

# Decision trees

Szymon Bobek

Institute of Applied Computer science  
AGH University of Science and Technology

<http://geist.agh.edu.pl>



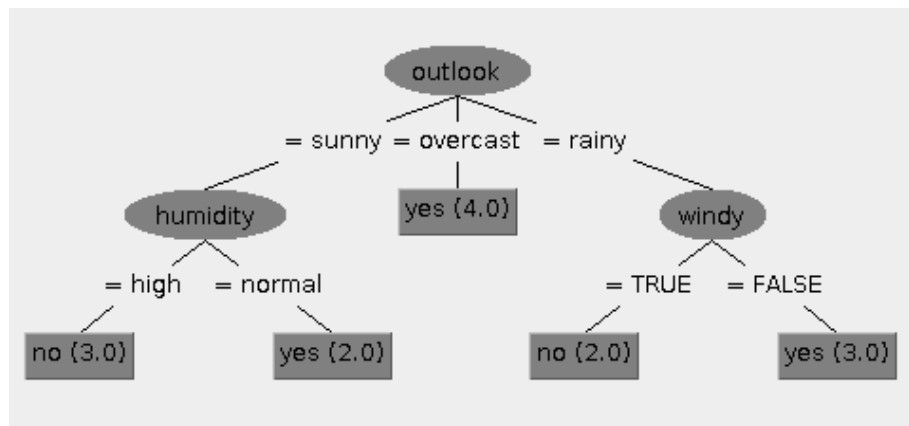
# Outline I

- 1 What is a decision tree
- 2 Tree growing
- 3 Split criterion based on impurity
  - Entropy
  - Gini index
  - Handling numerical values
- 4 Pruning
  - Pre-pruning/Early stopping
  - Post-pruning
- 5 Regression trees and clustering
  - Entropy and variance
  - KD trees
- 6 Summary

# Presentation Outline

- 1 What is a decision tree
- 2 Tree growing
- 3 Split criterion based on impurity
- 4 Pruning
- 5 Regression trees and clustering
- 6 Summary

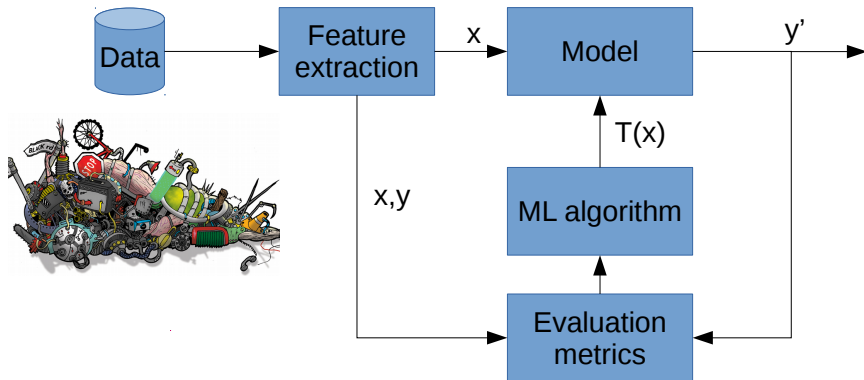
# Decision tree



# It comes from data

Outlook	AirTemp	Humidity	Windy	Water	Forecast	Enjoy
sunny	warm	normal	TRUE	warm	same	yes
sunny	warm	high	TRUE	warm	same	yes
rainy	cold	high	TRUE	warm	change	no
sunny	warm	high	TRUE	cool	change	no
overcast	warm	normal	FALSE	warm	same	yes
overcast	cold	high	FALSE	cool	same	no
...	...	...	...	...	...	...

# It comes from garbage



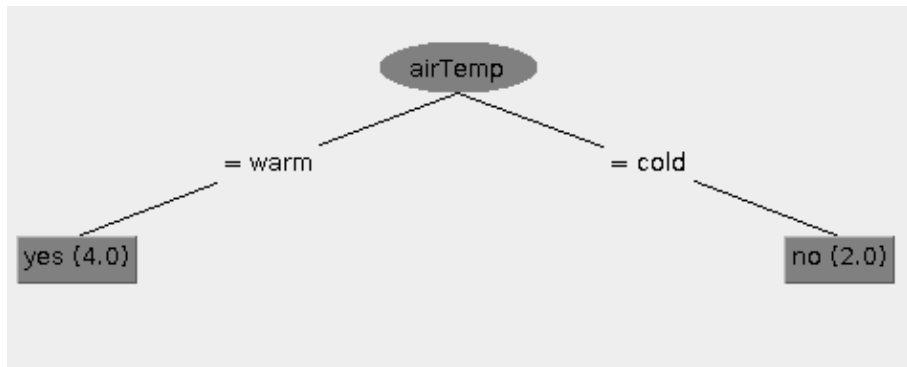
# Presentation Outline

- 1 What is a decision tree
- 2 Tree growing
- 3 Split criterion based on impurity
- 4 Pruning
- 5 Regression trees and clustering
- 6 Summary

