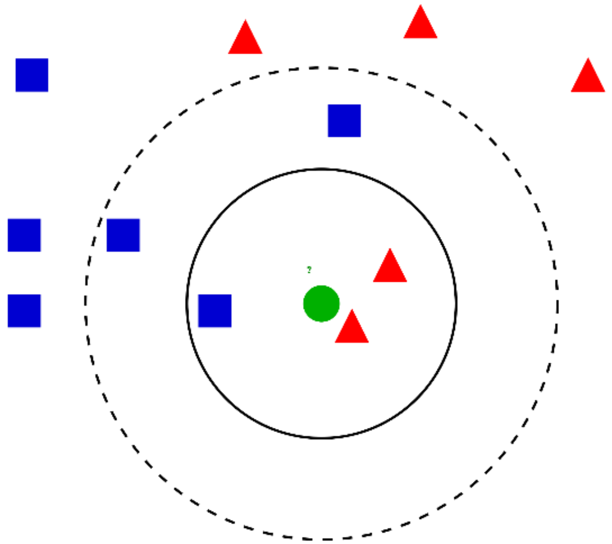
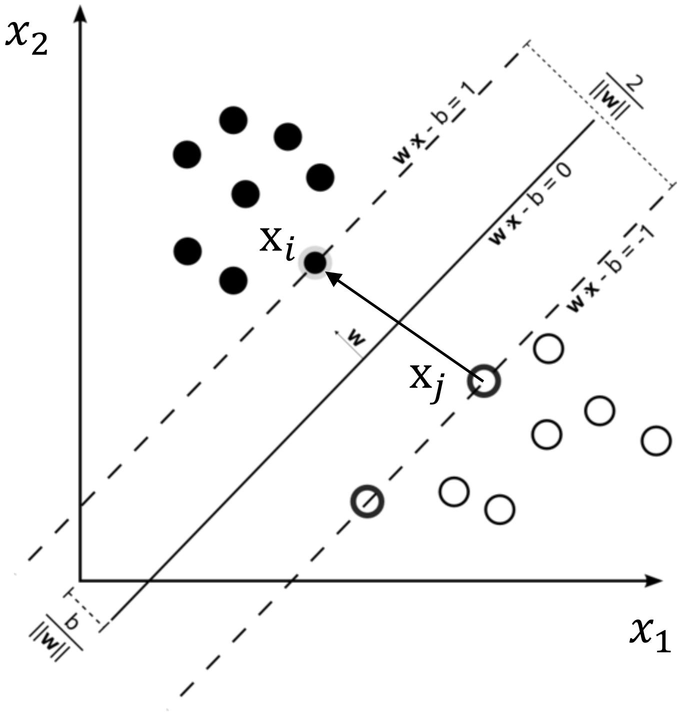
Types of machine learning algorithms

