CS 480/680
Introduction to Machine Learning

Lecture 10
Decision Trees

Kathryn Simone

10 October 2024

UNIVERSITY OF WATERLOO | FACULTY OF MATHEMATICS

Interpretability is a concern when human life is on the line

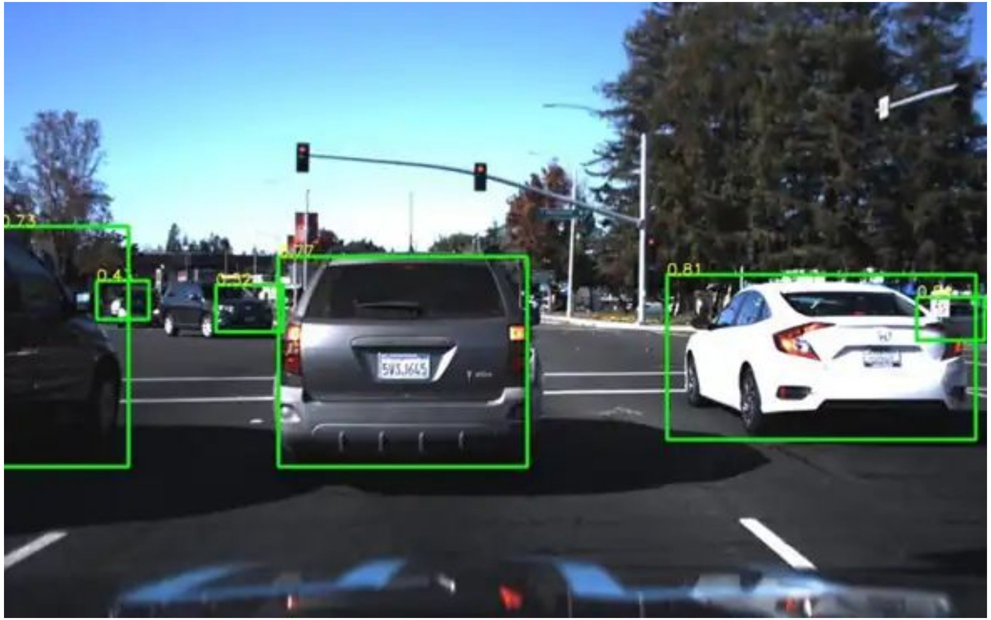
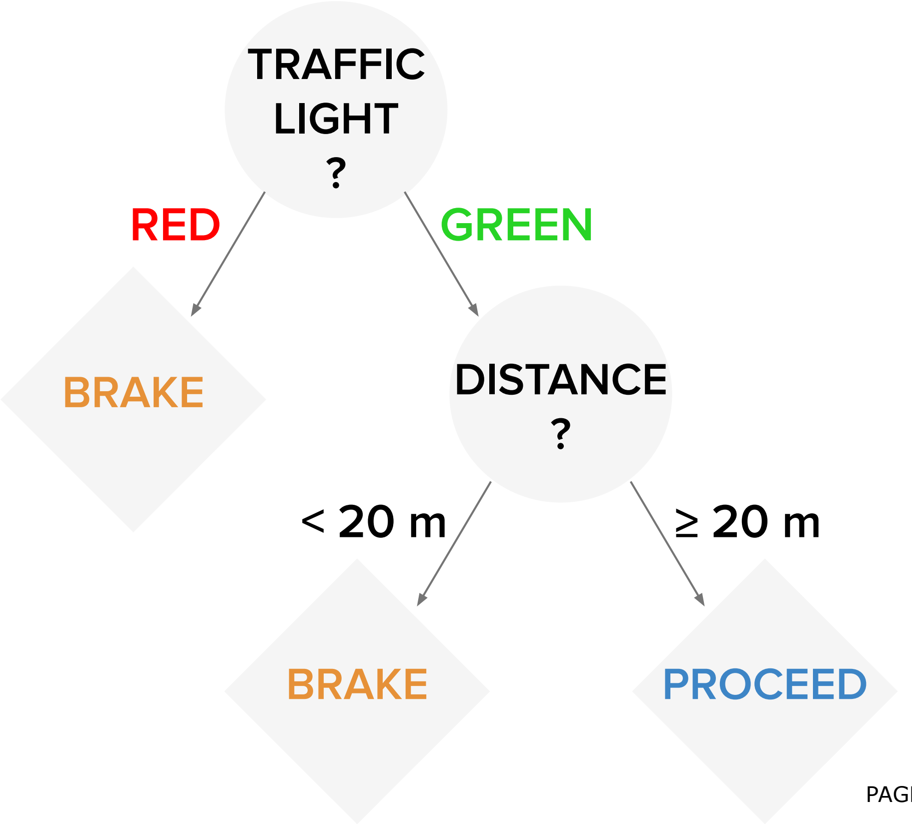
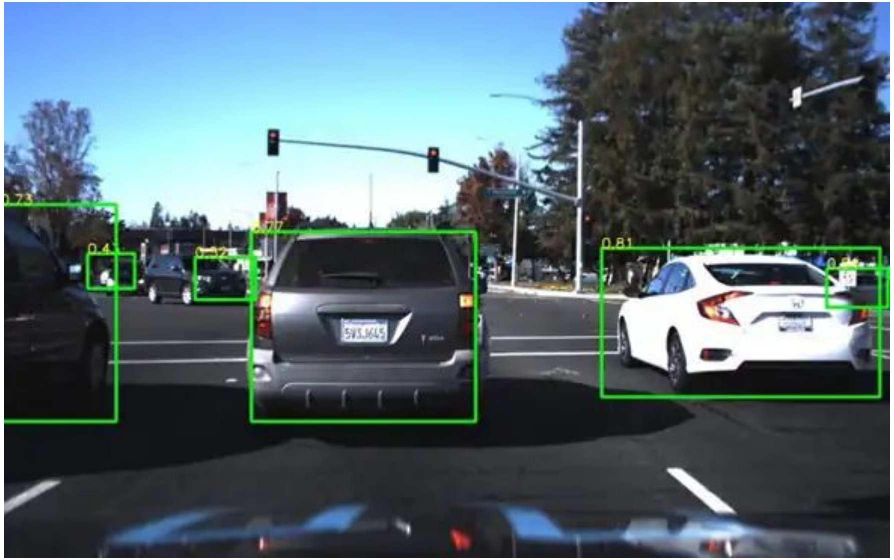