# So, how do we do this?

Outlook	AirTemp	Humidity	Windy	Water	Forecast	Enjoy
sunny	warm	normal	TRUE	warm	same	yes
sunny	warm	high	TRUE	warm	same	yes
rainy	cold	high	TRUE	warm	change	no
sunny	warm	high	TRUE	cool	change	yes
overcast	warm	normal	FALSE	warm	same	yes
overcast	cold	high	FALSE	cool	same	no

# Outcome from learning algorithm



# How to find the best tree?

## The best tree

The best tree is the one that has best quality metric. For example it minimizes a classification error  $E(Data)$  on data:

$$E(Data) =$$

## It is NP-hard problem

Exponentially large number of possible trees makes decision tree learning NP-hard.

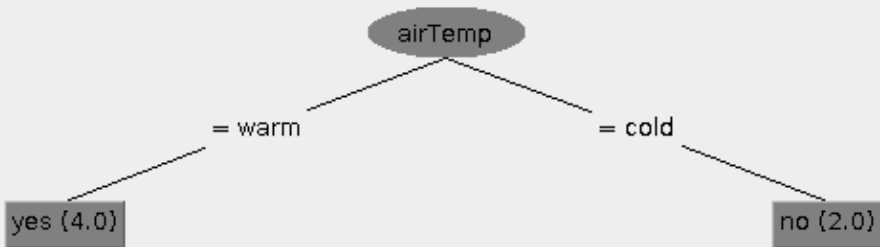
# What should be the root?

## The most important question

Not because the root of the tree is somehow special, but because it is not special at all!

## NP-hardness

We will focus on greedy approaches for building a tree.



# Greedy algorithm for growing trees

---

**Algorithm**  $\text{GrowTree}(D, F)$  – grow a feature tree from training data.

---

**Input** : data  $D$ ; set of features  $F$ .

**Output:** feature tree  $T$  with labelled leaves.

```
1 if  $\text{Homogeneous}(D)$  then return  $\text{Label}(D)$ ;  
2  $S \leftarrow \text{BestSplit}(D, F)$  split  $D$  into subsets  $D_i$  according to the literals in  $S$ ;  
3 for each  $i$  do  
4   | if  $D_i \neq \emptyset$  then  $T_i \leftarrow \text{GrowTree}(D_i, F)$  ;  
5   | else  $T_i$  is a leaf labelled with  $\text{Label}(D)$ ;  
6 end  
7 return a tree whose root is labelled with  $S$  and whose children are  $T_i$ 
```

---

# Impurity-based best split

---

**Algorithm**  $\text{BestSplit-Class}(D, F)$  – find the best split for a decision tree.

---

**Input** : data  $D$ ; set of features  $F$ .

**Output**: feature  $f$  to split on.

```
1  $l_{\min} \leftarrow 1$ ;  
2 for each  $f \in F$  do  
3   | split  $D$  into subsets  $D_1, \dots, D_l$  according to the values  $v_j$  of  $f$ ;  
4   | if  $\text{Imp}(\{D_1, \dots, D_l\}) < l_{\min}$  then  
5   |   |  $l_{\min} \leftarrow \text{Imp}(\{D_1, \dots, D_l\})$ ;  
6   |   |  $f_{\text{best}} \leftarrow f$ ;  
7   | end  
8 end  
9 return  $f_{\text{best}}$ 
```

---

$$\text{Imp}(\{D_1, \dots, D_l\}) = \sum_{j=1}^l \frac{|D_j|}{|D|} \text{Imp}(D_j)$$

# Presentation Outline

- 1 What is a decision tree
- 2 Tree growing
- 3 Split criterion based on impurity
  - Entropy
  - Gini index
  - Handling numerical values
- 4 Pruning
- 5 Regression trees and clustering
- 6 Summary

- 1 What is a decision tree
- 2 Tree growing
- 3 Split criterion based on impurity
  - Entropy
  - Gini index
  - Handling numerical values
- 4 Pruning
  - Pre-pruning/Early stopping
  - Post-pruning
- 5 Regression trees and clustering
  - Entropy and variance
  - KD trees
- 6 Summary

# Entropy interpretation

$$H(D) = - \sum_{c \in C} p(c) \log_2 p(c)$$

Where:

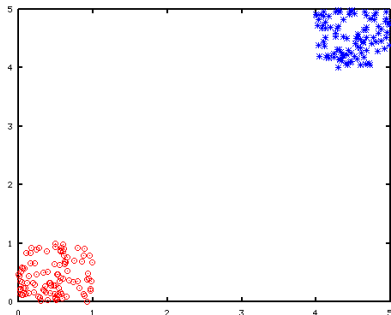
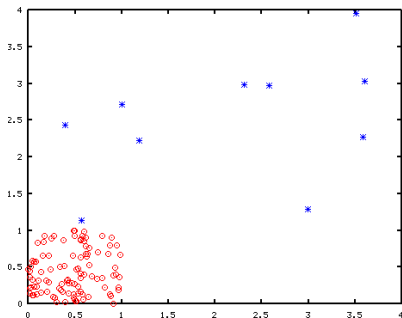
- $D$  - Current dataset for which the entropy is calculated (for every node it will be different set)
- $C$  - Set of class labels in dataset  $D$
- $p(c)$  - Probability of observing item with class label  $c$  in  $D$

## Possible values of entropy

Entropy values are not between 0 and 0.5. The lower bound is 0, but the upper bound depends on data and equals:

$$\log_2(|D|)$$

# Entropy measures the diversity of data



## Entropy wrt. class attribute

In DT generation problem we calculate entropy of the set with respect to the class label, not features.

$$Gain(D) = \text{Imp}(D) - \sum_{v \in \text{Values}(F)} \frac{|D_v|}{|D|} \text{Imp}(D_v)$$

## Entropy-based

$$H(D) = - \sum_{c \in C} p(c) \log_2 p(c)$$

$$Gain(D) = H(D) - \sum_{v \in \text{Values}(F)} \frac{|D_v|}{|D|} H(D_v)$$

# Example

Outlook	AirTemp	Humidity	Windy	Water	Forecast	Enjoy
sunny	warm	normal	TRUE	warm	same	yes
sunny	warm	high	TRUE	warm	same	yes
rainy	cold	high	TRUE	warm	change	no
sunny	warm	high	TRUE	cool	change	yes
overcast	warm	normal	FALSE	warm	same	yes
overcast	cold	high	FALSE	cool	same	no

$$H(D) =$$

$$H(D_{AirTemp}) =$$

$$H(D_{Windy}) =$$

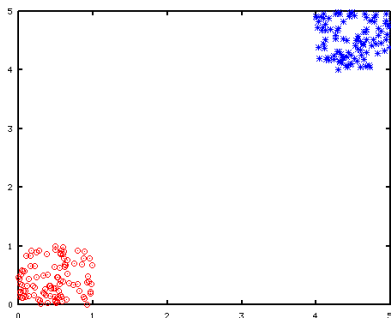
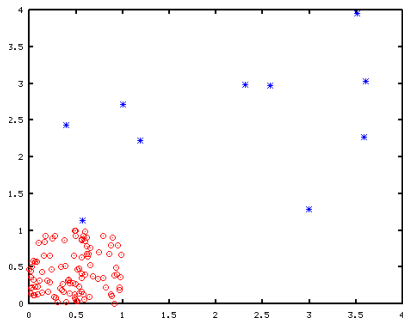
- 1 What is a decision tree
- 2 Tree growing
- 3 Split criterion based on impurity
  - Entropy
  - Gini index
  - Handling numerical values
- 4 Pruning
  - Pre-pruning/Early stopping
  - Post-pruning
- 5 Regression trees and clustering
  - Entropy and variance
  - KD trees
- 6 Summary

## Gini index

It says what is the probability of misclassification of data in a dataset if all elements were classified incorrectly according to their distribution in dataset:

$$G(D) = \sum_{c \in C} p(c)(1 - p(c)) =$$

# Gini index measures the diversity of data



## Gini wrt. class attribute

In DT generation problem we calculate impurity of the set with respect to the class label, not features.