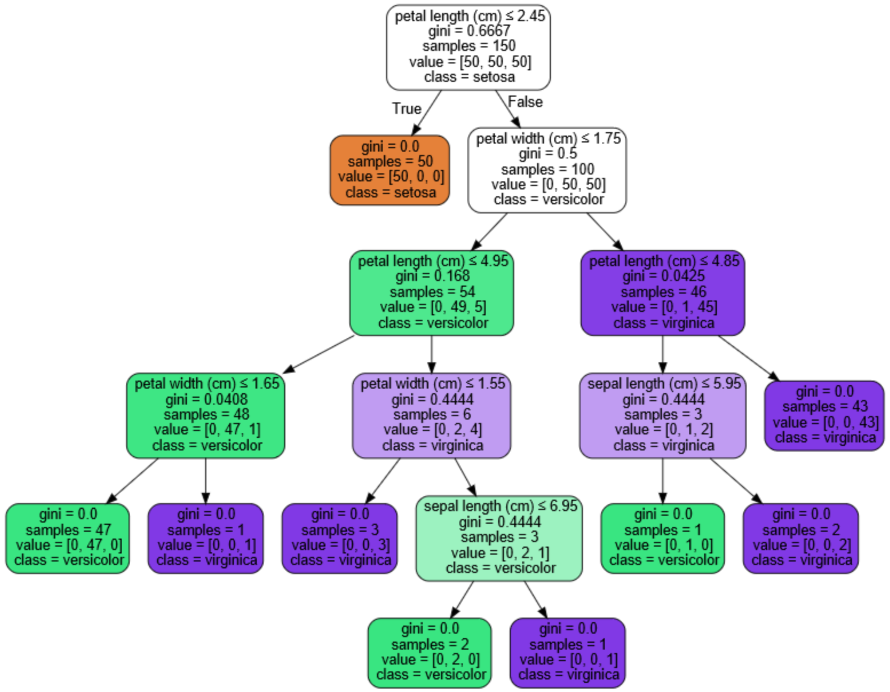
Instance-based





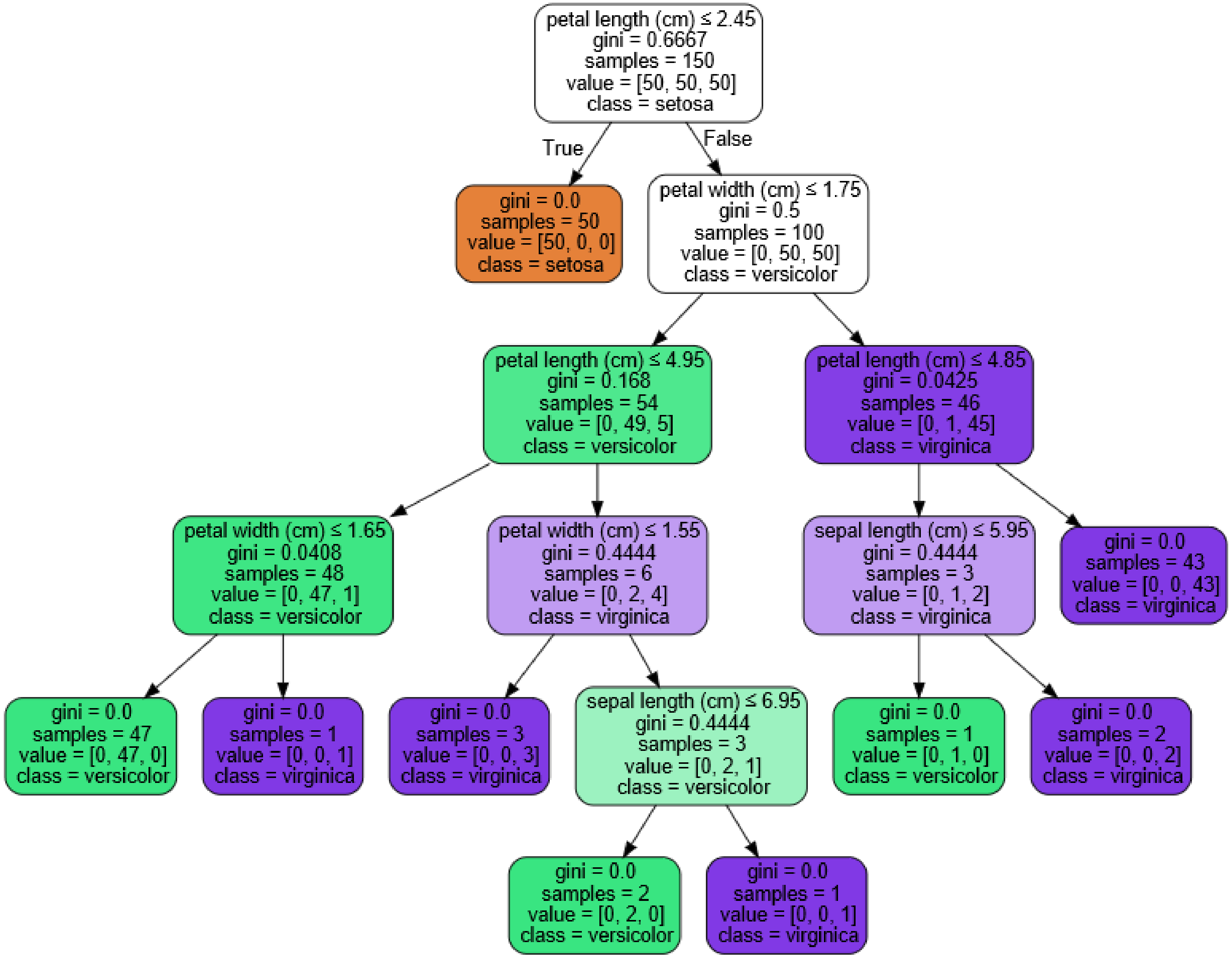
Linear

Rule-based



Decision trees

Decision trees



Advantages of decision trees

- Interpretability
- Almost no need for preprocessing
- Numerical and categorical data
- Resilience

Split evaluation

Information Gain:

$$IG = \frac{|\mathrm{X}_{node}|}{|\mathrm{X}_{\text{total}}|} I(\mathrm{X}_{node}) - \frac{|\mathrm{X}_{right}|}{|\mathrm{X}_{\text{total}}|} I(\mathrm{X}_{right}) - \frac{|\mathrm{X}_{left}|}{|\mathrm{X}_{\text{total}}|} I(\mathrm{X}_{left})$$

Split evaluation

Information Gain:

$$IG = \frac{|\mathrm{X}_{node}|}{|\mathrm{X}_{\text{total}}|} I(\mathrm{X}_{node}) - \frac{|\mathrm{X}_{right}|}{|\mathrm{X}_{\text{total}}|} I(\mathrm{X}_{right}) - \frac{|\mathrm{X}_{left}|}{|\mathrm{X}_{\text{total}}|} I(\mathrm{X}_{left})$$