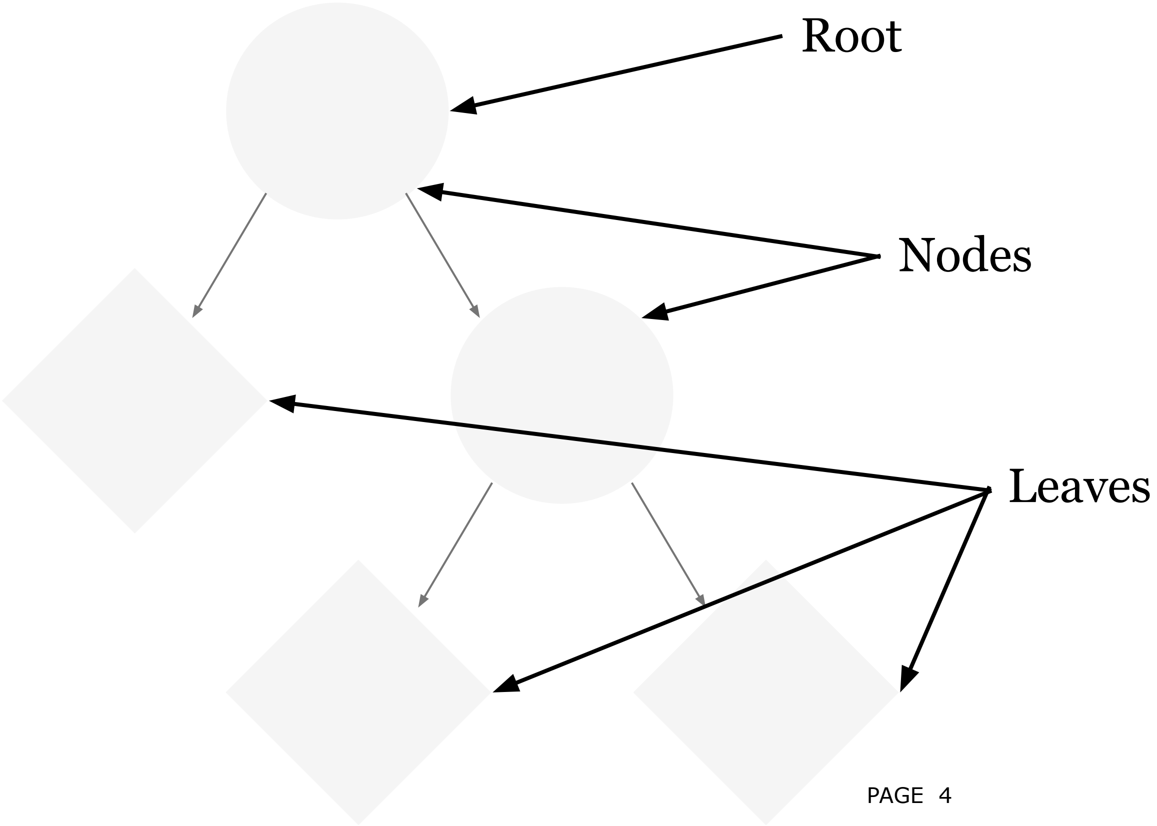
Figure: www.labellerr.com

A decision tree is a recursive partitioning model

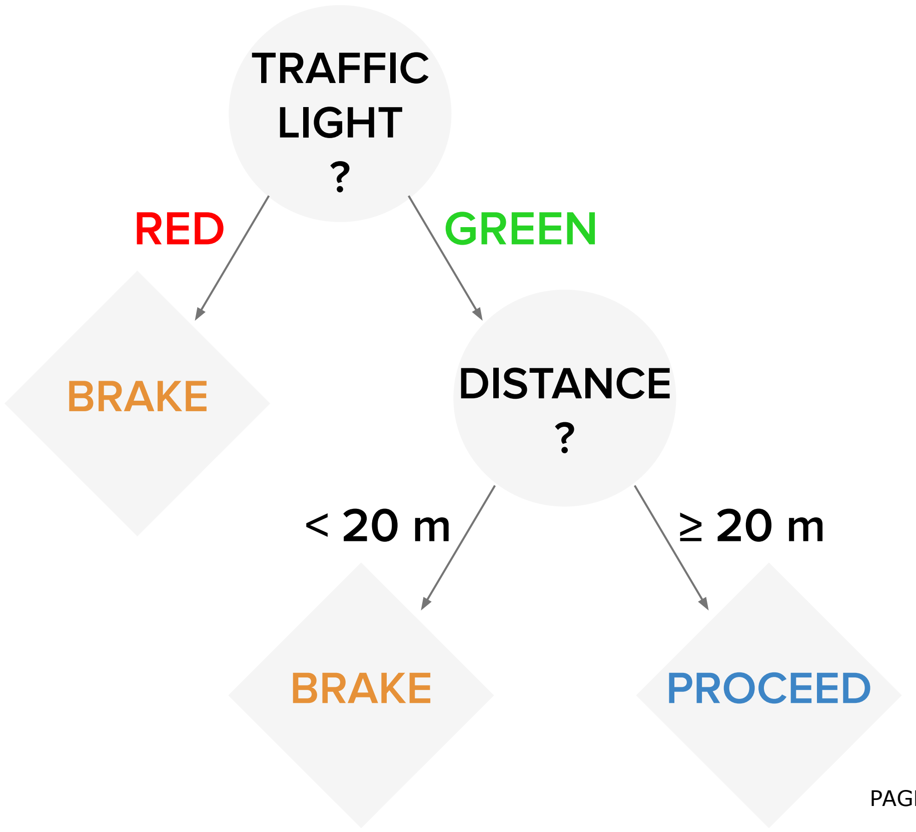


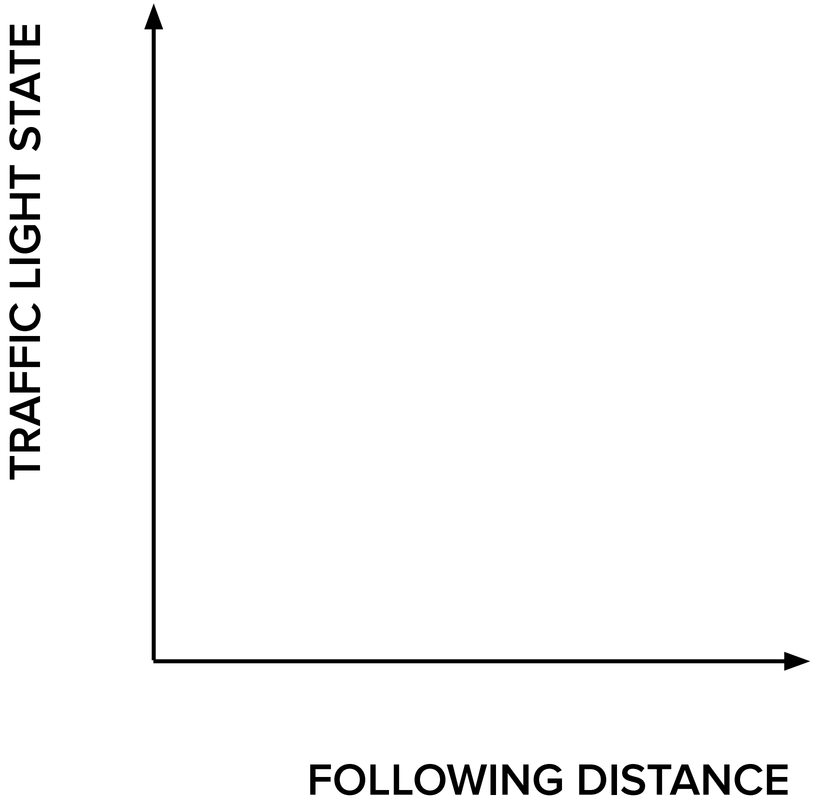


Anatomy of a tree

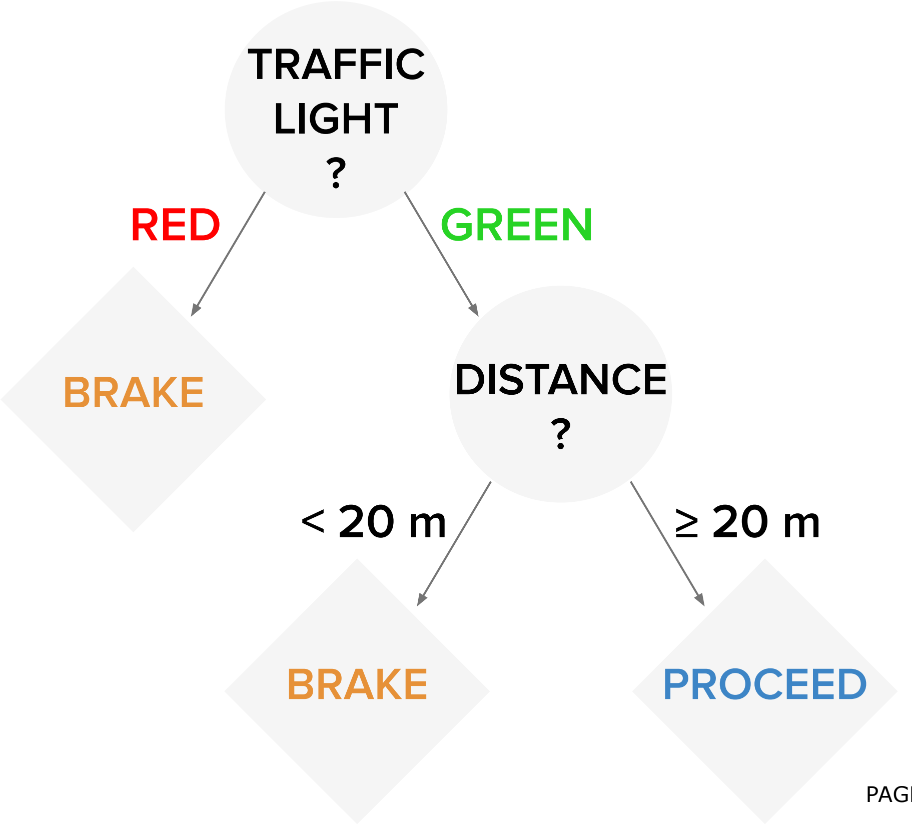


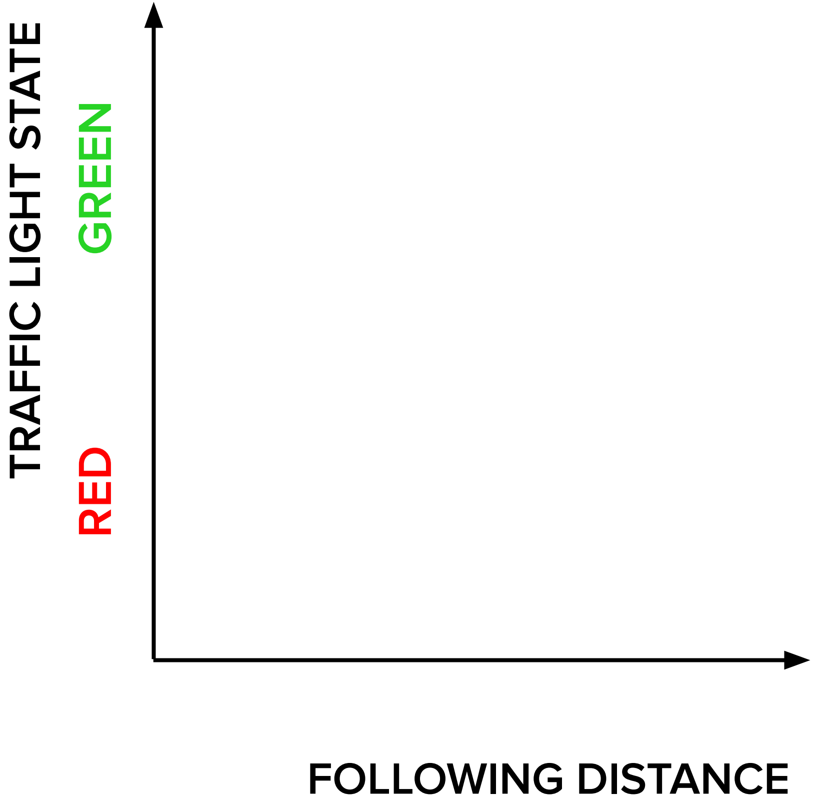
Visualizing decision trees



