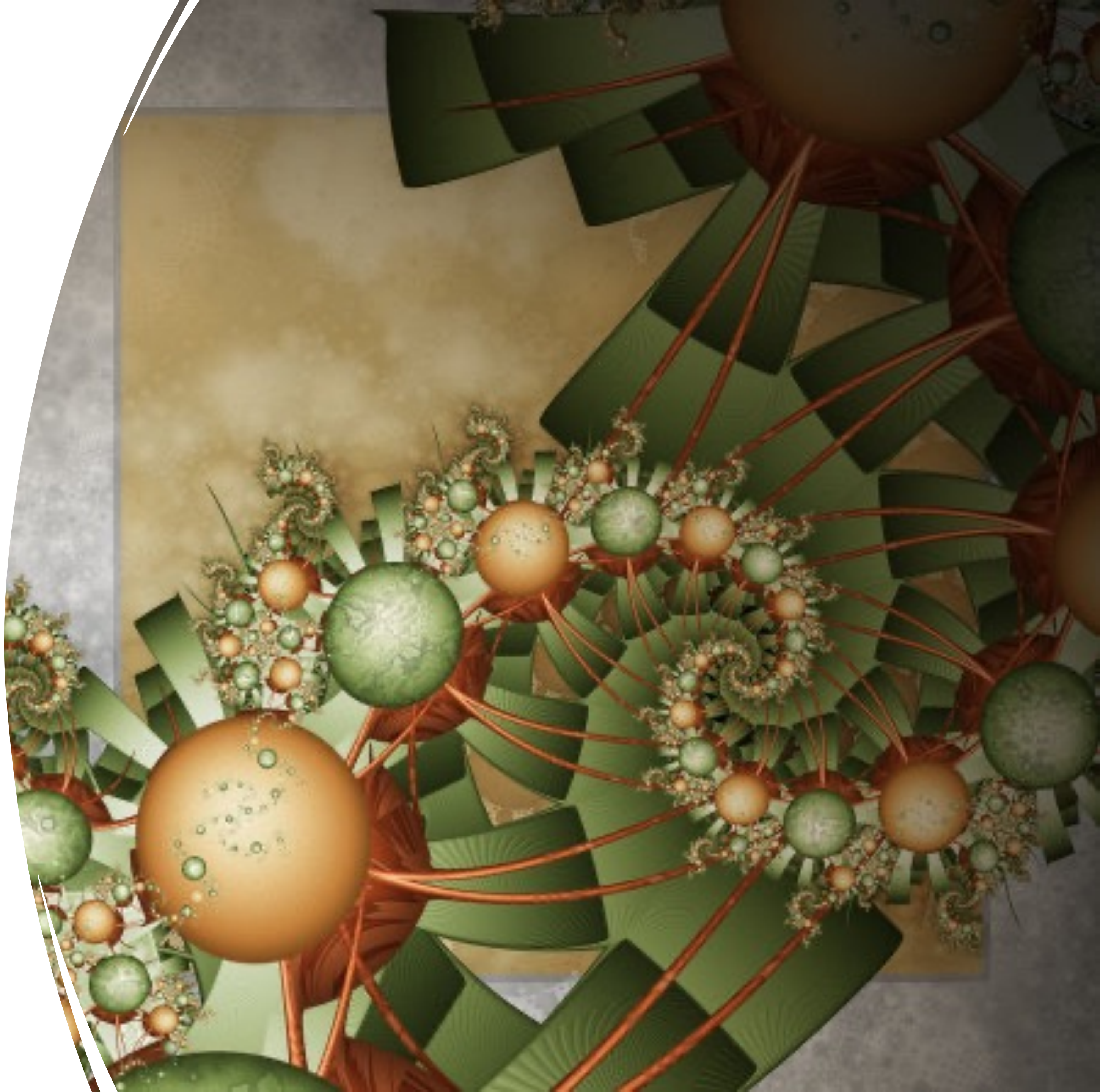
# Introduction to decision trees and random forests

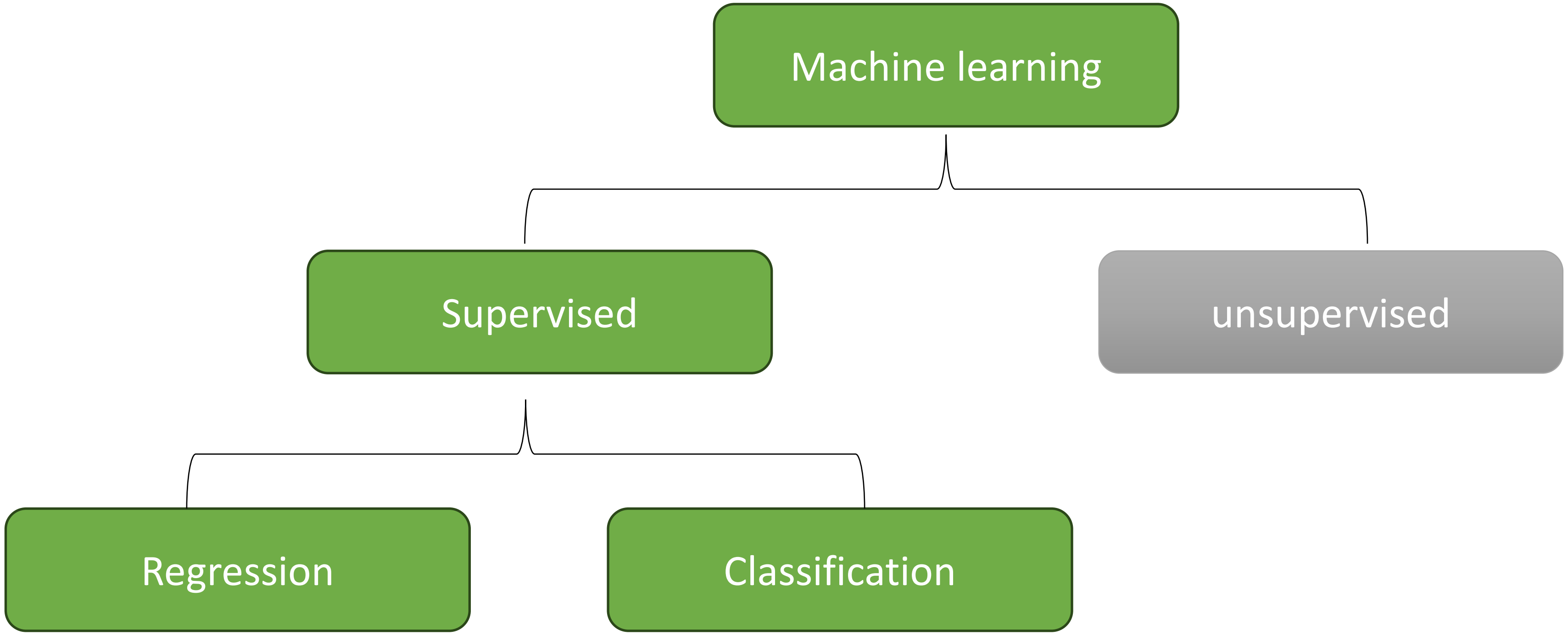
MSB 2023

# Overview & class objectives

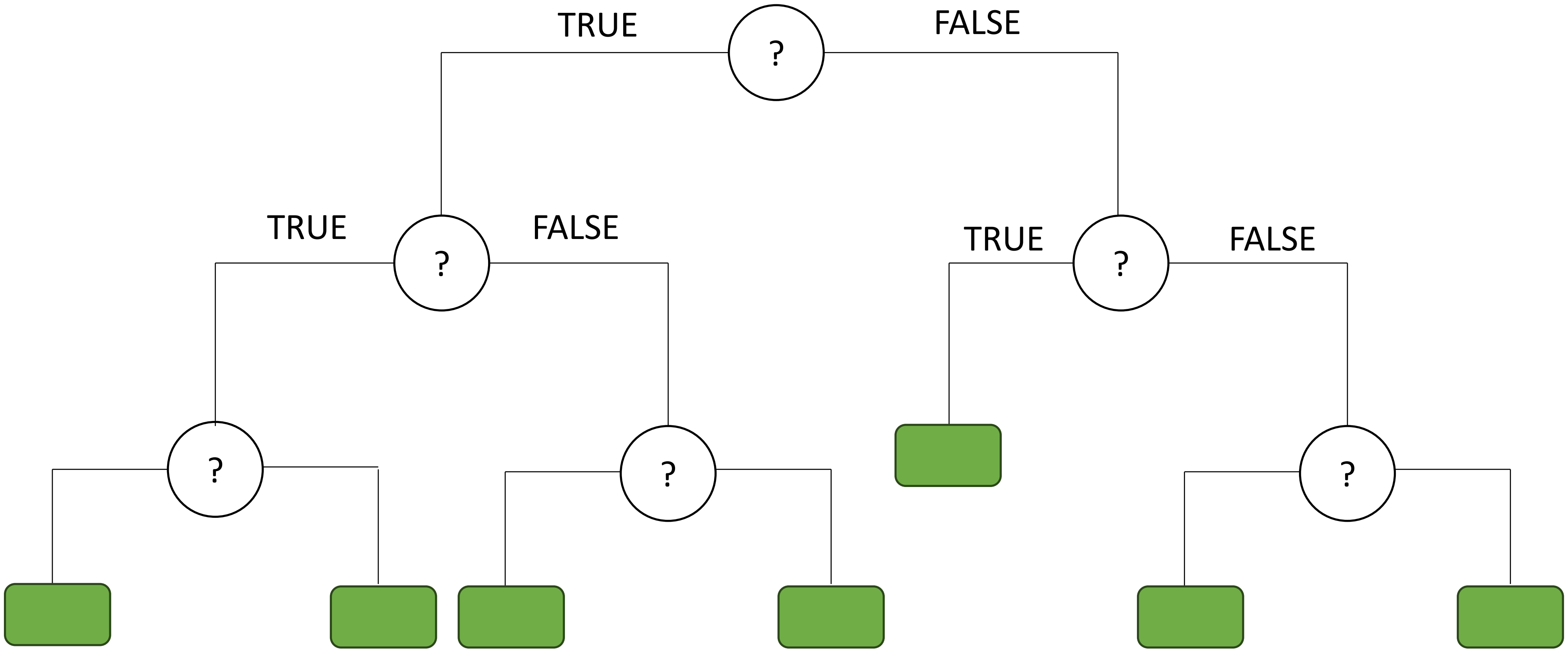
- **Objectives:**
  - Discuss the main concepts between decision trees
  - Understand regression and classification trees
  - Discover random forests and ensemble methods