Misclassification Error:

$$I_E(\mathrm{X}) = 1 - \max\{p(y)\} = 1 - \max_{\mathrm{y}} \left(\frac{|\mathrm{x}_i : y_i = y|}{|\mathrm{X}|} \right)$$

Split evaluation

Information Gain:

$$IG = \frac{|\mathrm{X}_{node}|}{|\mathrm{X}_{\mathrm{total}}|} I(\mathrm{X}_{node}) - \frac{|\mathrm{X}_{right}|}{|\mathrm{X}_{\mathrm{total}}|} I(\mathrm{X}_{right}) - \frac{|\mathrm{X}_{left}|}{|\mathrm{X}_{\mathrm{total}}|} I(\mathrm{X}_{left})$$

Entropy:

$$I_H(\mathrm{X}) = -\sum_{y \in \mathrm{Y}} p(y) \log_2\big(p(y)\big) = -\sum_{y \in \mathrm{Y}} \frac{|\mathrm{x}_i \colon y_i = y|}{|\mathrm{X}|} \times \log_2\left(\frac{|\mathrm{x}_i \colon y_i = y|}{|\mathrm{X}|}\right)$$

Split evaluation

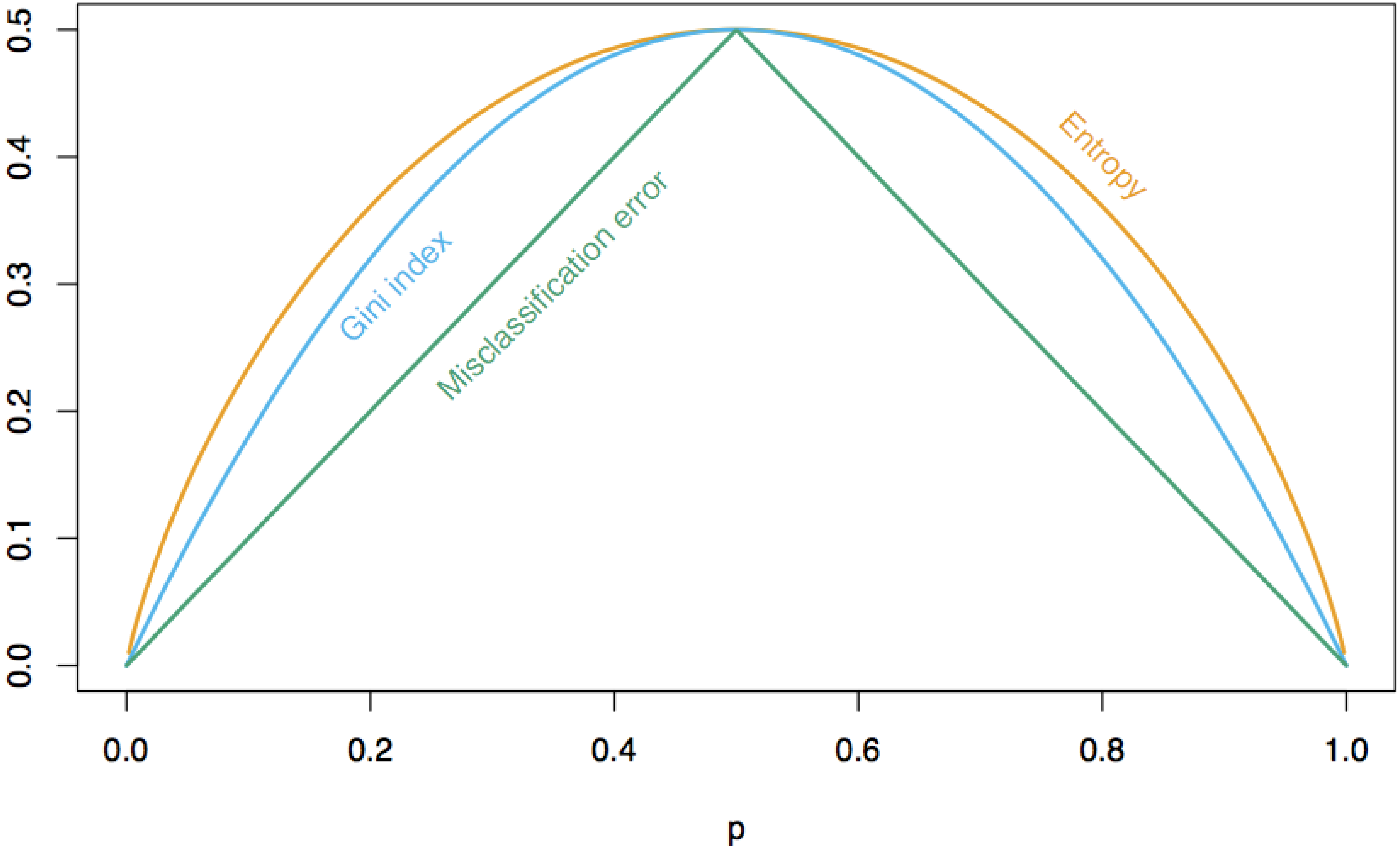
Information Gain:

$$IG = \frac{|\mathrm{X}_{node}|}{|\mathrm{X}_{\mathrm{total}}|} I(\mathrm{X}_{node}) - \frac{|\mathrm{X}_{right}|}{|\mathrm{X}_{\mathrm{total}}|} I(\mathrm{X}_{right}) - \frac{|\mathrm{X}_{left}|}{|\mathrm{X}_{\mathrm{total}}|} I(\mathrm{X}_{left})$$