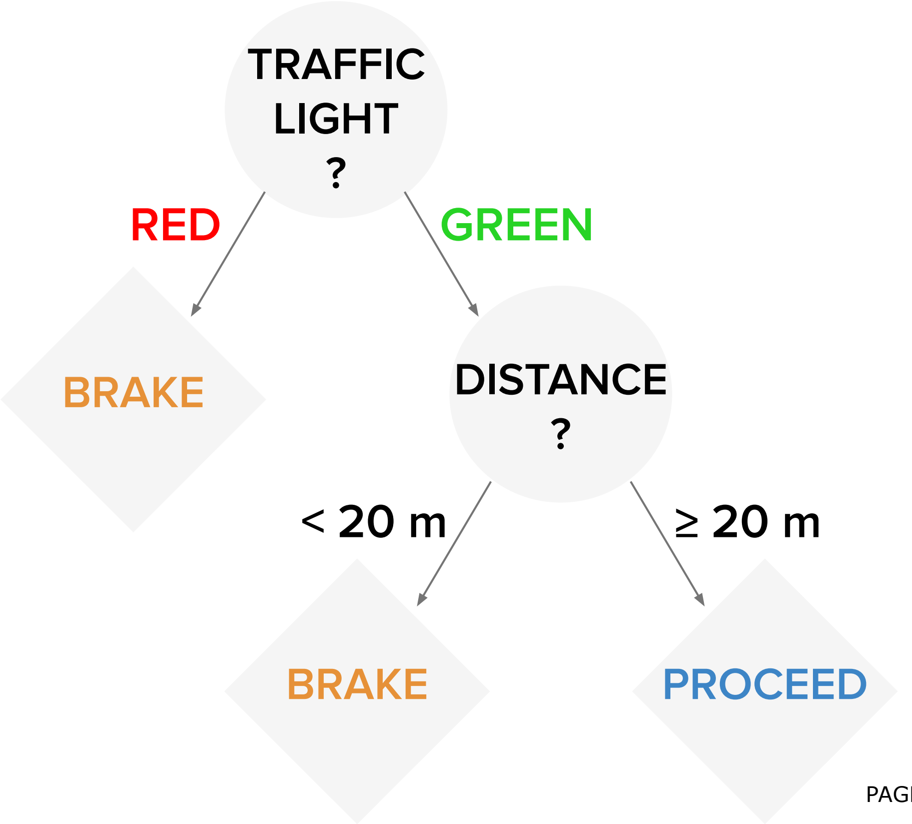


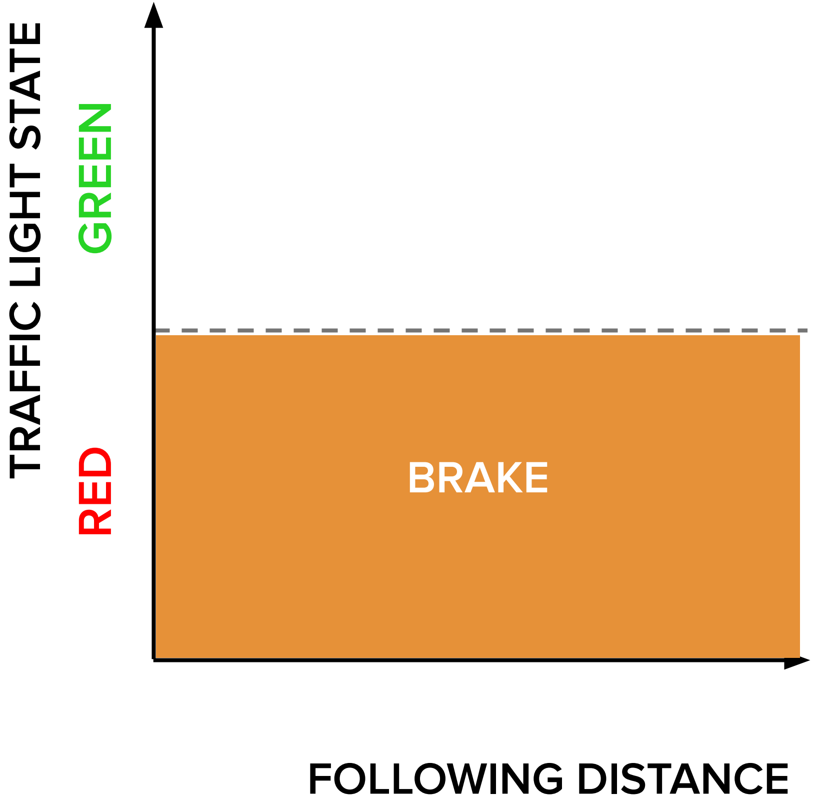
Visualizing decision trees



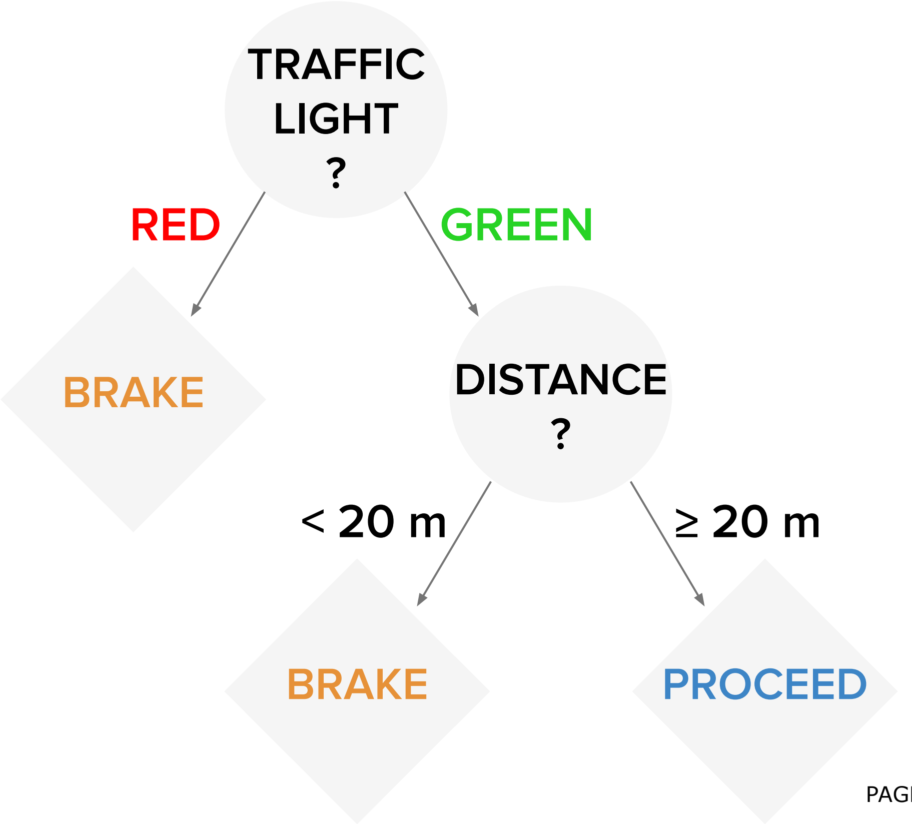


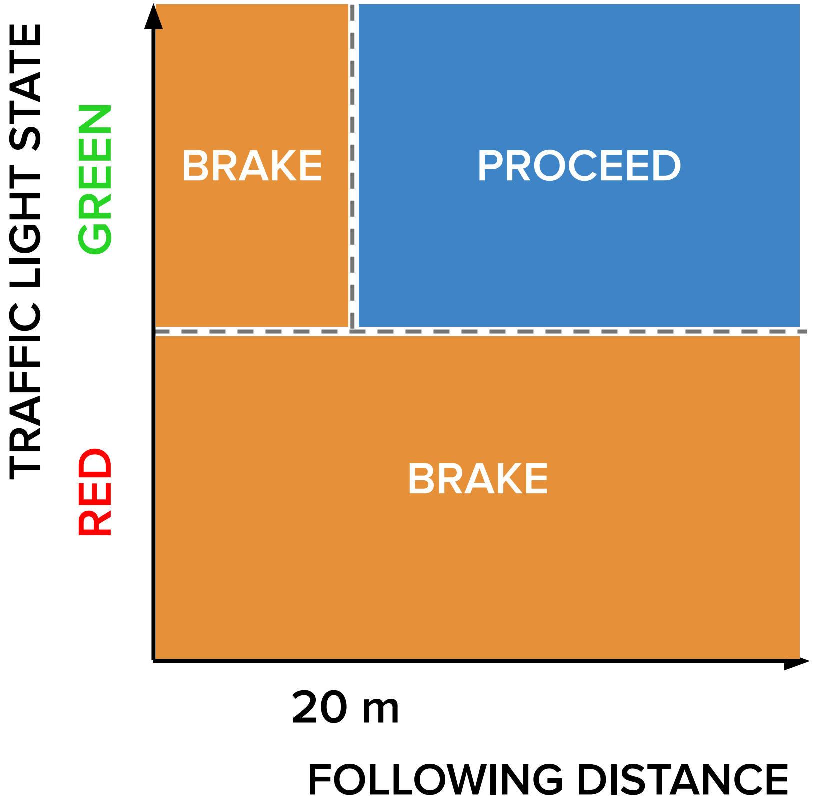
Visualizing decision trees





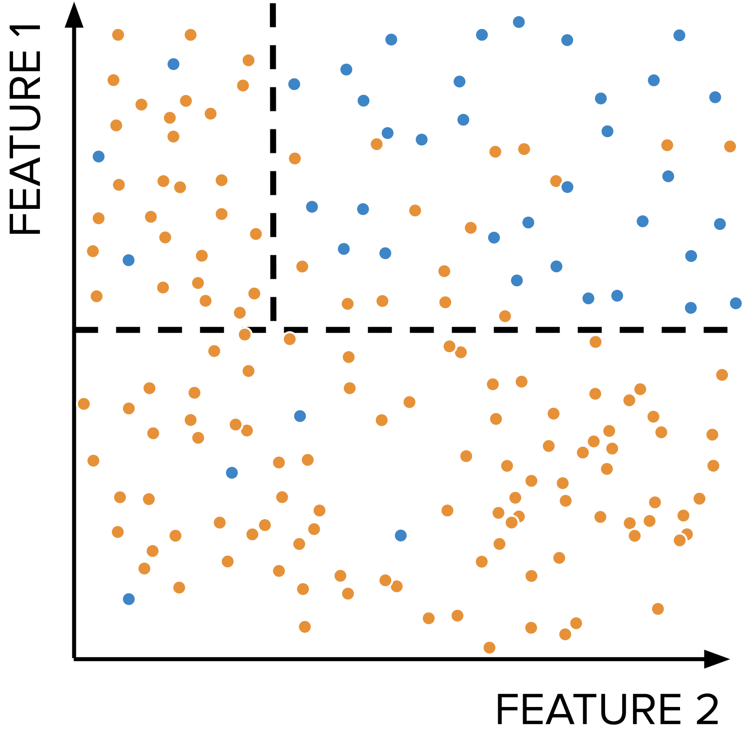
Visualizing decision trees



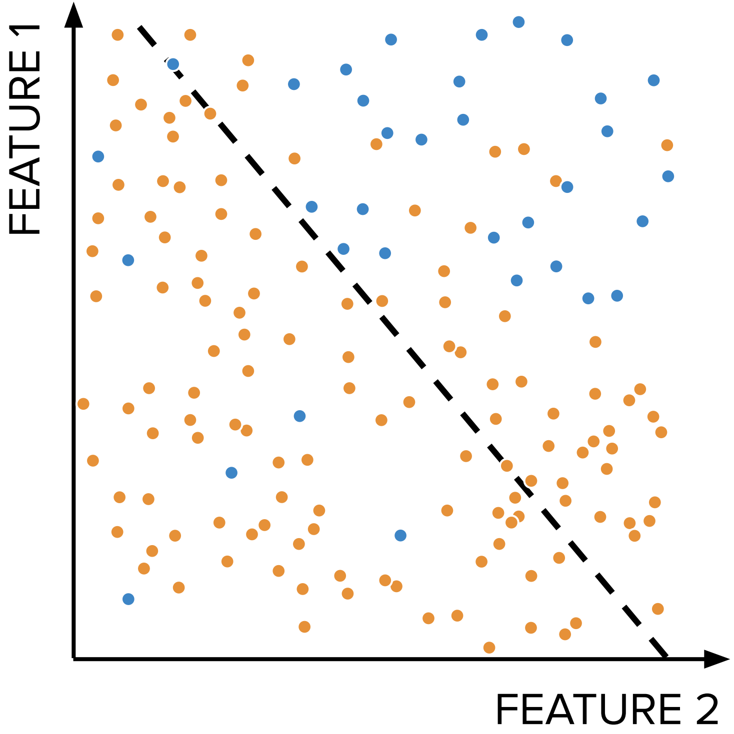