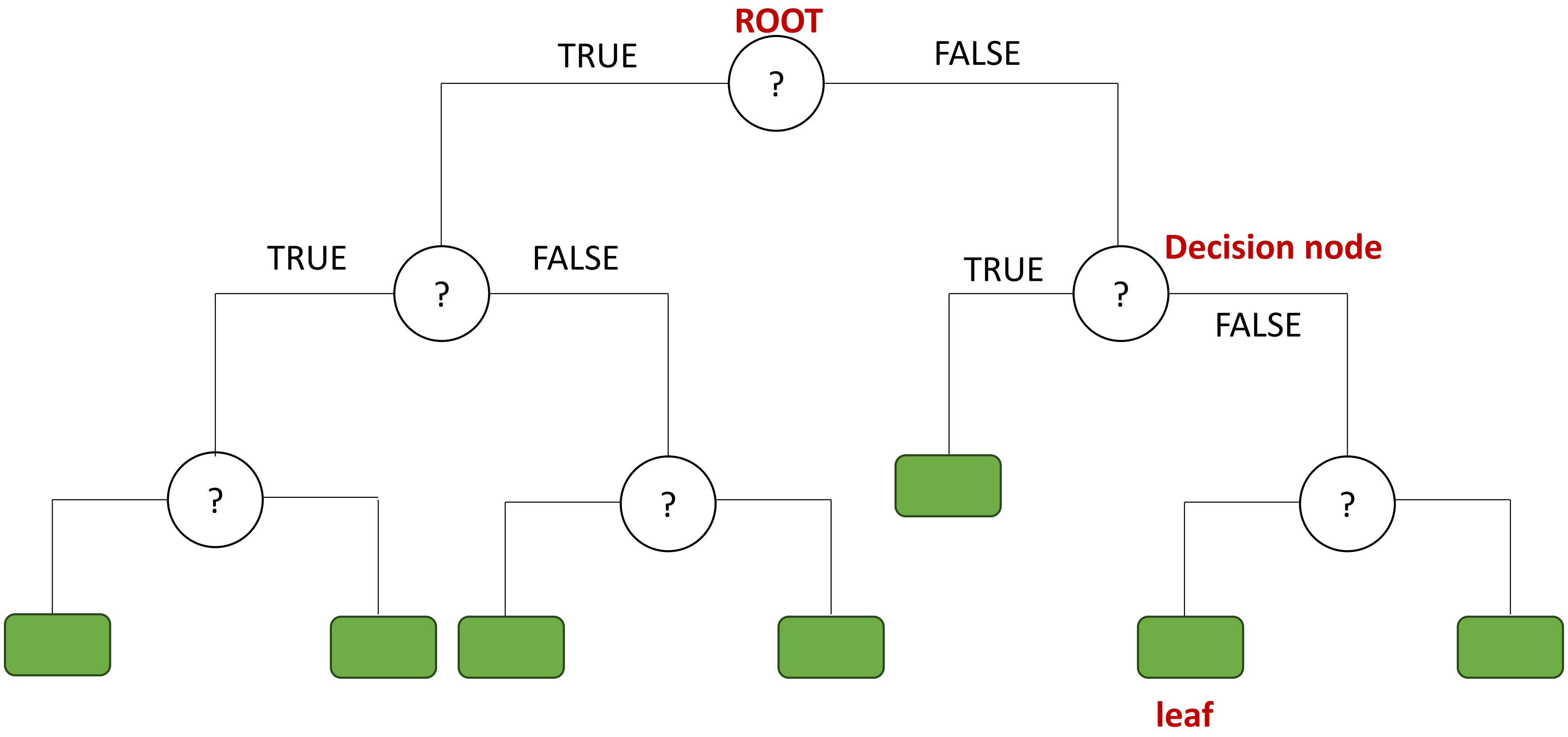
# Decision tree – basic definitions



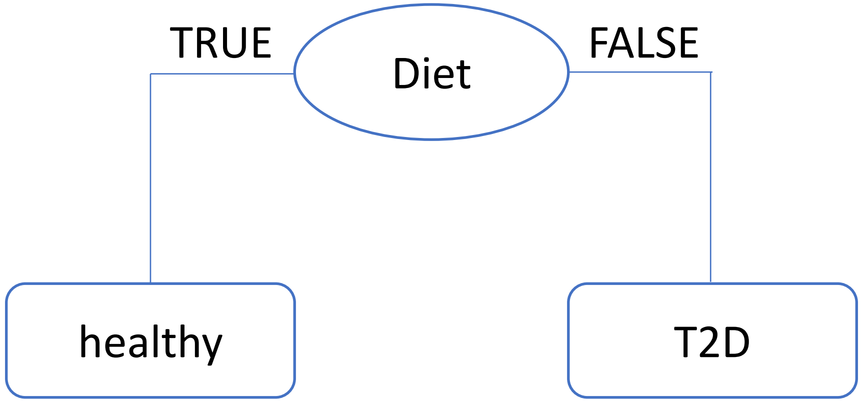
# Decision tree – basic definitions



# Decision tree – basic definitions



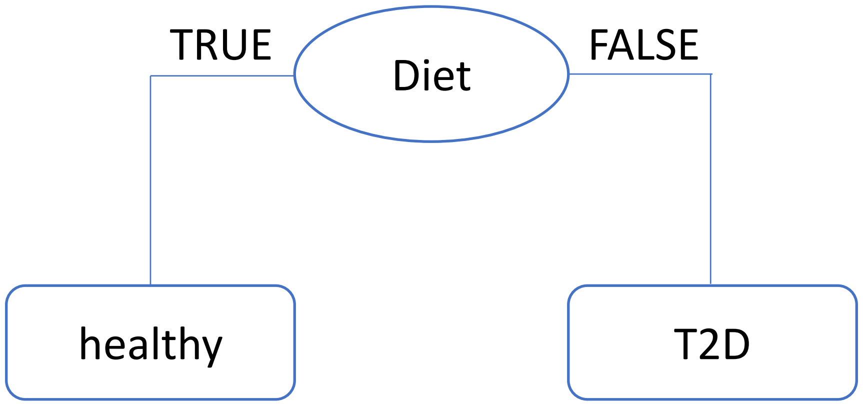
# Decision tree – basic definition




A decision tree can be used to perform a classification task

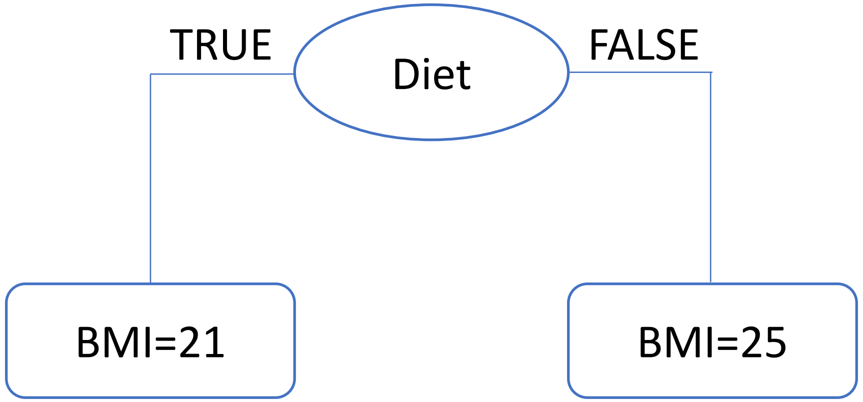
Classification tree

# Decision tree – basic definition




A decision tree can be used to perform a classification task

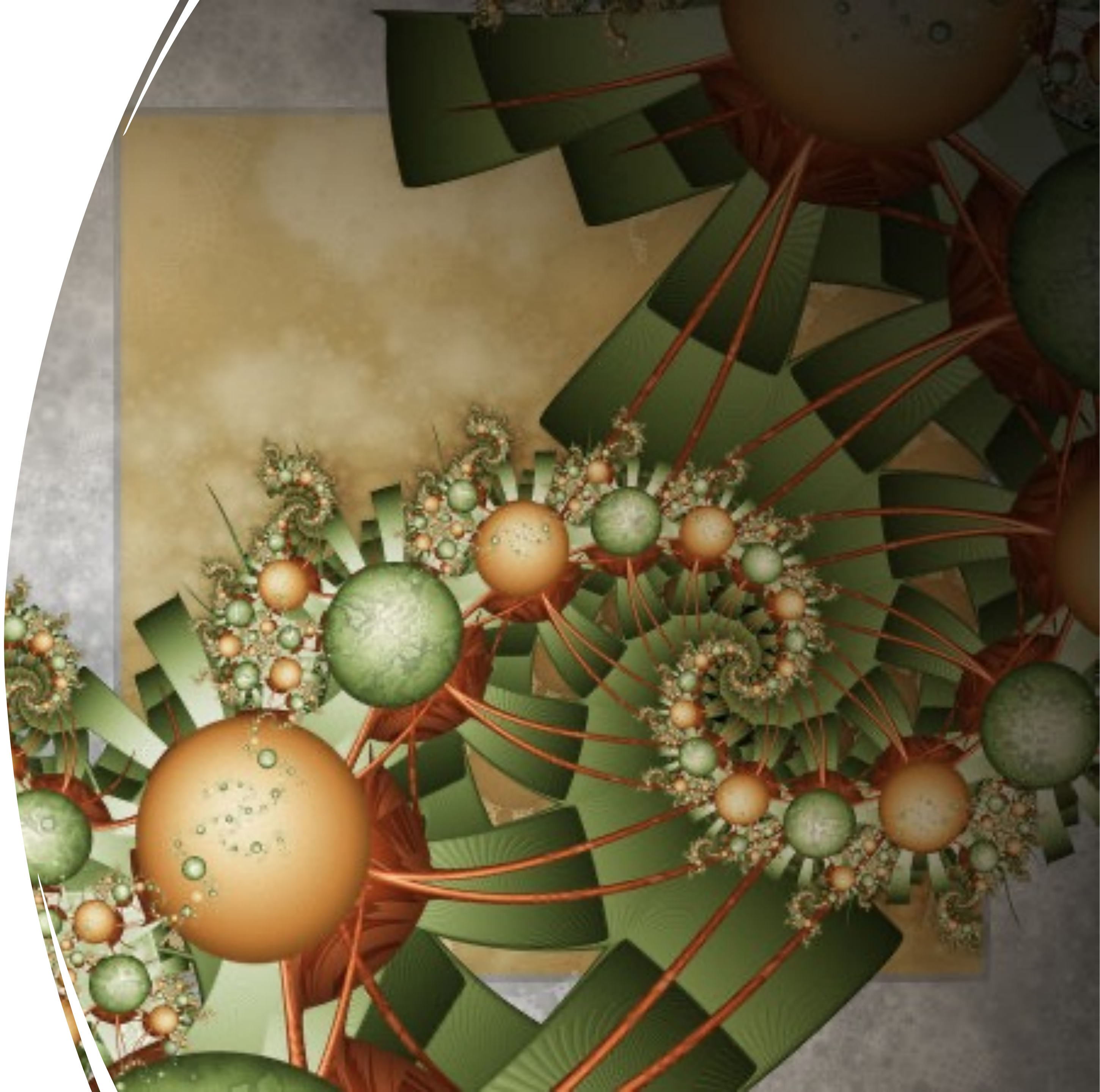
Classification tree



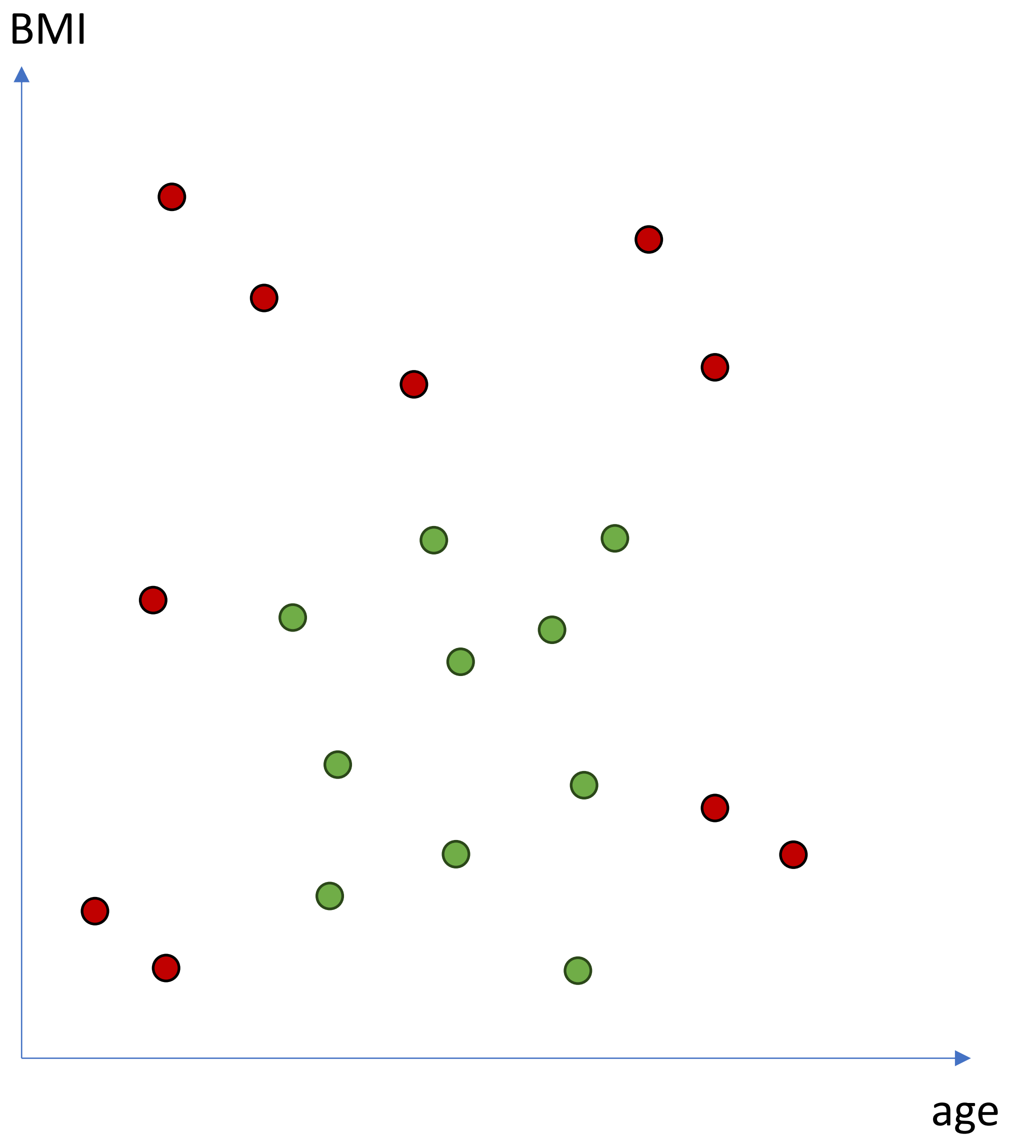

A decision tree can be used to perform a regression task

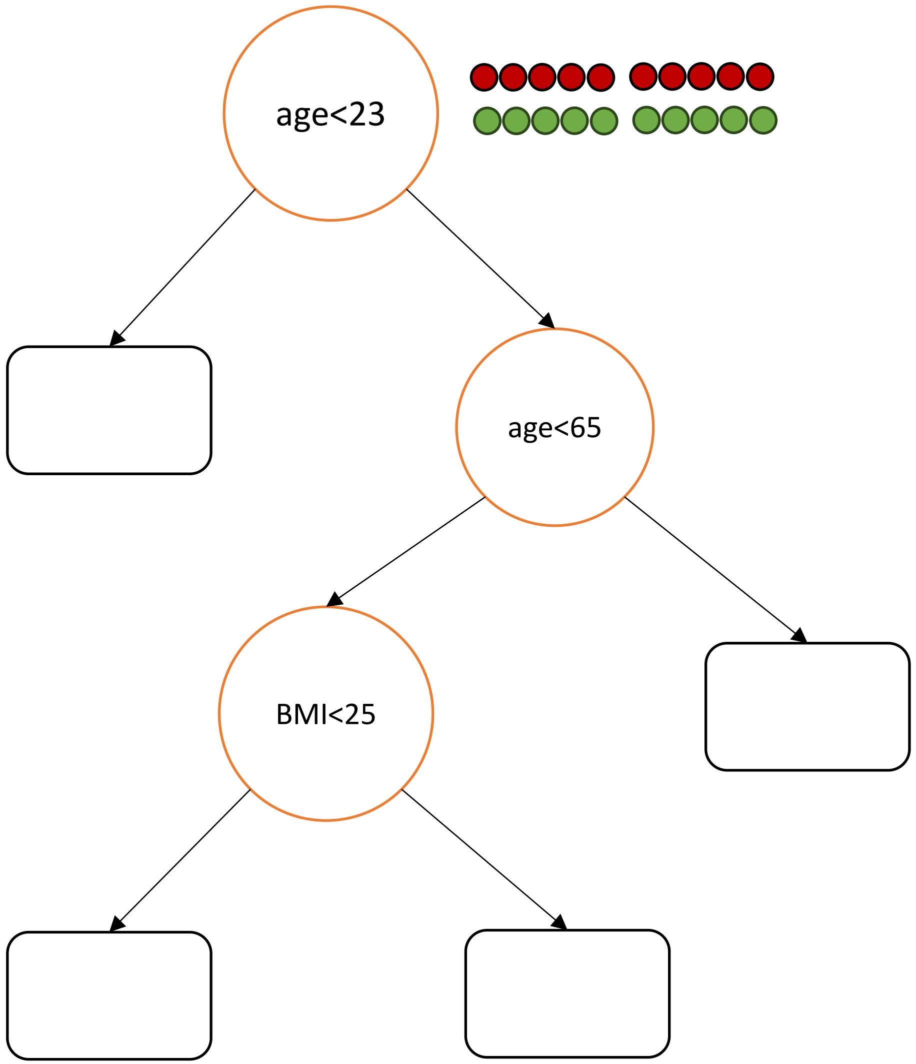
regression tree

# Classification trees

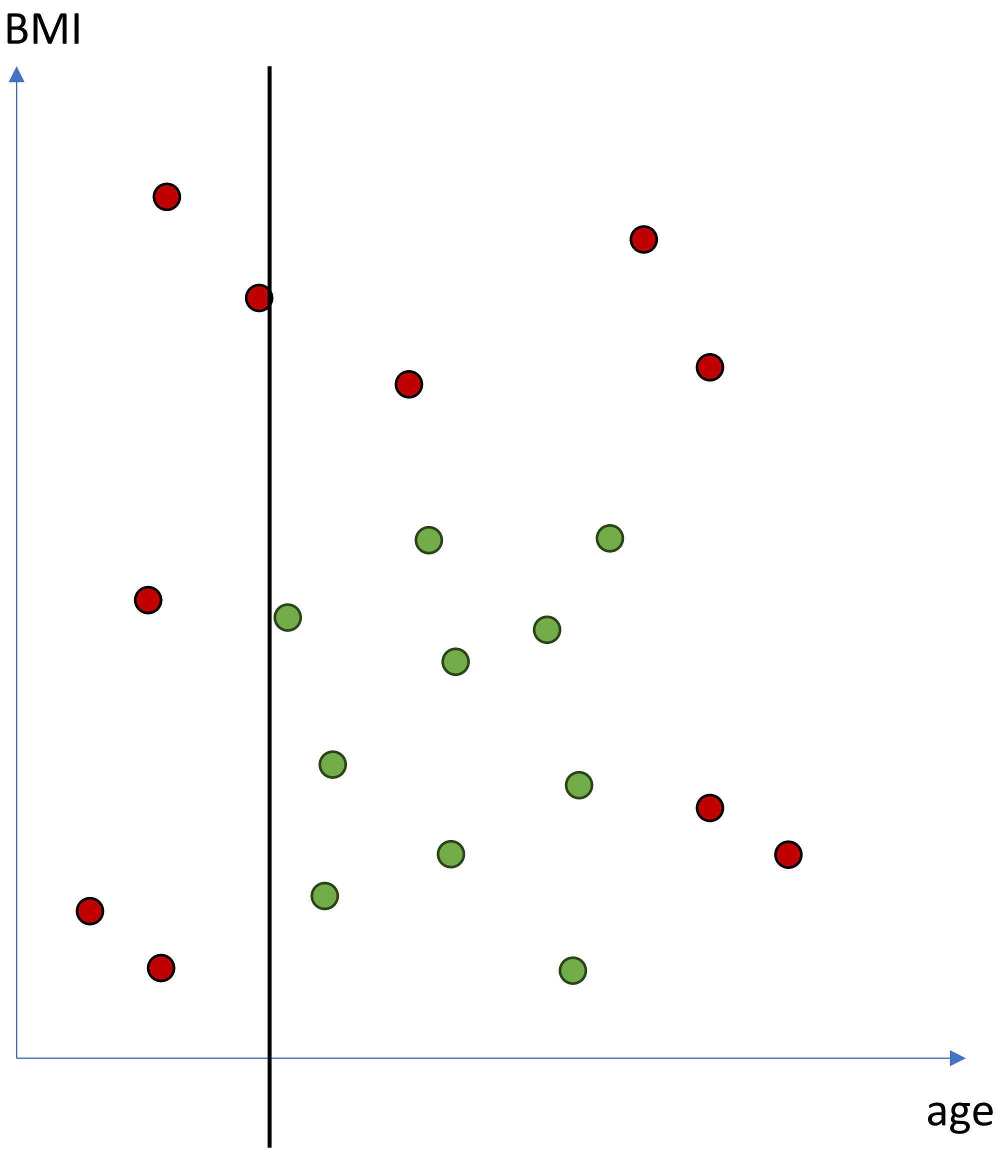


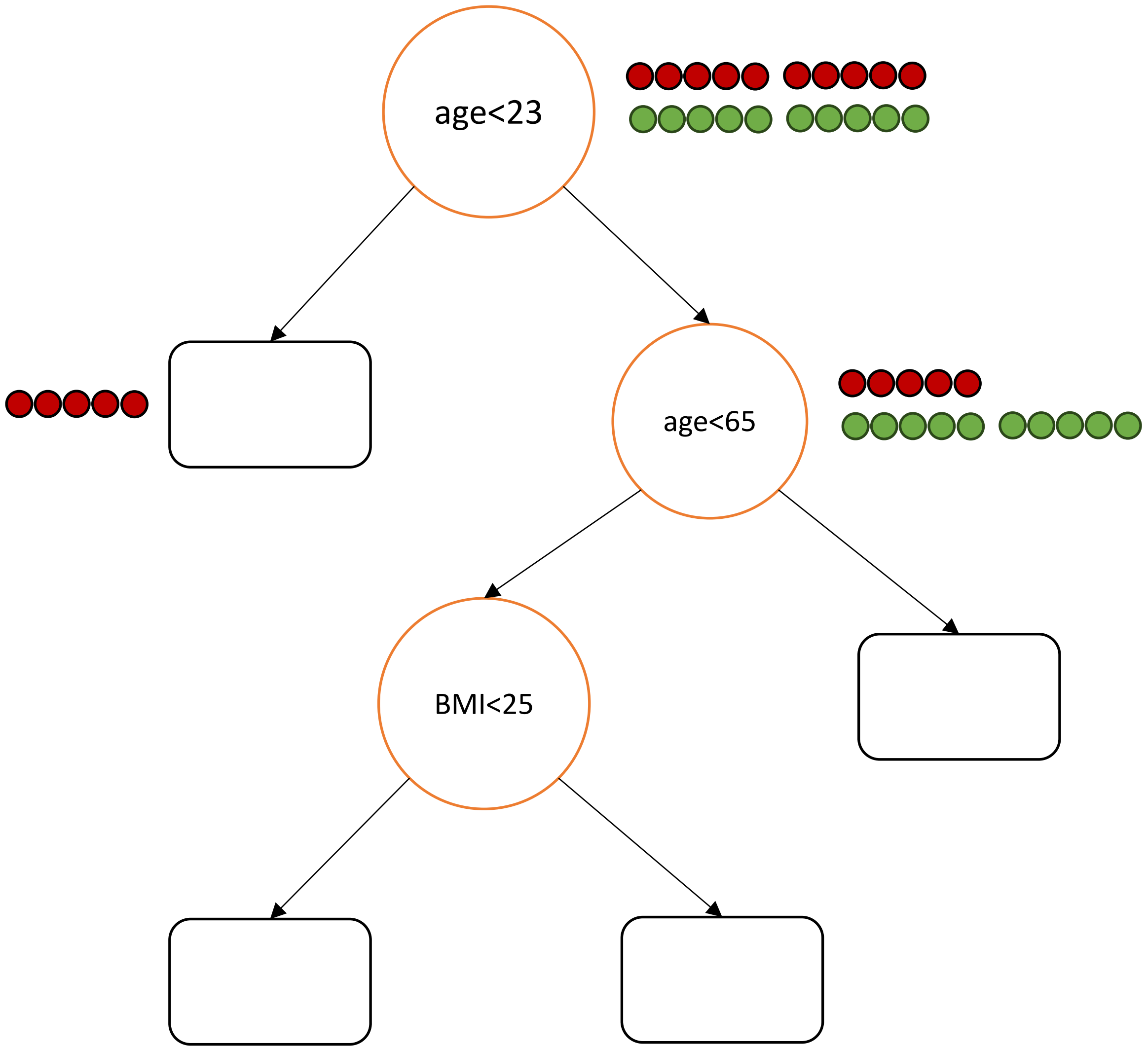
# Classification decision tree



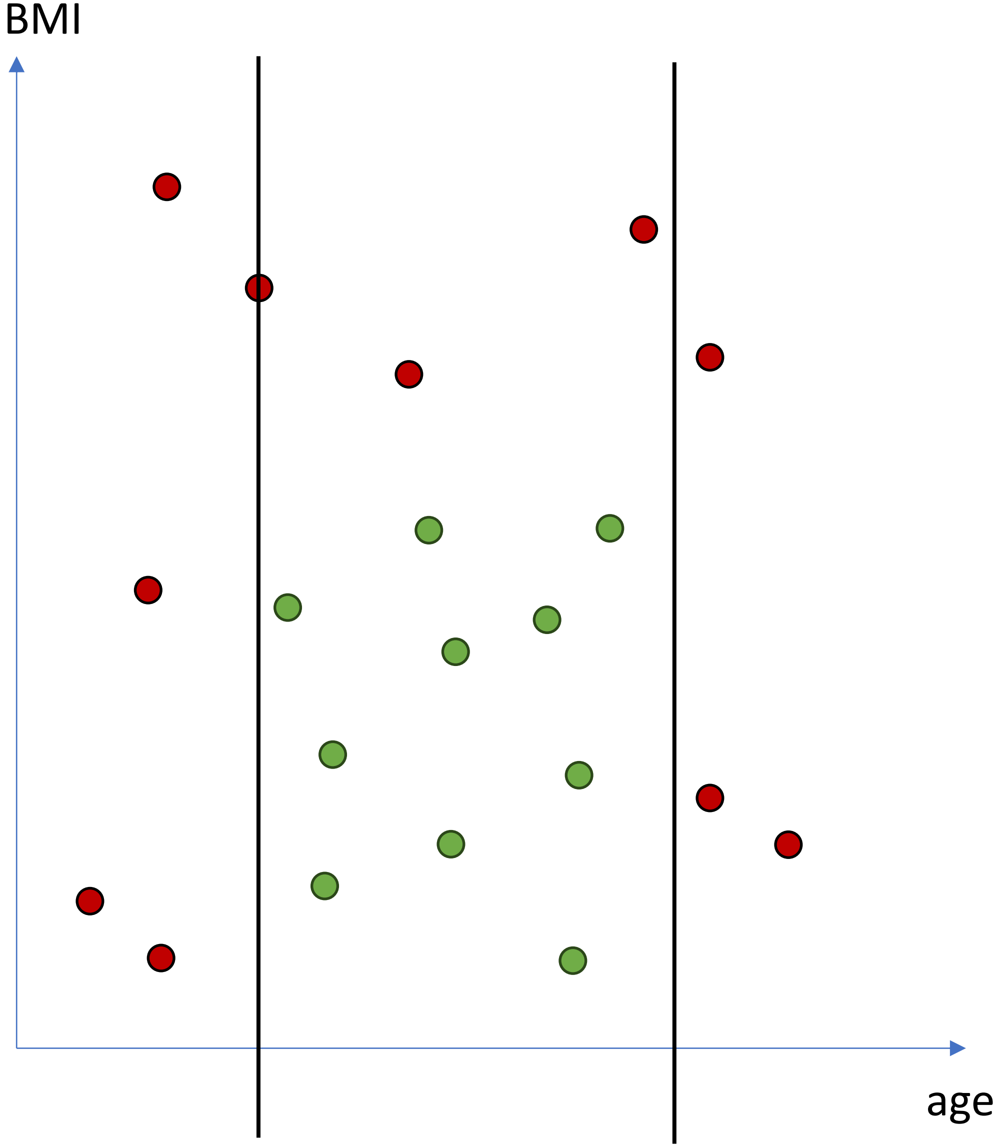