Gini Impurity:

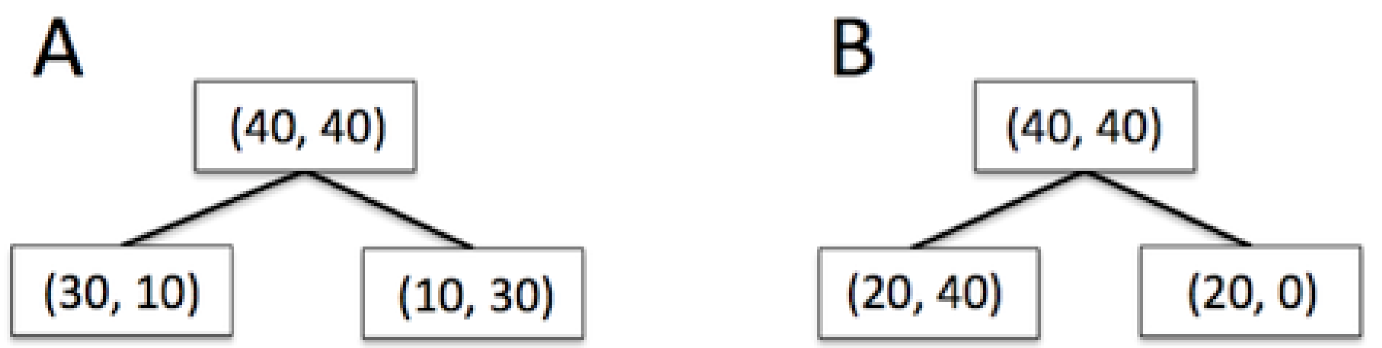
$$I_G(\mathrm{X}) = \sum_{y \in \mathrm{Y}} p(y)\big(1 - p(y)\big) = \sum_{y \in \mathrm{Y}} \frac{|\mathrm{x}_i : y_i = y|}{\mathrm{X}} \frac{|\mathrm{x}_i : y_i \neq y|}{\mathrm{X}}$$

Comparison



Entropy
Gini index
Misclassification error
0.0
0.1
0.2
0.3
0.4
0.5
0.0
0.2
0.4
0.6
0.8
1.0
p