# Example

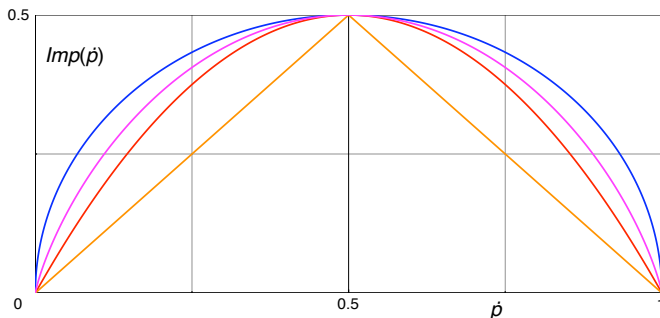
Outlook	AirTemp	Humidity	Windy	Water	Forecast	Enjoy
sunny	warm	normal	TRUE	warm	same	yes
sunny	warm	high	TRUE	warm	same	yes
rainy	cold	high	TRUE	warm	change	no
sunny	warm	high	TRUE	cool	change	yes
overcast	warm	normal	FALSE	warm	same	yes
overcast	cold	high	FALSE	cool	same	no

$$G(D) =$$

$$G(D_{AirTemp}) =$$

$$G(D_{Windy}) =$$

# Other impurity measures



## Comparison for two-valued class

From the bottom: the relative size of the minority class,  $\min(\dot{p}, 1 - \dot{p})$ ; the Gini index,  $2\dot{p}(1 - \dot{p})$ ; entropy,  $-\dot{p} \log_2 \dot{p} - (1 - \dot{p}) \log_2 (1 - \dot{p})$  (divided by 2 so that it reaches its maximum in the same point as the others); and the (rescaled) square root of the Gini index,  $\sqrt{\dot{p}(1 - \dot{p})}$  – notice that this last function describes a semi-circle.

Suppose you have 10 positives and 10 negatives, and you need to choose between the two splits  $[8+, 2-][2+, 8-]$  and  $[10+, 6-][0+, 4-]$ .

- You duly calculate the weighted average entropy (or **information gain** of both splits and conclude that the first split is the better one.
- Just to be sure, you also calculate the average **Gini index**, and again the first split wins.
- You then remember somebody telling you that the square root of the Gini index was a better impurity measure, so you decide to check that one out as well. Lo and behold, it favours the second split...! What to do?

You then remember that mistakes on the positives are about ten times as costly as mistakes on the negatives.

- You're not quite sure how to work out the maths, and so you decide to simply have ten copies of every positive: the splits are now  $[80+, 2-][20+, 8-]$  and  $[100+, 6-][0+, 4-]$ .
- You recalculate the three splitting criteria and now all three favour the second split.

# Data distribution and impurity measures

You then remember that mistakes on the positives are about ten times as costly as mistakes on the negatives.

- You're not quite sure how to work out the maths, and so you decide to simply have ten copies of every positive: the splits are now  $[80+, 2-][20+, 8-]$  and  $[100+, 6-][0+, 4-]$ .
- You recalculate the three splitting criteria and now all three favour the second split.

## Unbalanced datasets

Entropy and Gini index are sensitive to fluctuations in the class distribution,  $\sqrt{Gini}$  isn't.

- 1 What is a decision tree
- 2 Tree growing
- 3 Split criterion based on impurity
  - Entropy
  - Gini index
  - Handling numerical values
- 4 Pruning
  - Pre-pruning/Early stopping
  - Post-pruning
- 5 Regression trees and clustering
  - Entropy and variance
  - KD trees
- 6 Summary

# Splits on numerical values

Outlook	AirTemp	Humidity	Windy	Water	Forecast	Enjoy
sunny	30	normal	TRUE	warm	same	yes
sunny	35	high	TRUE	warm	same	yes
rainy	15	high	TRUE	warm	change	no
sunny	25	high	TRUE	cool	change	yes
overcast	28	normal	FALSE	warm	same	yes
overcast	17	high	FALSE	cool	same	no



# Splits on numerical values

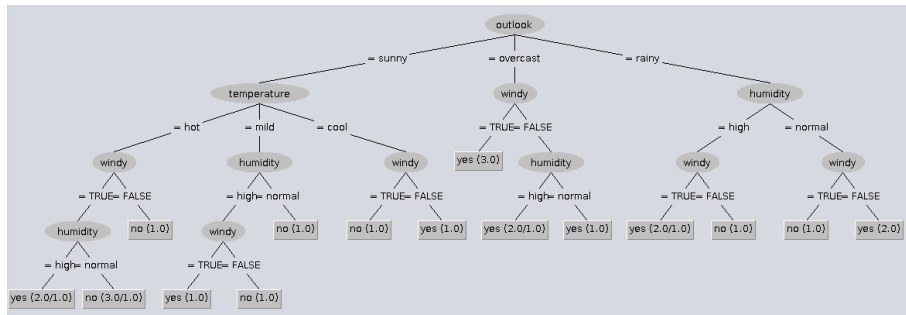
Outlook	AirTemp	Humidity	Windy	Water	Forecast	Enjoy
sunny	30	normal	TRUE	warm	same	yes
sunny	35	high	TRUE	warm	same	yes
rainy	15	high	TRUE	warm	change	no
sunny	25	high	TRUE	cool	change	yes
overcast	28	normal	FALSE	warm	same	yes
overcast	17	high	FALSE	cool	same	no