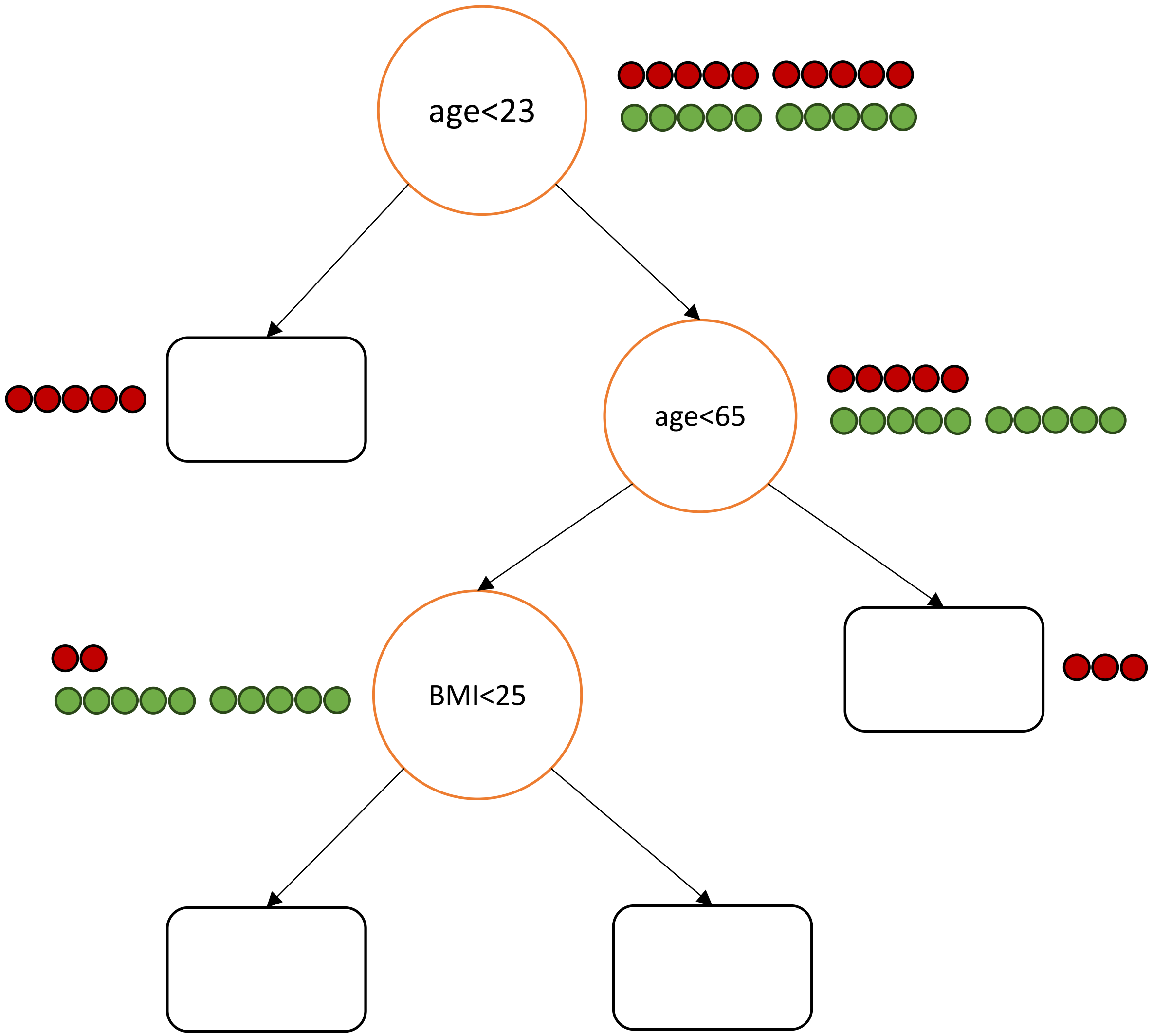
# Classification decision tree



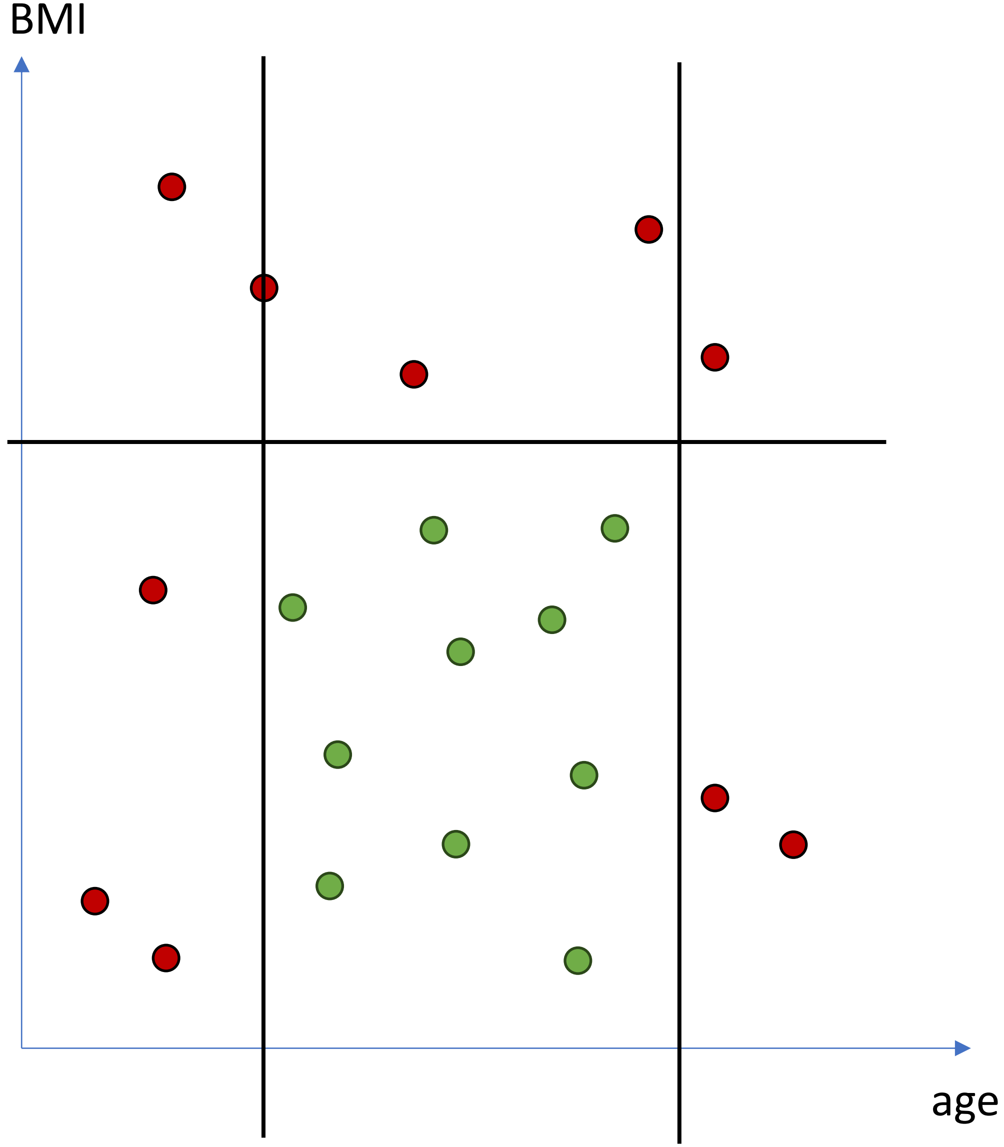



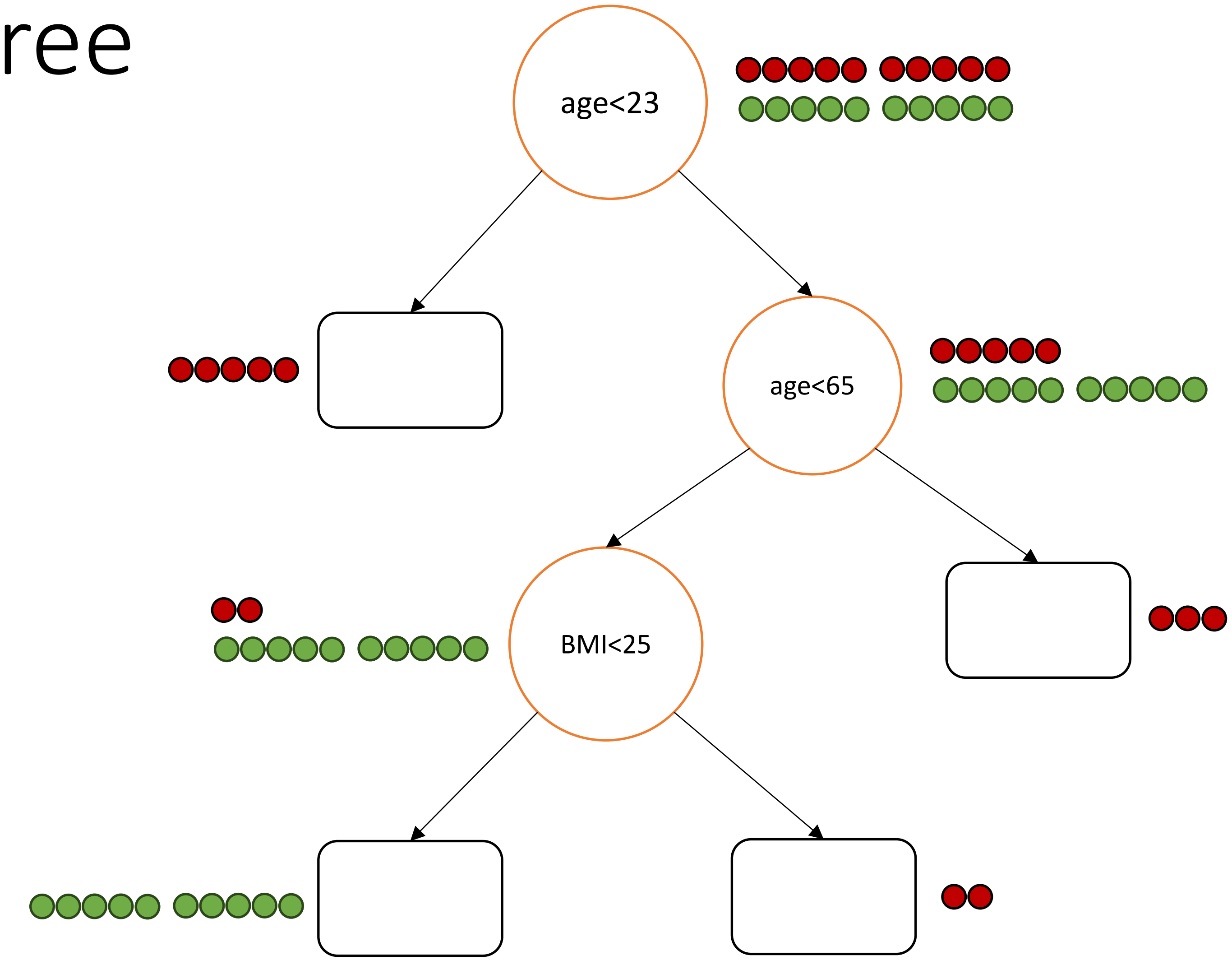
# Classification decision tree



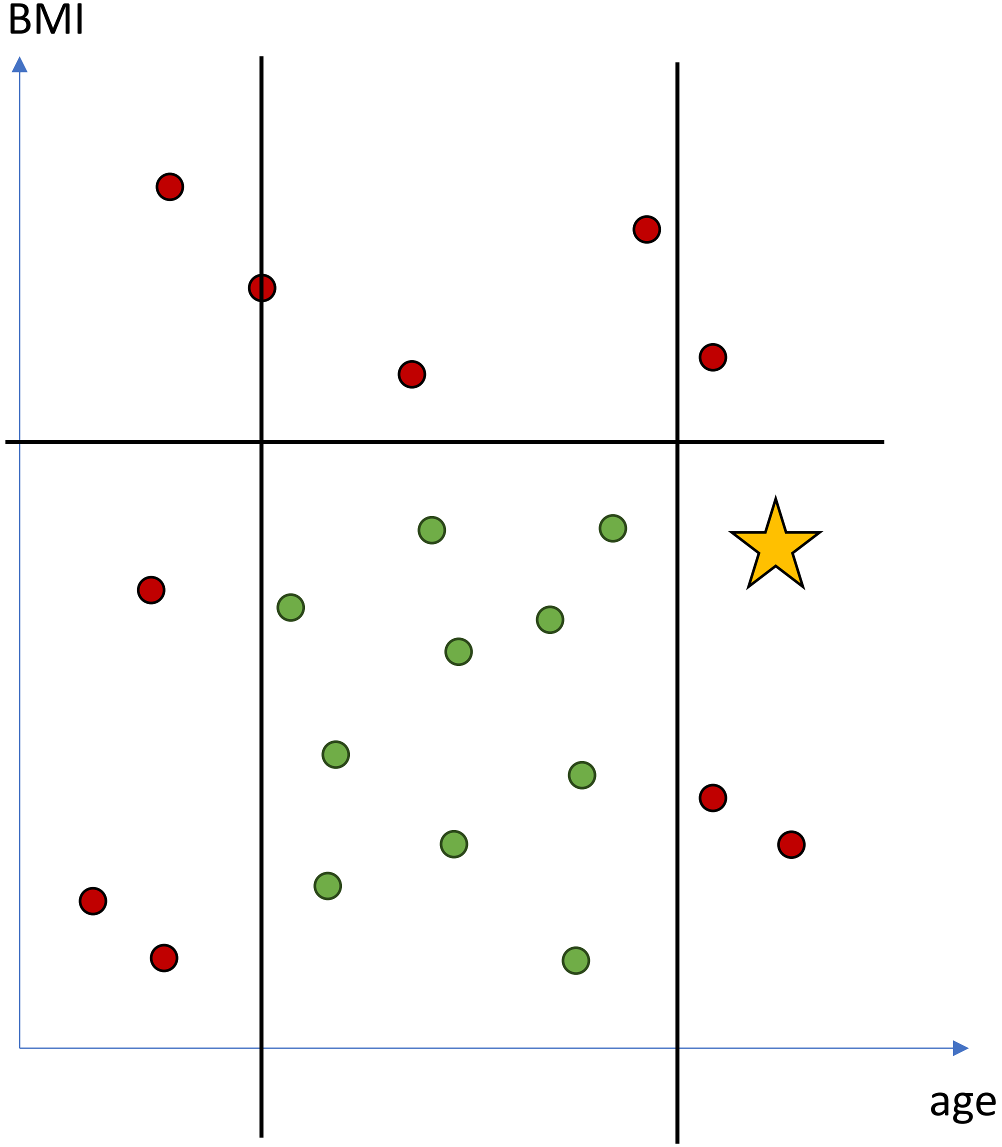



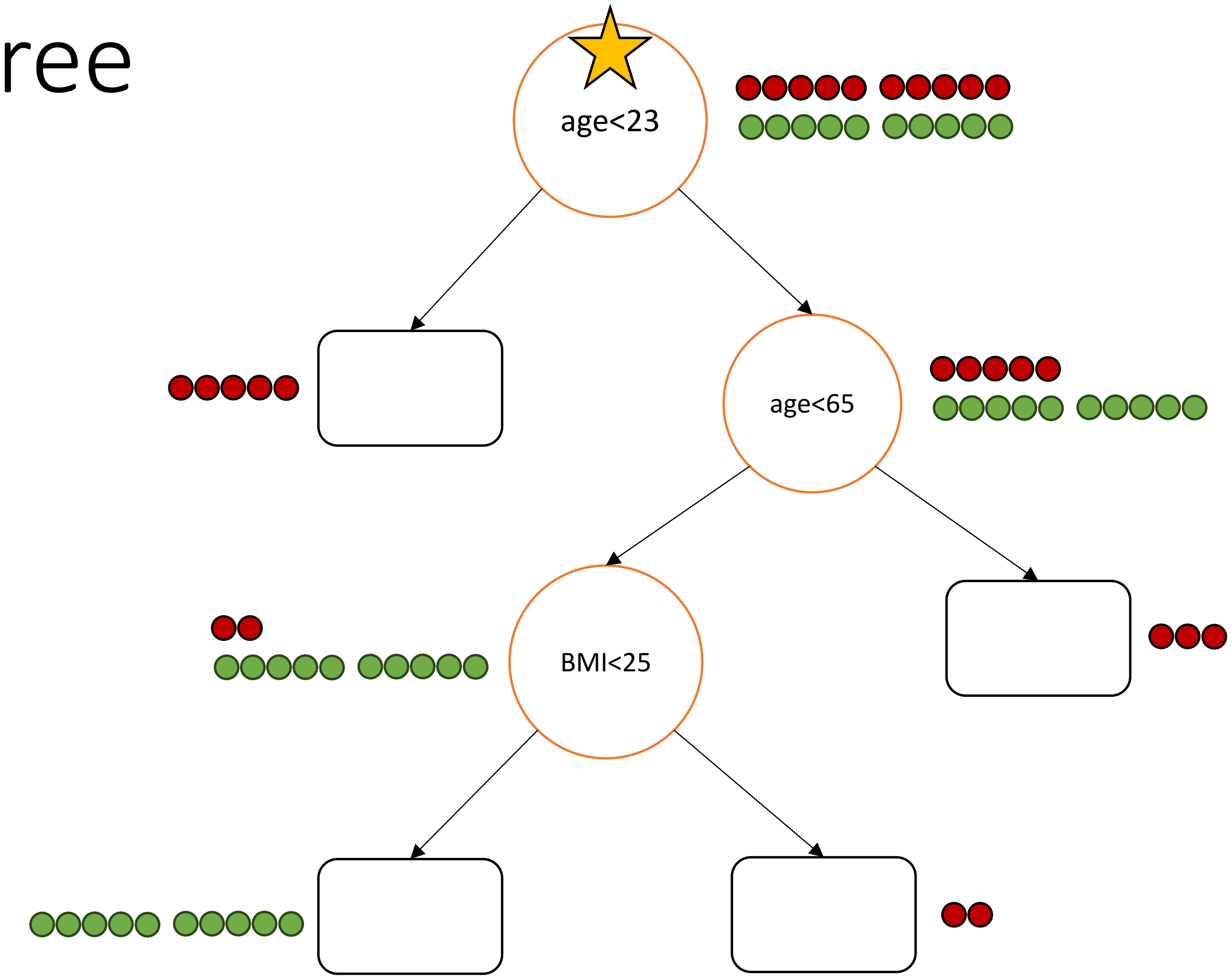
# Classification decision tree



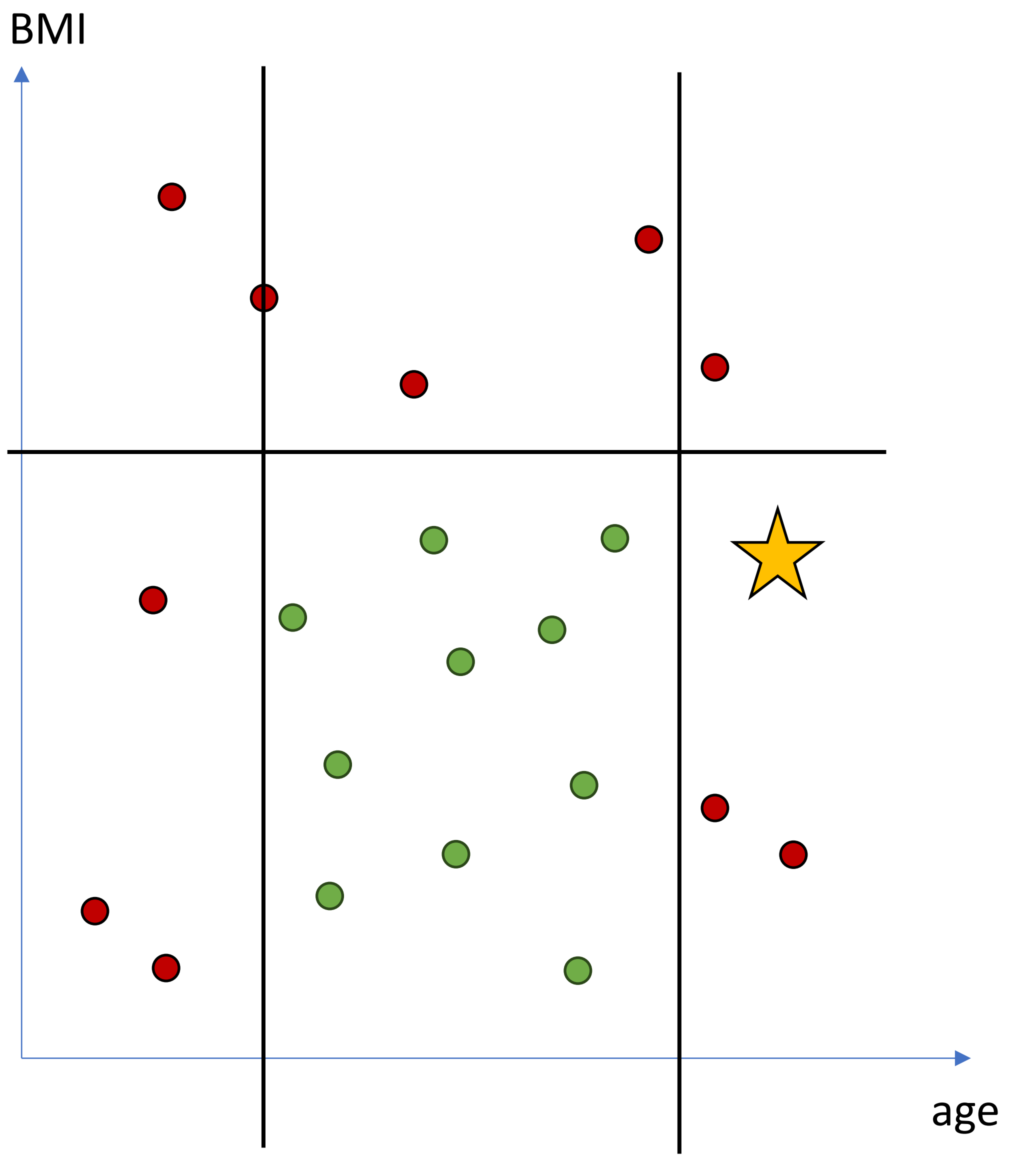



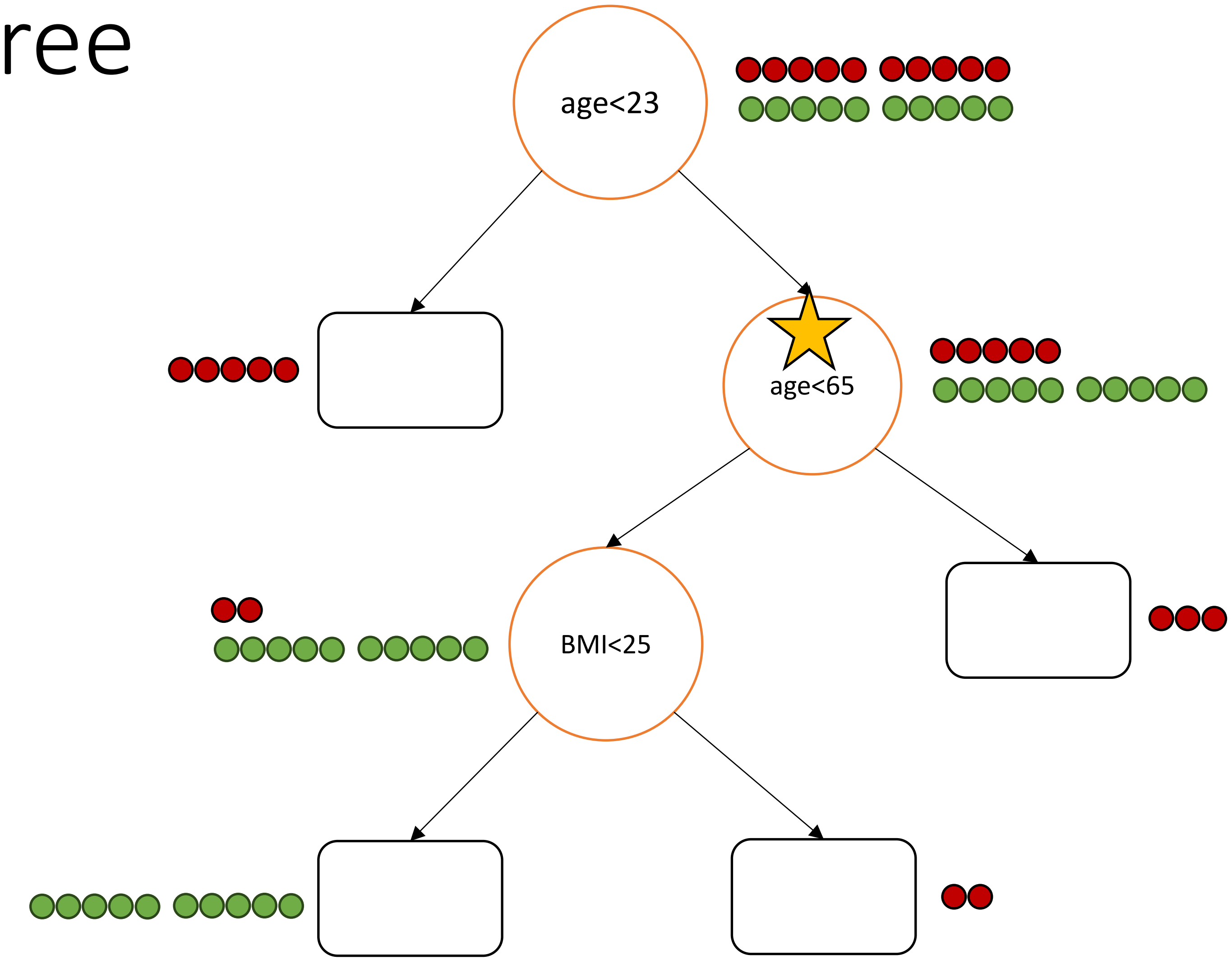
# Classification decision tree



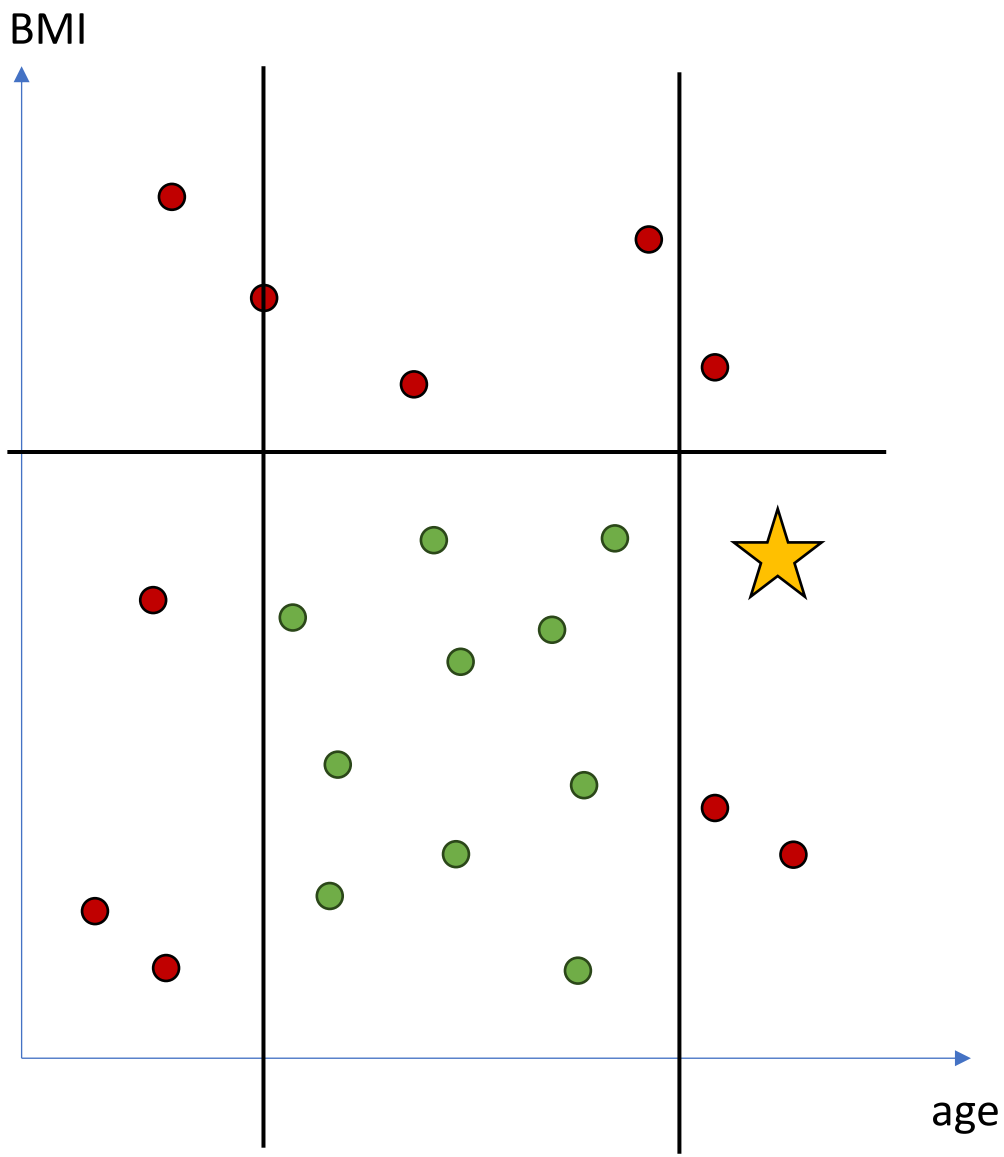



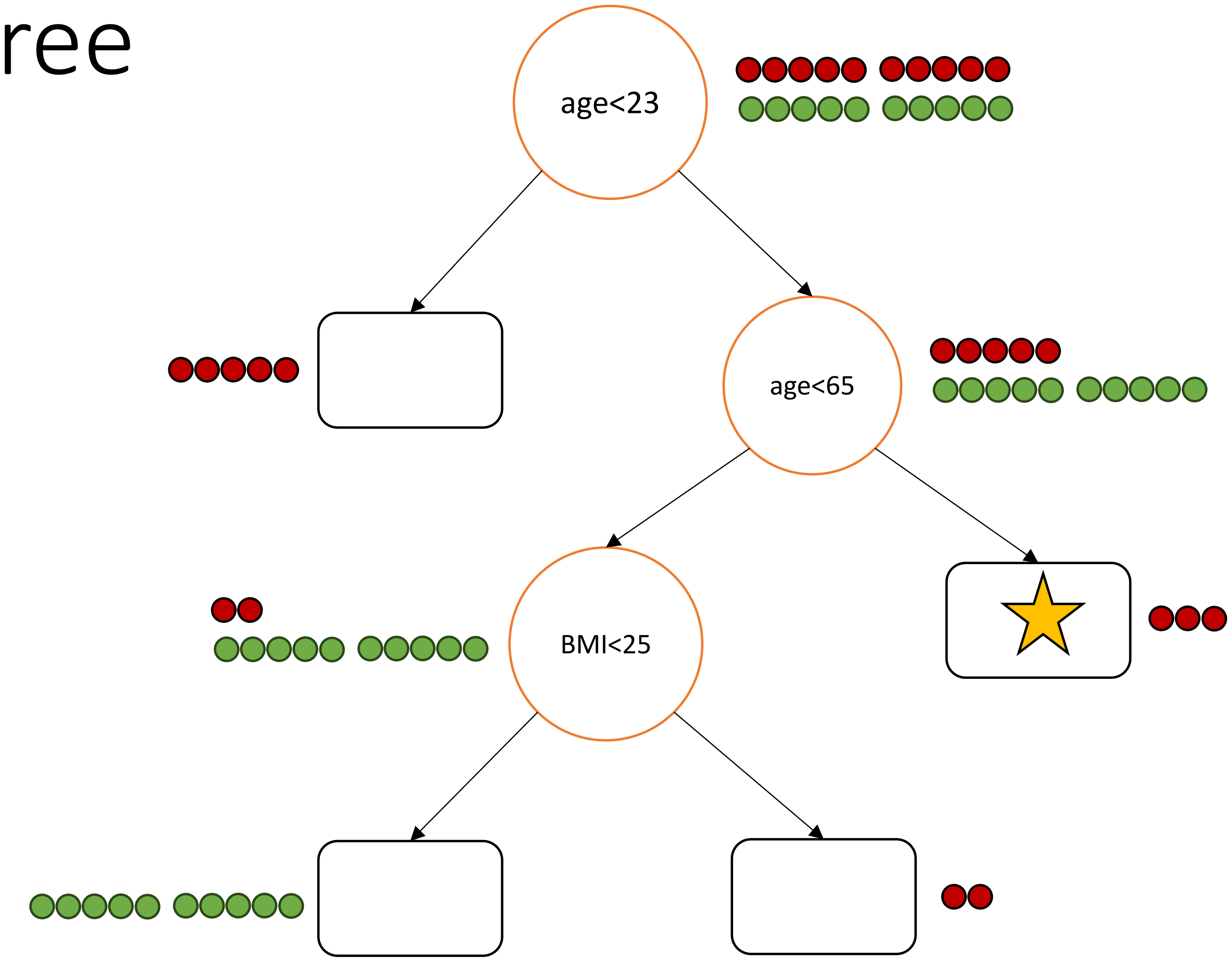
# Classification decision tree



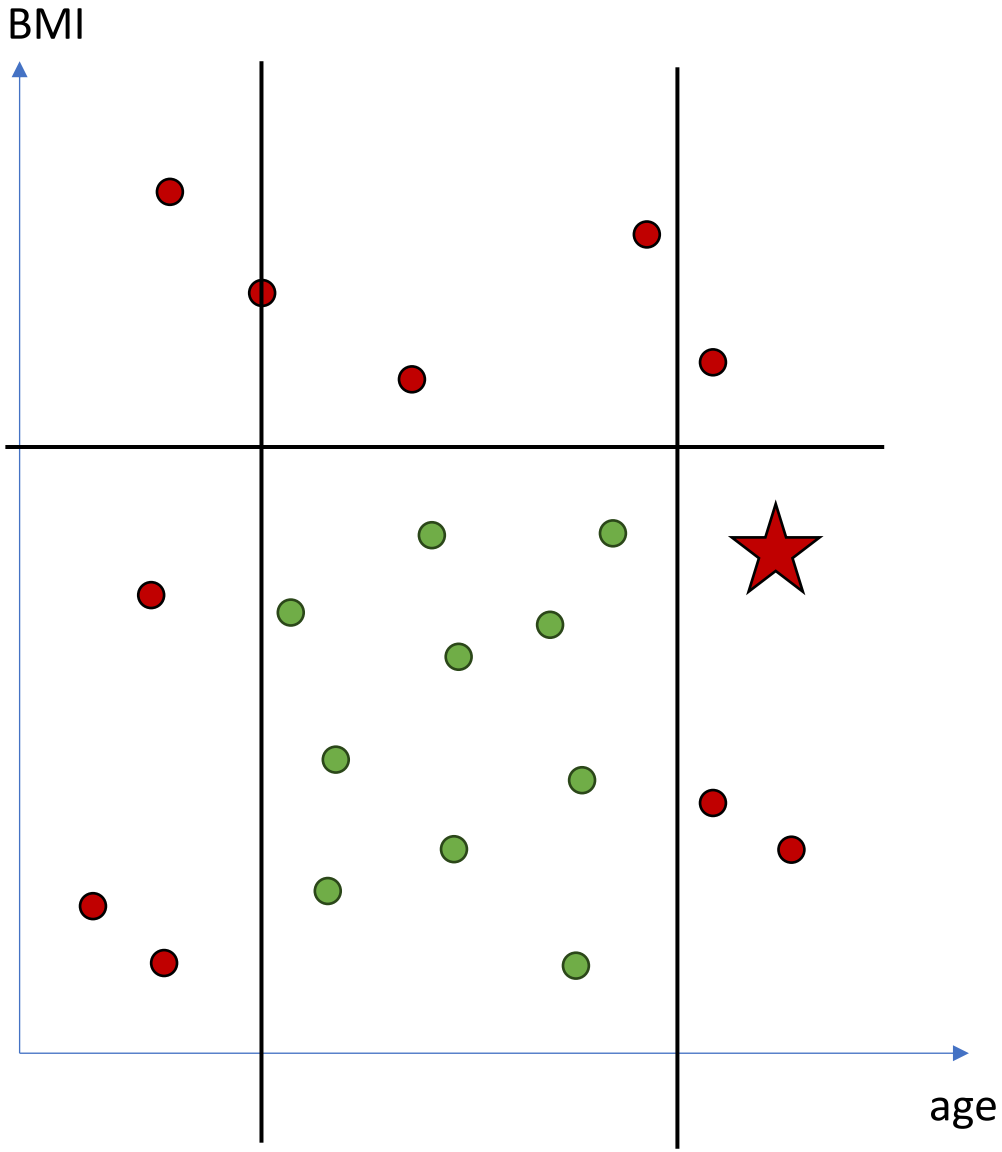



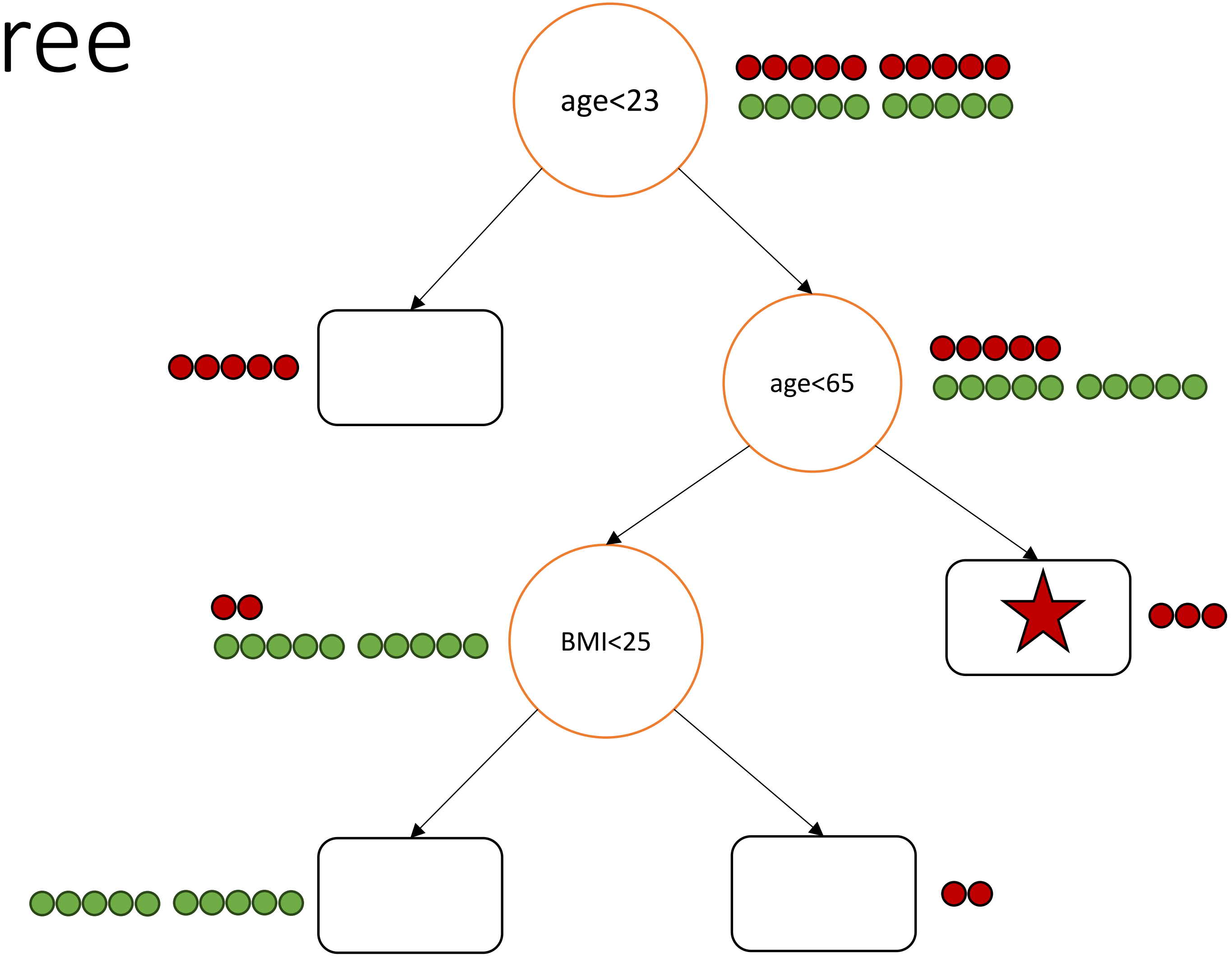
# Classification decision tree



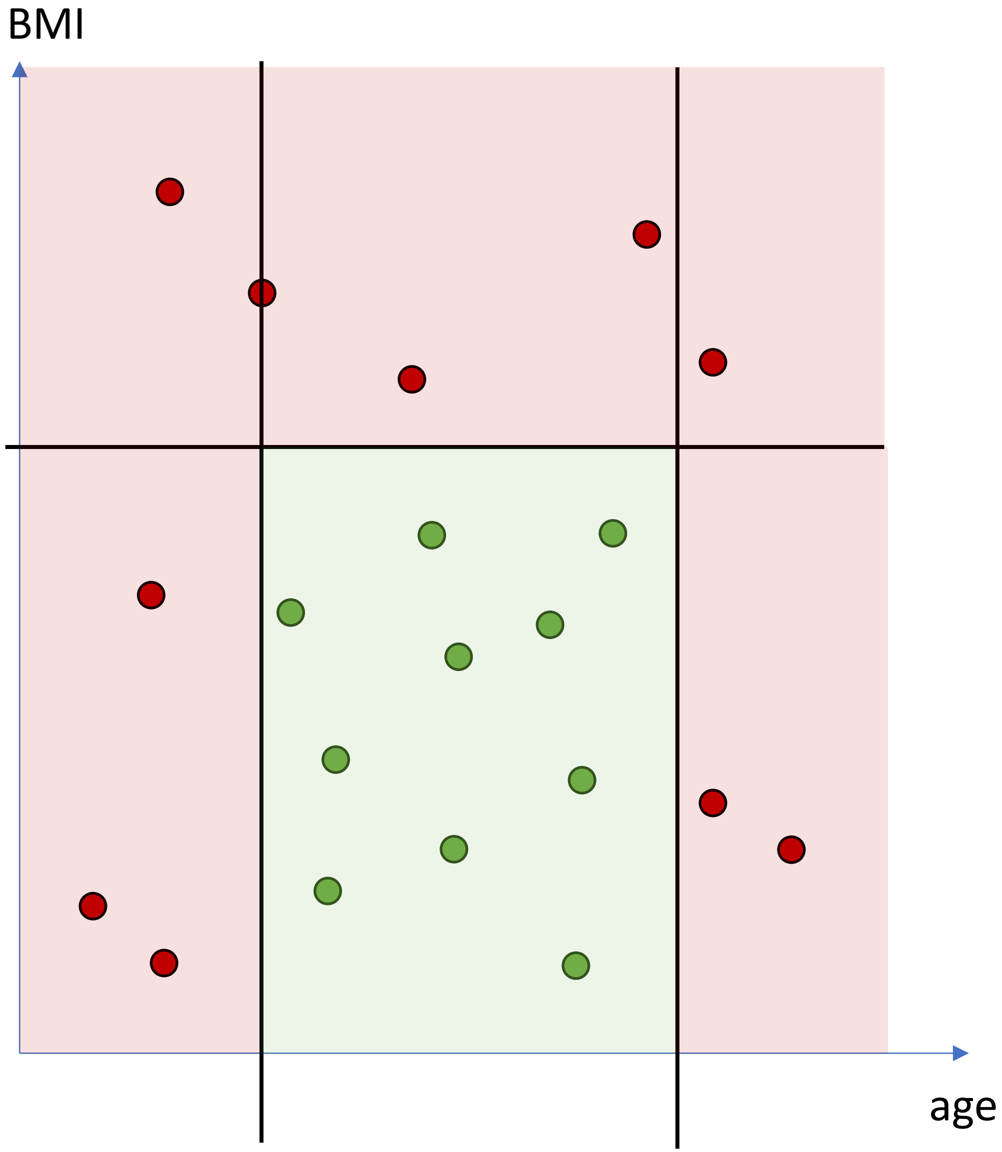



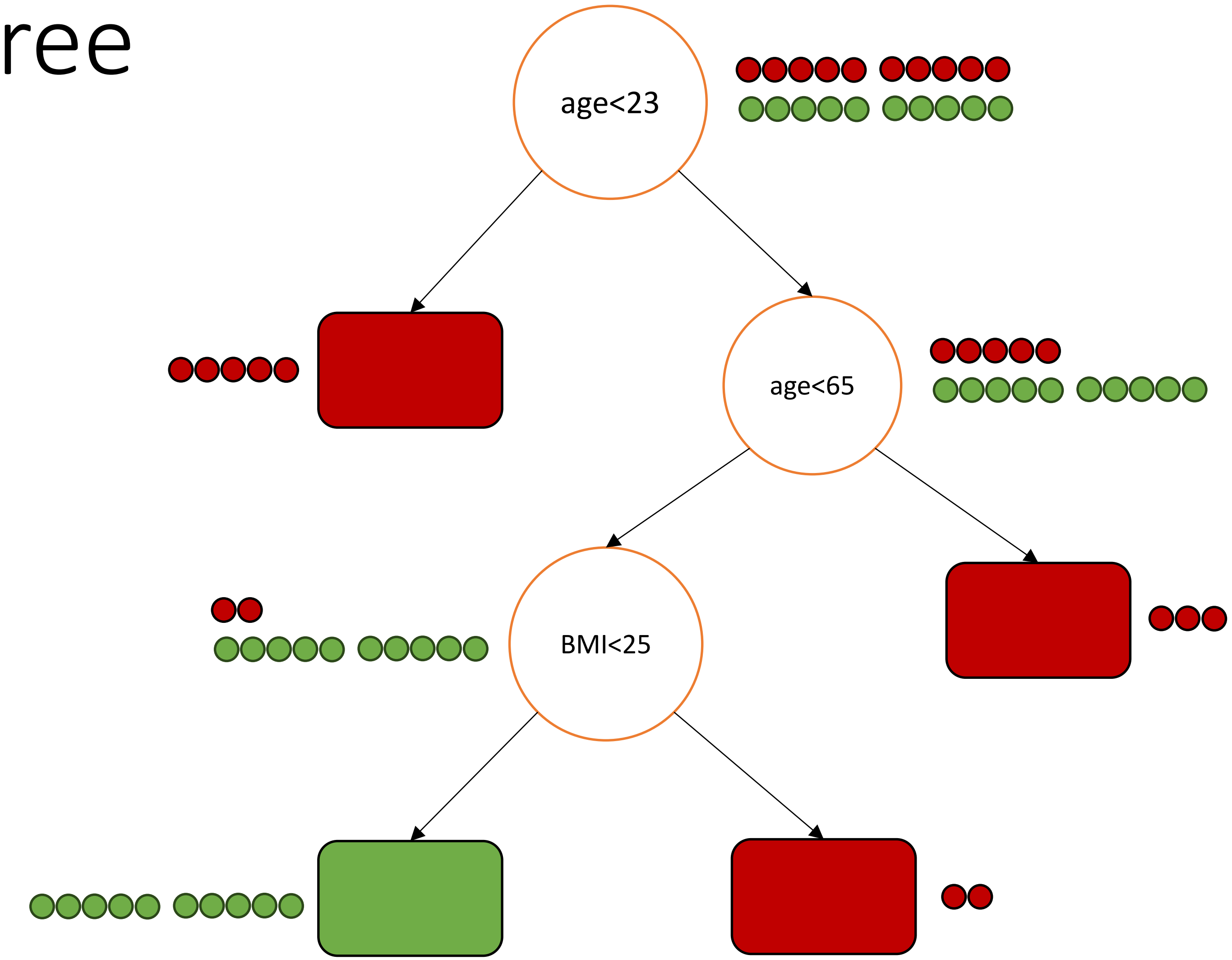
# Classification decision tree



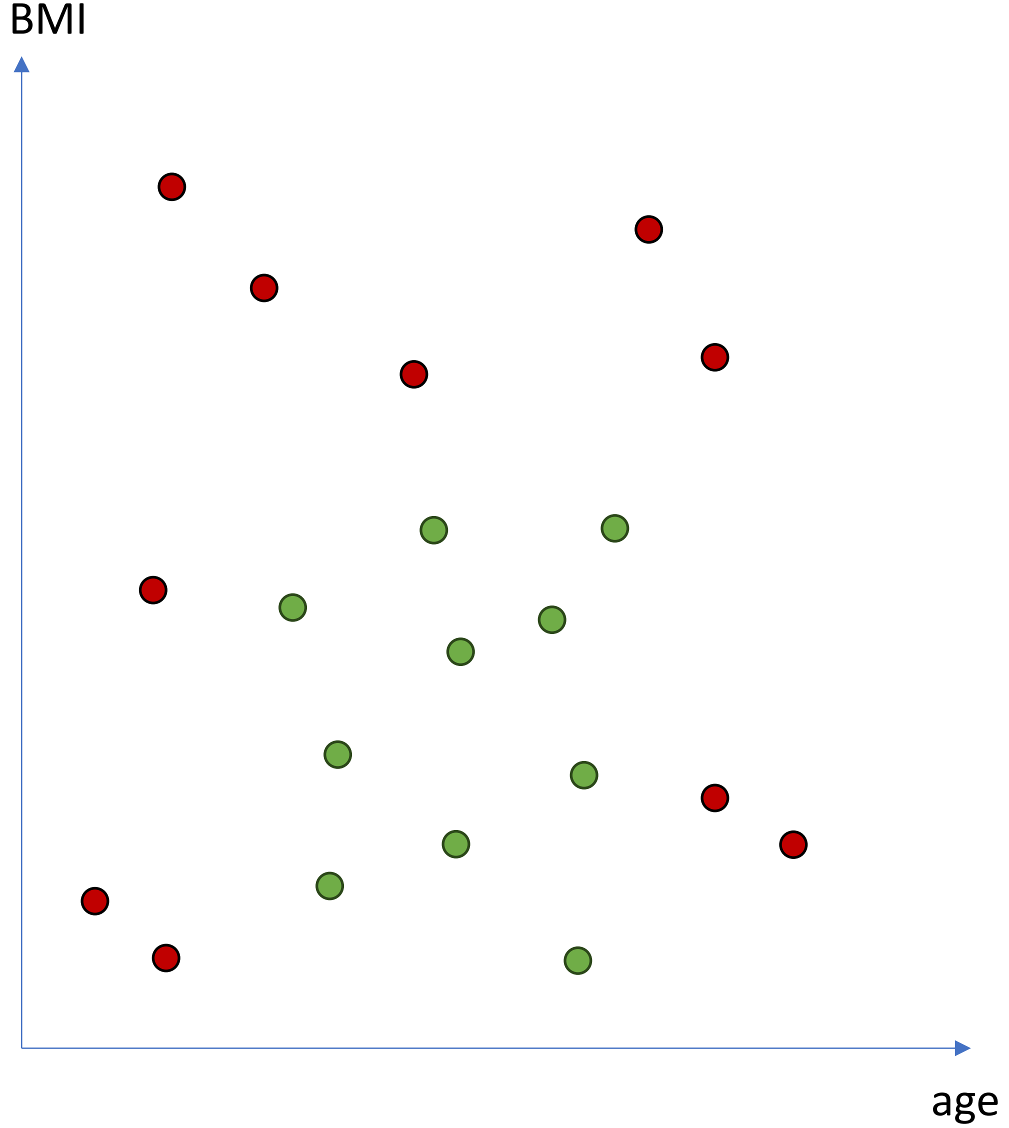



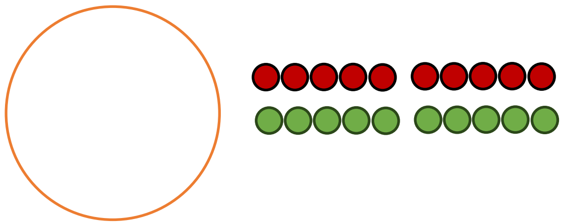
# Classification decision tree



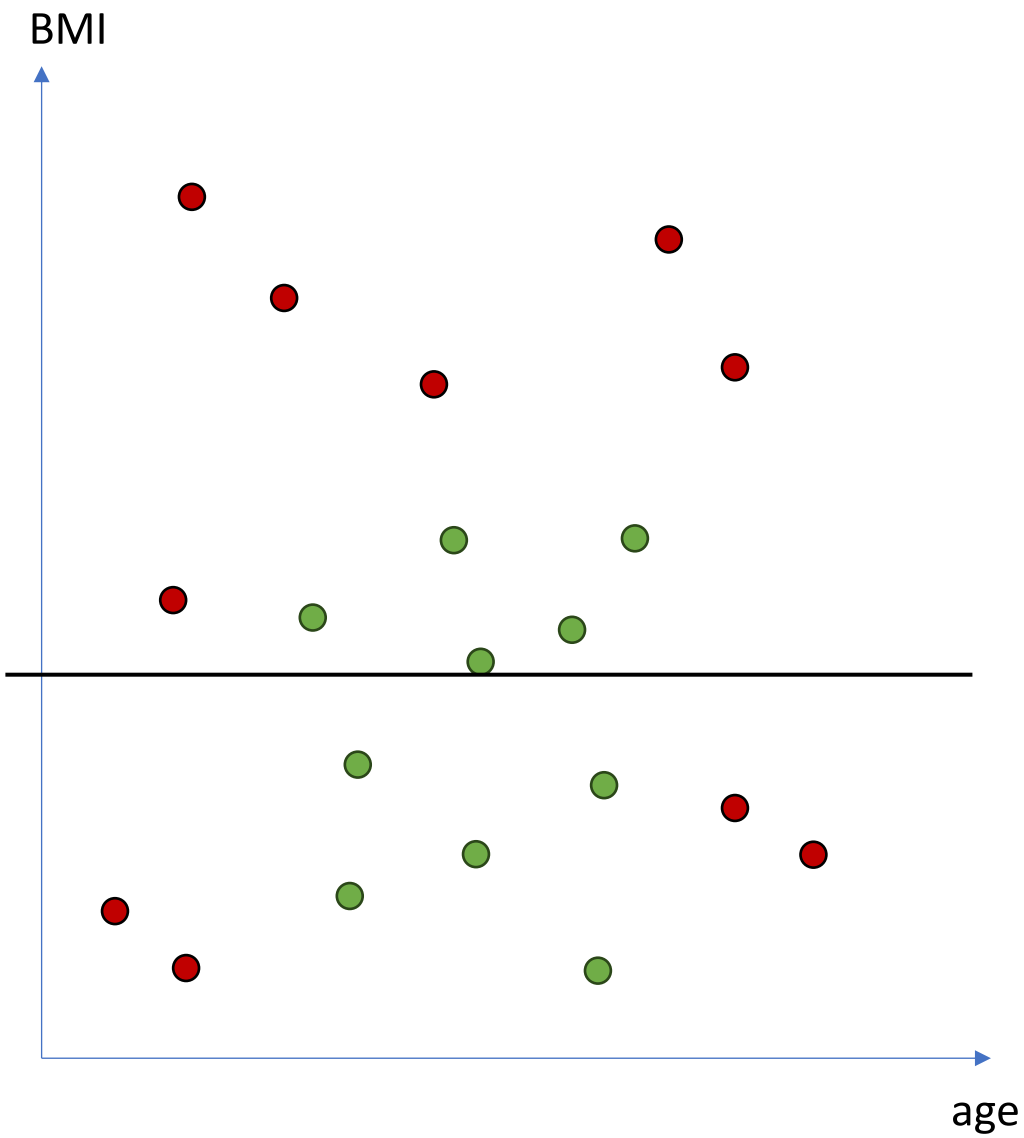