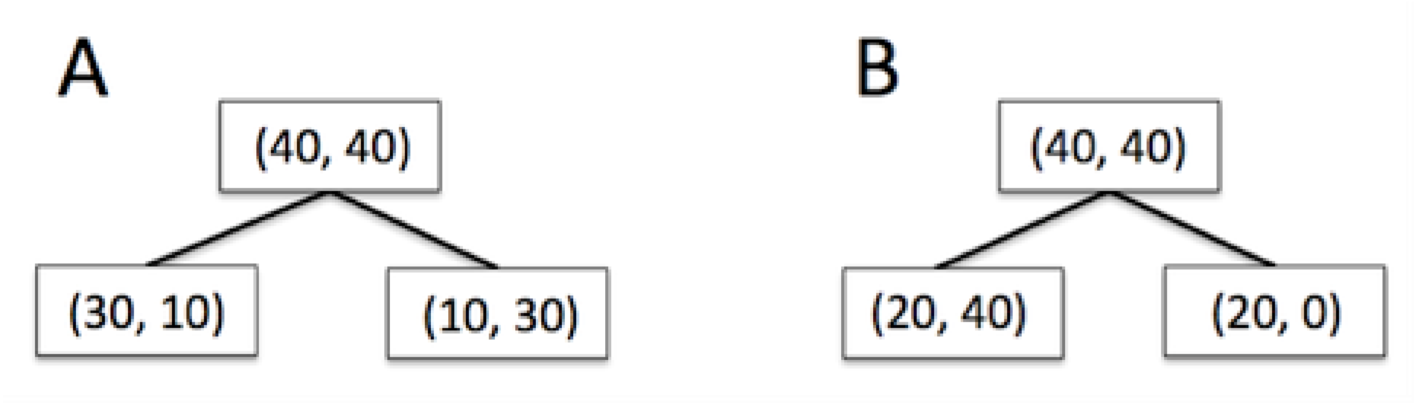
Example



$$IG_E(A) = 1 * 0.5 - 0.5 * 0.25 - 0.5 * 0.25 = 0.25$$

$$IG_E(B) = 1 * 0.5 - 0.75 * 0.33 - 0.25 * 0 = 0.25$$

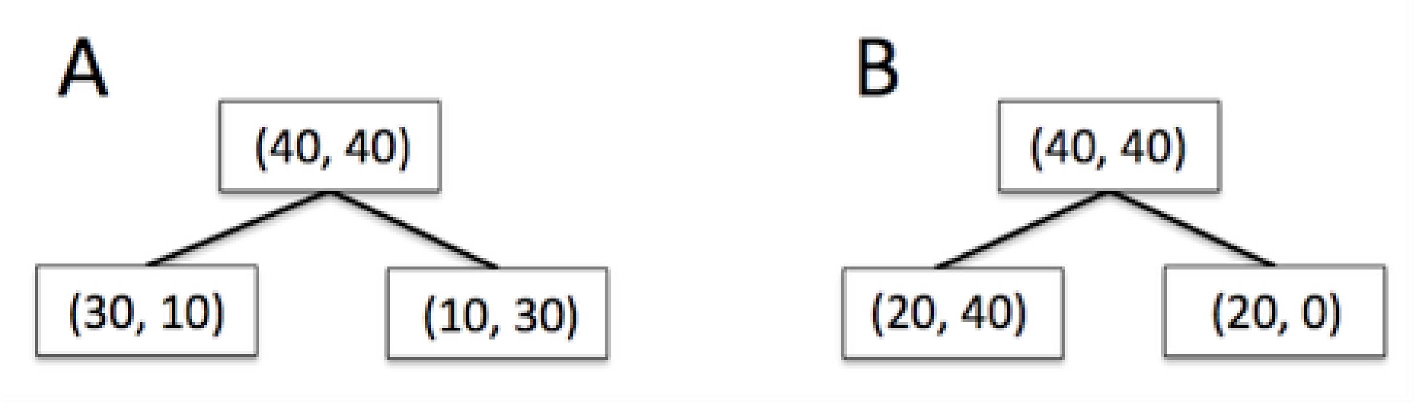
Example



$$IG_H(A) = 1 * 1 - 0.5 * 0.81 - 0.5 * 0.81 = 0.19$$

$$IG_H(B) = 1 * 1 - 0.75 * 0.92 - 0.25 * 0 = 0.31$$

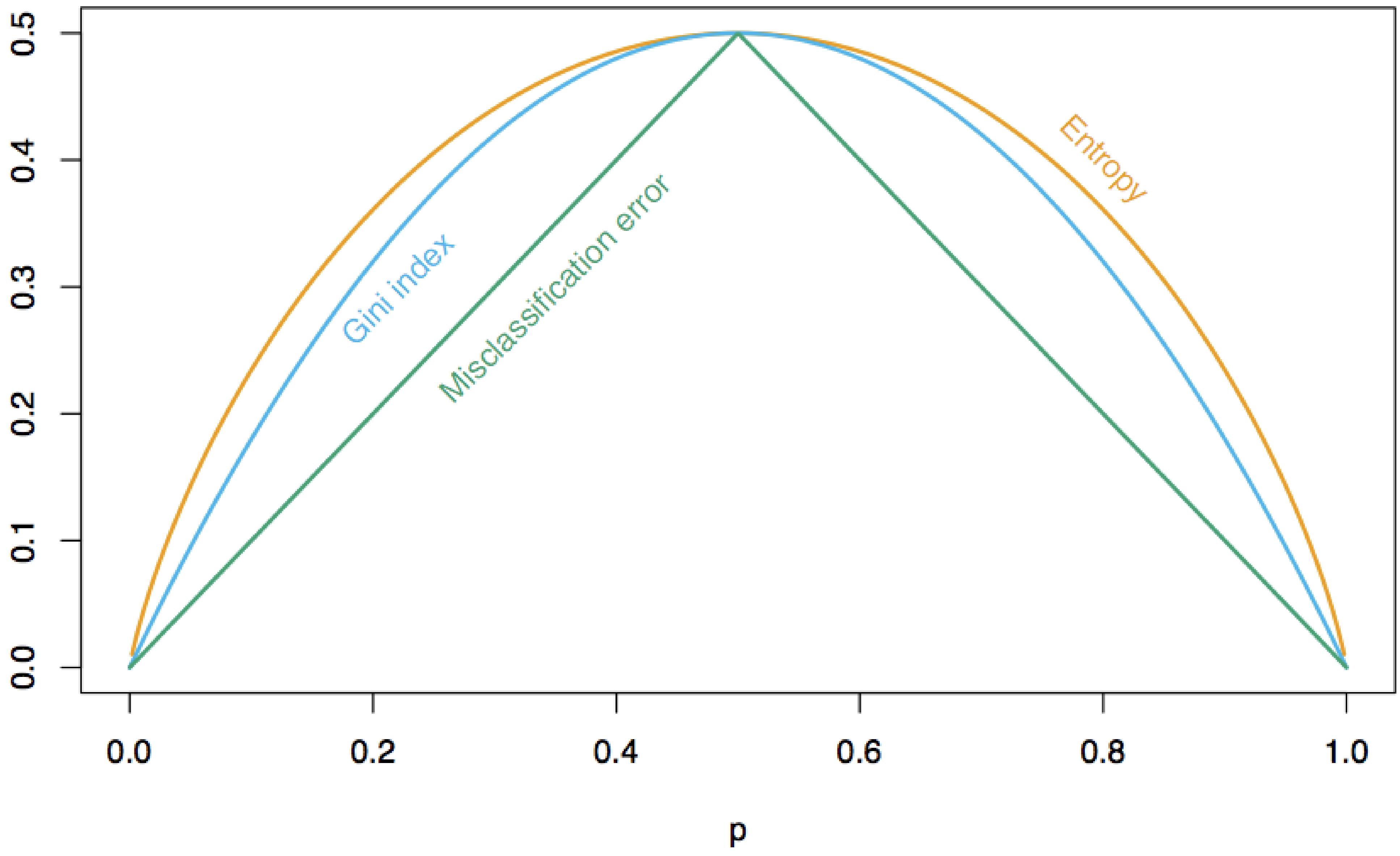
Example



$$IG_G(A) = 1 * 0.5 - 0.5 * 0.375 - 0.5 * 0.375 = 0.125$$

$$IG_G(B) = 1 * 0.5 - 0.75 * 0.4 - 0.25 * 0 = 0.16$$