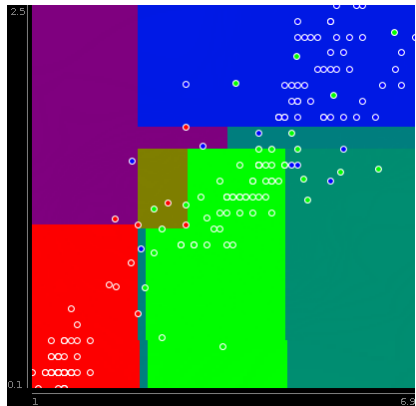
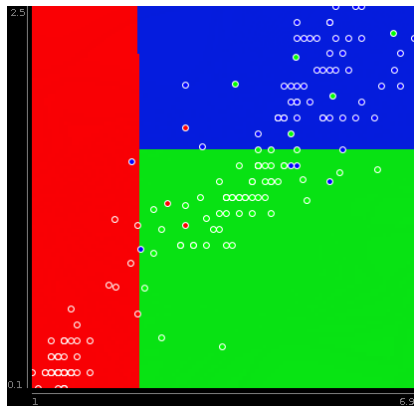
# Presentation Outline

- 1 What is a decision tree
- 2 Tree growing
- 3 Split criterion based on impurity
- 4 **Pruning**
  - Pre-pruning/Early stopping
  - Post-pruning
- 5 Regression trees and clustering
- 6 Summary