UNIVERSITY OF WATERLOO | FACULTY OF MATHEMATICS

Decision trees can approximate certain nonlinear functions



DECISION TREE



LINEAR MODEL

Predictions correspond to the majority class within a region

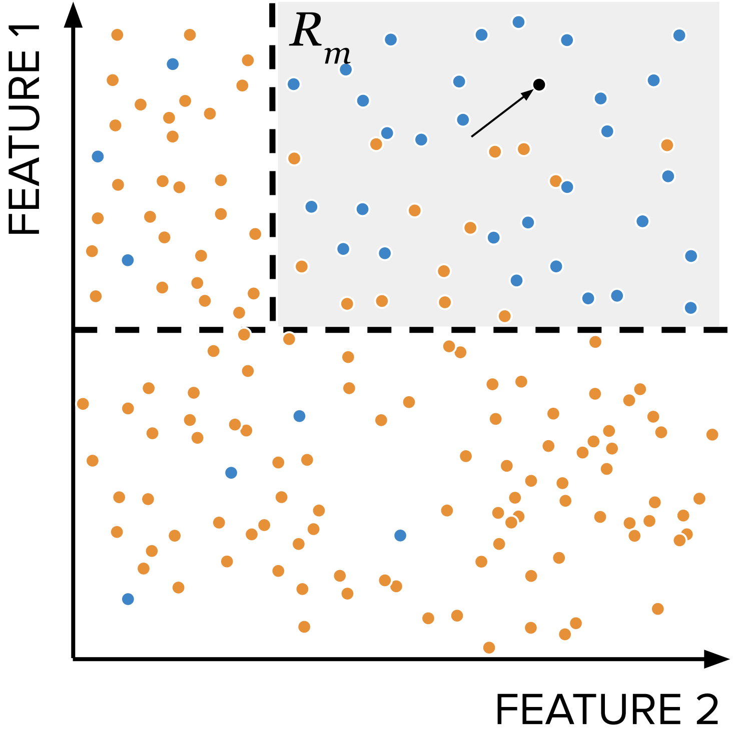
The prediction made for an observation x_i within a subregion R_m of the domain of the data is the majority class within that region:

$$\hat{y}_i = \underset{k}{\operatorname{argmax}}\, \bar{p}_{mk}$$

where $\bar{p}_{mk}$ is the empirical fraction of observations with label k within the subregion R_m:

$$\bar{p}_{mk} = \frac{1}{N_m} \sum_{x_i \in R_m} \mathbb{1}(y_i = k)$$

and N_m is the number of observations within partition R_m.



Growing a tree means defining the next node

$$(\hat{j}, \hat{t}) = \underset{j,t}{\operatorname{argmin}} |S_0| l(S_0) + |S_1| l(S_1)$$

$$\begin{aligned} l(S) &= |S_0| l(S_0) + |S_1| l(S_1) \\ &= |S_0| l\left(\{(x_i, y_i) \in S_0 : x_{ij} \leq t\}\right) \\ &+ |S_1| l\left(\{(x_i, y_i) \in S_1 : x_{ij} > t\}\right) \end{aligned}$$

Loss functions: misclassification error

$$l(S_m) = \frac{1}{N_m} \sum_{x_i \in R_m} \mathbb{1}(y_i \neq \hat{y}_i)$$
$$= 1 - \max_k \bar{p}_{mk}$$

Characterization	Example 1	Example 2
Predicted labels $\{\hat{y}_i\}$	$\{ 0, 0, 0, 0 \}$	$\{ 0, 0, 0, 0 \}$
True labels $\{y_i\}$	$\{ 0, 0, 0, 0 \}$	$\{ 0, 0, 1, 1 \}$
$\bar{p}_0$	1	0.5
l_0	0	0.5
Comments	Perfect Prediction	50% Misclassification