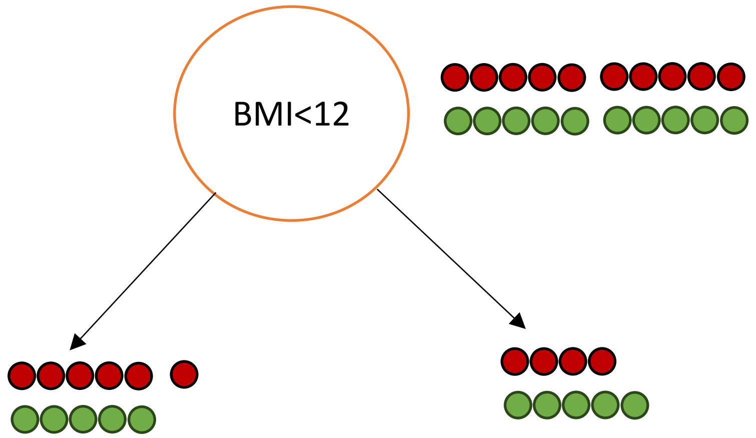
# Training a decision tree



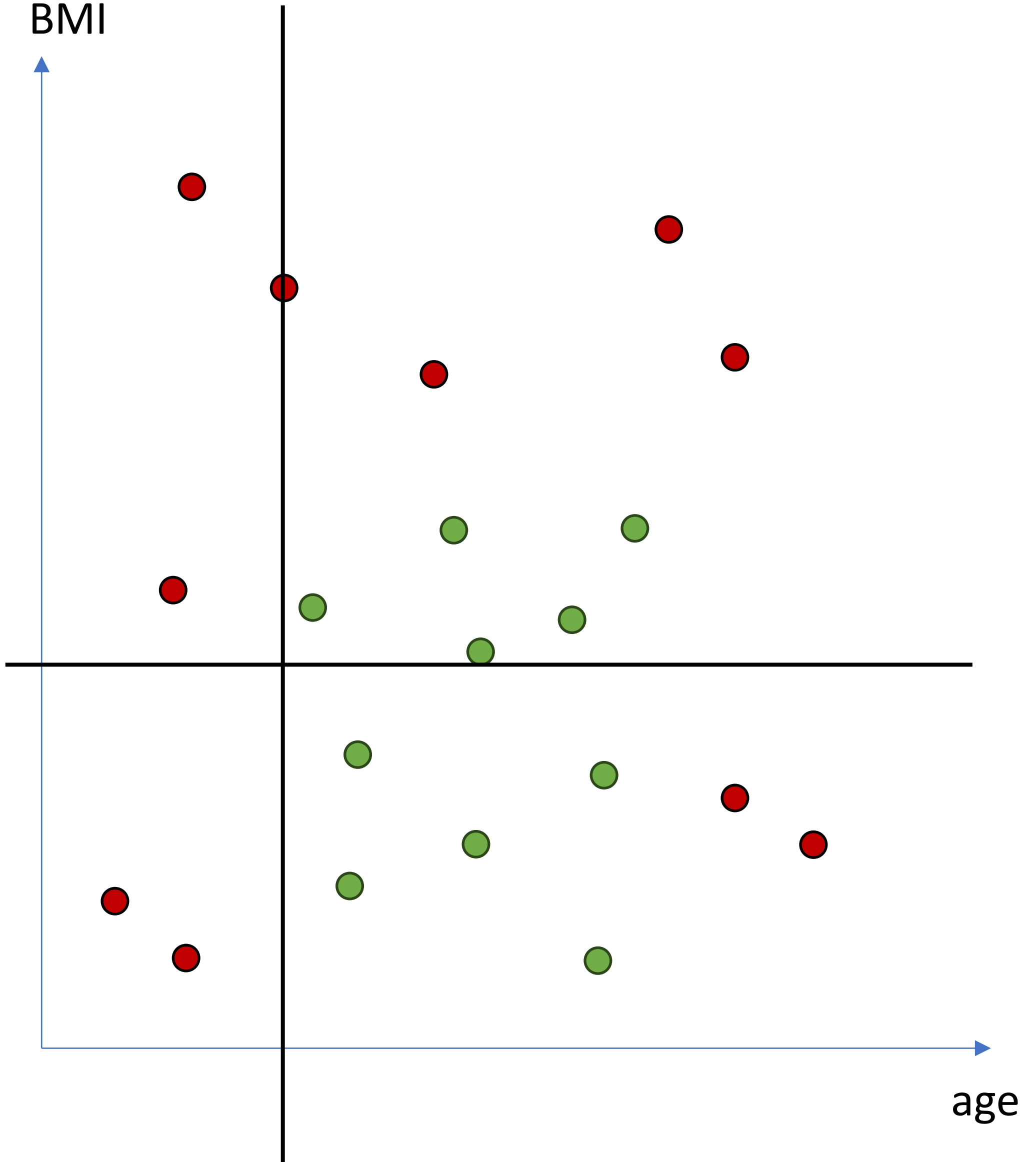



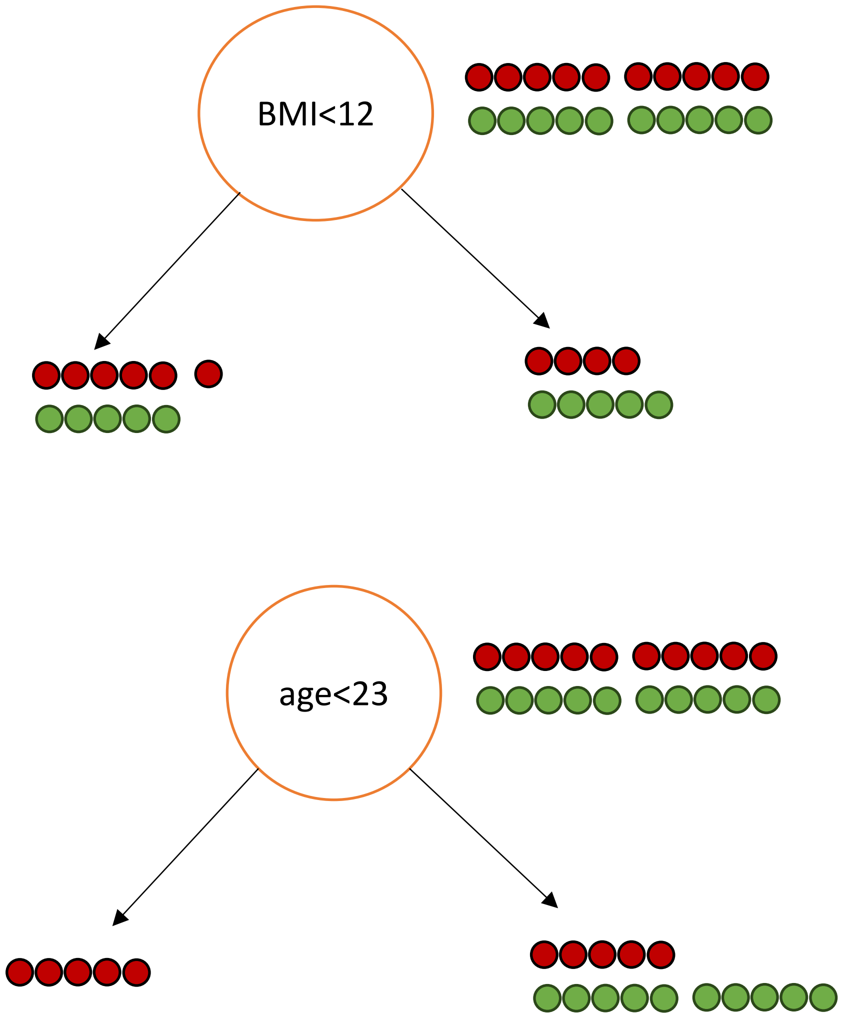
# Training a decision tree



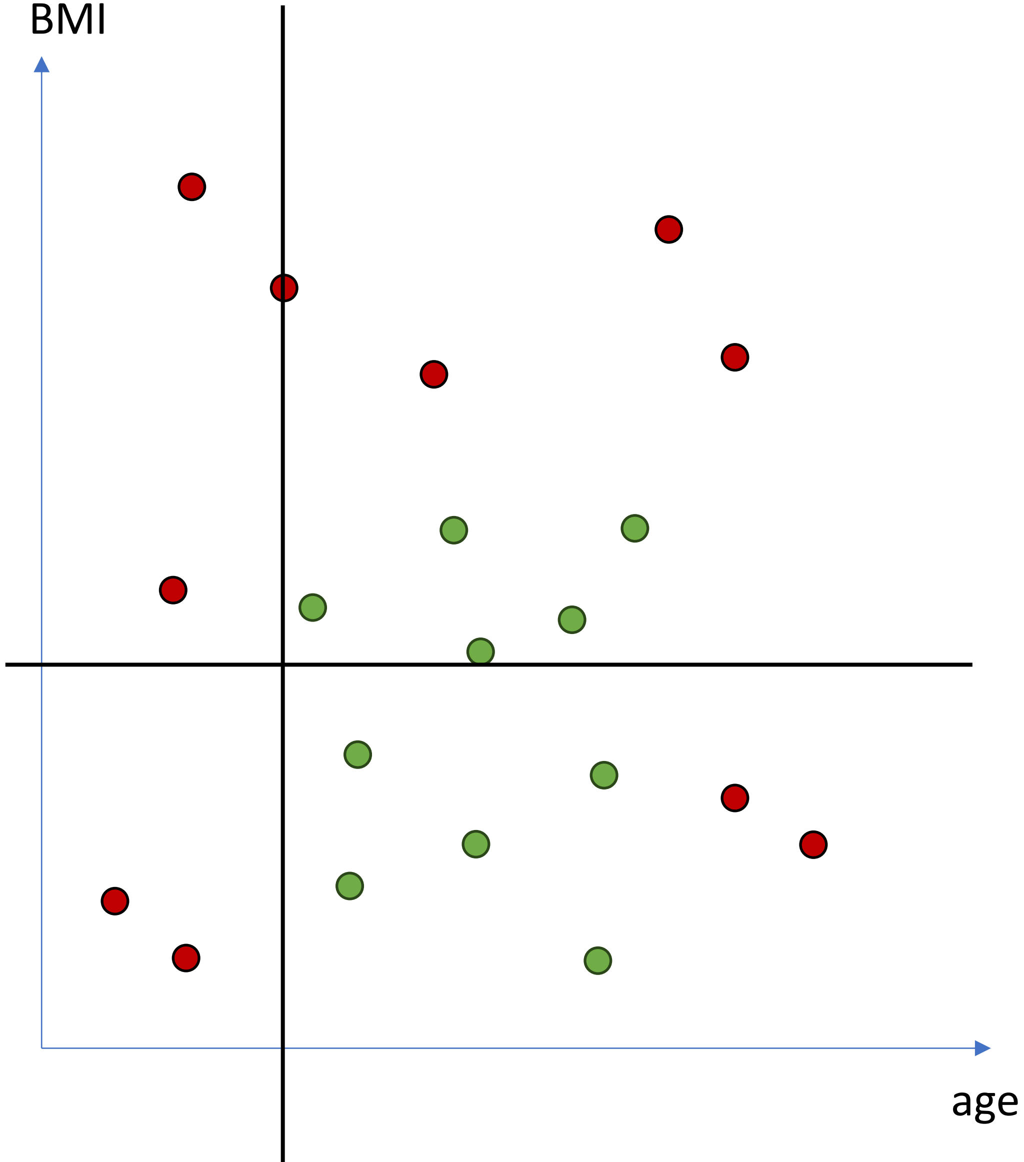
# Training a decision tree

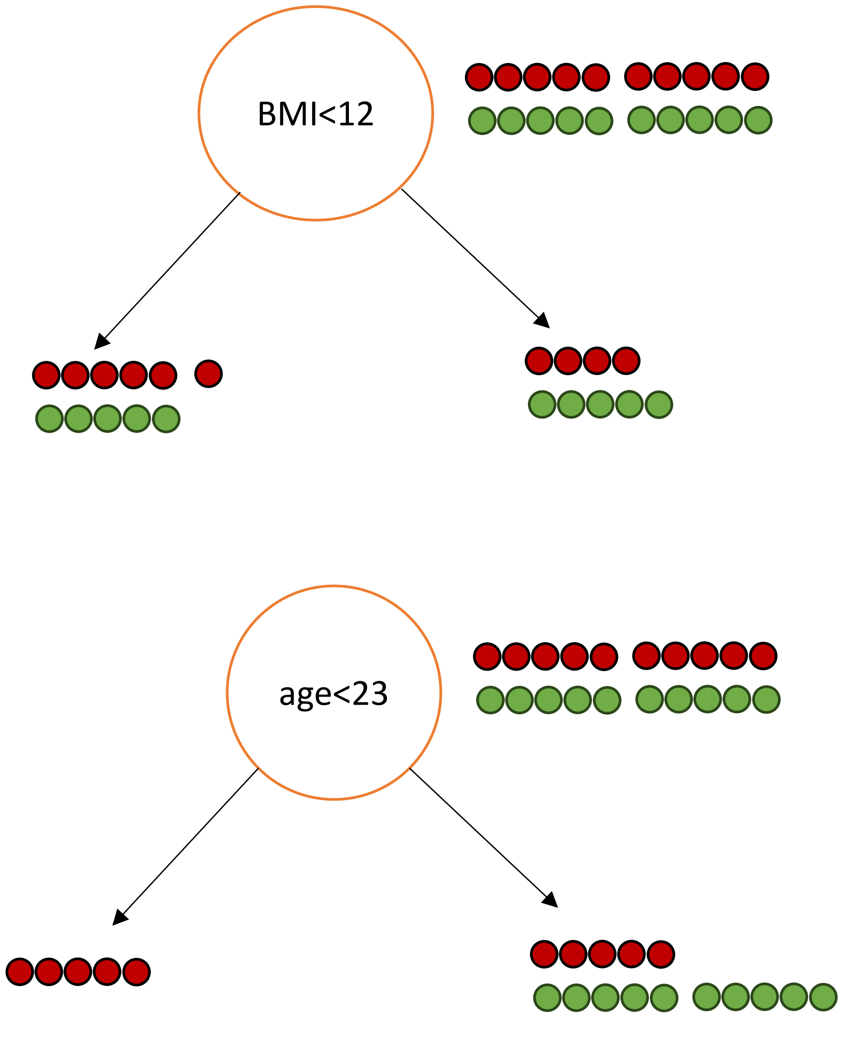



**Which split is better?**

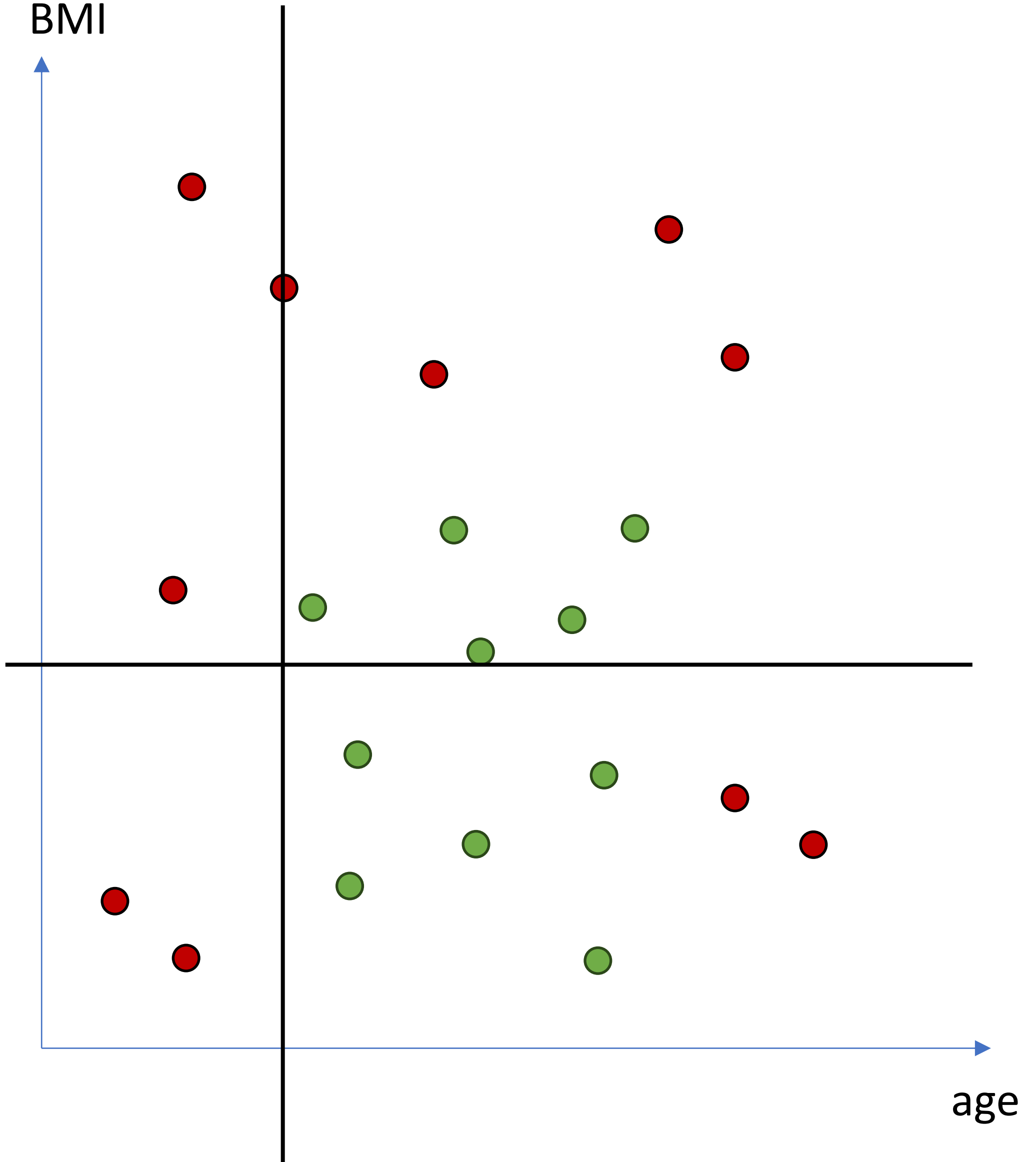
# Training a decision tree

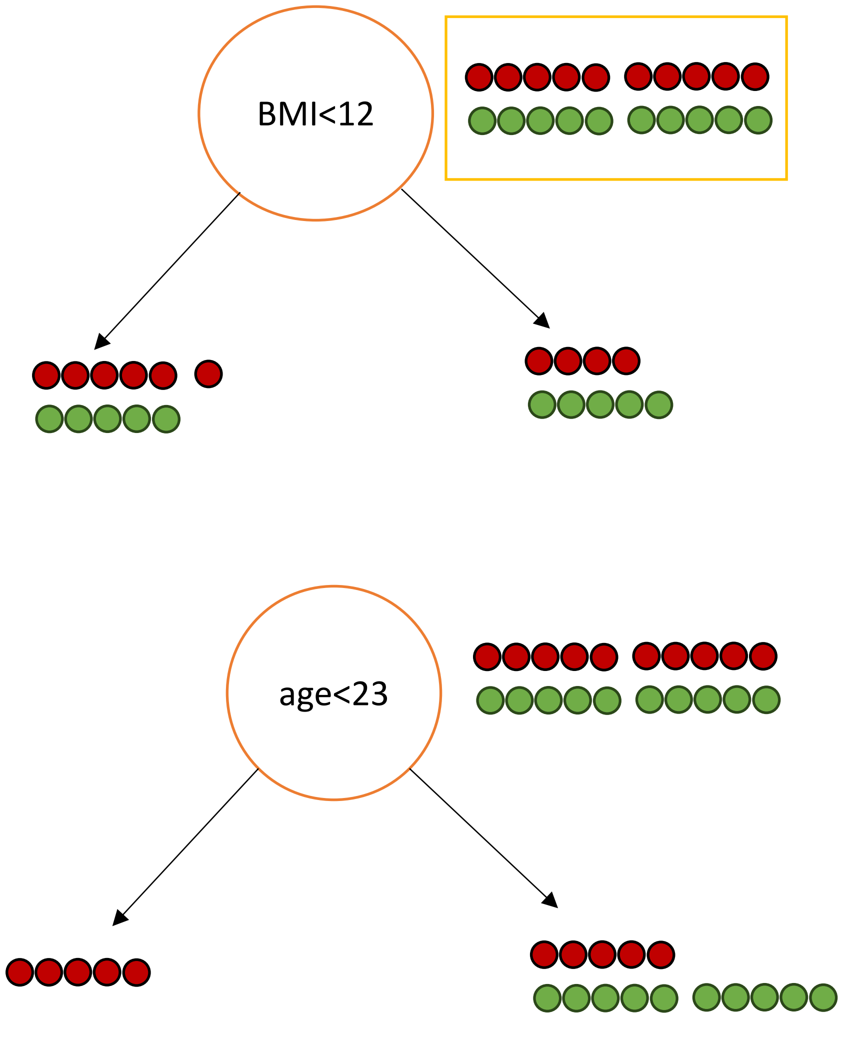



$$Entropy = \Sigma \ -p_i \, log(p_i)$$

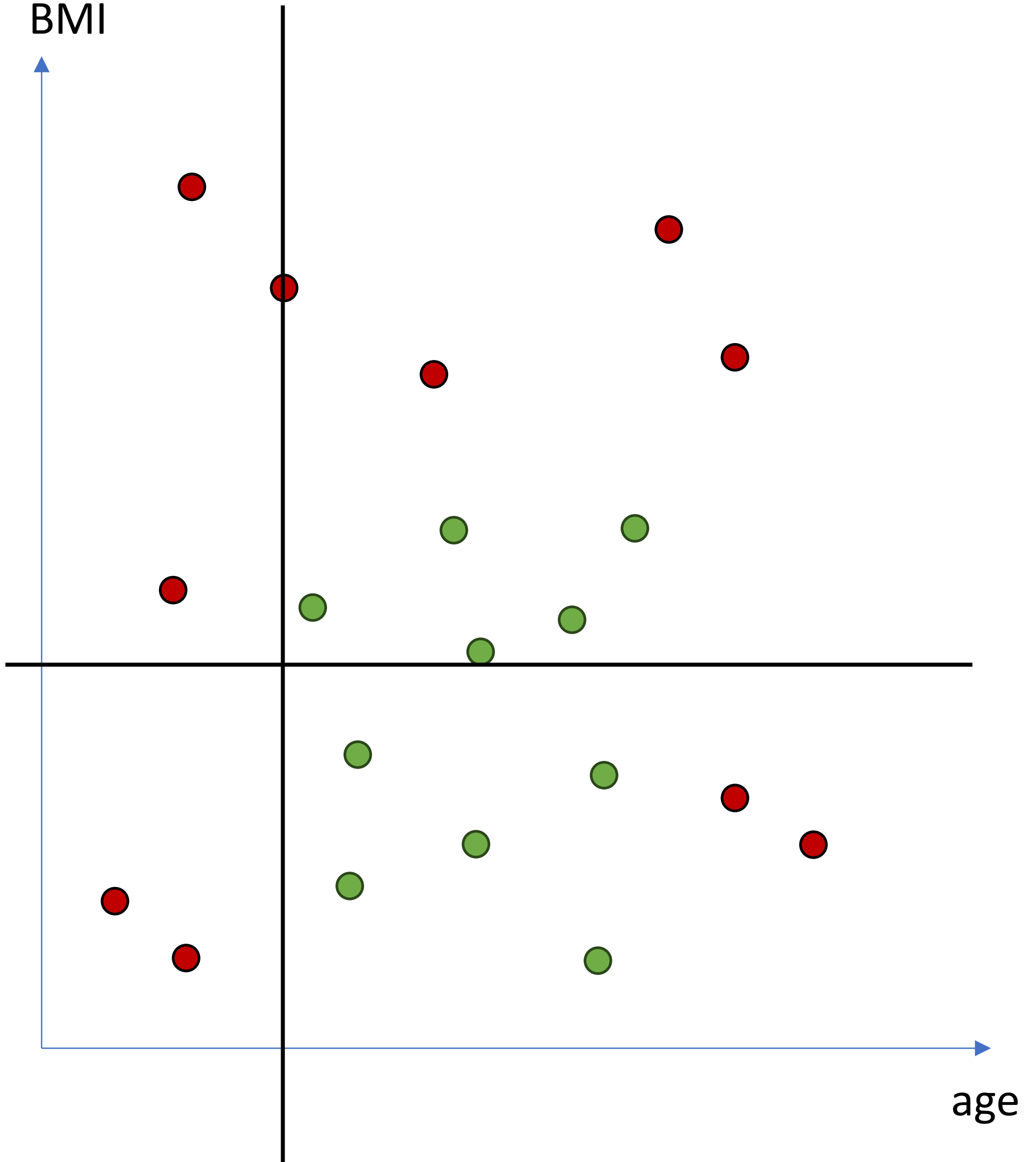
# Training a decision tree

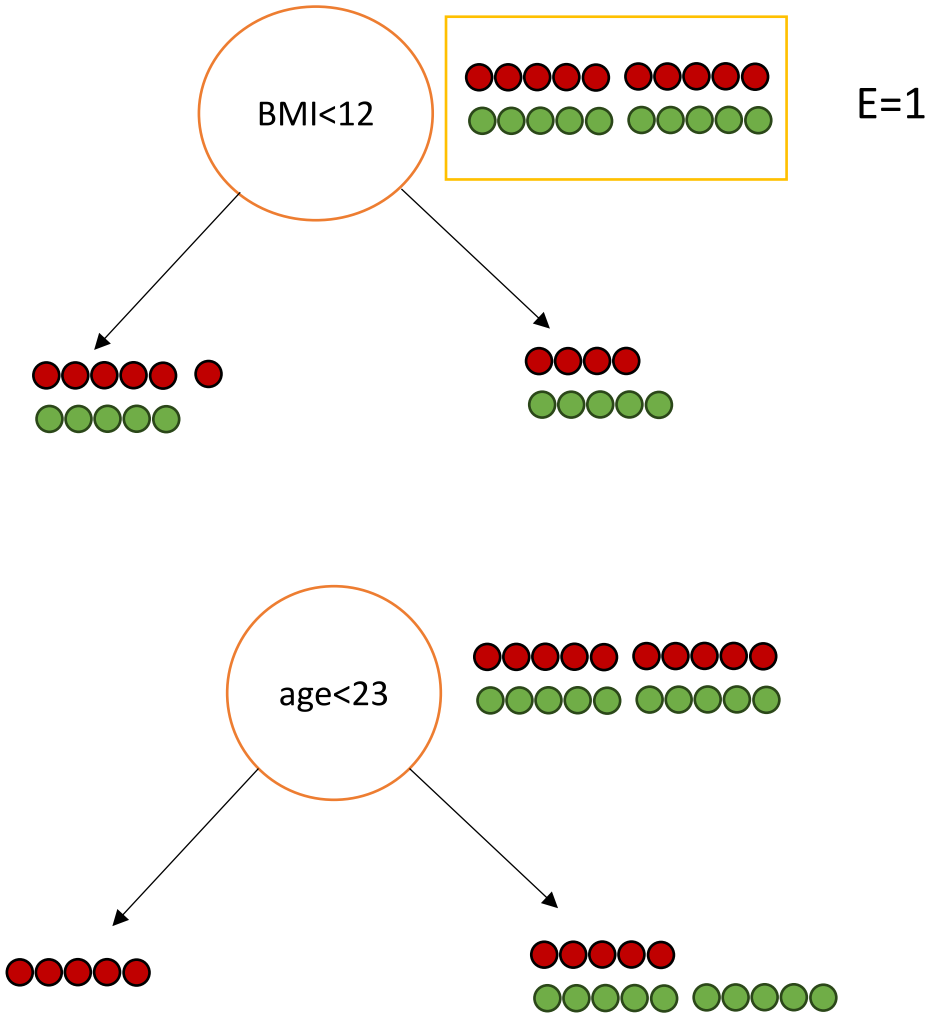



$$Entropy = -\,0.5\,log(0.5)\ -0.5\,log(0.5)$$

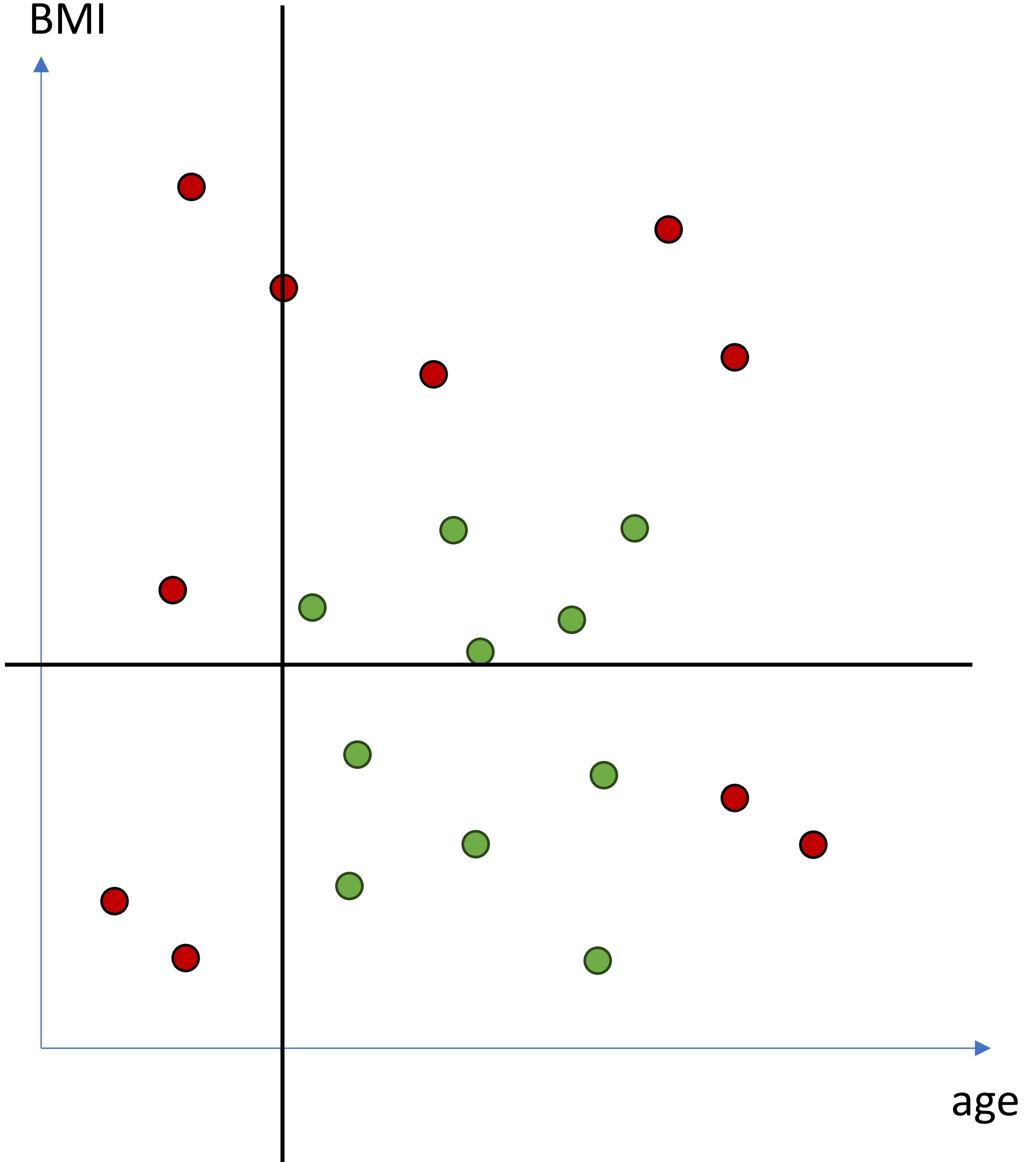
# Training a decision tree

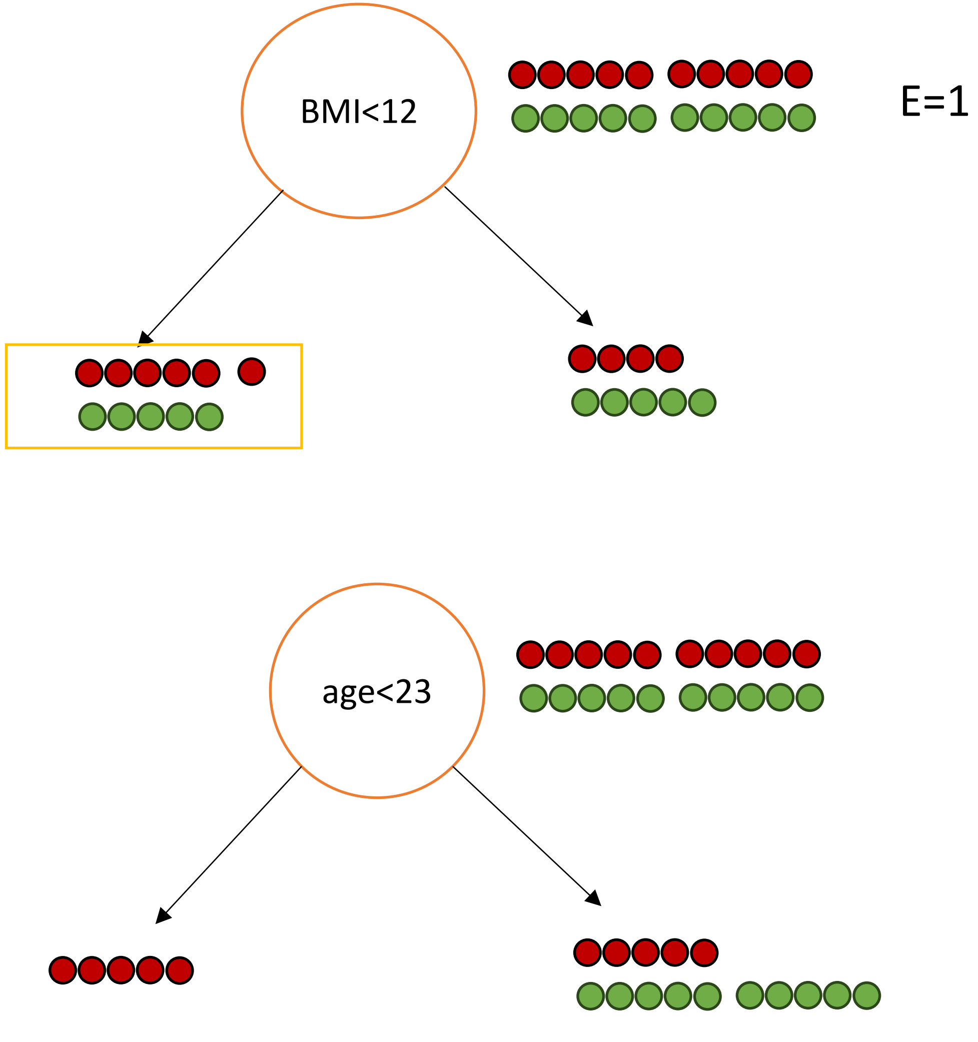



$$Entropy = -\,0.55\,log(0.55)\; - 0.45\,log(0.45)$$

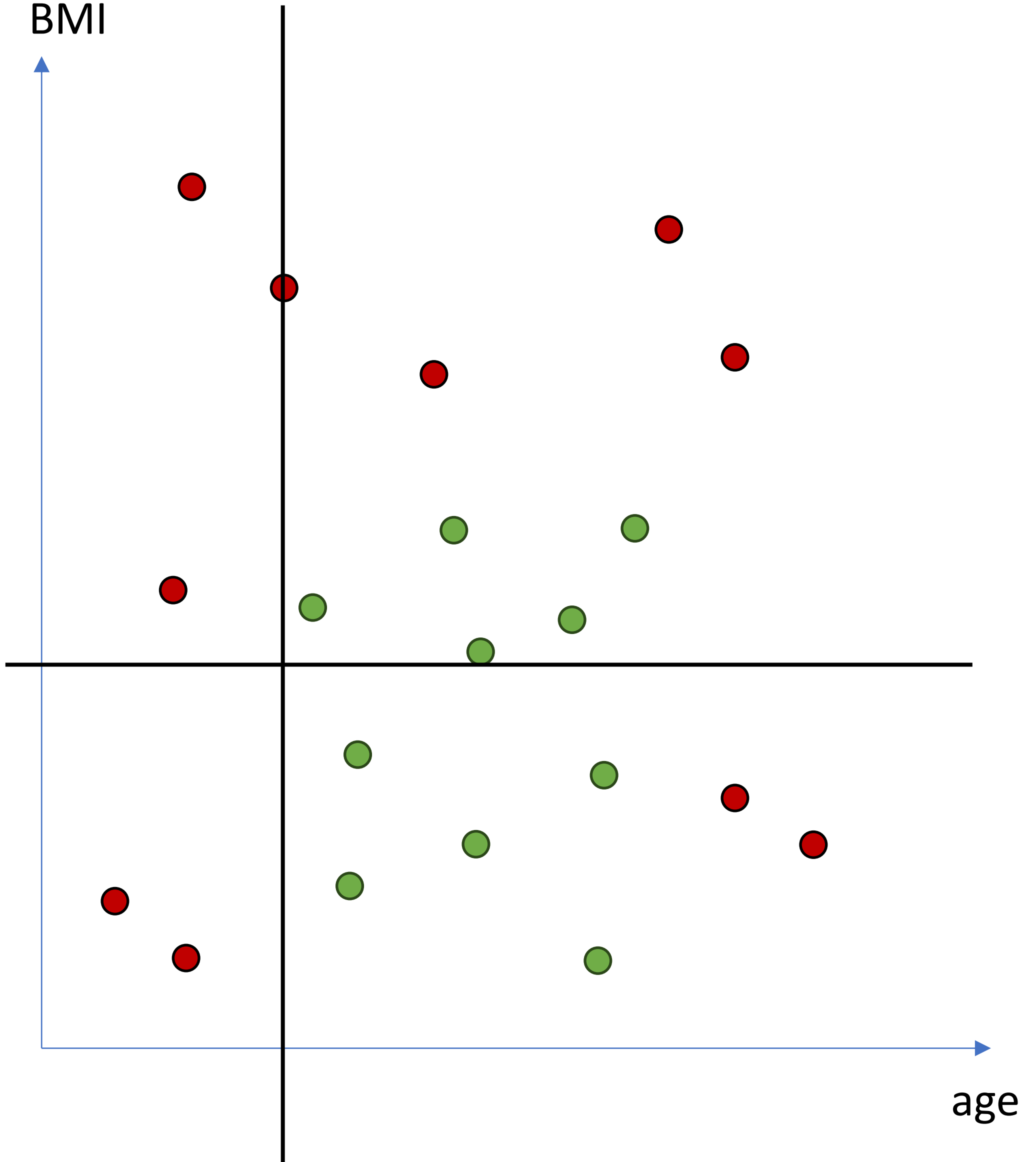
# Training a decision tree

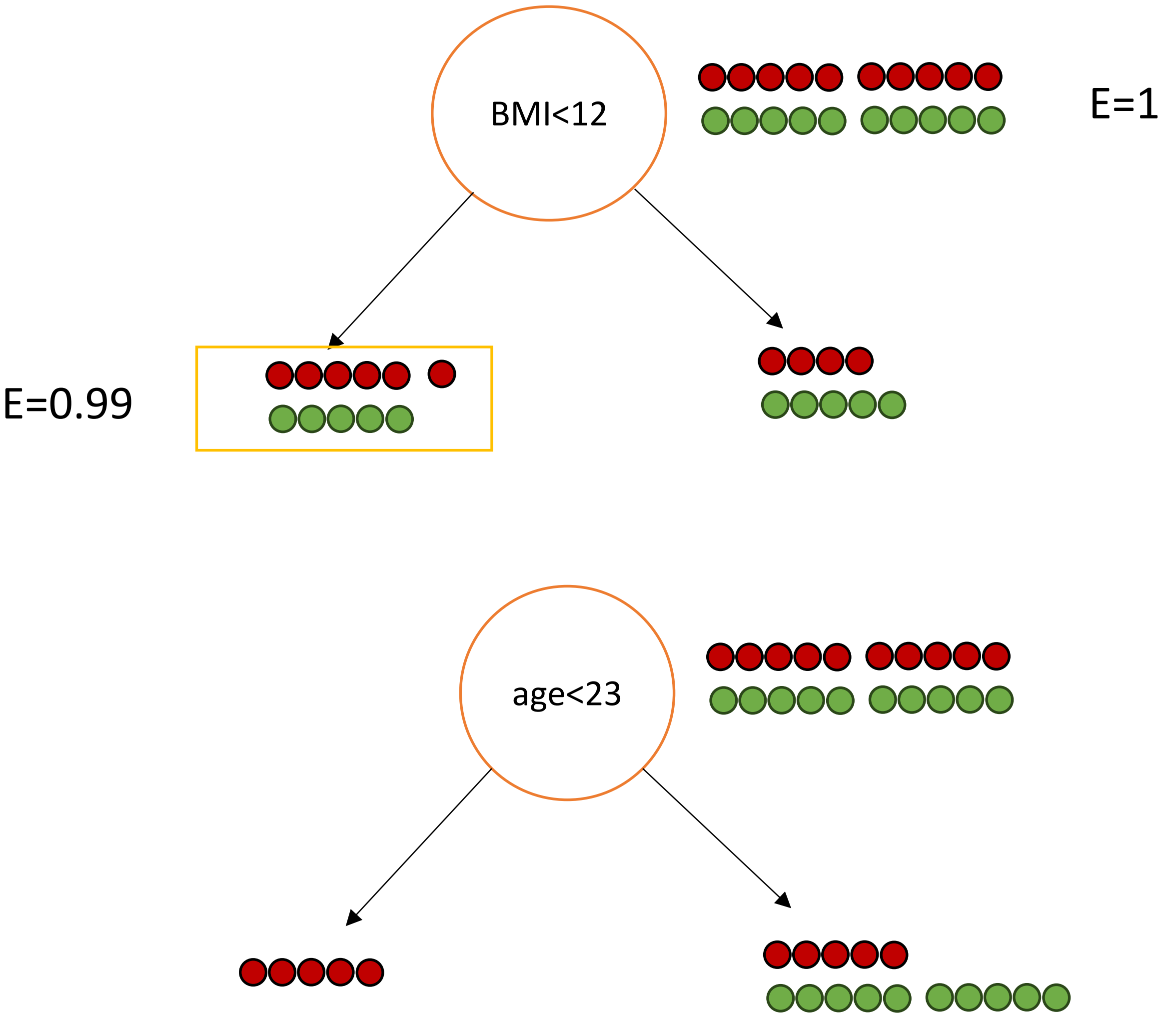