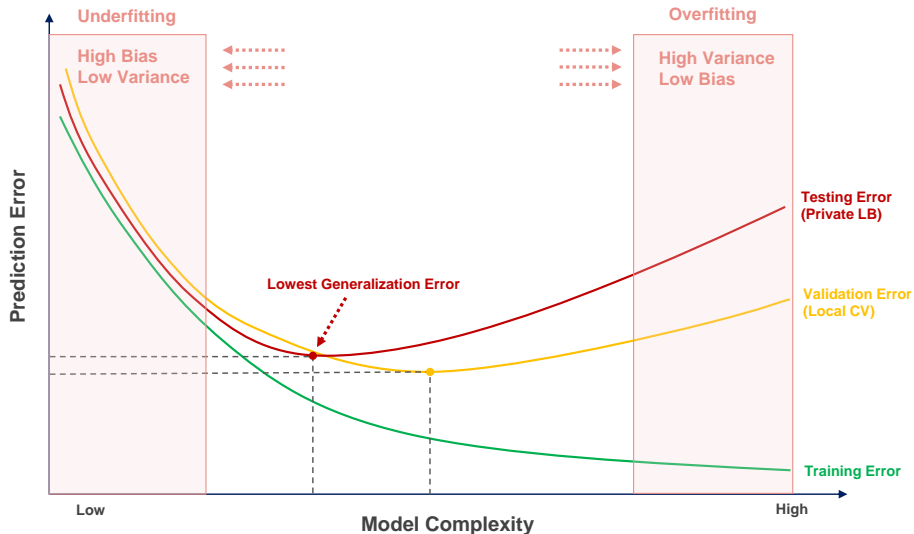
# Overfitting in decision trees



# Overfitting in decision trees



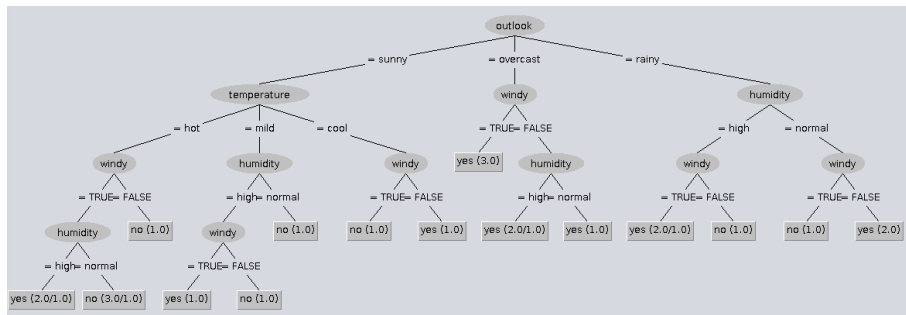
# Limiting tree-depth



# Limiting tree-depth

## General idea

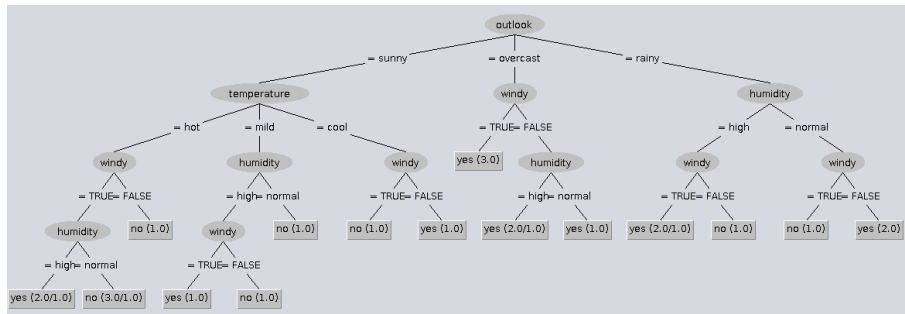
Stop splits after certain depth is reached. Decide what should be the maximum depth using validation set!



# Minimum node size

## General idea

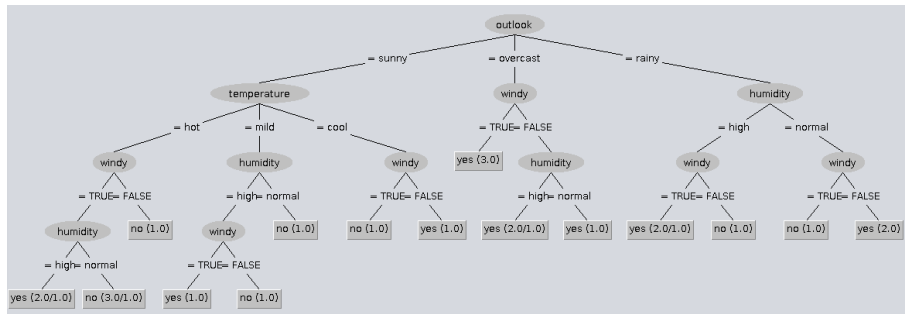
Do not split an intermediate node which contains too few data points. What does it mean *too few*?



# Classification error

## General idea

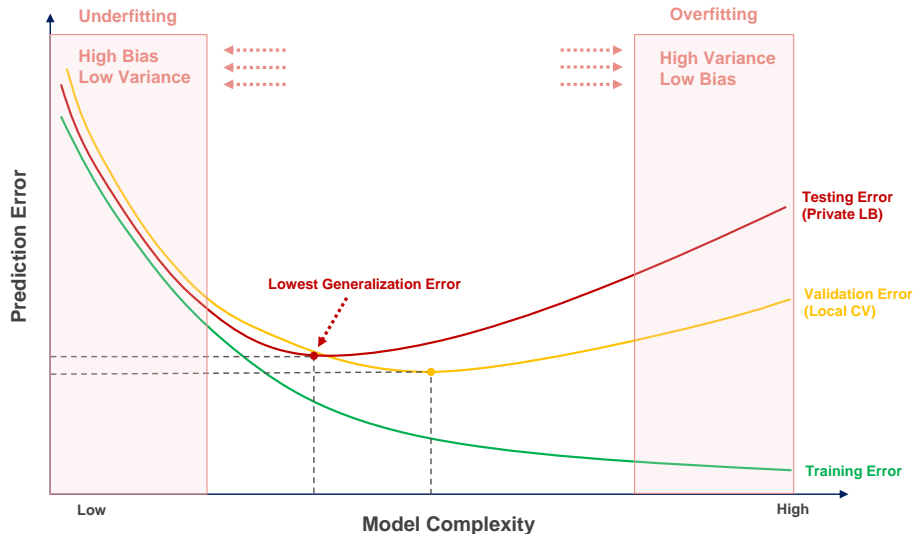
Do not consider any split that does not cause a sufficient decrease in classification error



# Outline

- 1 What is a decision tree
- 2 Tree growing
- 3 Split criterion based on impurity
  - Entropy
  - Gini index
  - Handling numerical values
- 4 Pruning
  - Pre-pruning/Early stopping
  - Post-pruning
- 5 Regression trees and clustering
  - Entropy and variance
  - KD trees
- 6 Summary

# Do not stop too early



# XOR

# Reduced-error pruning

---

**Algorithm**  $\text{PruneTree}(T, D)$  – reduced-error pruning of a decision tree.

---

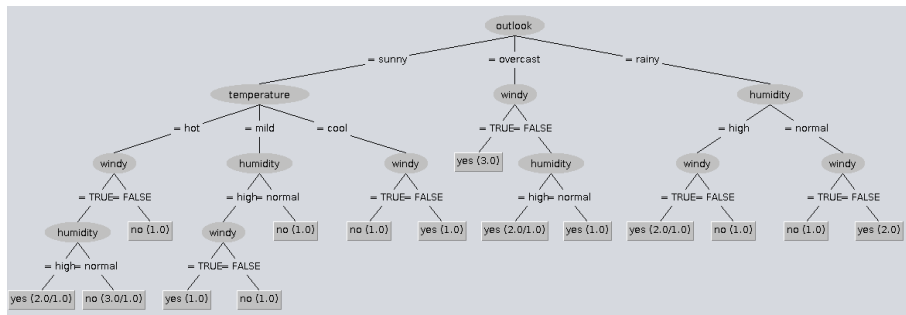
**Input** : decision tree  $T$ ; labelled data  $D$ .