Comparison



0.0
0.1
0.2
0.3
0.4
0.5
0.0
0.2
0.4
0.6
0.8
1.0
p
Gini index
Misclassification error
Entropy

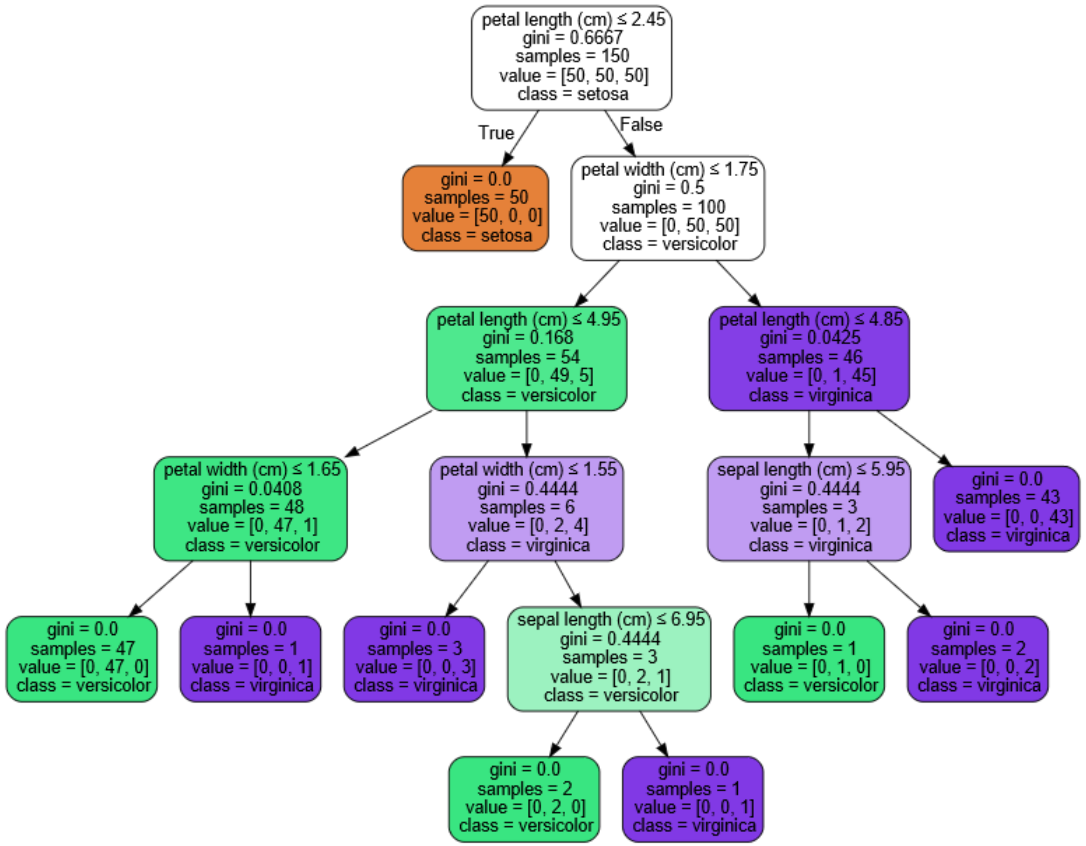
ID3, C4.5, CART

1. If all objects in the node belong to the same class -> mark the leaf as this class and stop.

2. Find the threshold with the best information gain. No threshold yields information gain -> mark the node with the majority class (or assign class probability) and stop.