$$\boldsymbol{Entropy = -0.55\,log(0.55)\ -0.45\,log(0.45)}$$

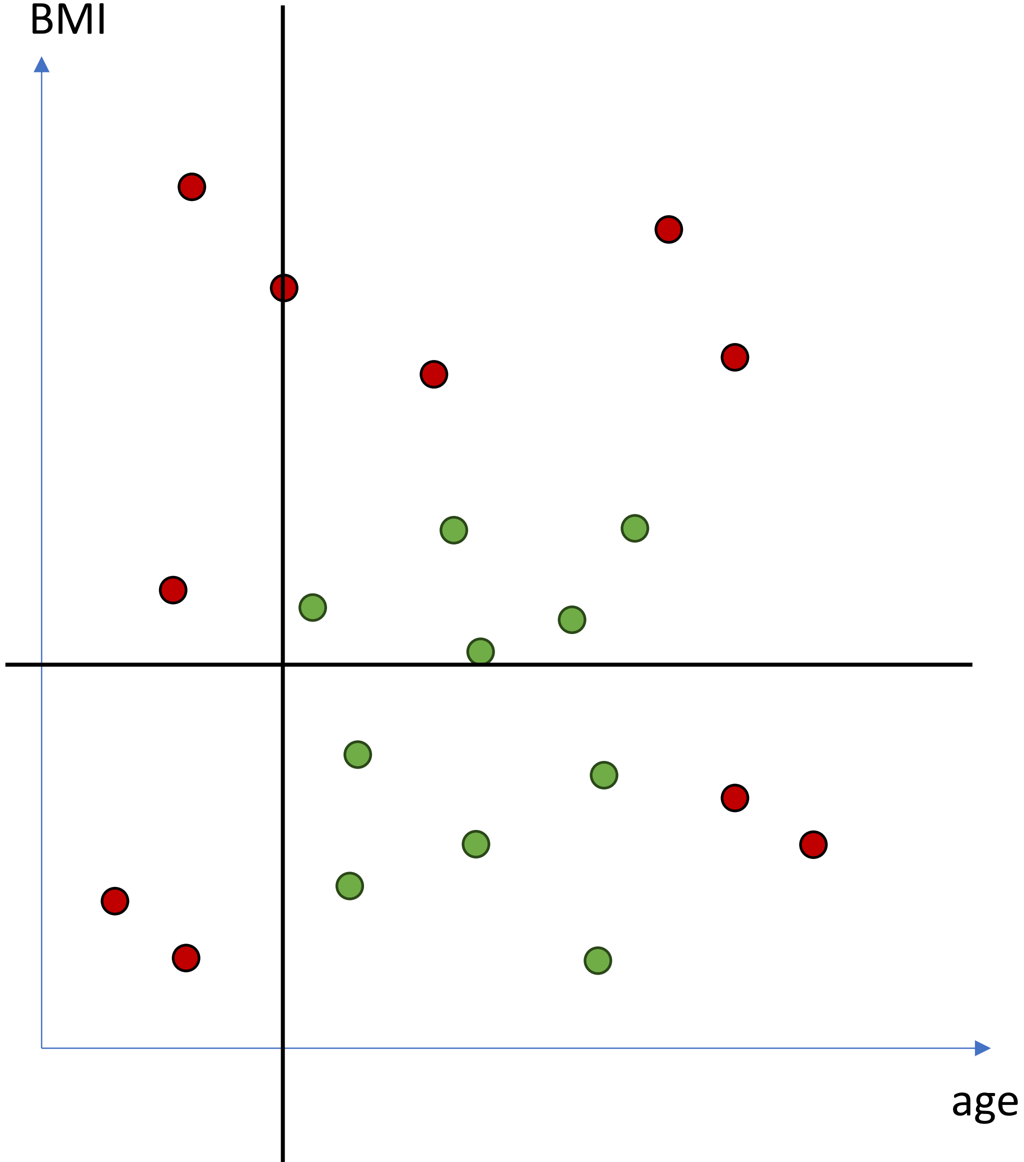
# Training a decision tree



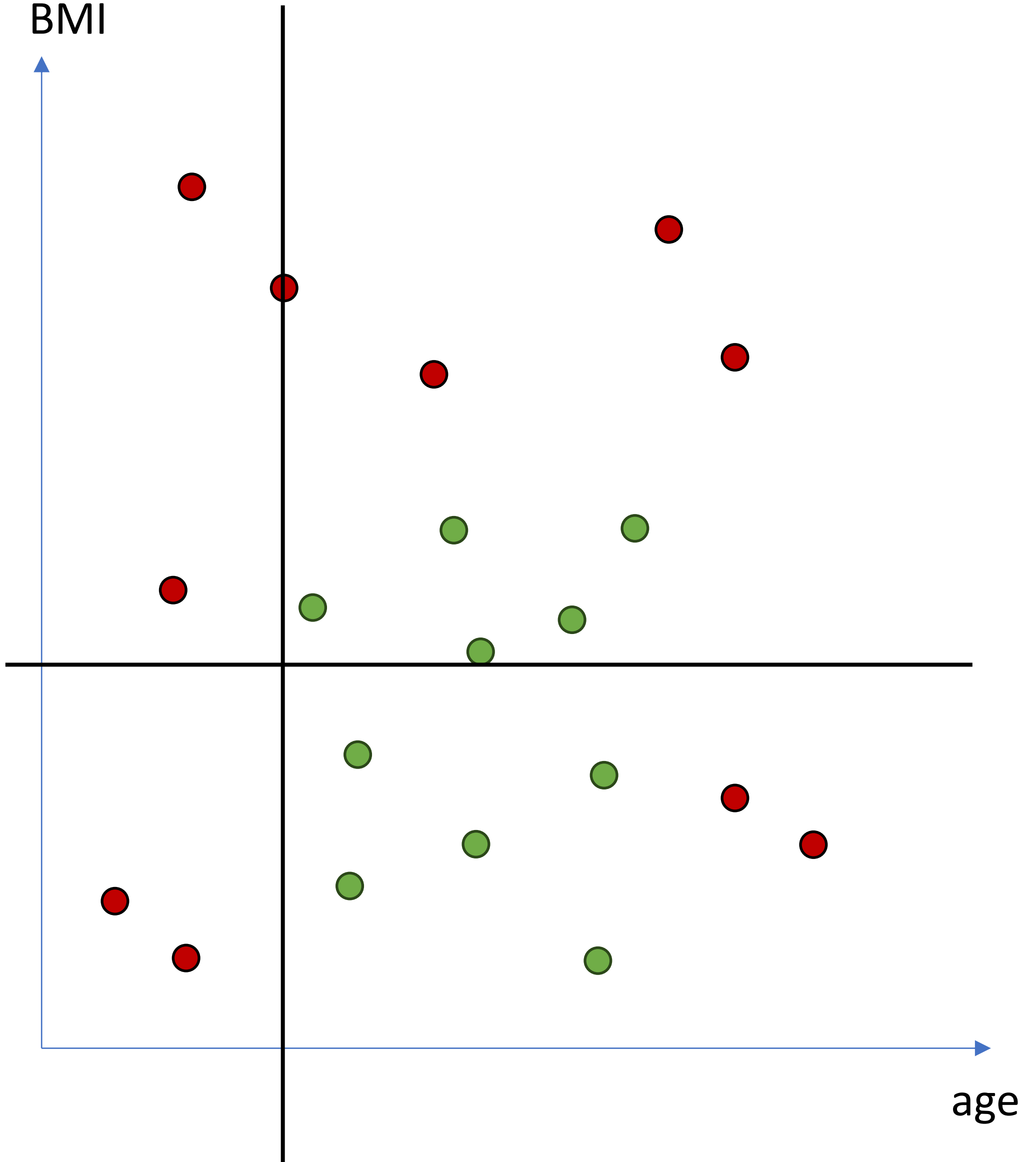



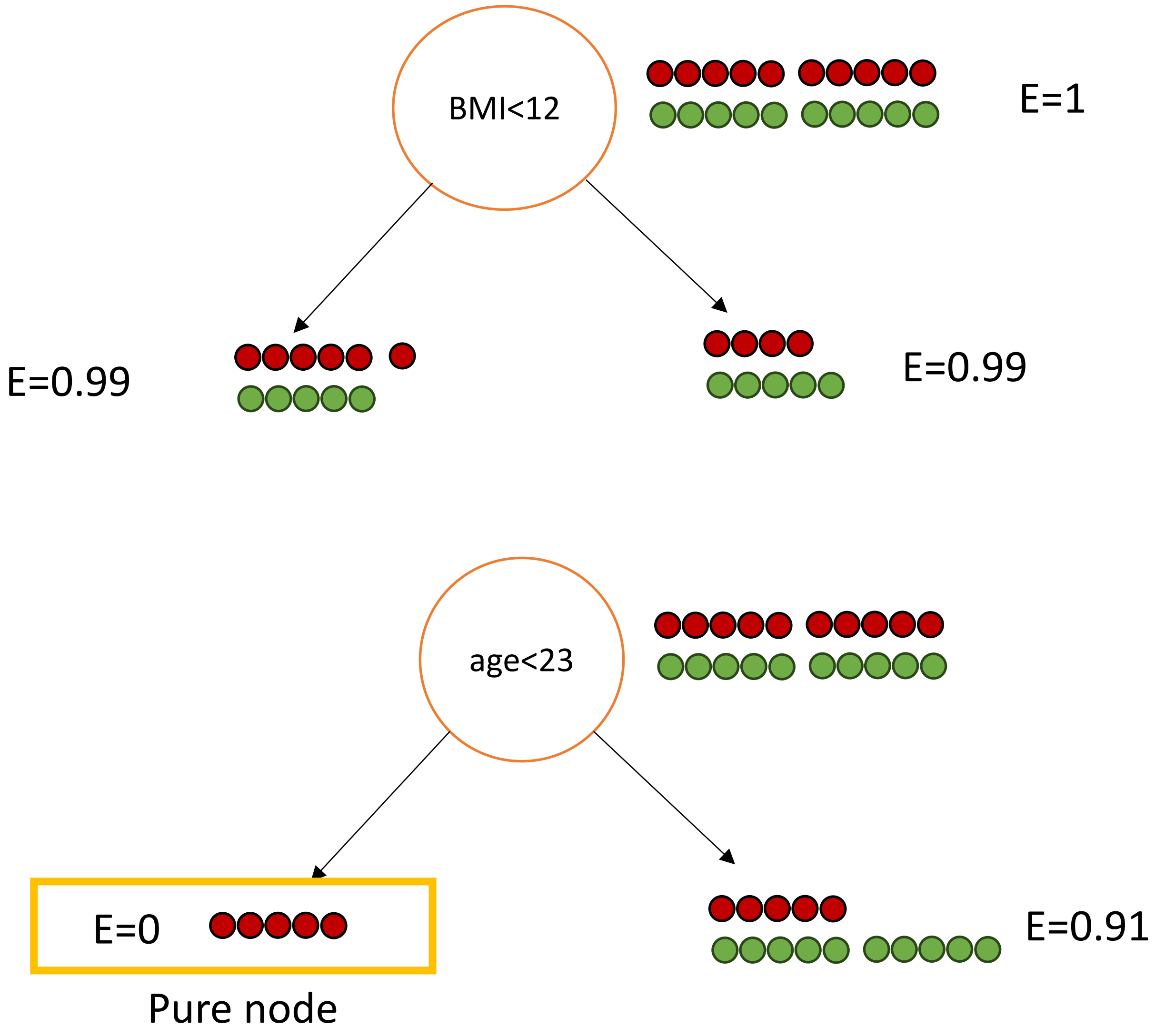

$$\boldsymbol{Entropy = -\,0.55\,log(0.55)\;-\,0.45\,log(0.45)}$$

# Training a decision tree

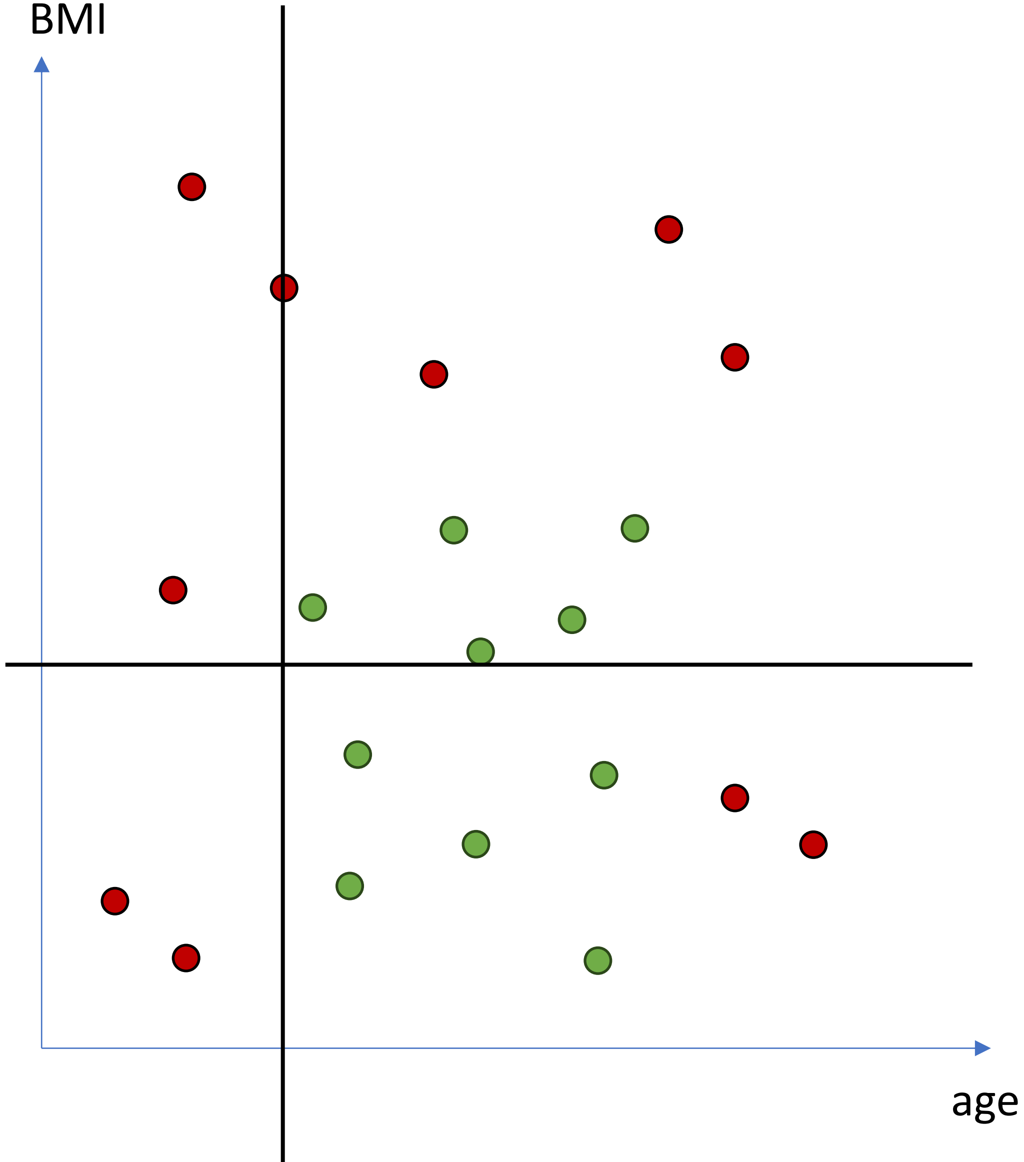



# Training a decision tree



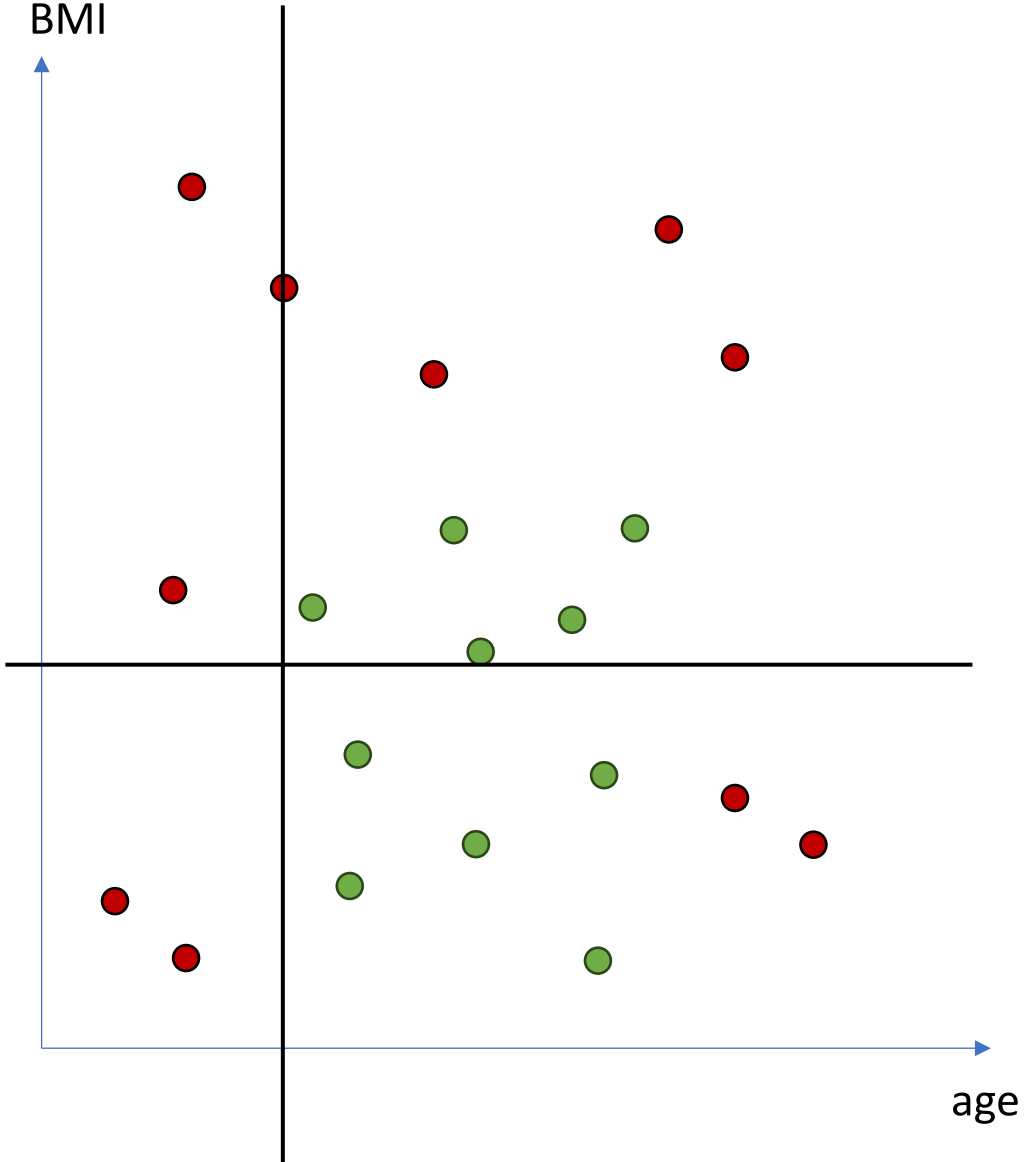
# Training a decision tree

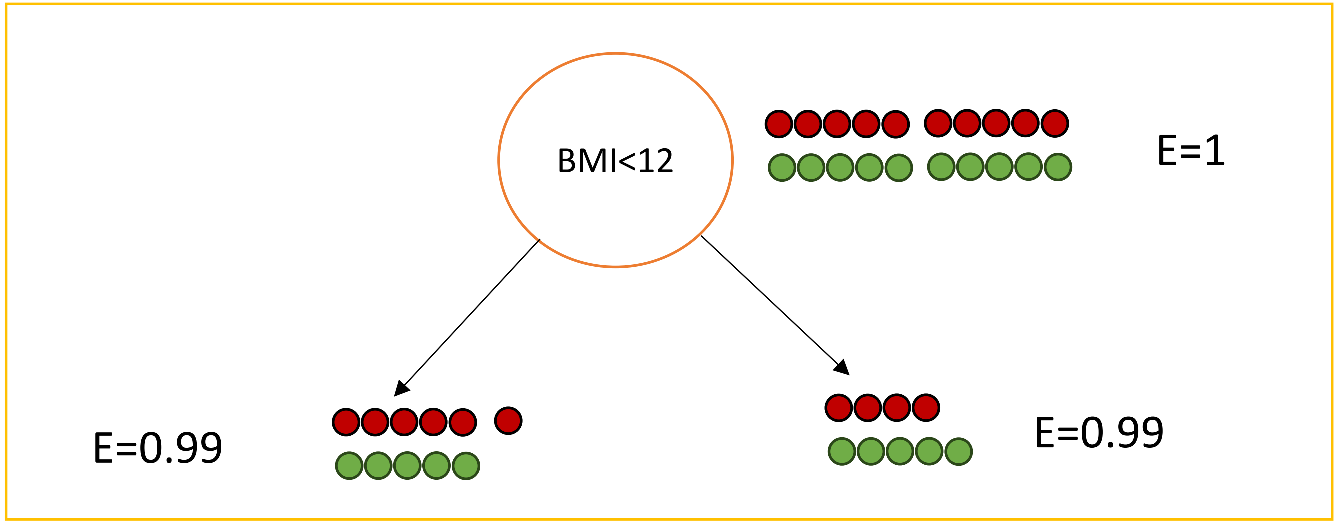


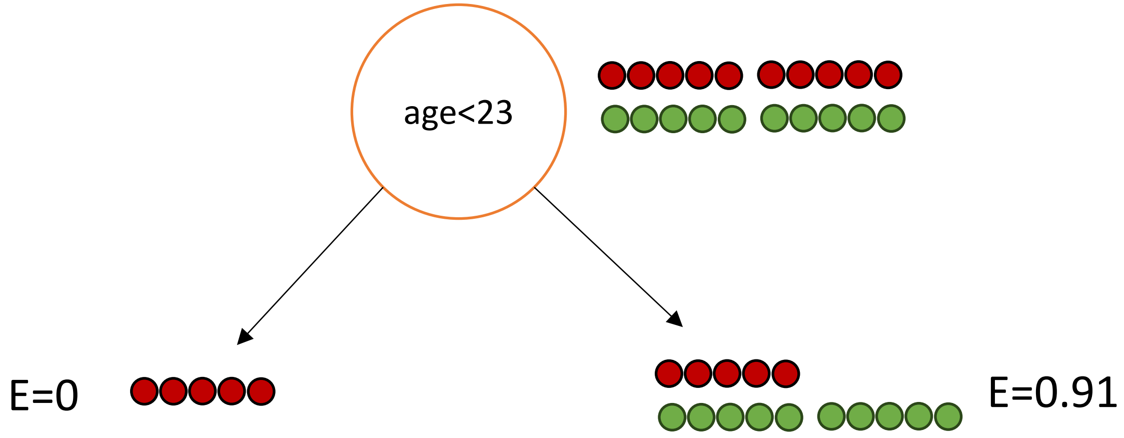


$$IG = E(parent) - \Sigma\, w_i\, E(child_i)$$

# Training a decision tree



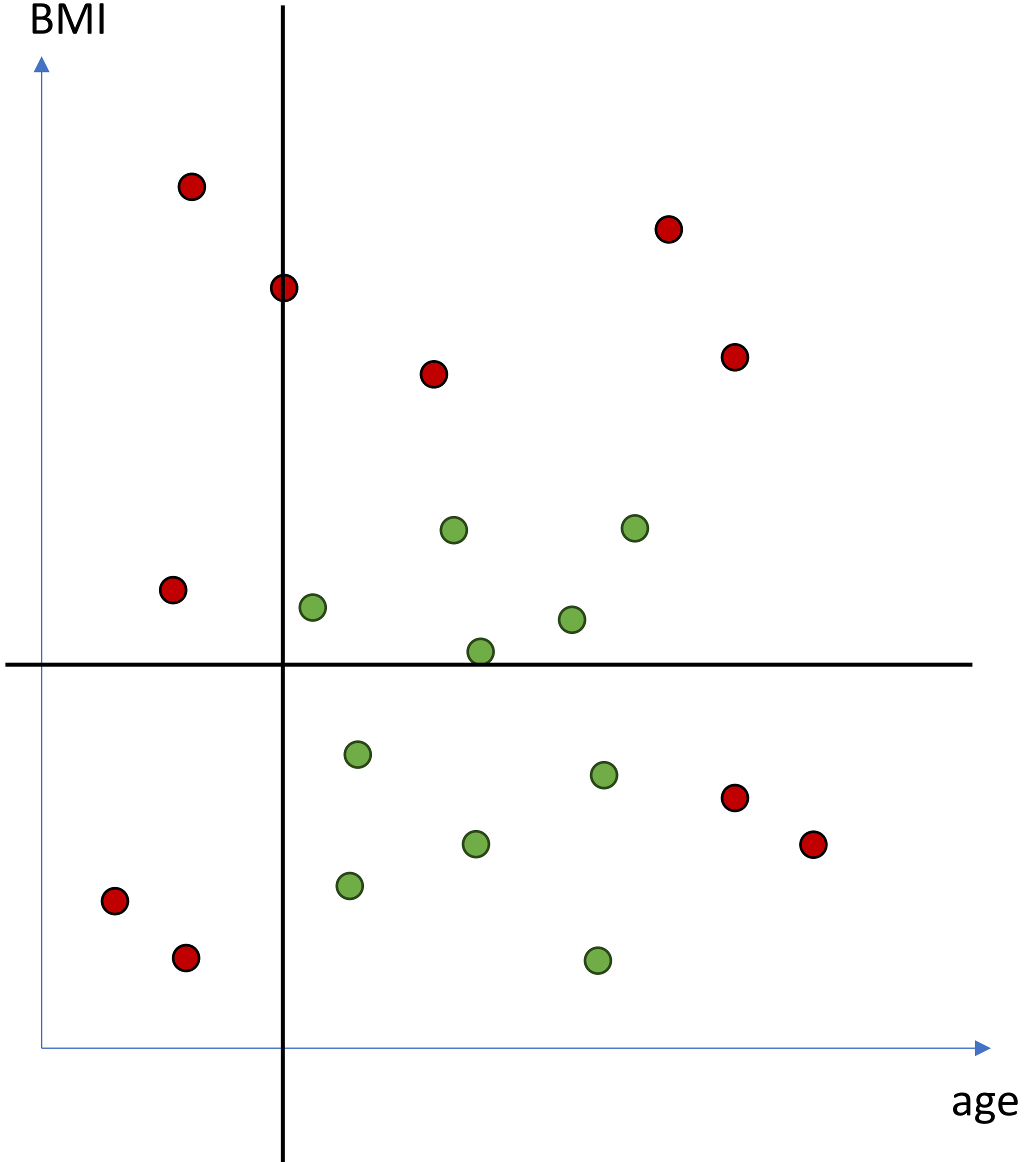



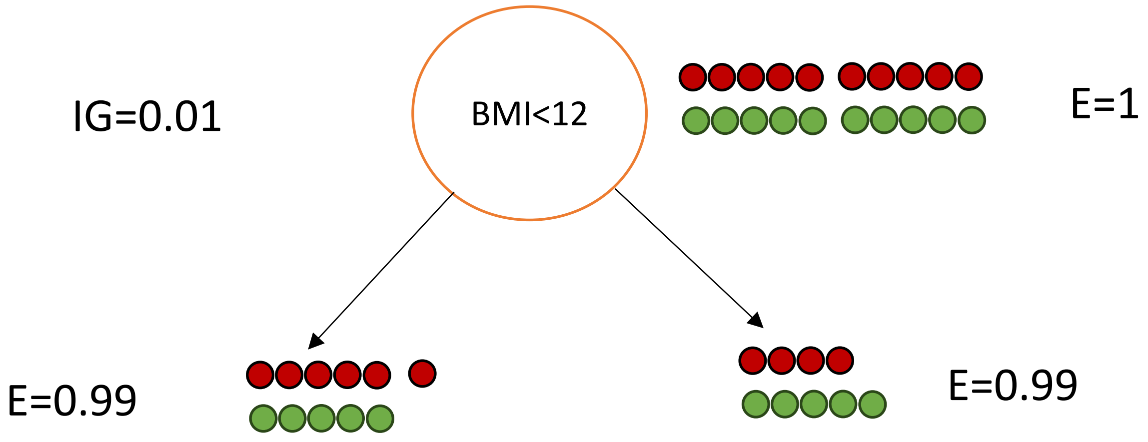



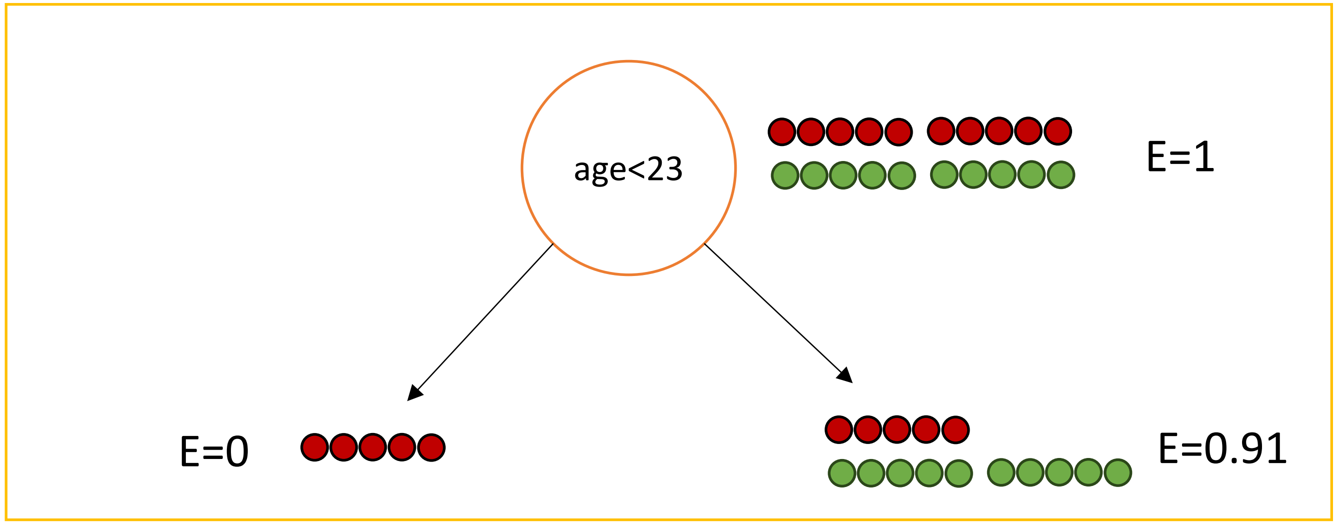

$$IG = E(parent) - \Sigma\, w_i\, E(child_i)$$

$$IG = 1 - \frac{11}{20} \times .99 - \frac{9}{20} \times 0.99 = 0.01$$

