution

Training data

Test set

data generating distribution
A sample data set

<table>
<thead>
<tr>
<th>Hour</th>
<th>Weather</th>
<th>Accident</th>
<th>Stall</th>
<th>Commute</th>
</tr>
</thead>
<tbody>
<tr>
<td>8 AM</td>
<td>Sunny</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>8 AM</td>
<td>Stormy</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>9 AM</td>
<td>Sunny</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>9 AM</td>
<td>Sunny</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>8 AM</td>
<td>Sunny</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>8 AM</td>
<td>Rainy</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>9 AM</td>
<td>Stormy</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>9 AM</td>
<td>Rainy</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>8 AM</td>
<td>Sunny</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
</tbody>
</table>

8 AM, Rainy, Yes, No? Can you describe a “model” that could be used to make decisions in general?

Decision trees

Tree with internal nodes labeled by features

Branches are labeled by tests on that feature

Leaves labeled with classes

Decision trees

Tree with internal nodes labeled by features

Branches are labeled by tests on that feature

Leaves labeled with classes
**To ride or not to ride, that is the question...**

<table>
<thead>
<tr>
<th>Terrain</th>
<th>Unicycle-type</th>
<th>Weather</th>
<th>Go-For-Ride?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Trail</td>
<td>Normal</td>
<td>Rainy</td>
<td>NO</td>
</tr>
<tr>
<td>Road</td>
<td>Normal</td>
<td>Sunny</td>
<td>YES</td>
</tr>
<tr>
<td>Trail</td>
<td>Mountain</td>
<td>Sunny</td>
<td>YES</td>
</tr>
<tr>
<td>Road</td>
<td>Normal</td>
<td>Rainy</td>
<td>YES</td>
</tr>
<tr>
<td>Road</td>
<td>Mountain</td>
<td>Rainy</td>
<td>YES</td>
</tr>
<tr>
<td>Trail</td>
<td>Normal</td>
<td>Snowy</td>
<td>NO</td>
</tr>
<tr>
<td>Road</td>
<td>Normal</td>
<td>Rainy</td>
<td>YES</td>
</tr>
<tr>
<td>Road</td>
<td>Mountain</td>
<td>Snowy</td>
<td>YES</td>
</tr>
<tr>
<td>Trail</td>
<td>Normal</td>
<td>Sunny</td>
<td>NO</td>
</tr>
<tr>
<td>Road</td>
<td>Normal</td>
<td>Snowy</td>
<td>NO</td>
</tr>
<tr>
<td>Trail</td>
<td>Mountain</td>
<td>Snowy</td>
<td>YES</td>
</tr>
</tbody>
</table>

**Recursive approach**

Base case: If all data belong to the same class, create a leaf node with that label.

Otherwise:
- calculate the “score” for each feature if we used it to split the data
- pick the feature with the highest score, partition the data based on that data value and call recursively
Partitioning the data

<table>
<thead>
<tr>
<th>Terrain</th>
<th>Unicycle-Type</th>
<th>Weather</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>Trail</td>
<td>Normal</td>
<td>Sunny</td>
<td>NO</td>
</tr>
<tr>
<td>Road</td>
<td>Normal</td>
<td>Sunny</td>
<td>YES</td>
</tr>
<tr>
<td>Trail</td>
<td>Mountain</td>
<td>Sunny</td>
<td>YES</td>
</tr>
<tr>
<td>Road</td>
<td>Mountain</td>
<td>Snowy</td>
<td>YES</td>
</tr>
<tr>
<td>Trail</td>
<td>Normal</td>
<td>Snowy</td>
<td>NO</td>
</tr>
<tr>
<td>Road</td>
<td>Normal</td>
<td>Snowy</td>
<td>NO</td>
</tr>
<tr>
<td>Trail</td>
<td>Mountain</td>
<td>Snowy</td>
<td>YES</td>
</tr>
</tbody>
</table>

**Terrain**

- Road: YES 4, NO 1; Trail: YES 2, NO 3
- YES: 2, NO: 3

**Unicycle-Type**

- Mountain: YES 4, NO 0; Normal: YES 2, NO 4
- YES: 2, NO: 4

**Weather**

- Sunny: YES 2, NO 1; Rainy: YES 2, NO 1
- YES: 2, NO: 2

---

**What score should we use?**

If we just stopped here, which tree would be best?

How could we make these into decision trees?
Training error: the average error over the training set

For classification, the most common "error" is the number of mistakes

Training error for each of these?