3. Separate objects into children nodes by the threshold rule.

4. Call 1. for every new node.



CART (**C**lassification **a**nd **R**egression **T**rees)

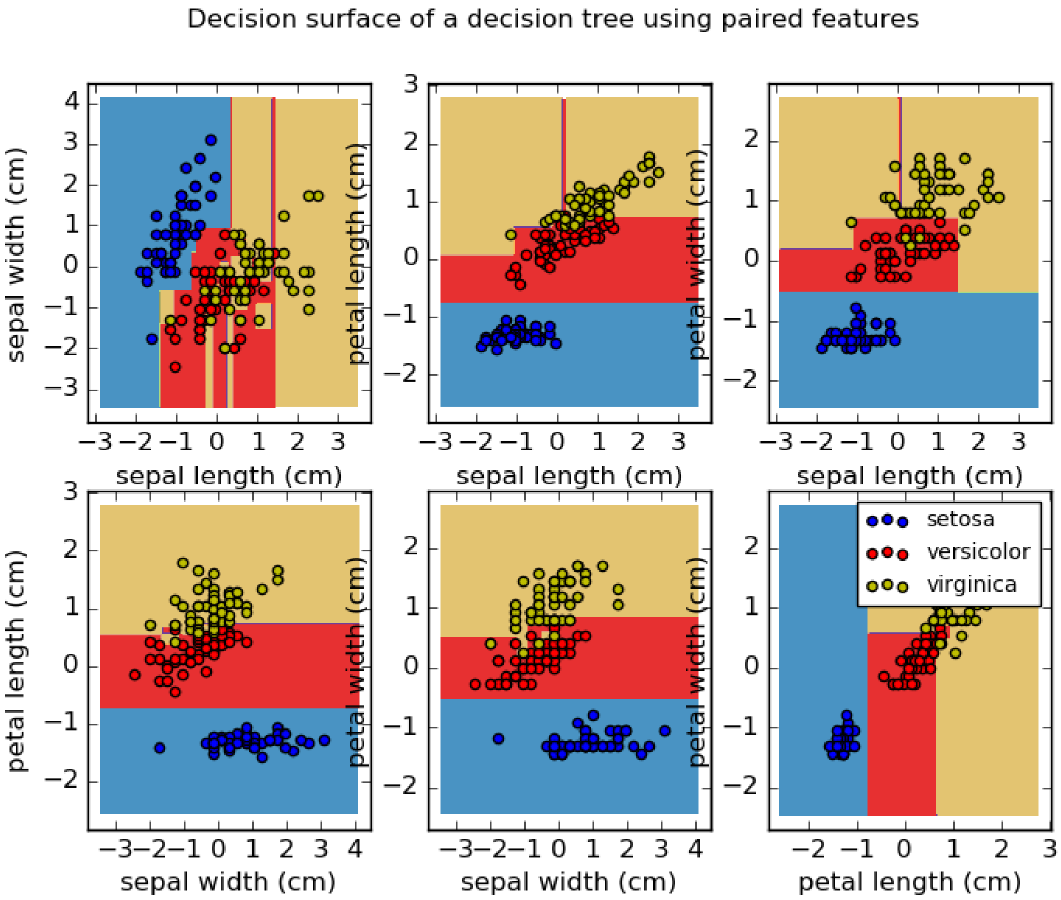
Gini Impurity for classification.

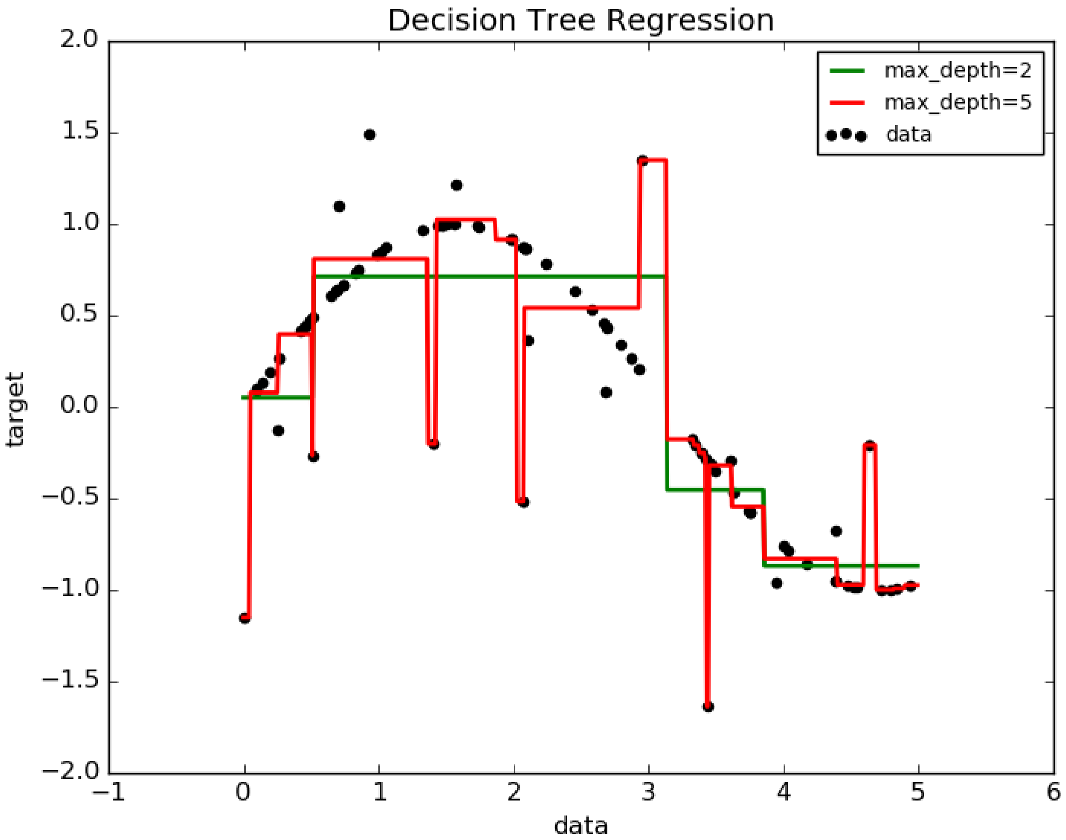
$$I_G(\mathrm{X}) = \sum_{y \in \mathrm{Y}} \frac{|\mathrm{x}_i : y_i = y|}{\mathrm{X}} \frac{|\mathrm{x}_i : y_i \neq y|}{\mathrm{X}}$$