# Training a decision tree



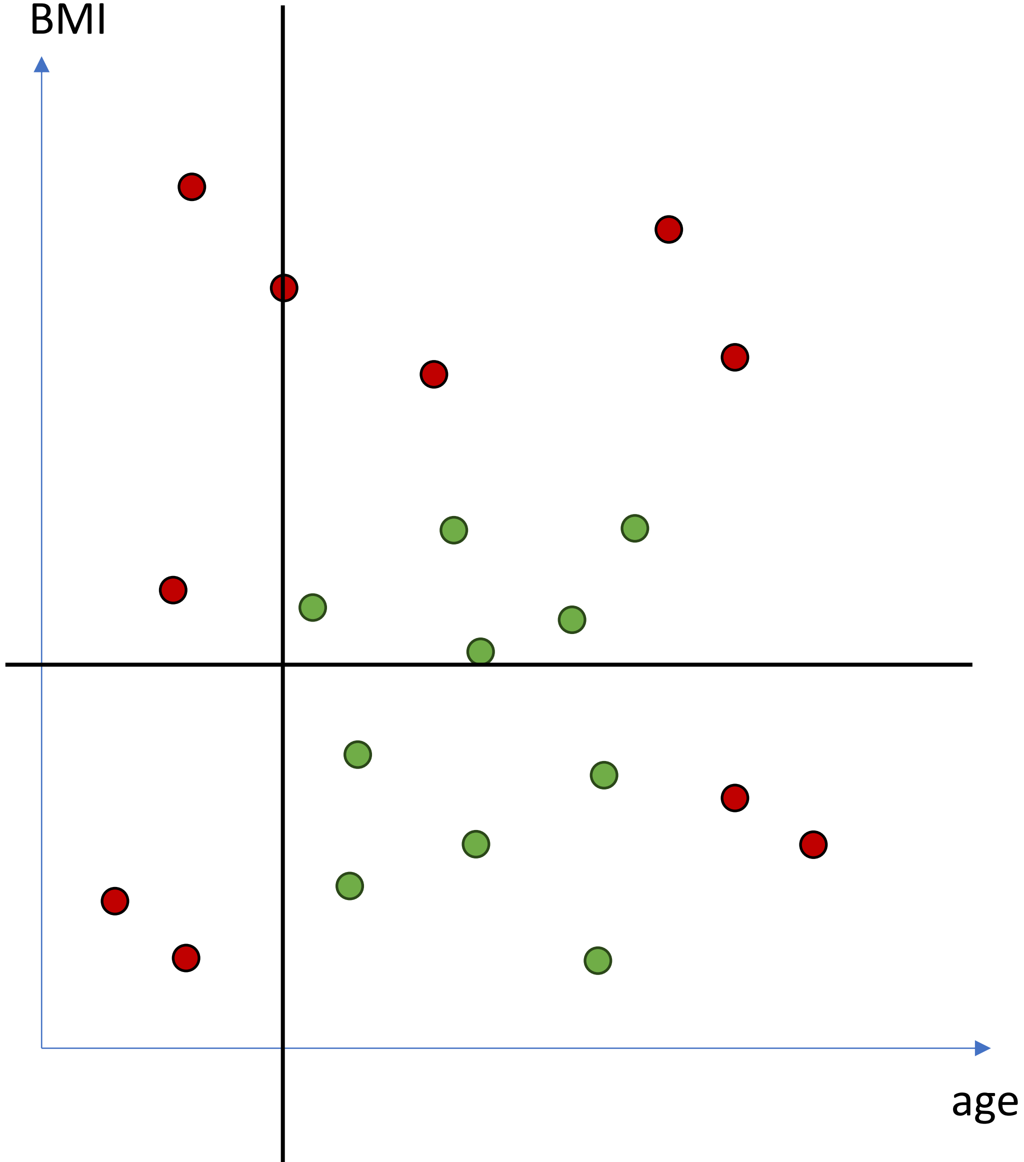



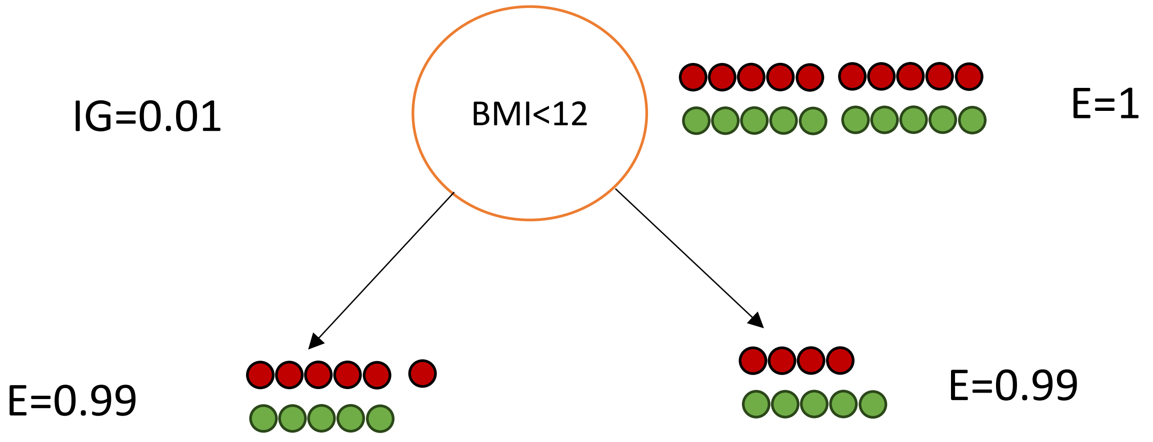



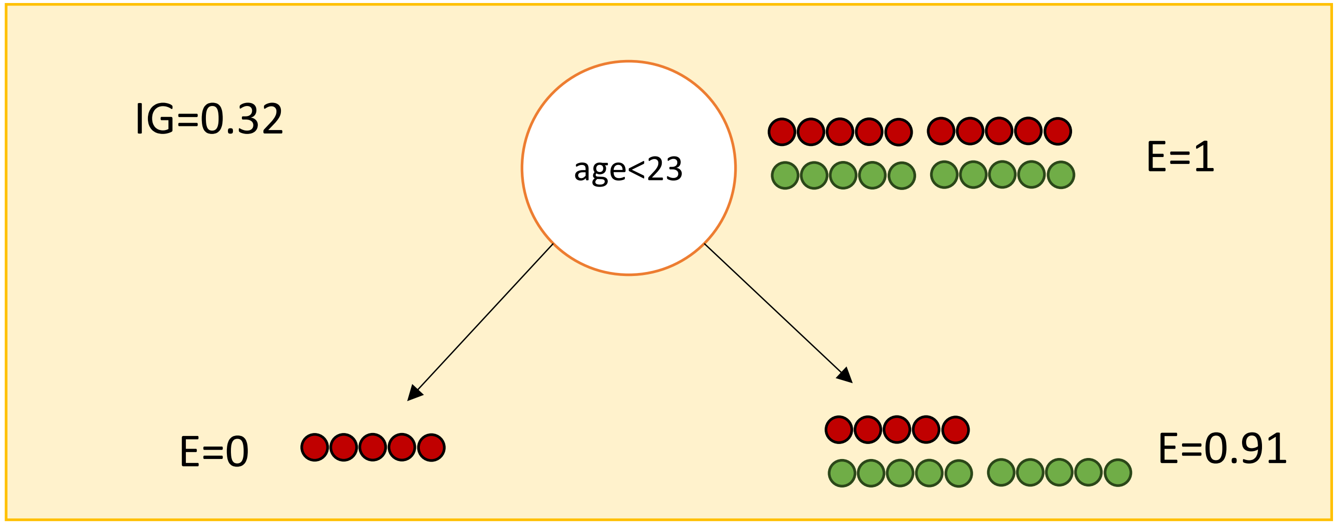

$$IG = E(parent) - \Sigma\, w_i\, E(child_i)$$

$$IG = 1 - \frac{5}{20} \times 0 - \frac{15}{20} \times 0.99 = 0.32$$

# Training a decision tree

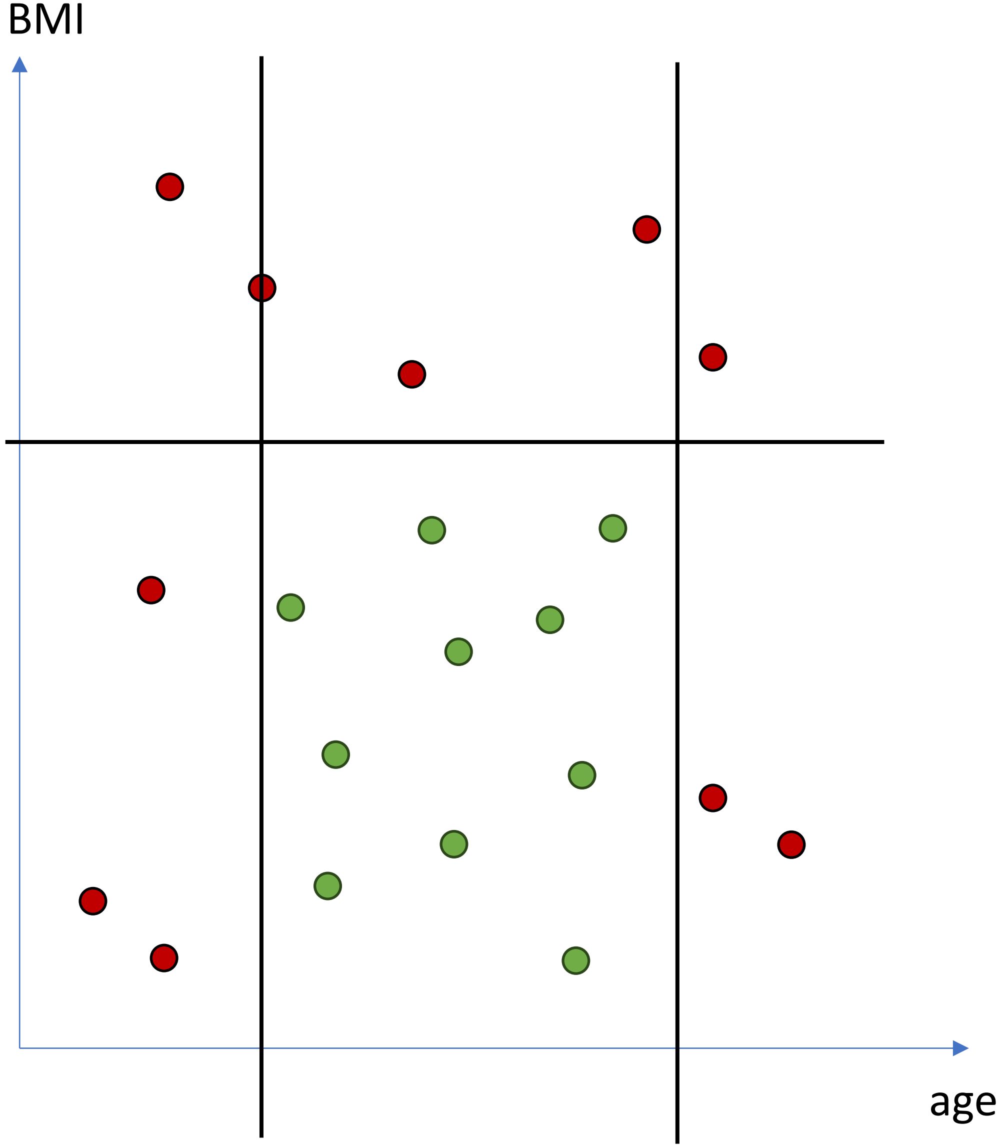


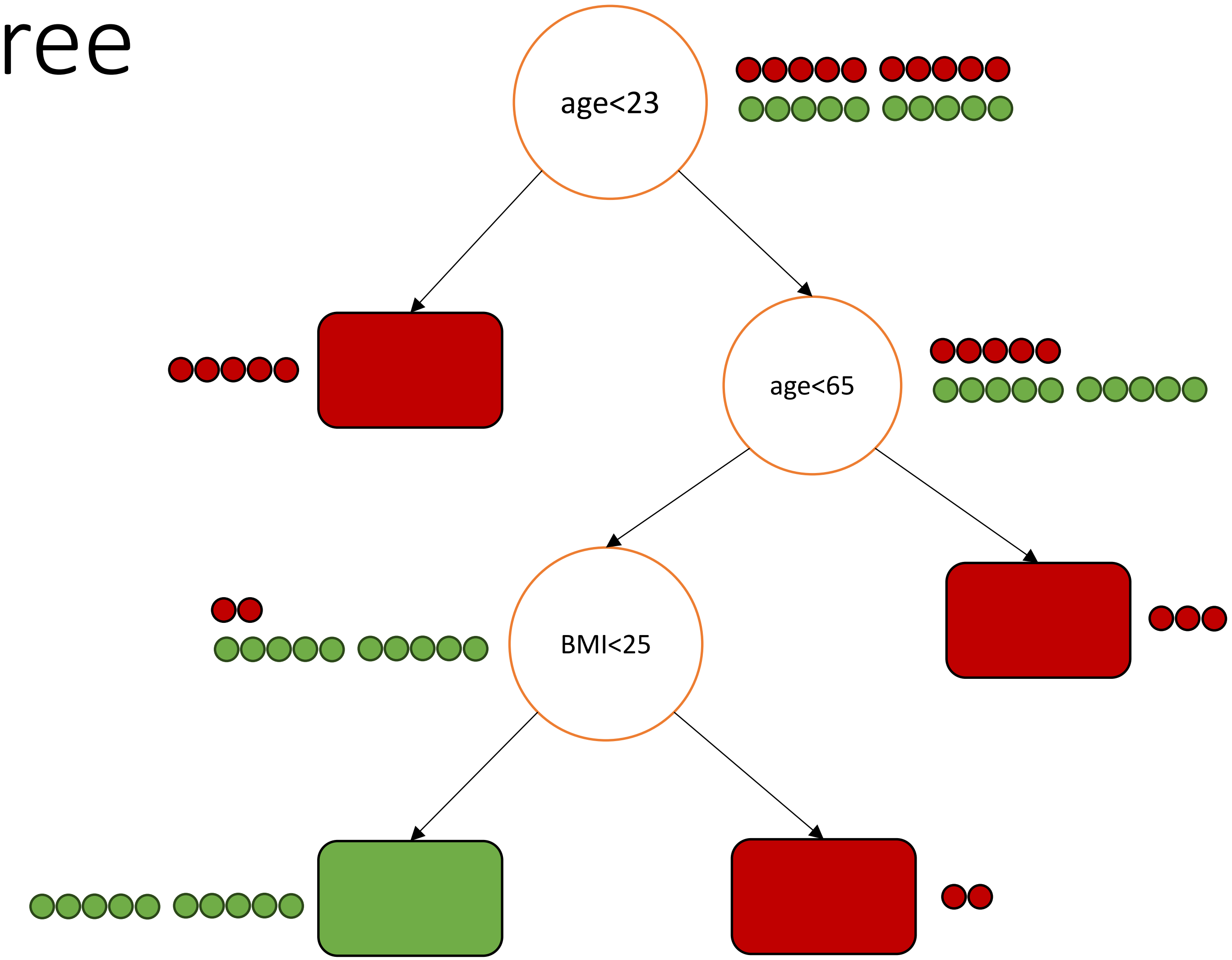



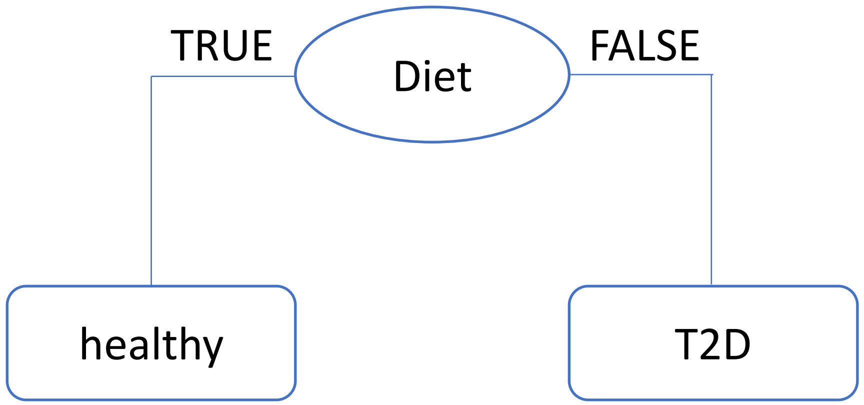

The training algorithm choses the split that maximizes the information gain

# Classification decision tree



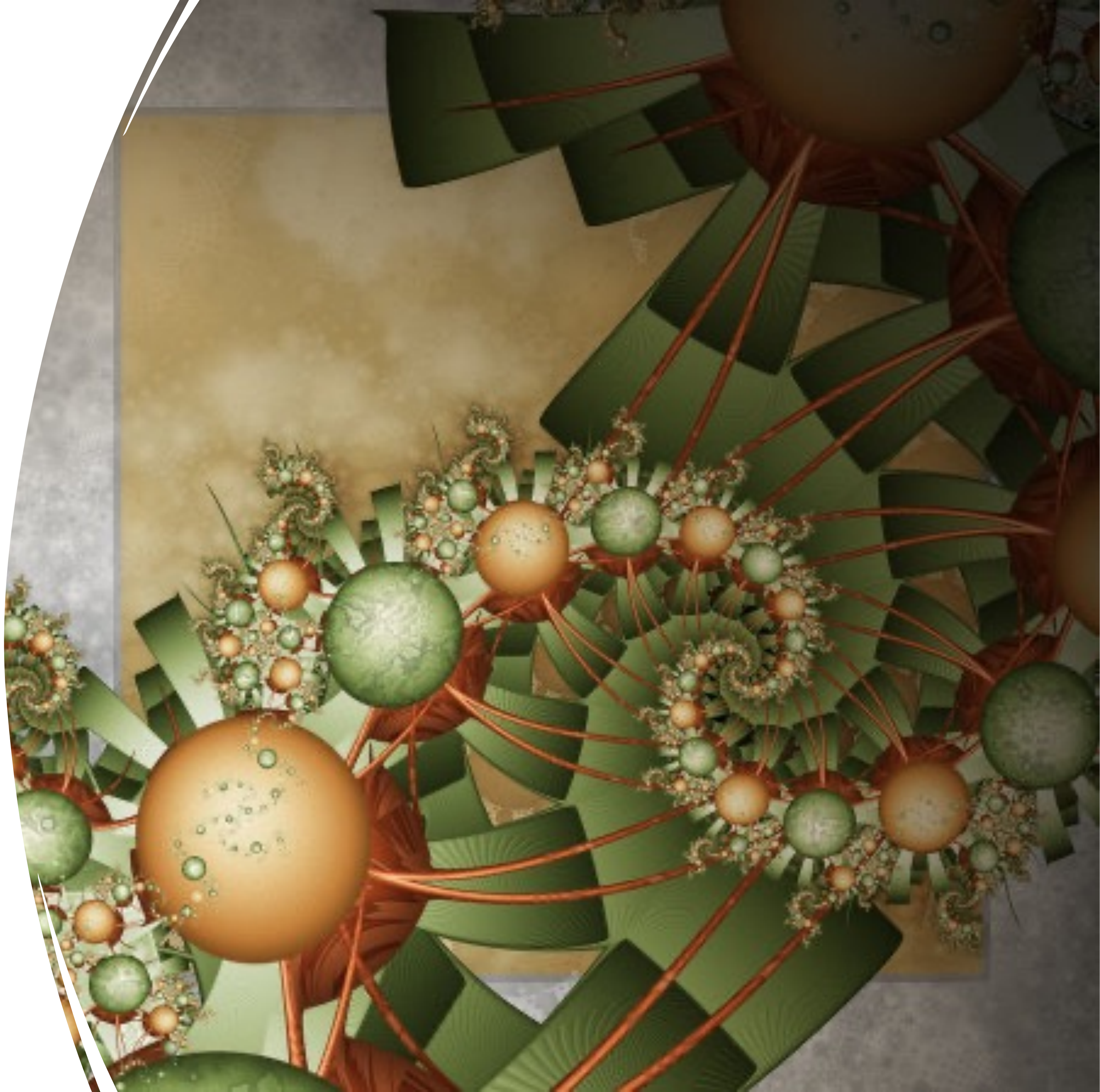



# Let’s recap!

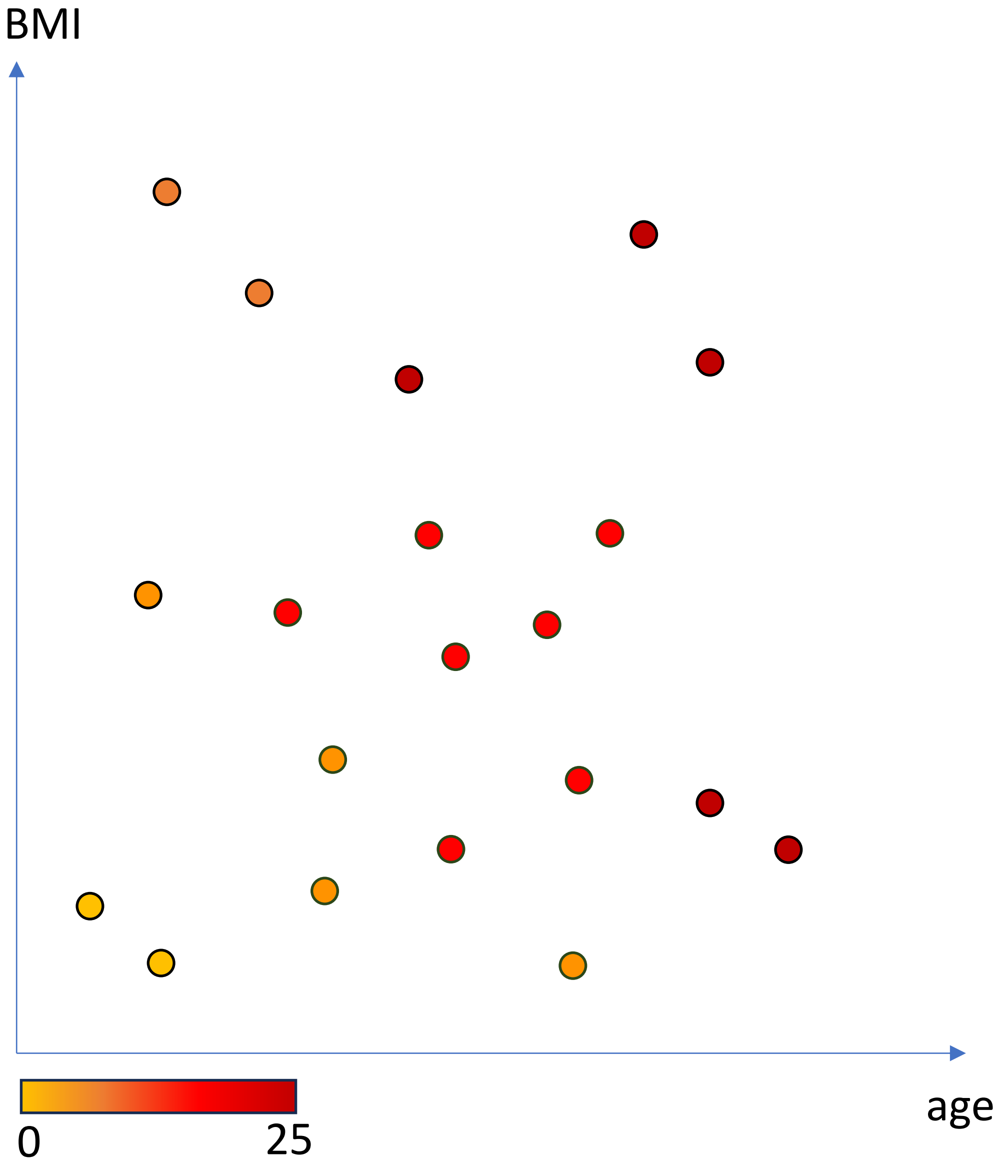



- Decision trees can be used to perform a classification task
- Training of a classification decision tree consists in finding the best data split that maximizes the information gain
- This training approach is a greedy algorithm, and does not guaranty you will train the best tree