**Output**: pruned tree  $T'$ .

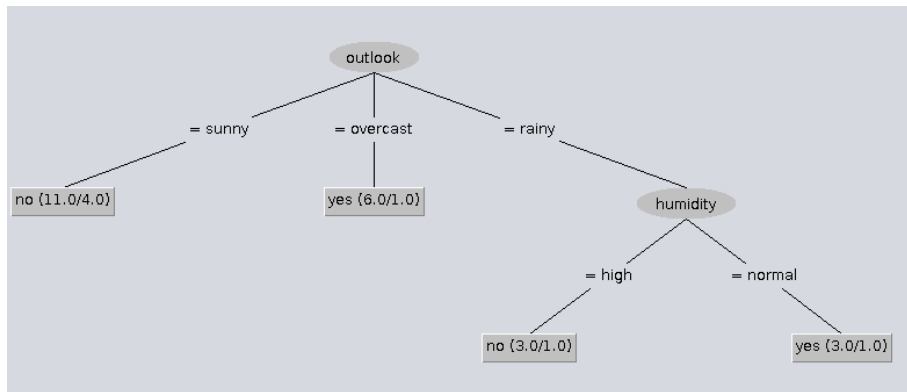
```
1 for every internal node  $N$  of  $T$ , starting from the bottom do
2    $T_N \leftarrow$  subtree of  $T$  rooted at  $N$ ;
3    $D_N \leftarrow \{x \in D \mid x \text{ is covered by } N\}$ ;
4   if accuracy of  $T_N$  over  $D_N$  is worse than majority class in  $D_N$  then
5     | replace  $T_N$  in  $T$  by a leaf labelled with the majority class in  $D_N$ ;
6   end
7 end
8 return pruned version of  $T$ 
```

---

# Illustrative example



# Illustrative example



# Presentation Outline

- 1 What is a decision tree
- 2 Tree growing
- 3 Split criterion based on impurity
- 4 Pruning
- 5 Regression trees and clustering**
  - Entropy and variance
  - KD trees
- 6 Summary

# Entropy as a variance of data

## Lowering impurity is decreasing variance

Entropy can be considered variance of nominal data. Thus, in case of numerical data, we can use variance as it is and proceed in order to reduce it.

In regression problems we can define the variance in the usual way:

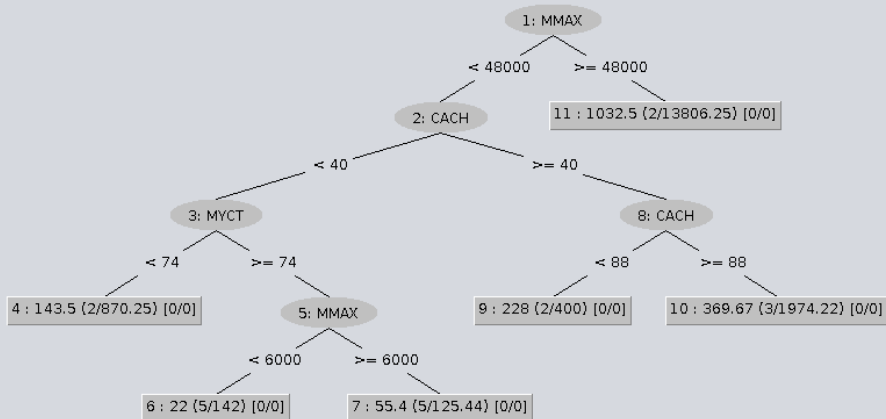
$$\text{Var}(Y) = \frac{1}{|Y|} \sum_{y \in Y} (y - \bar{y})^2$$

If a split partitions the set of target values  $Y$  into mutually exclusive sets  $\{Y_1, \dots, Y_I\}$ , the weighted average variance is then

$$\text{Var}(\{Y_1, \dots, Y_I\}) = \sum_{j=1}^I \frac{|Y_j|}{|Y|} \text{Var}(Y_j) = \dots = \frac{1}{|Y|} \sum_{y \in Y} y^2 - \sum_{j=1}^I \frac{|Y_j|}{|Y|} \bar{y}_j^2$$