Deviance for regression:

$$I_V(\mathrm{X}) = \sum_{\mathrm{x}_i \in \mathrm{X}} \sum_{\mathrm{x}_j \in \mathrm{X}} \frac{1}{2} \left(y_i - y_j \right)^2$$

Example


Decision surface of a decision tree using paired features
sepal width (cm)
sepal length (cm)
petal length (cm)
petal width (cm)
sepal width (cm)
petal length (cm)
setosa
versicolor
virginica


Decision Tree Regression
max_depth=2
max_depth=5
data
target
data

Regularization and Train/Validate/Test

Put aside a part of a dataset ($\approx 10 - 20\%$) for validation (**validation dataset**).

Train classifier on what is left (**training dataset**).

Optimize hyperparameters, such as tree depth, by the metric on validation dataset.

If we have many hyperparameters, there is a present danger of overfitting to validation metric. Put aside one more dataset (**test dataset**).

Pruning

- Minimum error pruning: remove nodes while validation error is not increasing.

- Cost complexity pruning: remove nodes while validation error is not increasing by more than α.

In other words, introduce a new, cumulative error function:

$$[error] + \alpha * [tree\ size]$$

Speeding up decision trees

The complexity of naïve implementation is $O\left(N_{features} N_{examples}^{3} \log\left(N_{examples}\right)\right)$.

We loop over $N_{features}$ features:

Sort the examples in $O\left(N_{examples} \log(N_{examples})\right)$

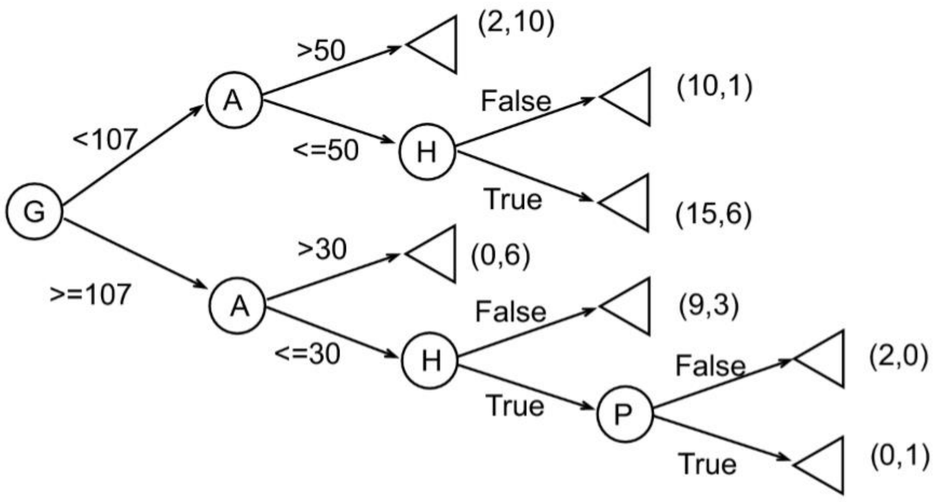
Go over $N_{examples} - 1$ thresholds, compute the gain in $O\left(N_{examples}\right)$

Speeding up decision trees

- Presorting and keeping the order of examples over all features.
- Binning samples by features.
- Incremental computation of class histograms.
- Parallelize everything!

Oblivious Decision Trees

Same feature on one level.



(G)lucose level
(A)ge
(H)ypertension
(P)regnancy

Fuzzy decision trees