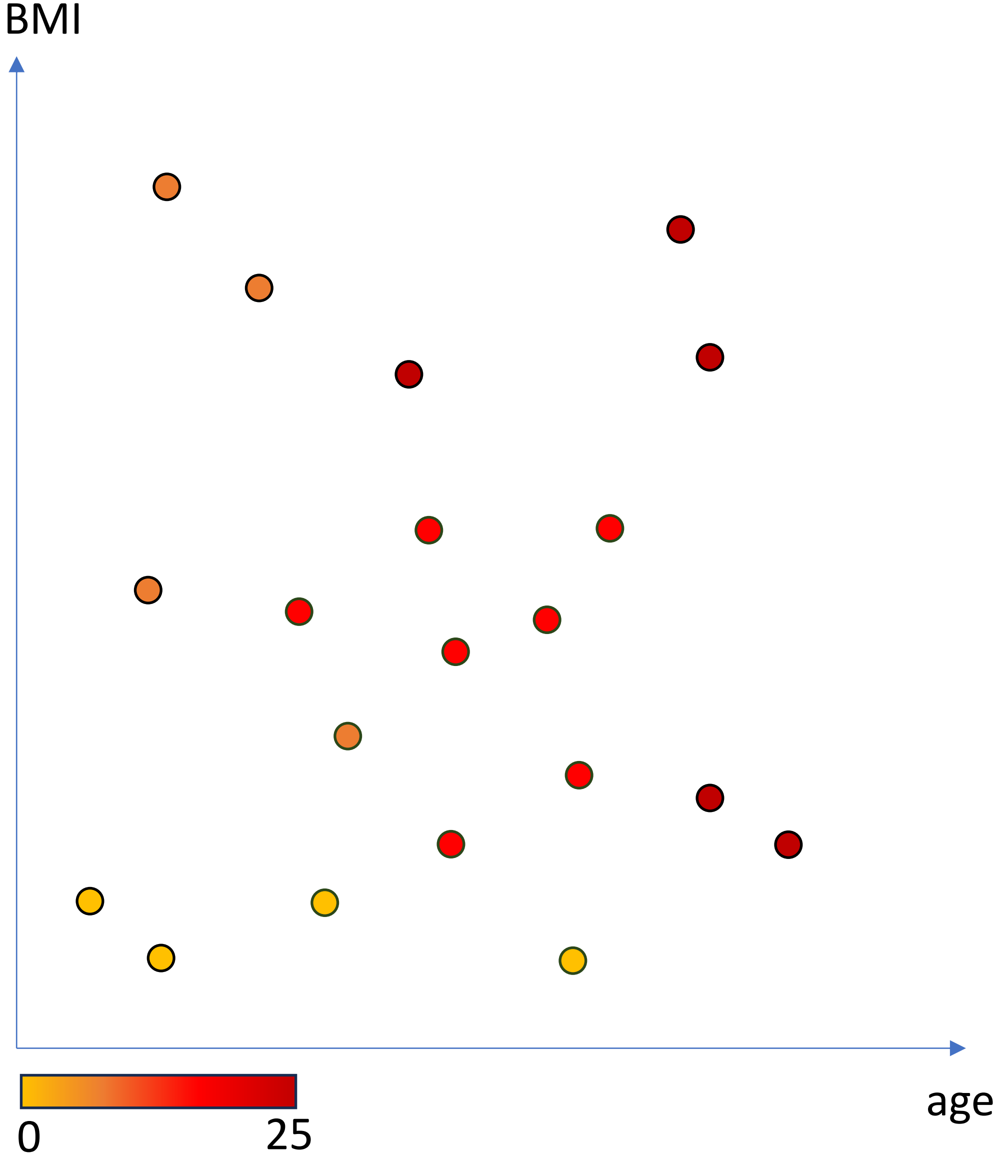
# Regression trees

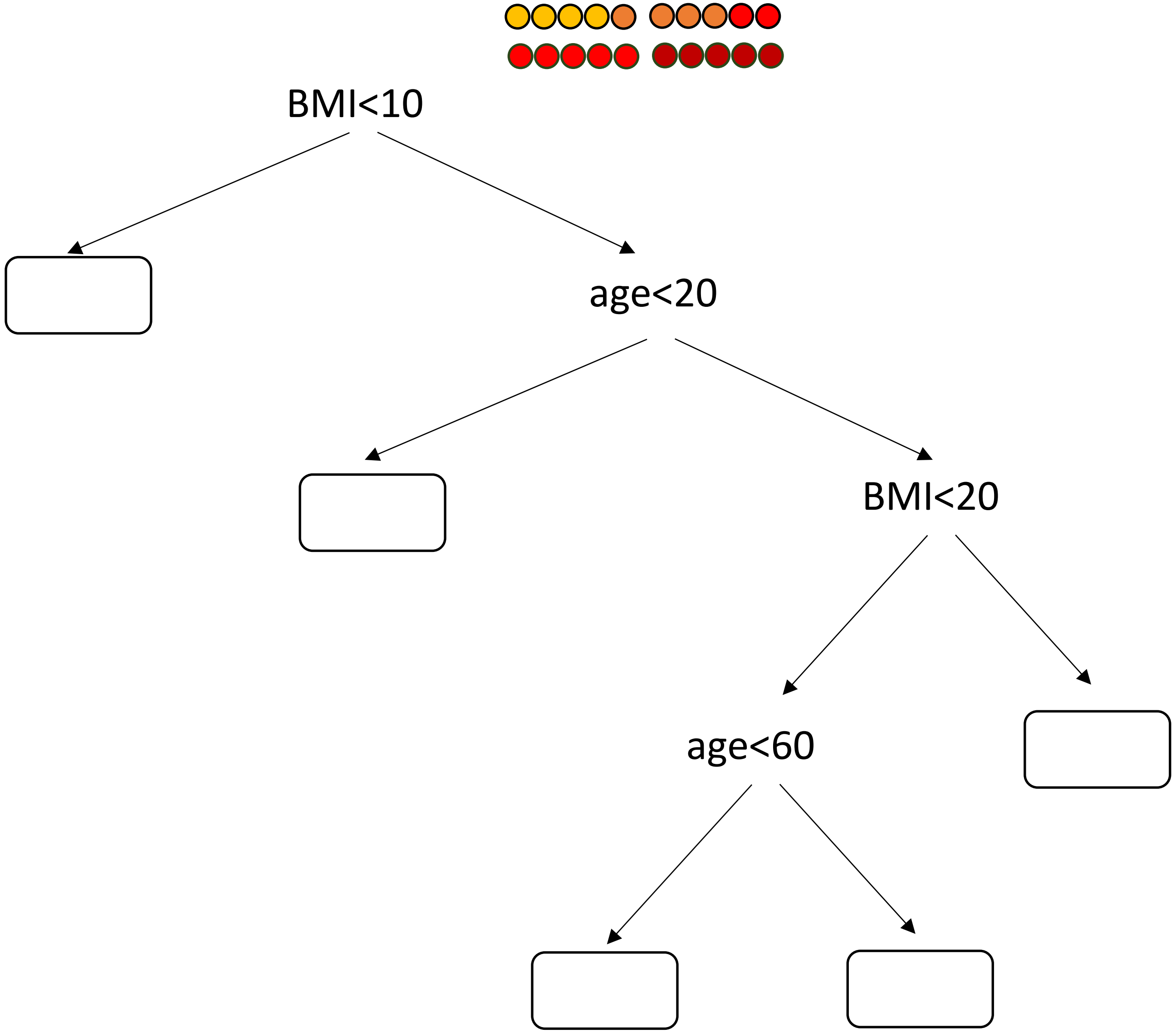


# Regression decision tree

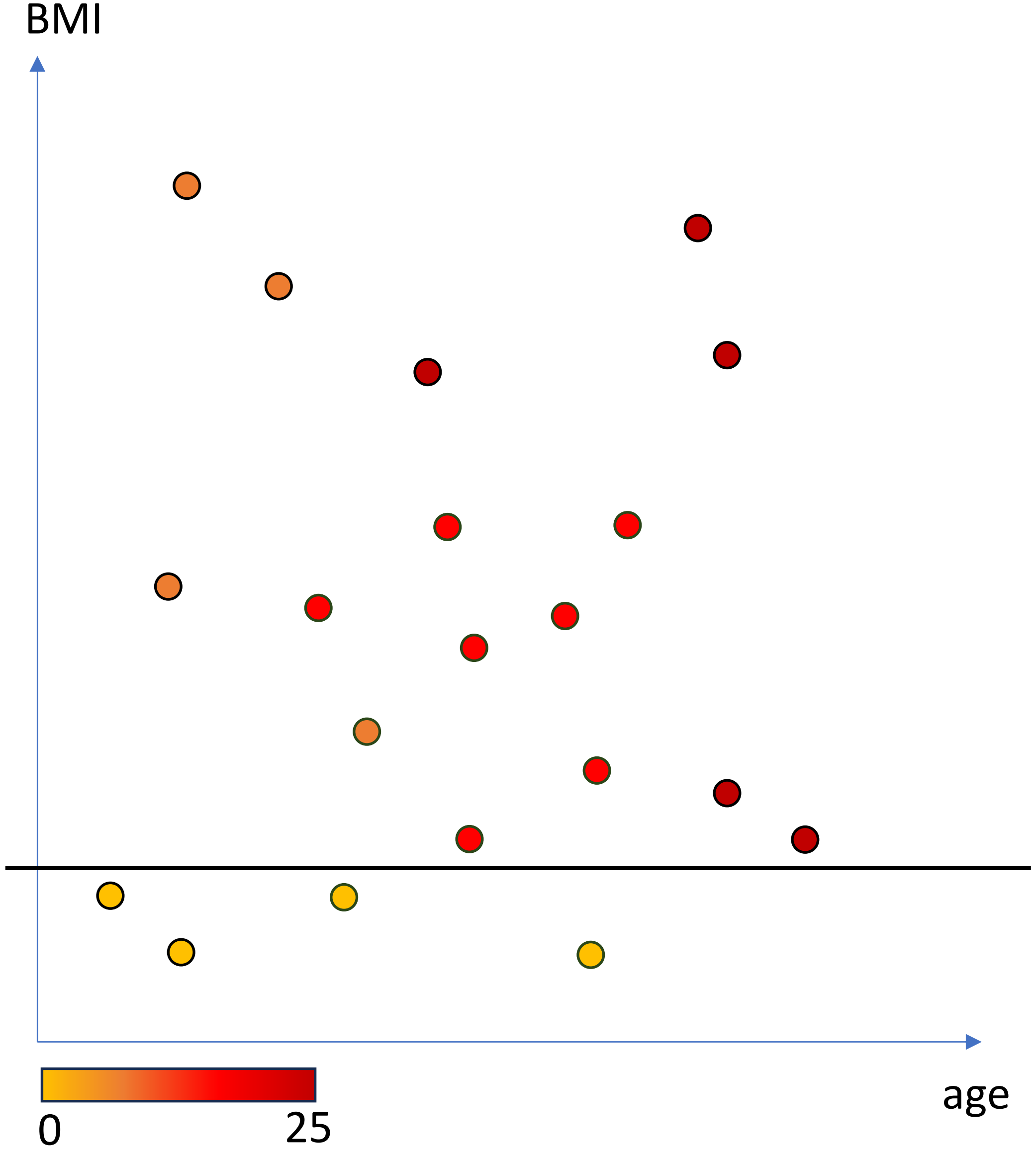


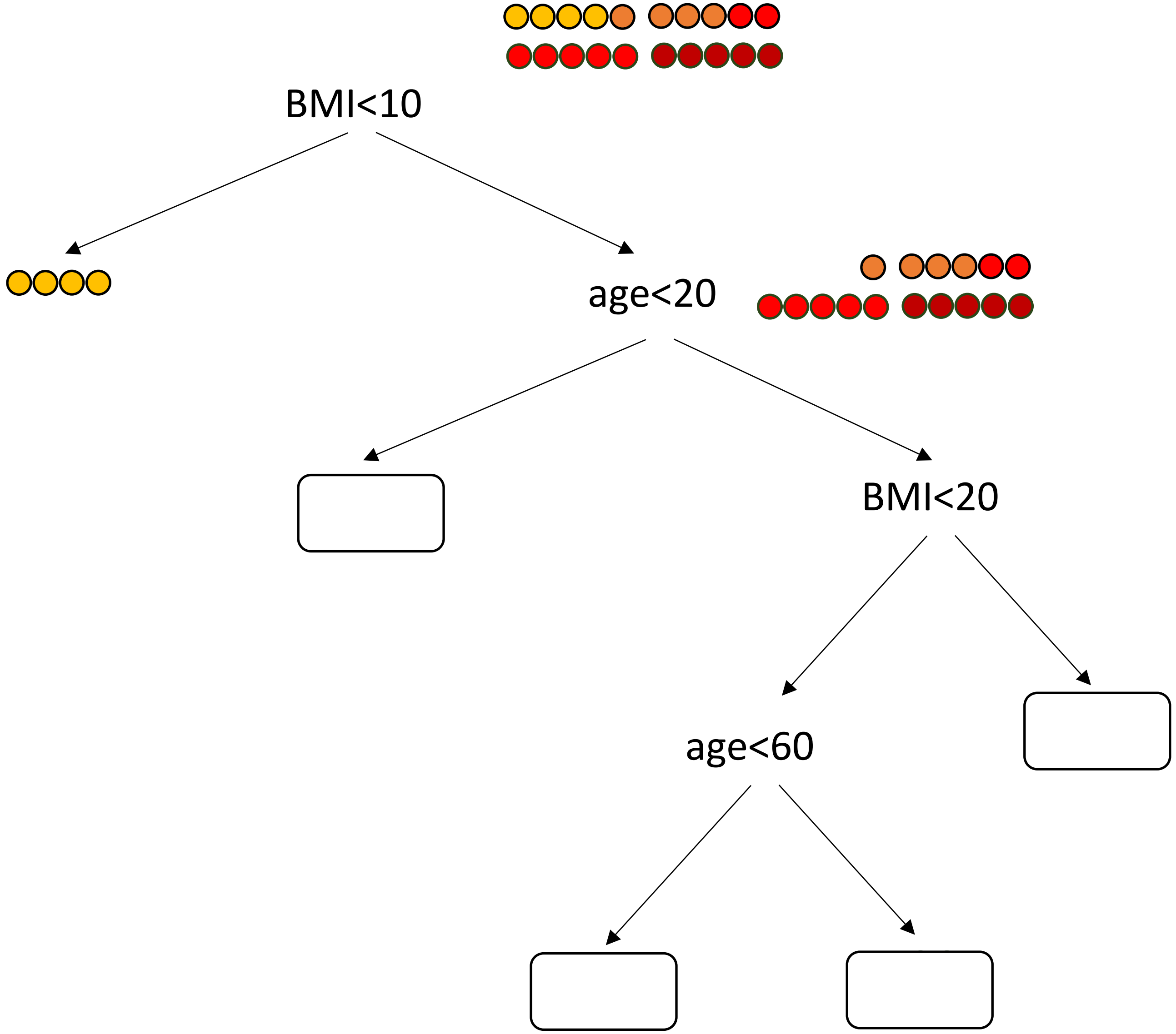
# Regression decision tree



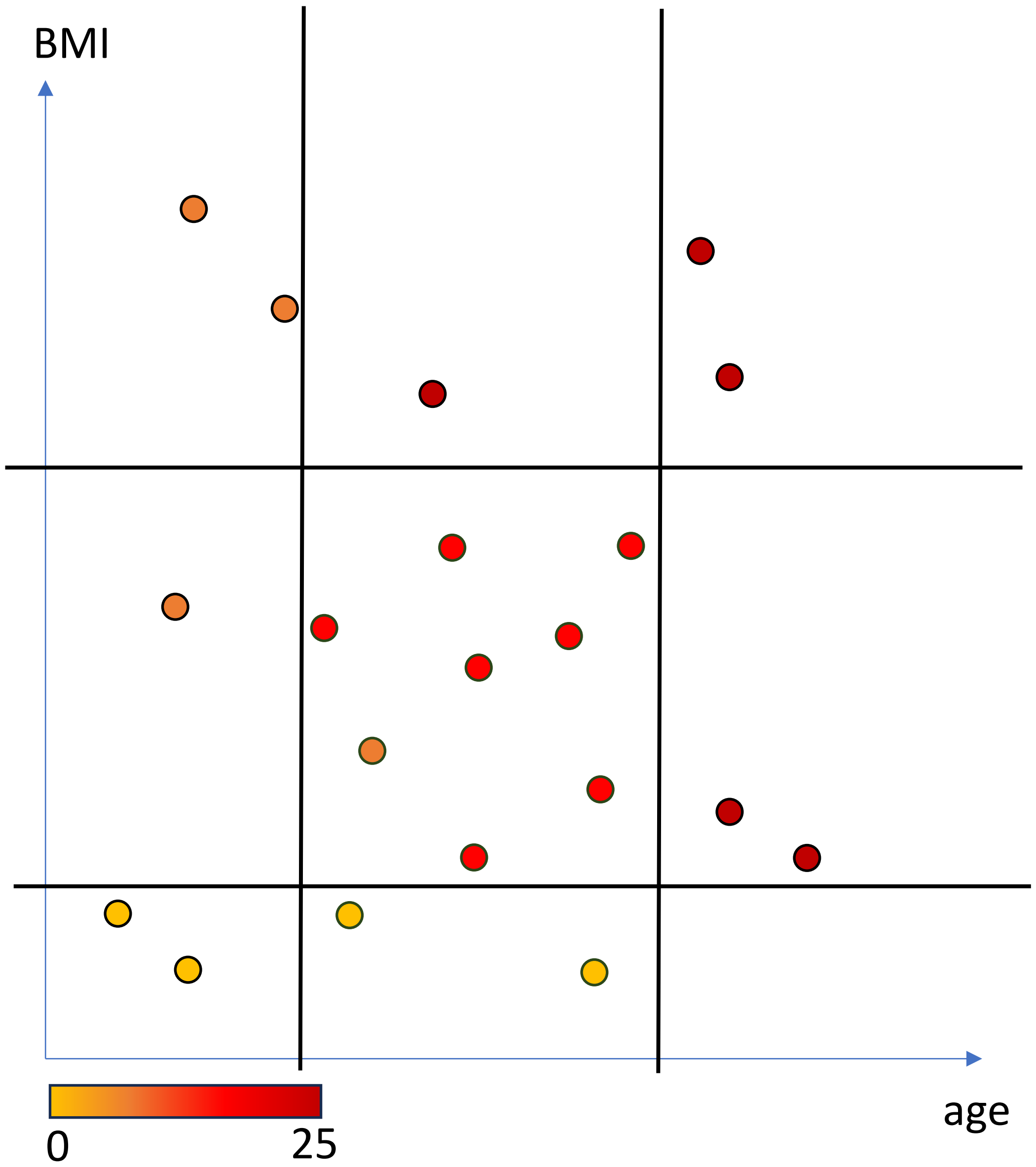



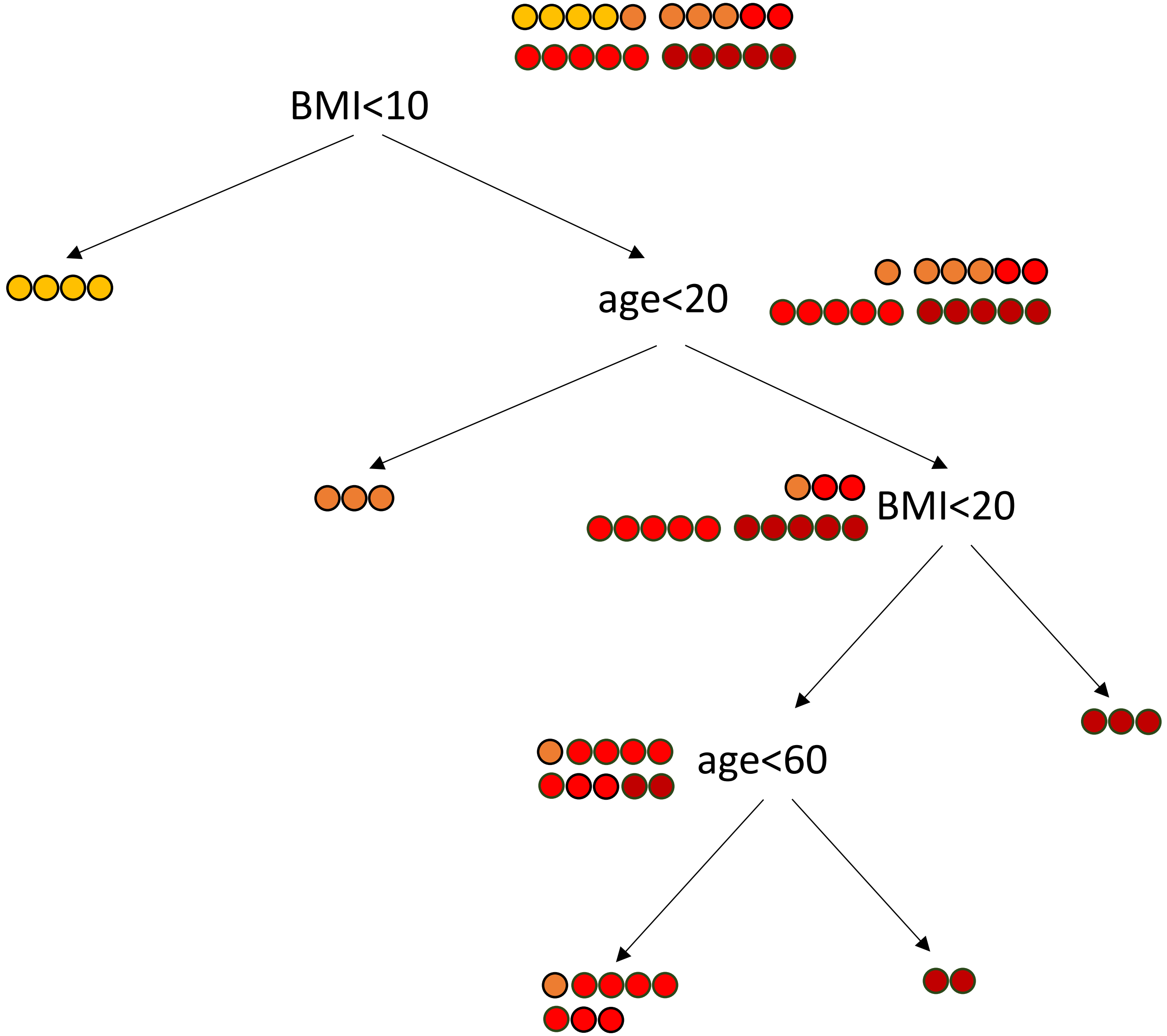
# Regression decision tree



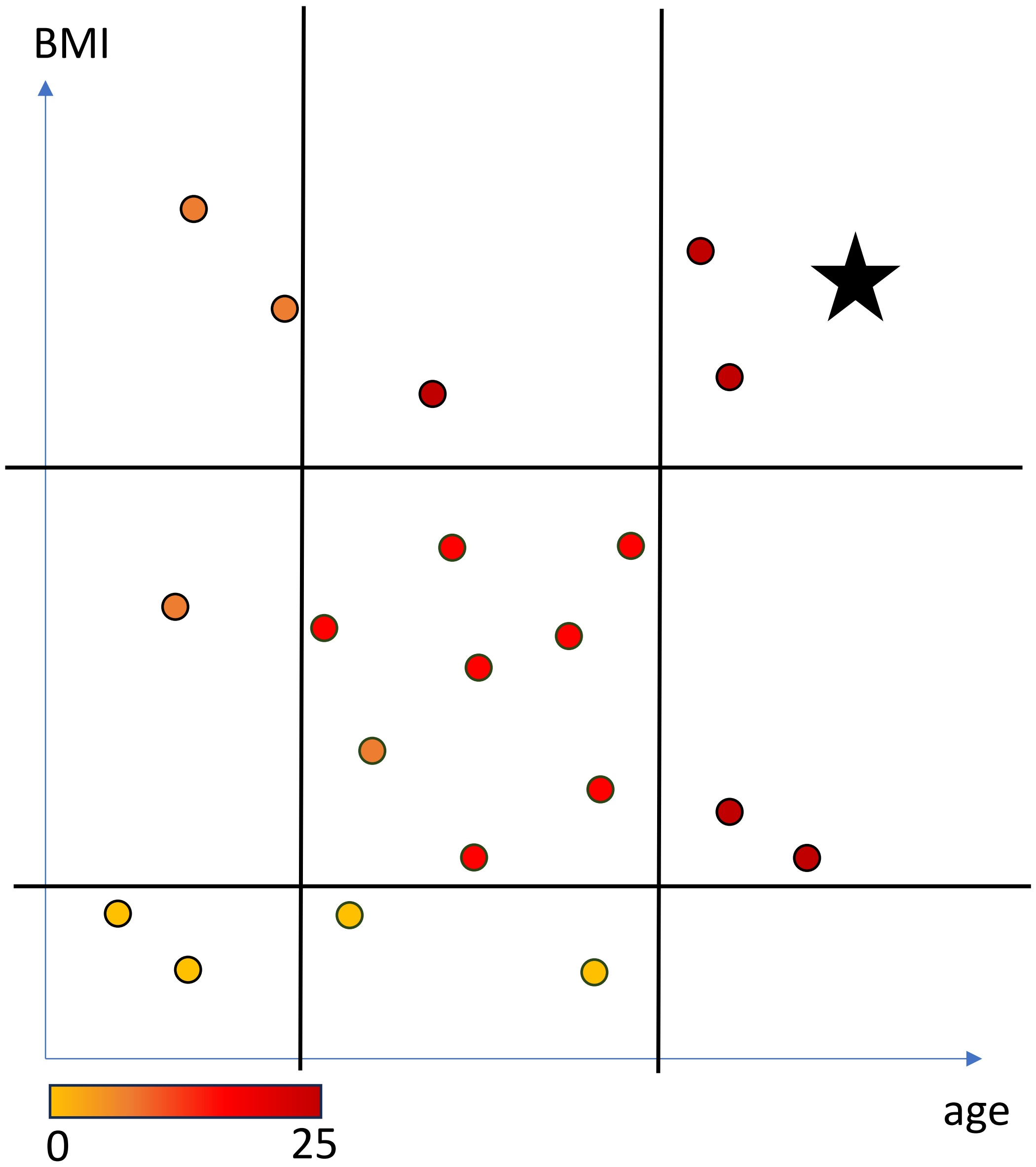



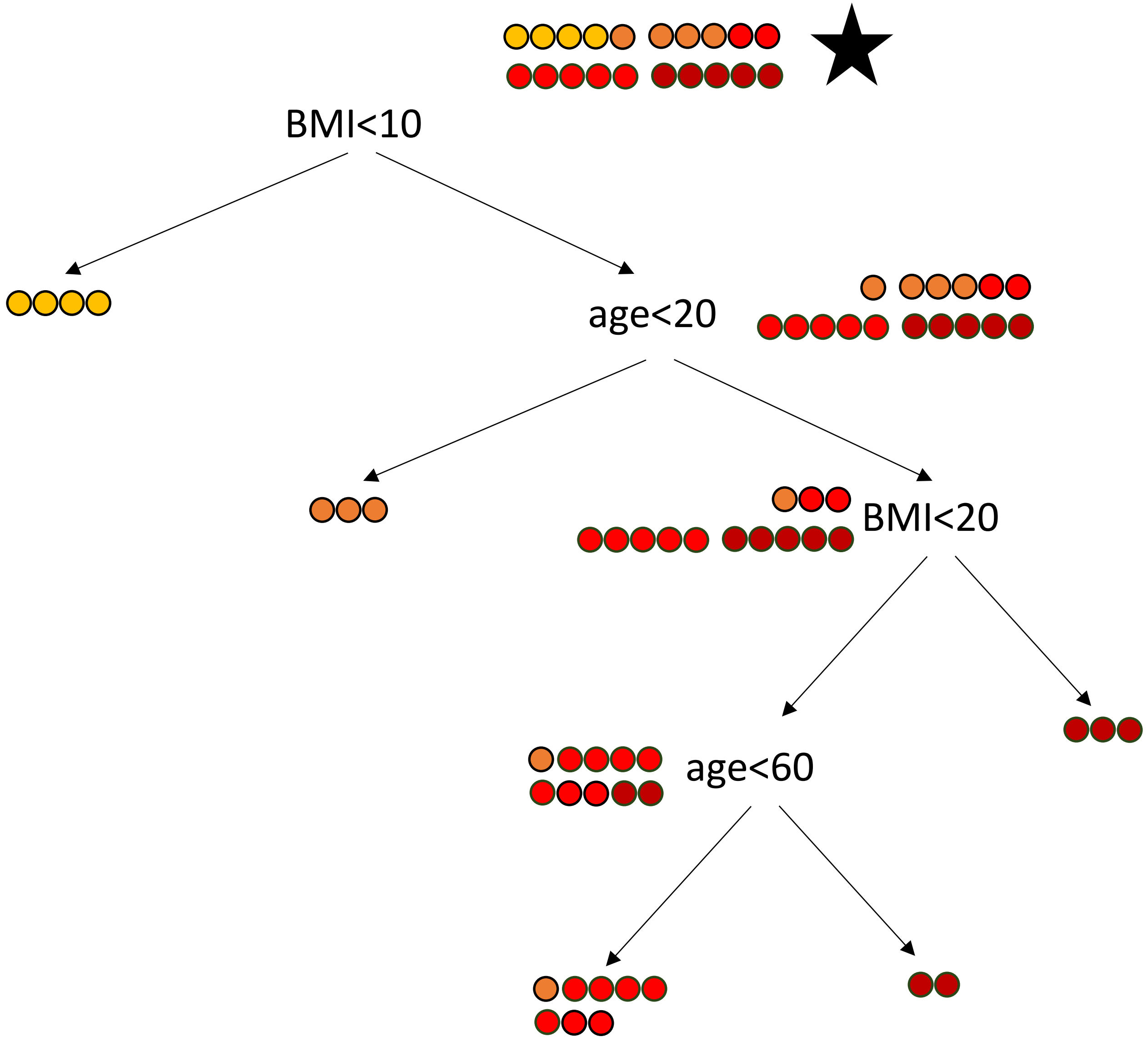
# Regression decision tree



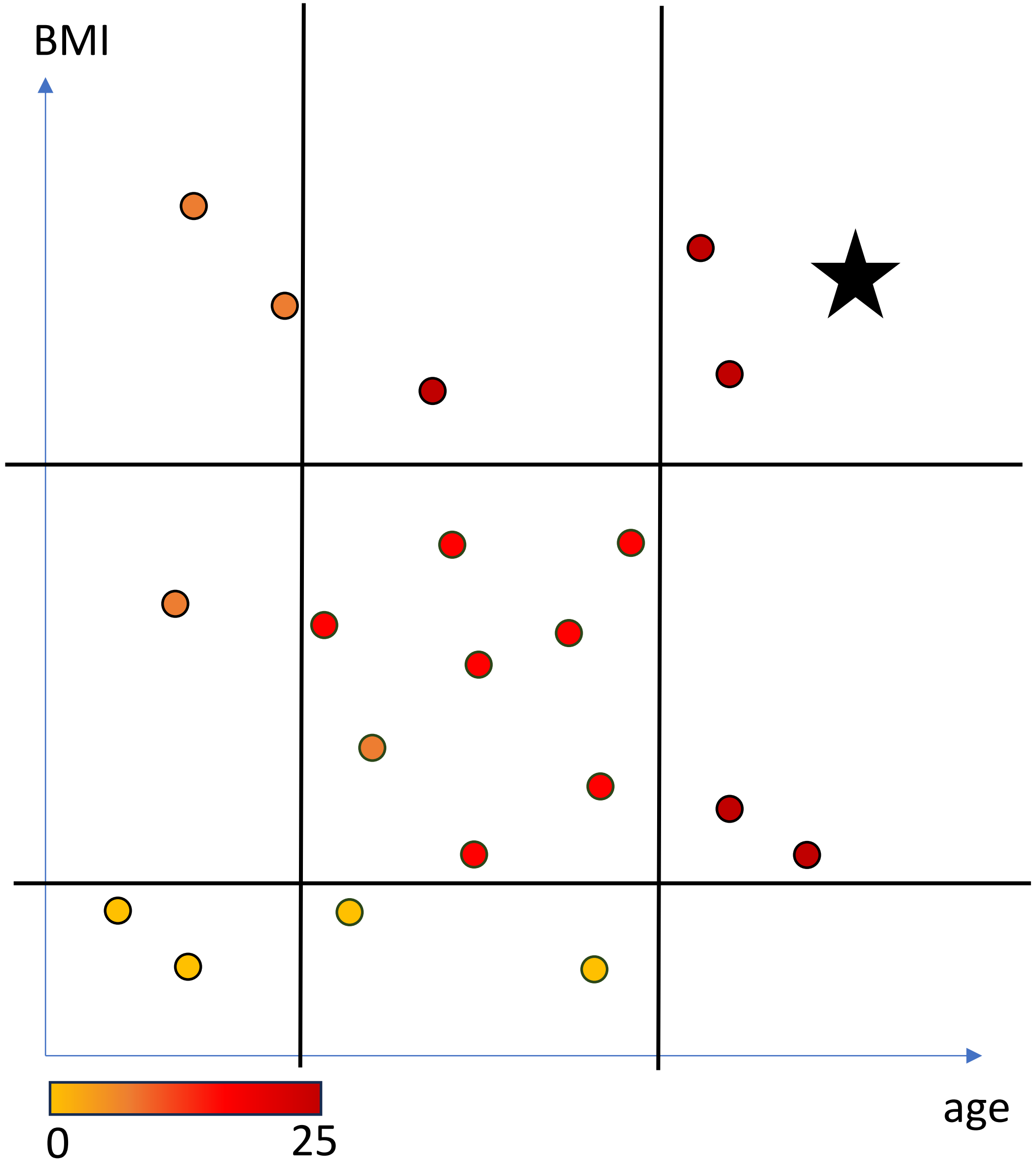



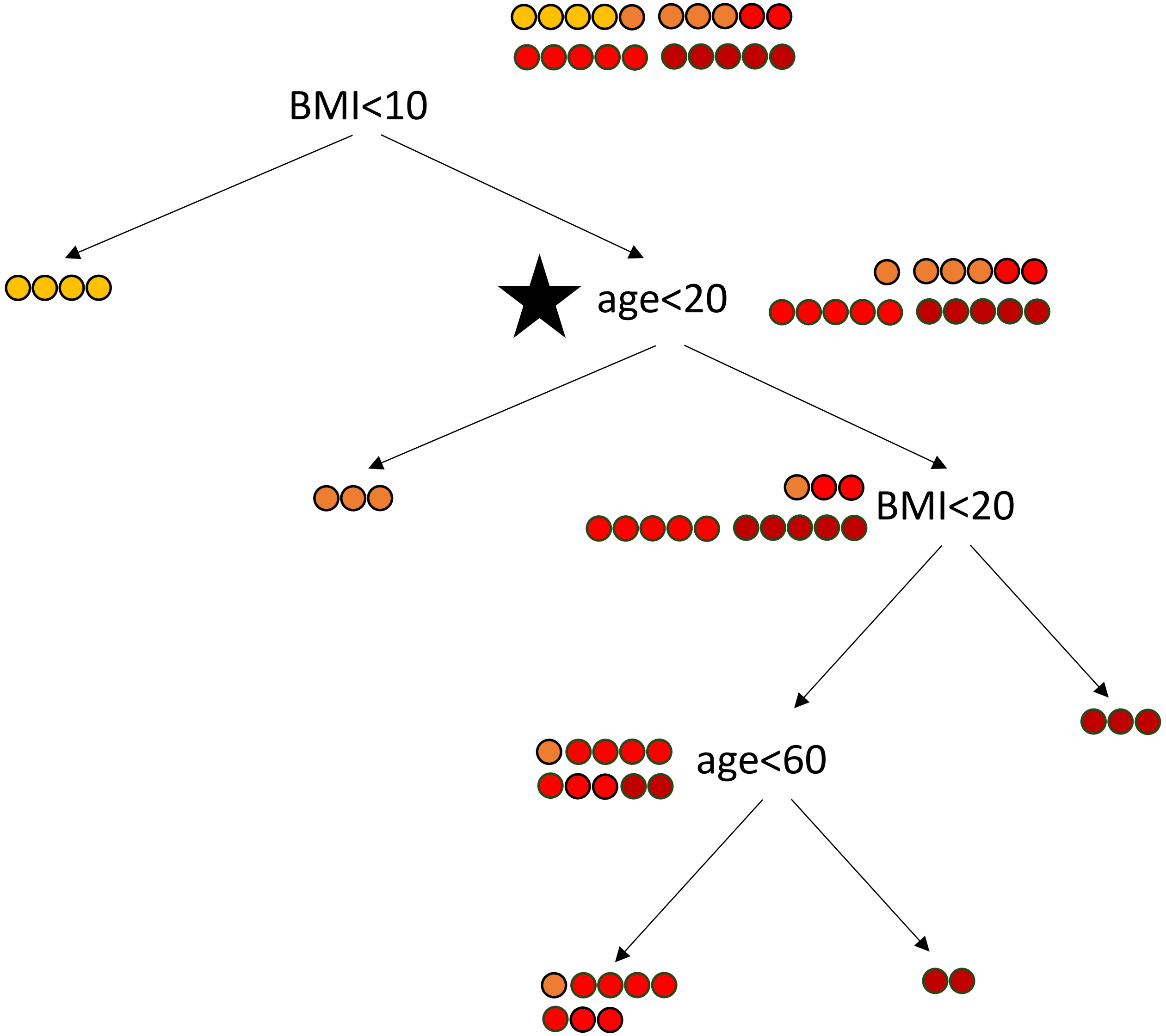
# Regression decision tree



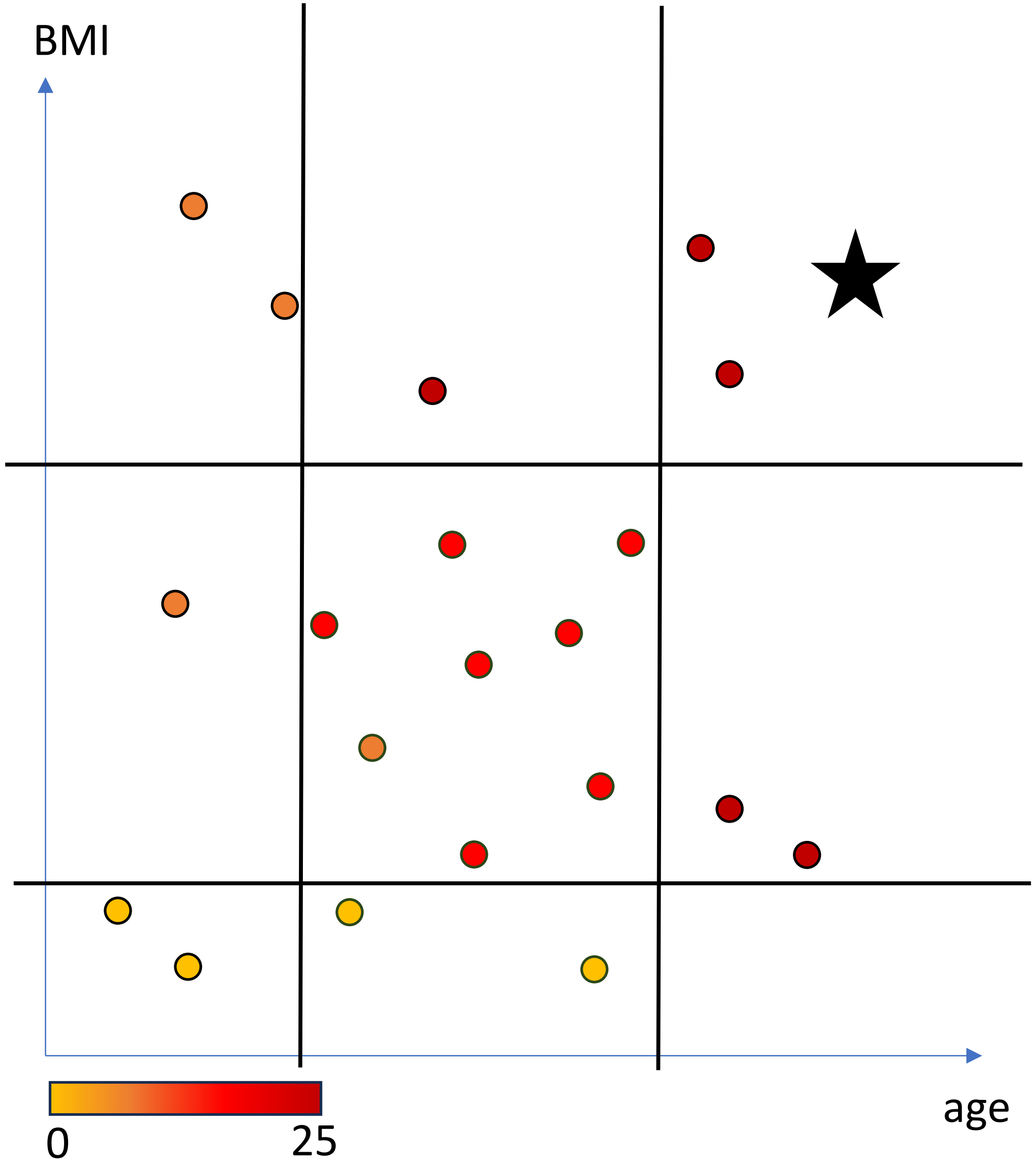



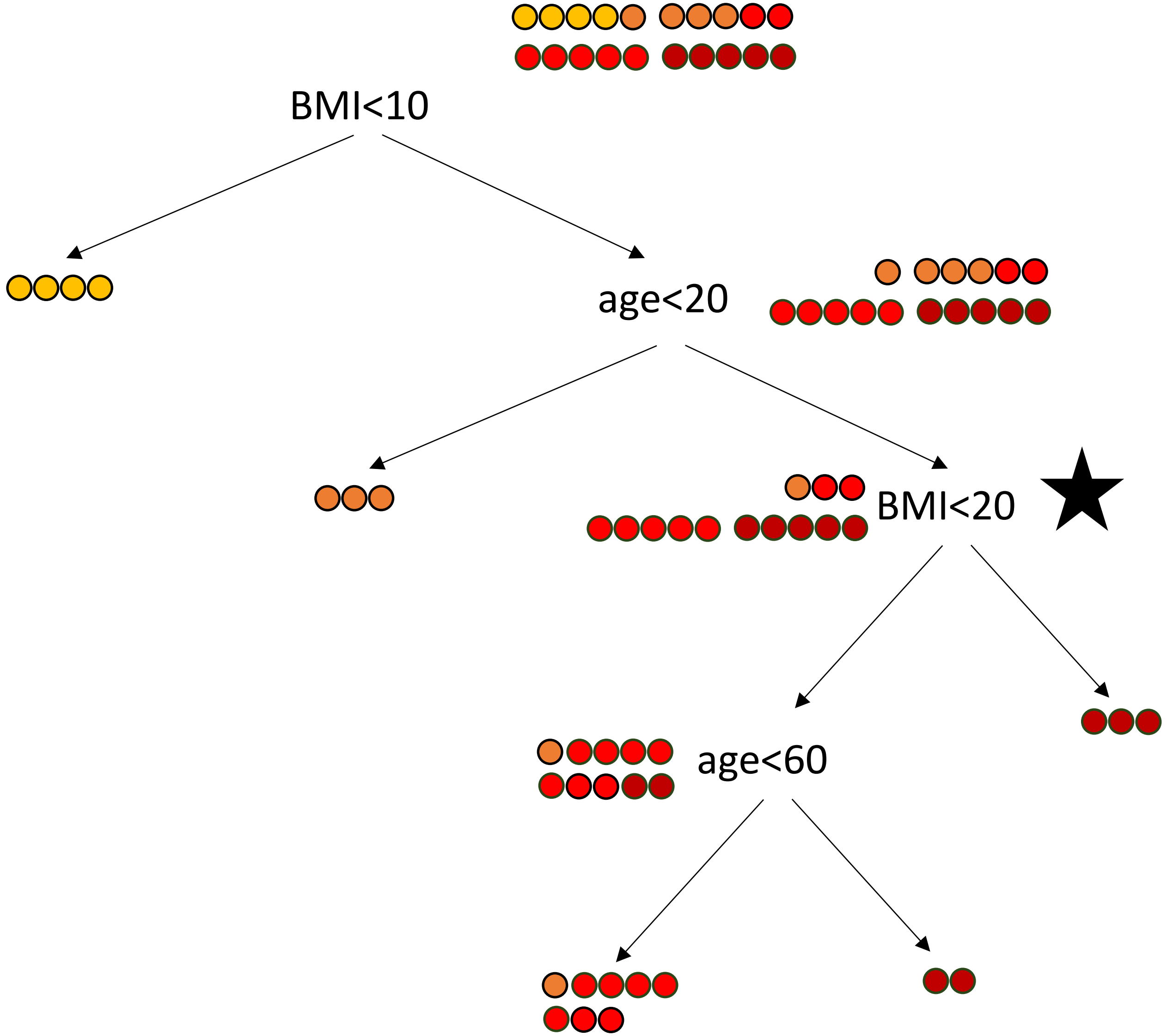
# Regression decision tree



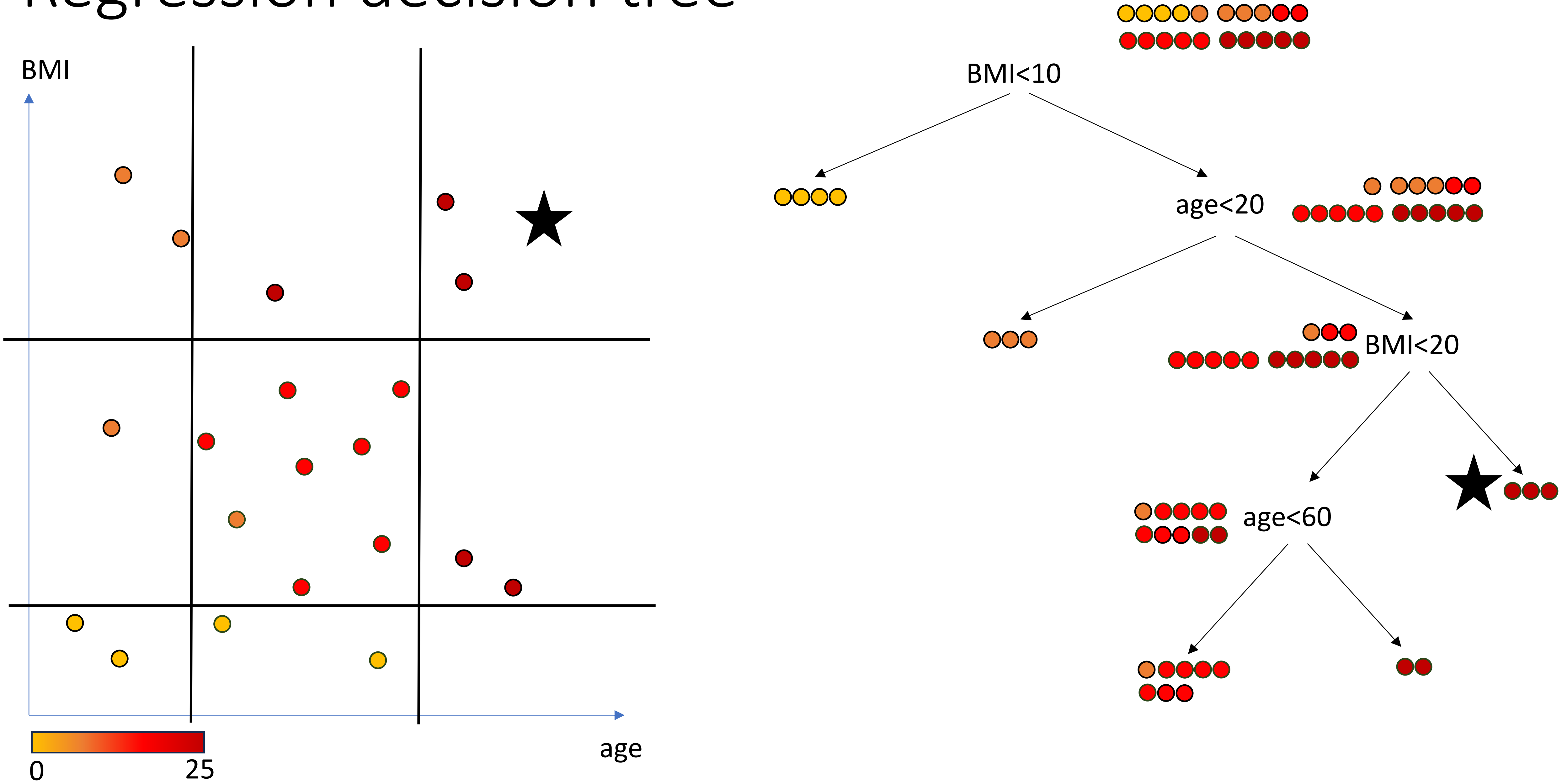



# Regression decision tree

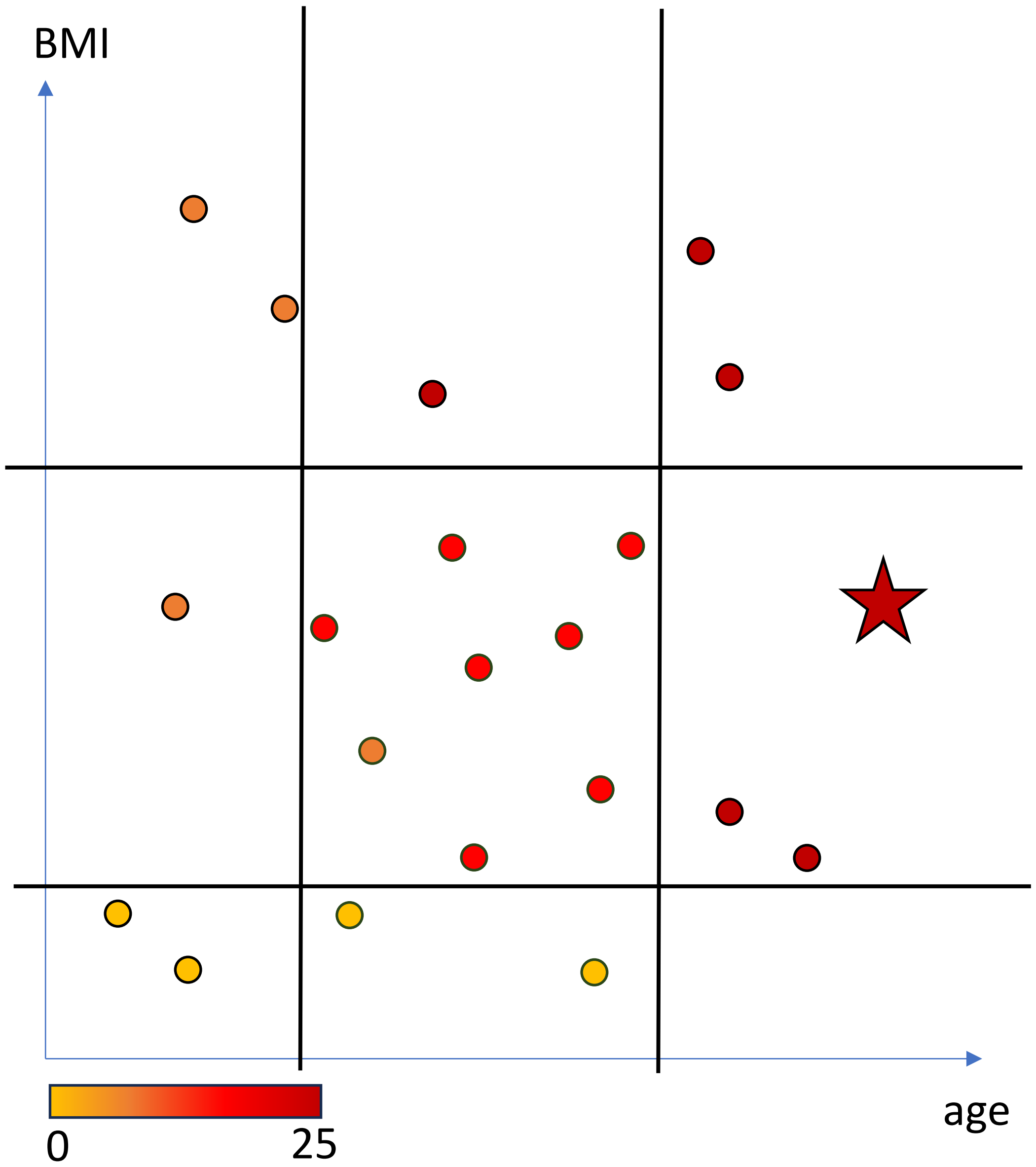


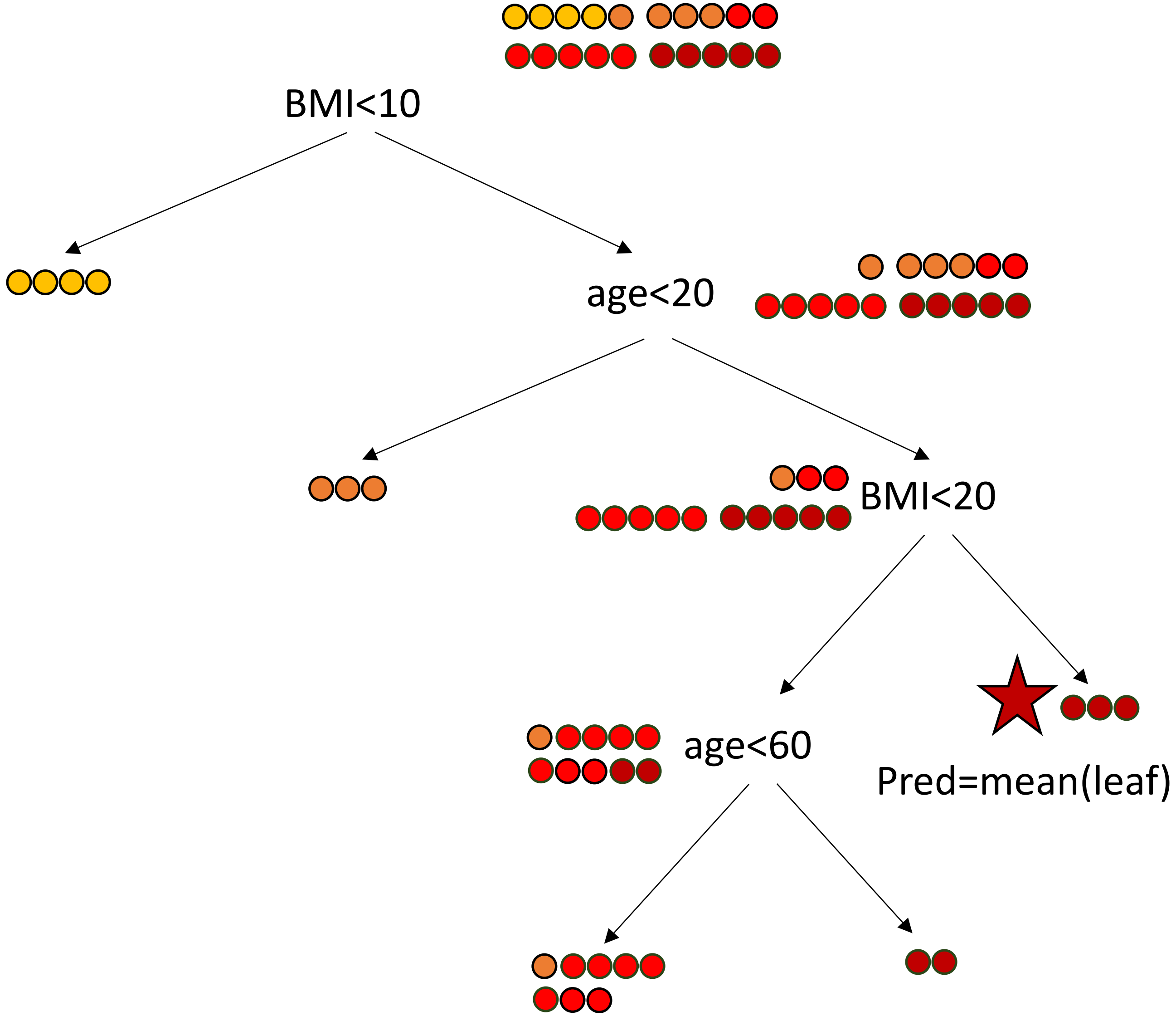



# Regression decision tree

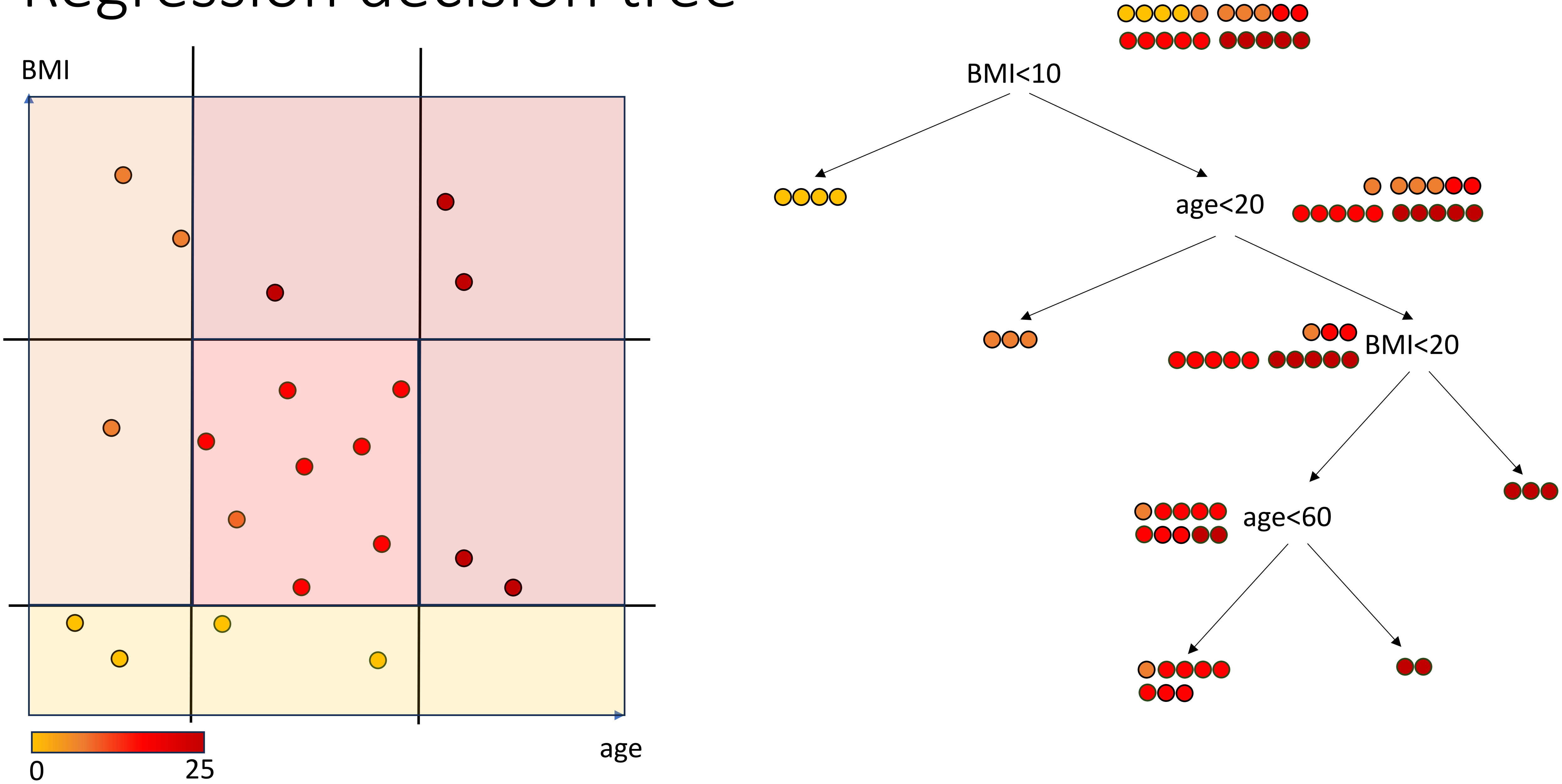


# Regression decision tree

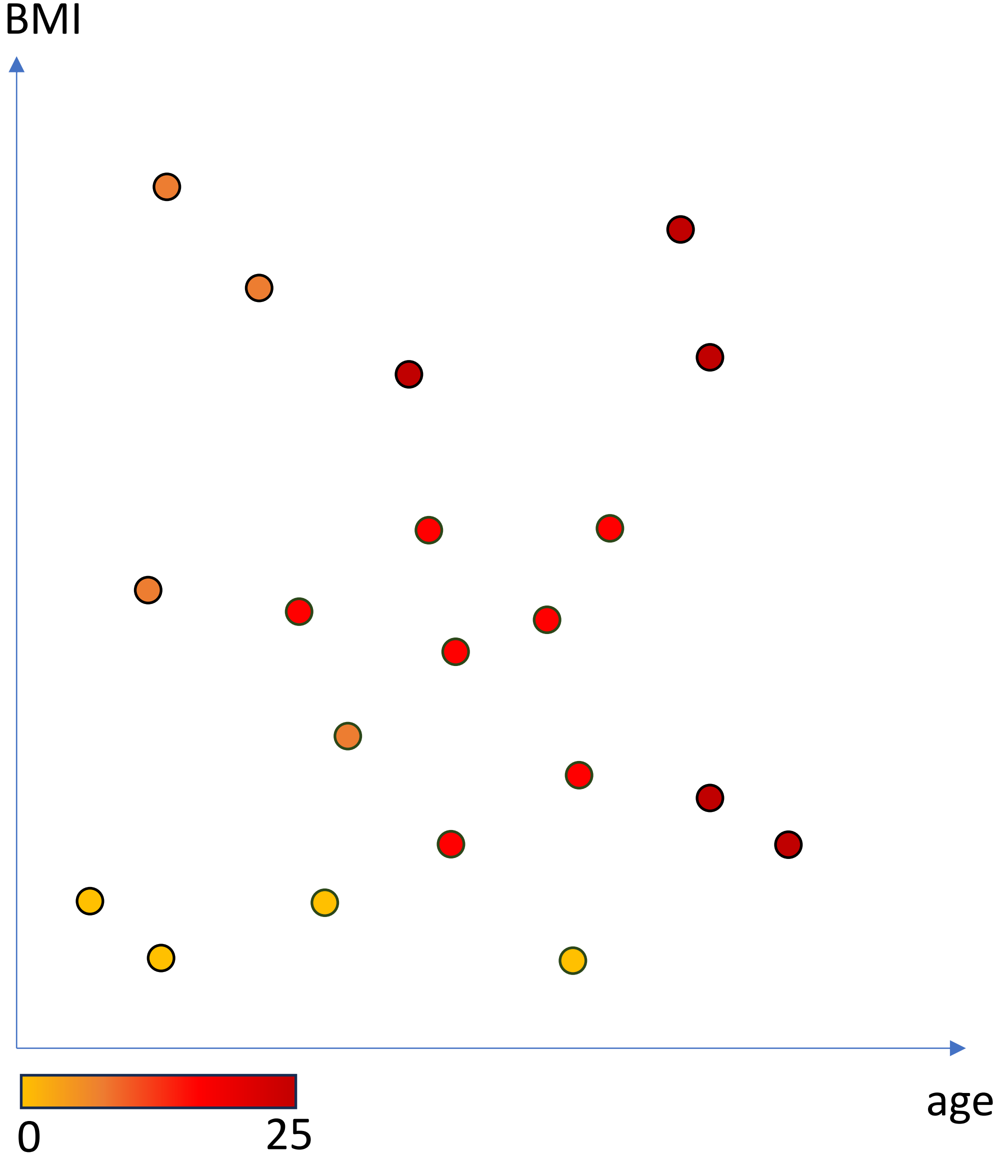


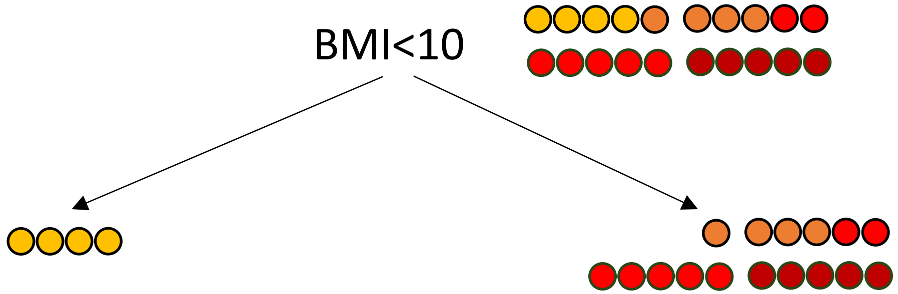



# Regression decision tree



# Training decision tree

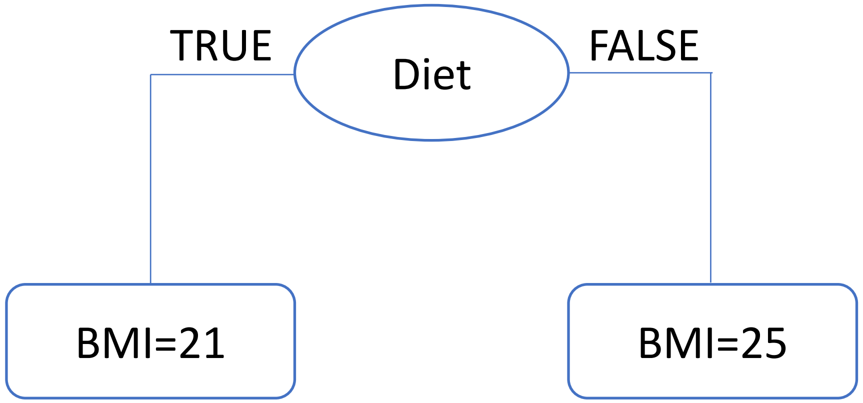



$$Var\ Red = Var(parent) - \Sigma\, w_i\, Var(child_i)$$

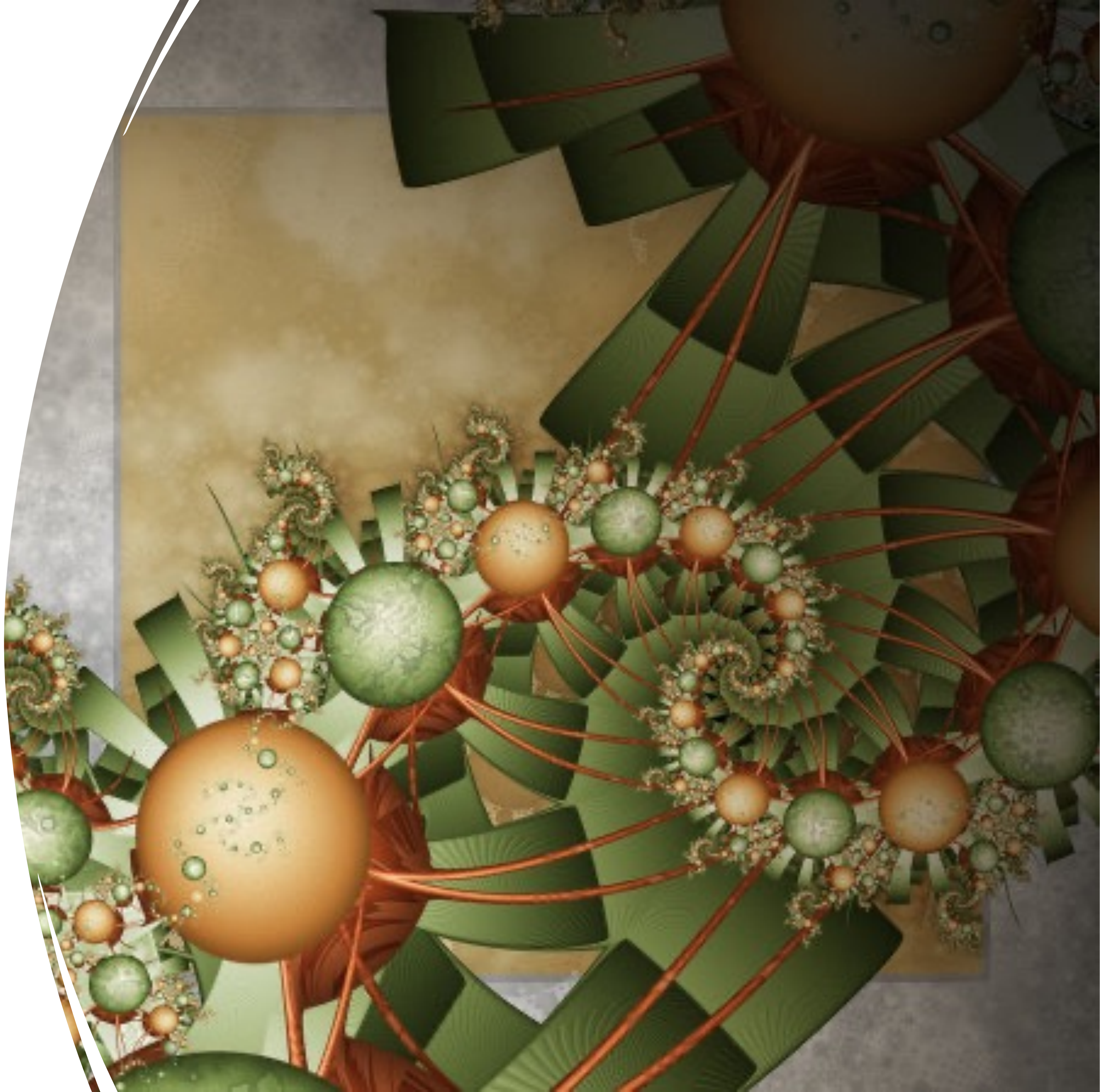
**The best split is the split that maximizes the variance reduction**