Other loss functions

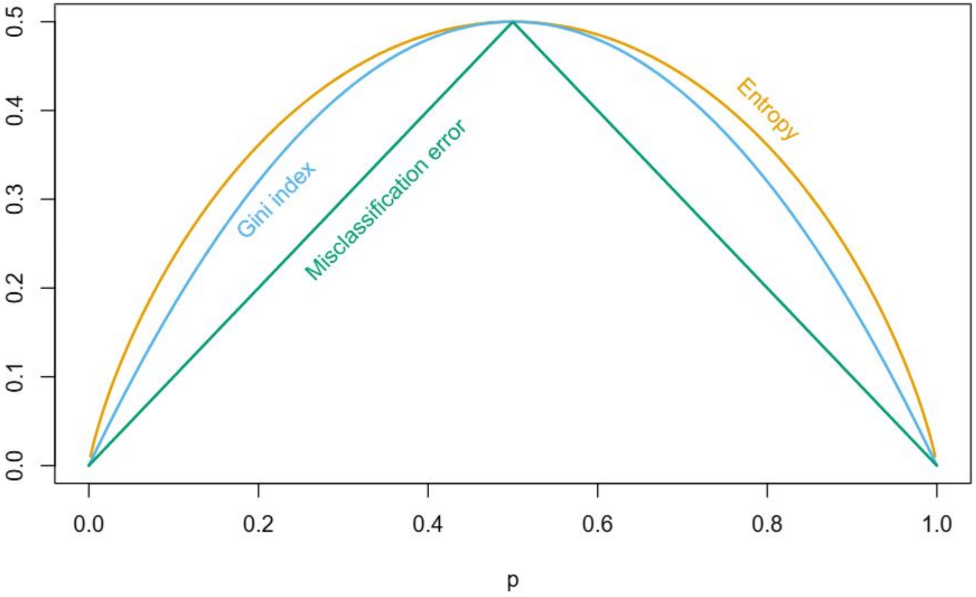
Entropy:

$$l(S_m) = - \sum_{k \in \{0,1\}} \bar{p}_{mk} \log \bar{p}_{mk}$$

Gini Index:

$$l(S_m) = \sum_{k \in \{0,1\}} \bar{p}_{mk}(1 - \bar{p}_{mk})$$

Comparing loss functions



Example: Let's learn a tree for this dataset

Traffic Light Color	Following Distance (m)	Vehicle Decision
Red	5.0	Brake
Green	5.0	Brake
Green	8.0	Brake
Green	10.0	Brake
Red	15.0	Brake
Green	20.0	Cruise
Red	30.0	Brake
Green	30.0	Cruise
Green	50.0	Cruise
Red	80.0	Cruise

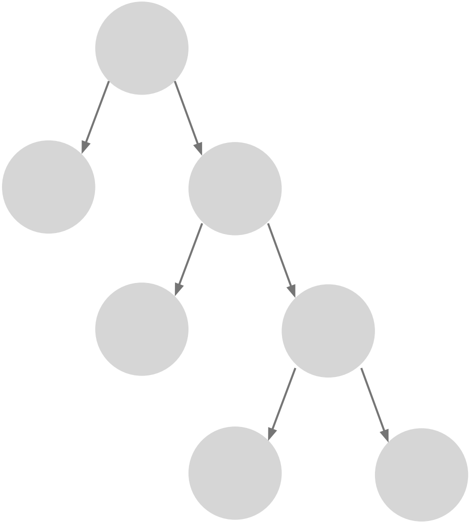
Stopping criteria

- Have achieved homogeneity in leaves
- Improvements are negligible
  - $\Delta = l(S_{OLD}) - (|S_0|\ l(S_0) + |S_1|\ l(S_1)) < \delta$
- Leaves are sparse
  - There are
- The tree has grown to a certain depth (height?)
  - Decision stump: One feature, one threshold
- The algorithm has run for some amount of time

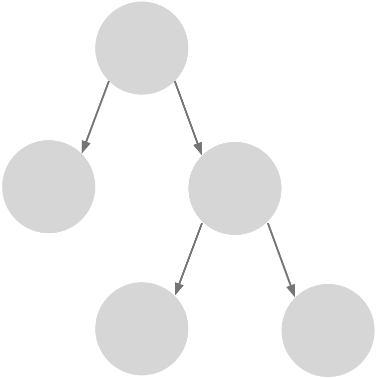
Pruning a tree

- Grow the tree fully, then regularize using hyperparameter α

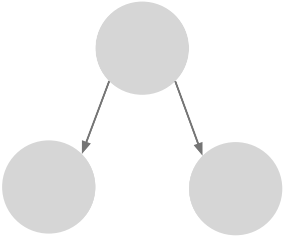
$$min \sum_v l_v(S) + \alpha N_v$$



$N_v = 4$



$N_v = 3$



$N_v = 2$

Now that we're at the end of the lecture, you should be able to…

★ Identify the components and structure of a decision tree, including **nodes, leaves, partitions**, and **thresholds**.
★ Implement a decision tree model by applying **recursive partitioning techniques**.
★ Differentiate between commonly-used loss functions and impurity measures (**entropy**, **Gini index**, and **misclassification error**).
★ Recognize when a decision tree can be used in **practical applications**.
★ Recommend strategies to **improve robustness**.