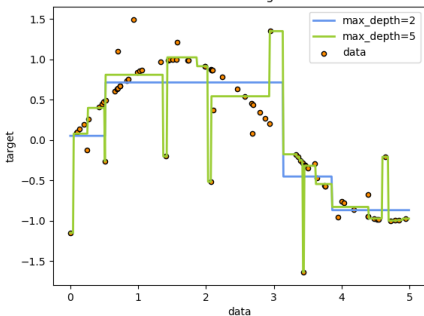
The first term is constant for a given set  $Y$  and so we want to maximise the weighted average of squared means in the children.

# Regression tree example

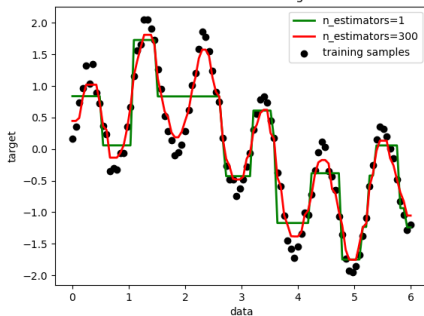


# Regression tree example

Decision Tree Regression



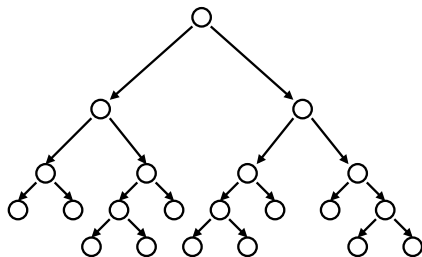
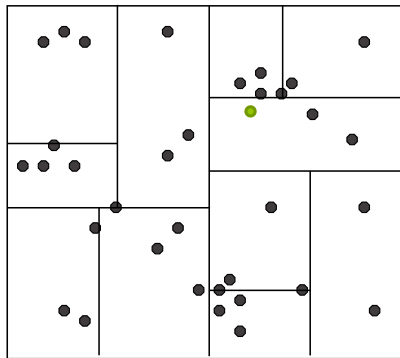
Boosted Decision Tree Regression



# Outline

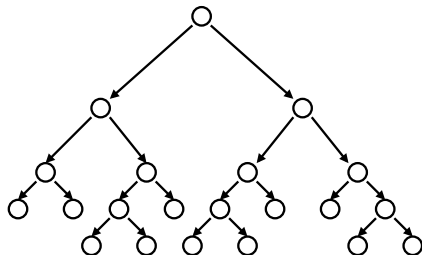
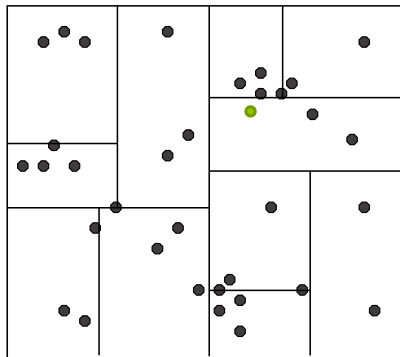
- 1 What is a decision tree
- 2 Tree growing
- 3 Split criterion based on impurity
  - Entropy
  - Gini index
  - Handling numerical values
- 4 Pruning
  - Pre-pruning/Early stopping
  - Post-pruning
- 5 Regression trees and clustering
  - Entropy and variance
  - KD trees
- 6 Summary

# KNN clustering with decision trees



- Go down the tree to the leaf where the query point is classified
- In the leaf, search for the nearest neighbour
- Backtrack, and search for new k-NN, but **do not** check points that are in bounding boxes further away than our k-NN so far.
- Prune whole branches that are further than k-NN

# KNN clustering with decision trees



## Pros and cons

- It saves search time for large  $N$
- It is usually pointless for large  $D$

# Presentation Outline

- 1 What is a decision tree
- 2 Tree growing
- 3 Split criterion based on impurity
- 4 Pruning
- 5 Regression trees and clustering
- 6 Summary**

## What we've learned?

- Linear regression (prediction/regression)
- Bias-variance tradeoff
- Logistic regression (classification)
- Support Vector Machines (classification)
- Naive Bayes and Bayesian Networks (classification)
- Decision trees (all of the above :) )

## What could be next?

- Learning from data streams (online learning)?
- ANN and Deep learning?
- Reinforcement learning?
- Something else?

# Thank you!

**Szymon Bobek**

Institute of Applied Computer Science

AGH University of Science and Technology

21 March 2017

<http://geist.agh.edu.pl>