# Let’s recap!

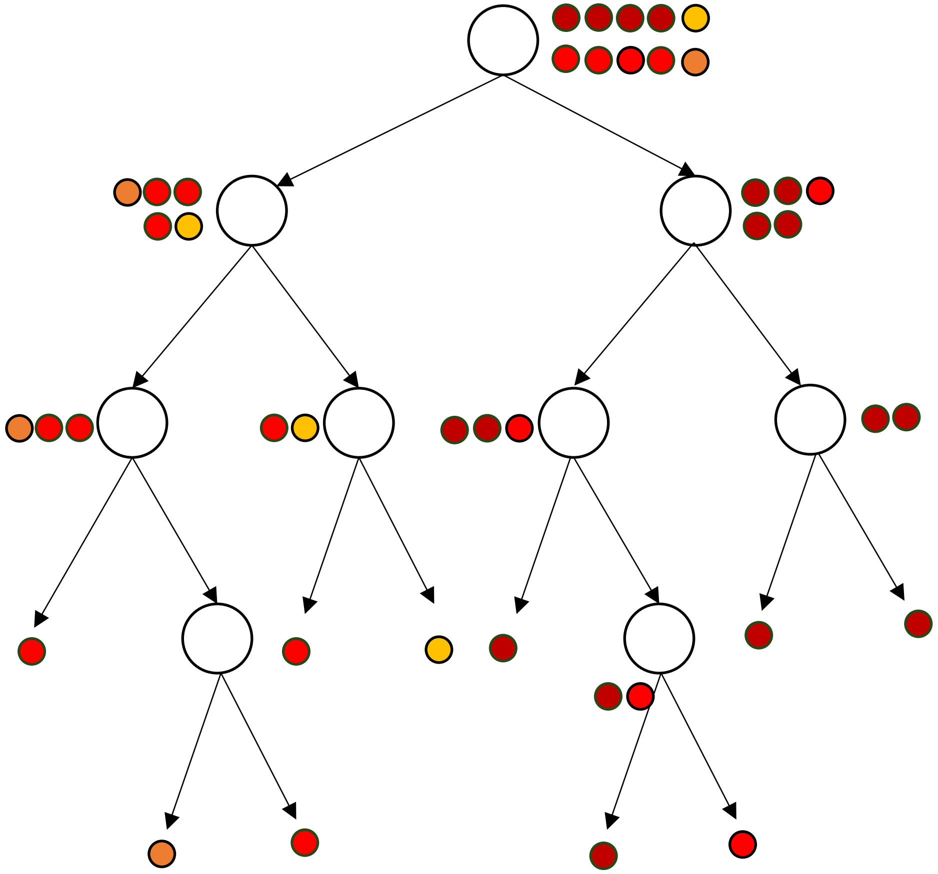



- Decision trees can be used to perform a regression task
- Training of a regression decision tree consists in finding the best data split that maximizes the variance reduction
- One of the main advantage of a decision tree is that you can combine categorical and numerical variables in the same model

# Overfitting and decision trees

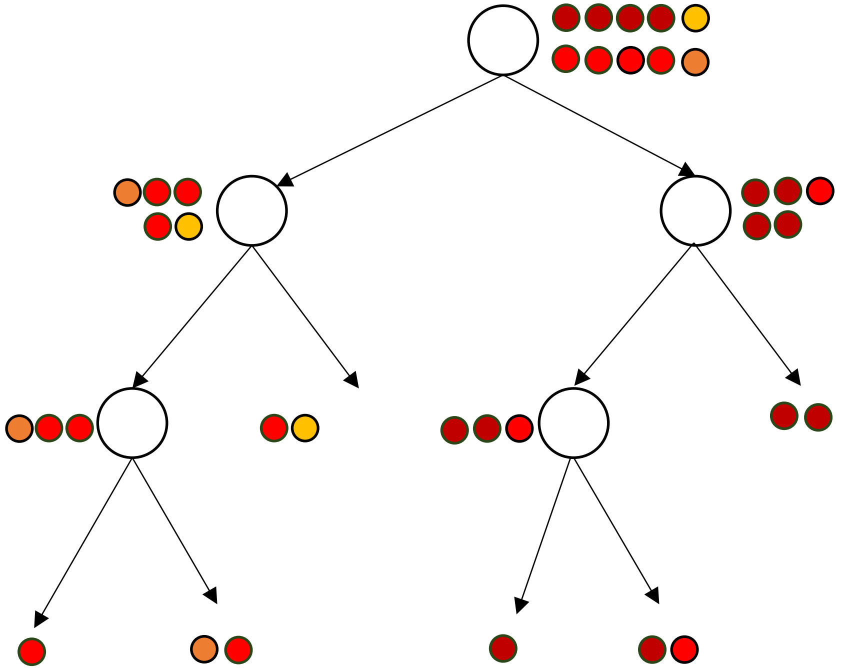


# Regression decision tree



**Overfitting:** the decision tree fits too perfectly the training dataset

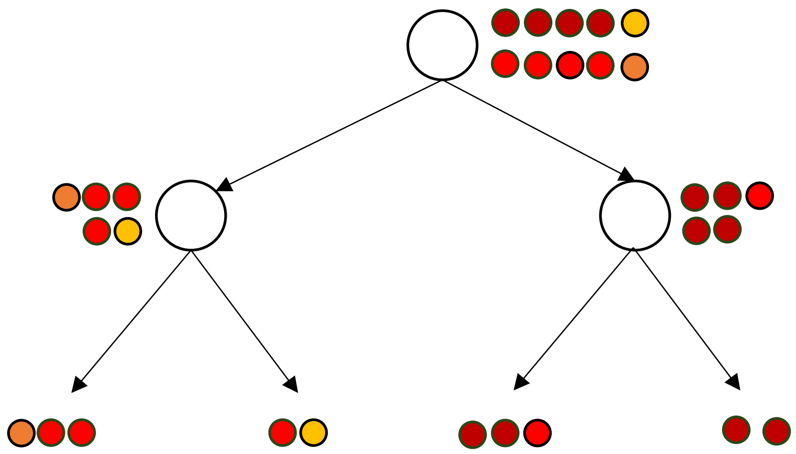
# Regression decision tree



**Overfitting:** the decision tree fits too perfectly the training dataset

**<u>- Pruning:</u>** remove branches that do not reduce the variance below a certain cutoff

# Regression decision tree

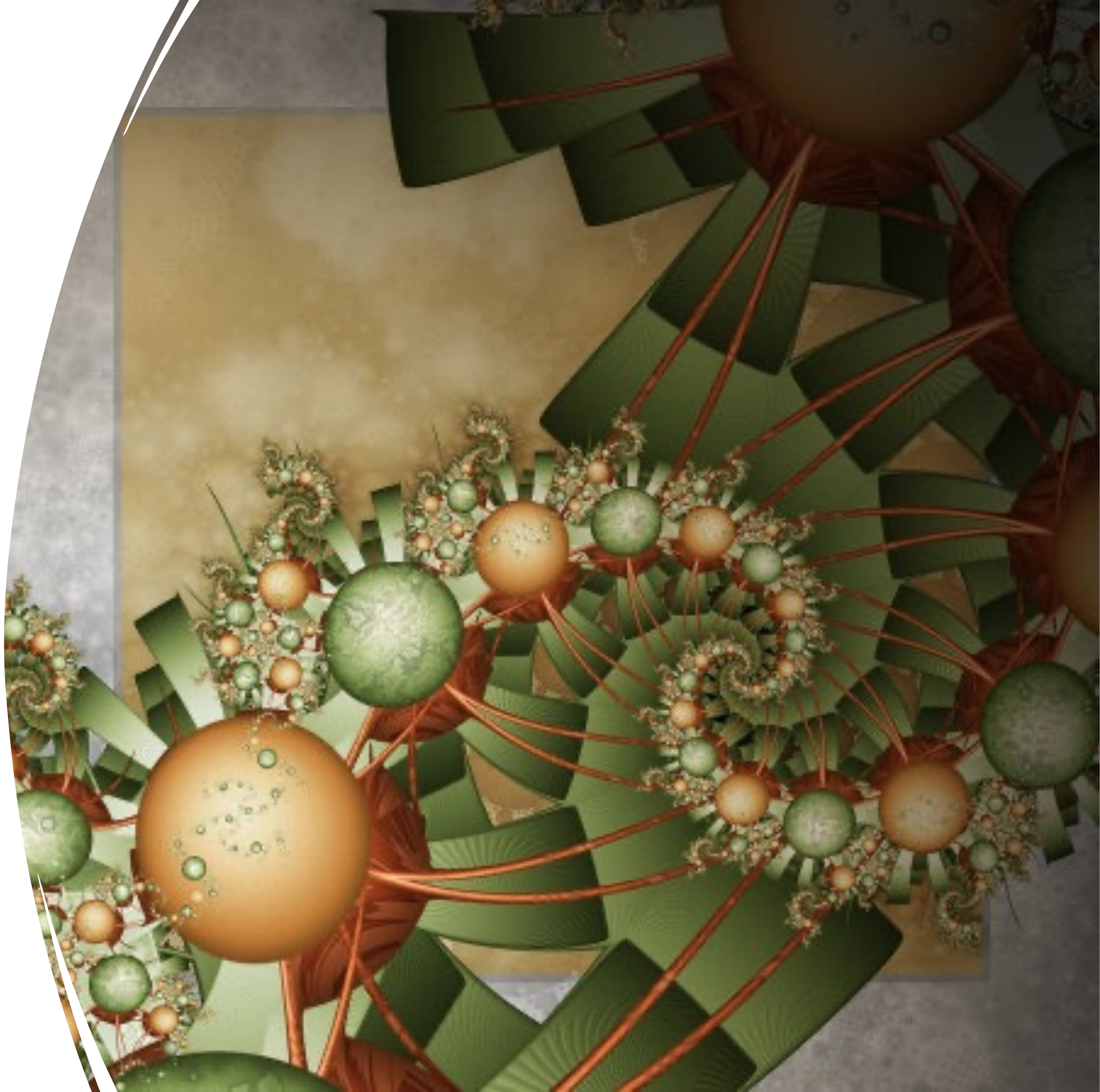


**Overfitting:** the decision tree fits too perfectly the training dataset

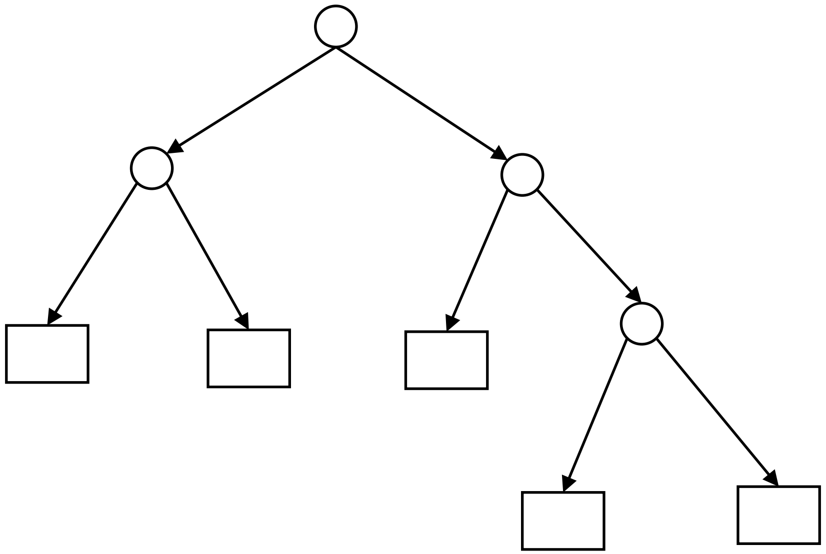
**<u>- Pruning: </u>**remove branches that do not reduce the variance below a certain cutoff

**- Max depth:** stop the tree training after a certain number of nodes

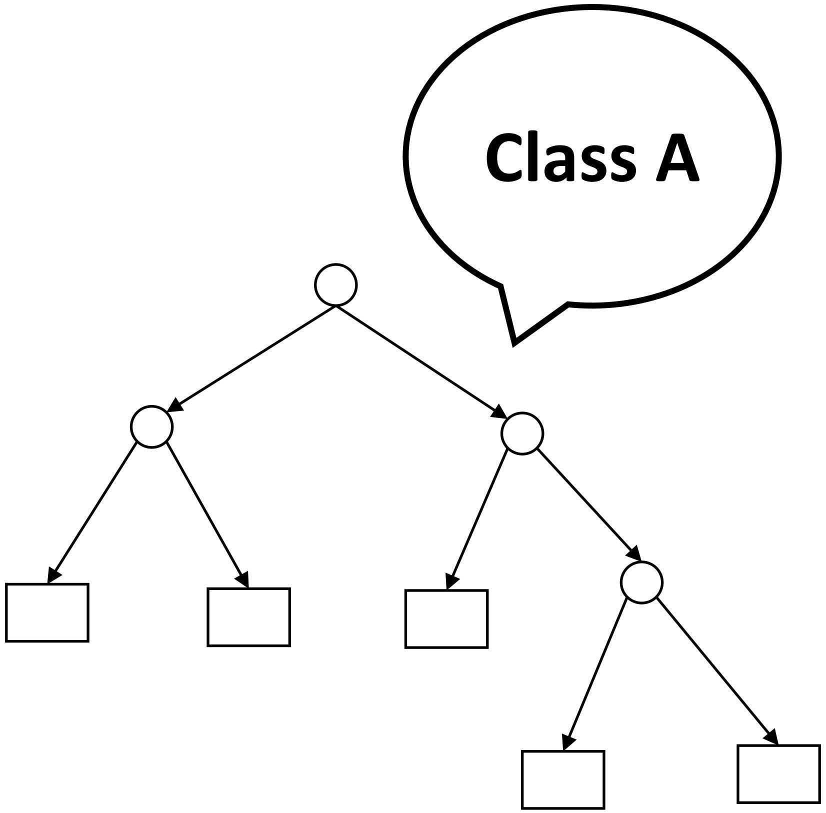
# Random forests



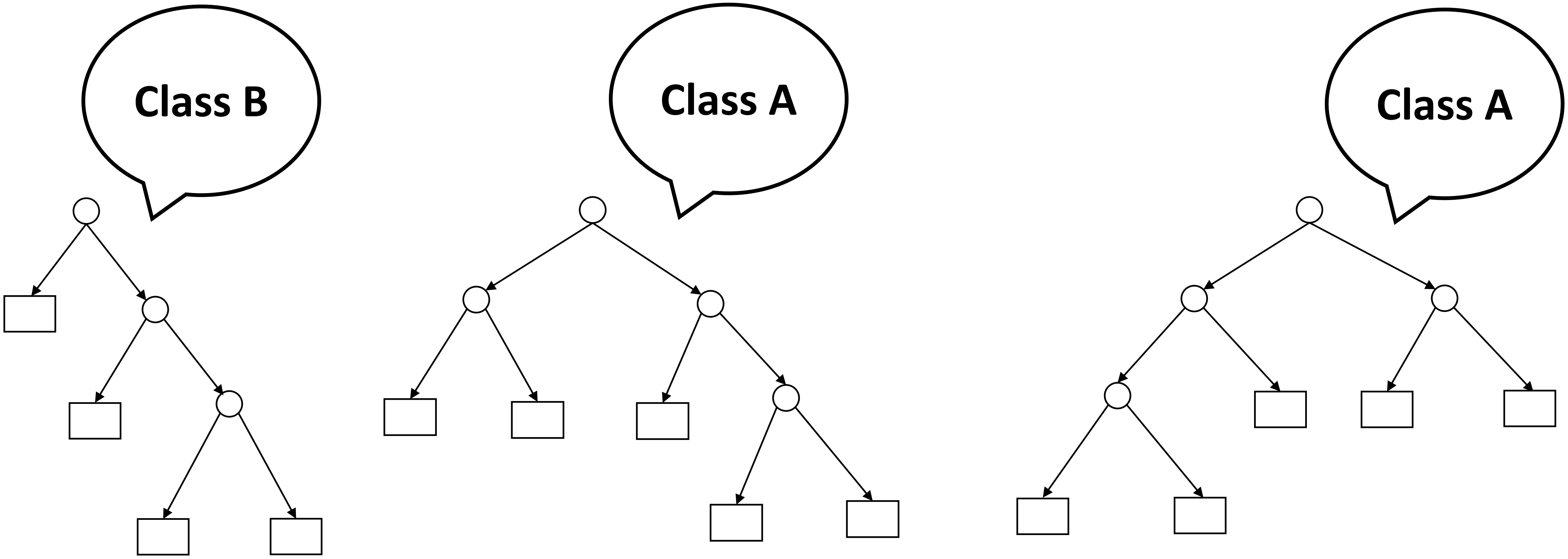
# Ensemble learning methods



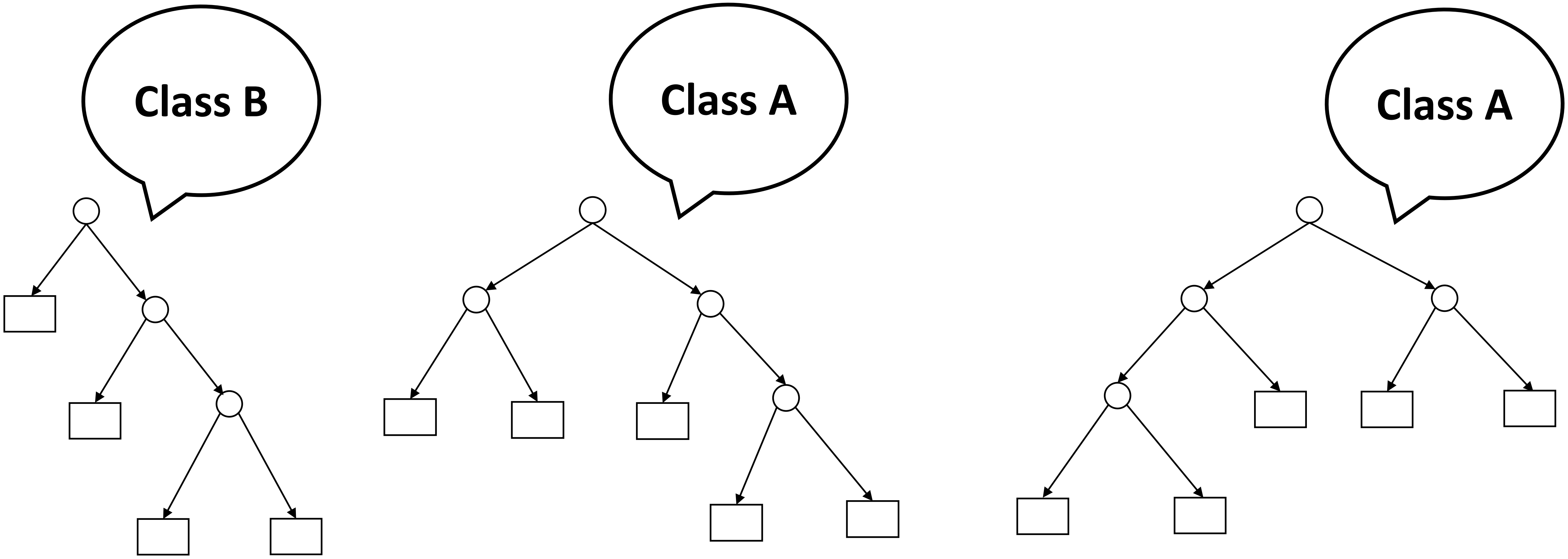
# Ensemble learning methods



# Ensemble learning methods



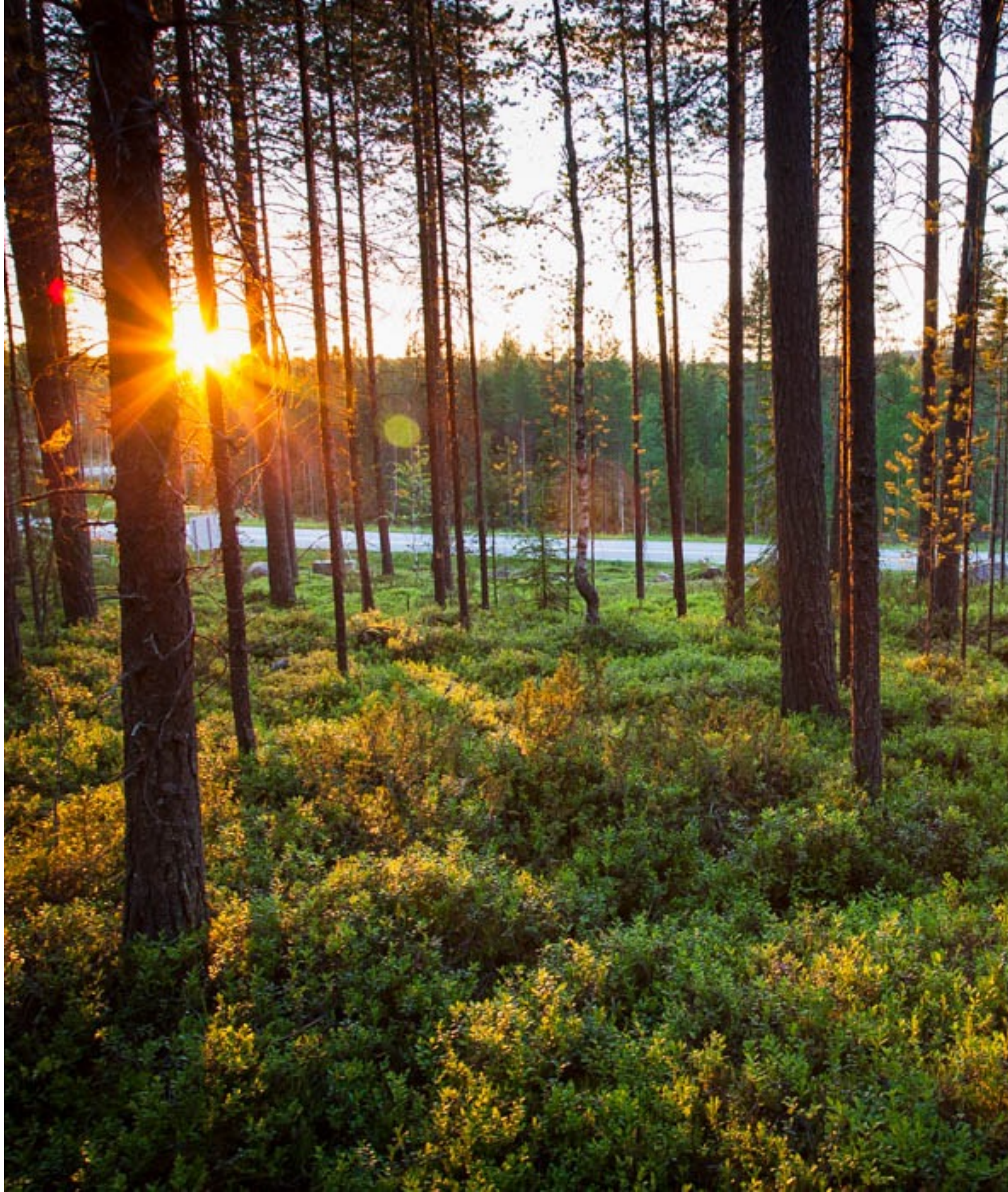
# Ensemble learning methods




**Prediction: Class with the most votes**

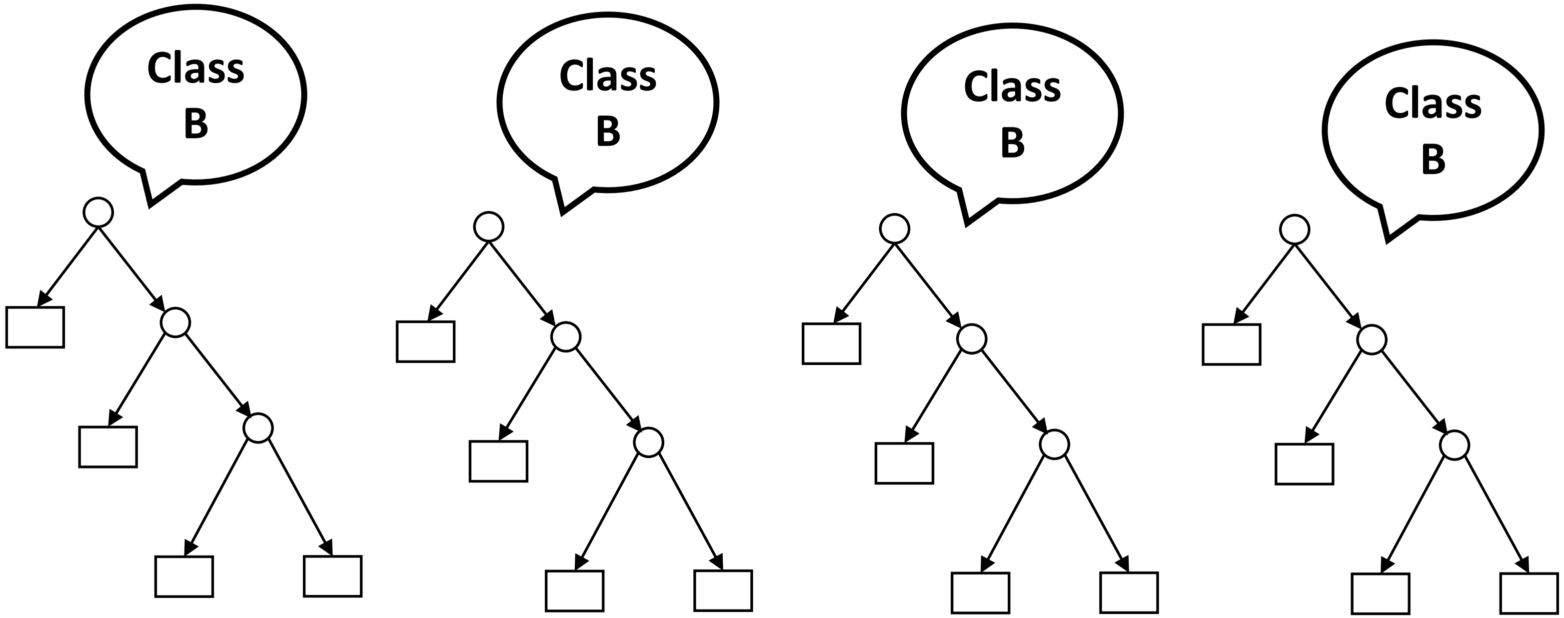
# Ensemble learning methods

Ensemble learning methods are based on the idea that a prediction from a crowd of models is more stable and accurate than the prediction of any one model alone



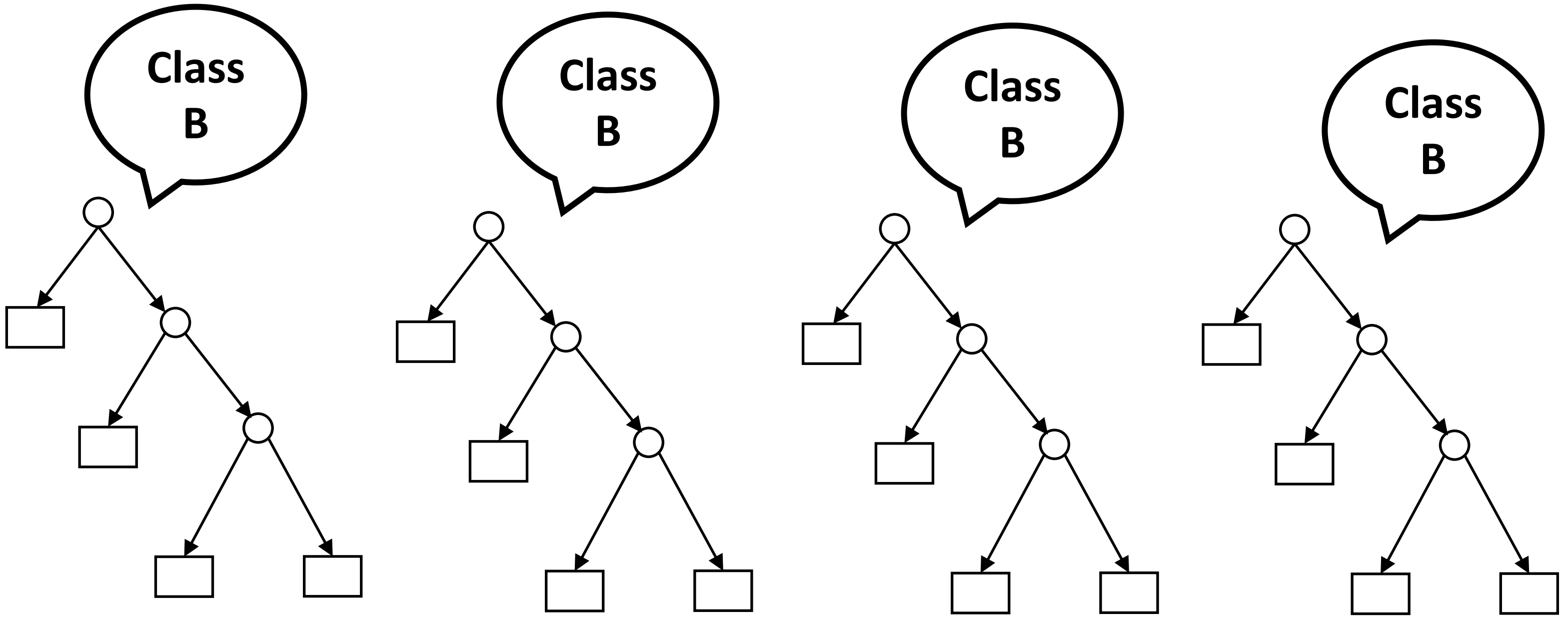
# Building different models

We need to build models different enough to make the combined predictions valuable




**Prediction: Class with the most votes**

# Building different models

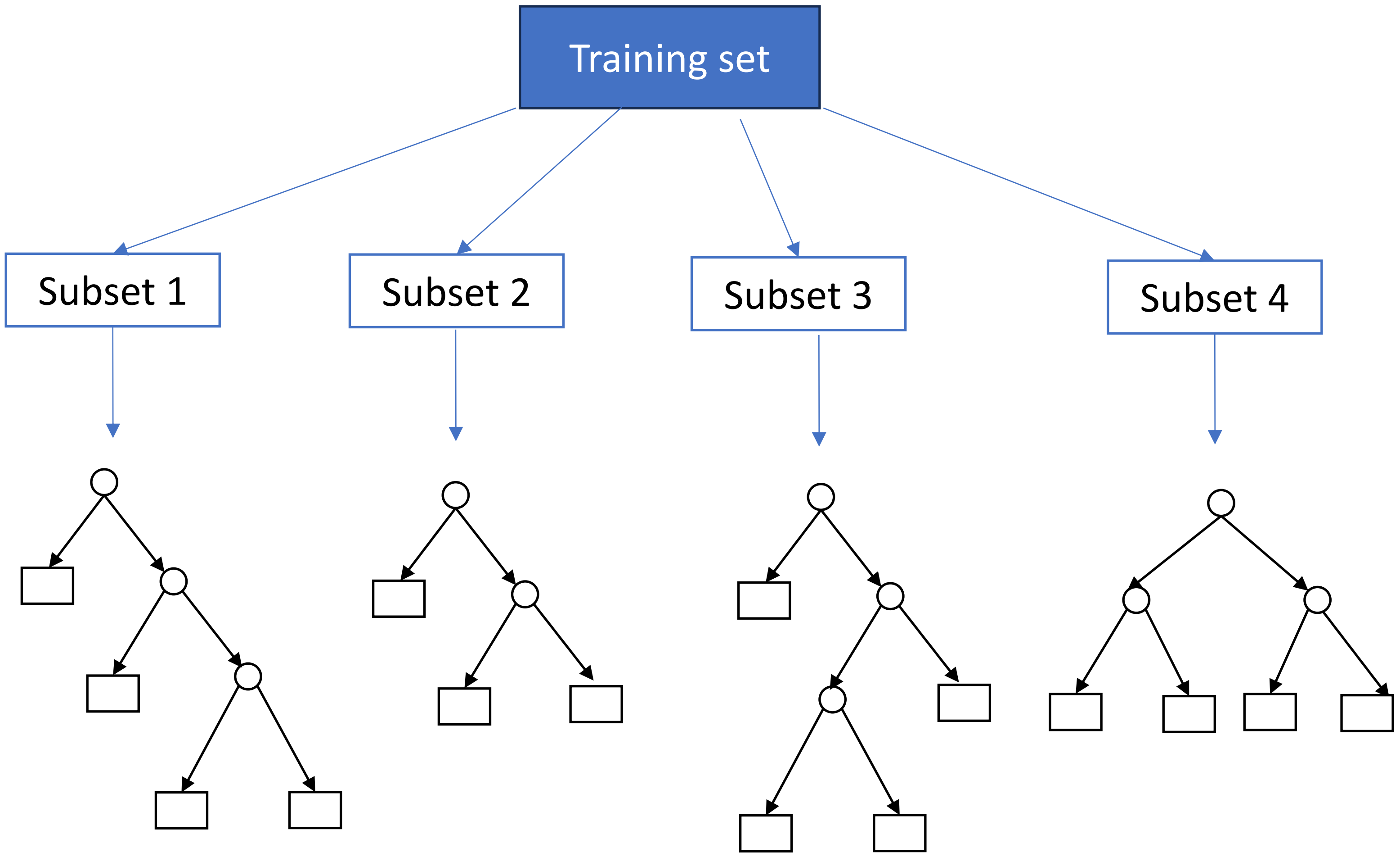



**Prediction: Class with the most votes**

We need to build models different enough to make the combined predictions valuable

How do you build different models with one unique training set?

# Building different models

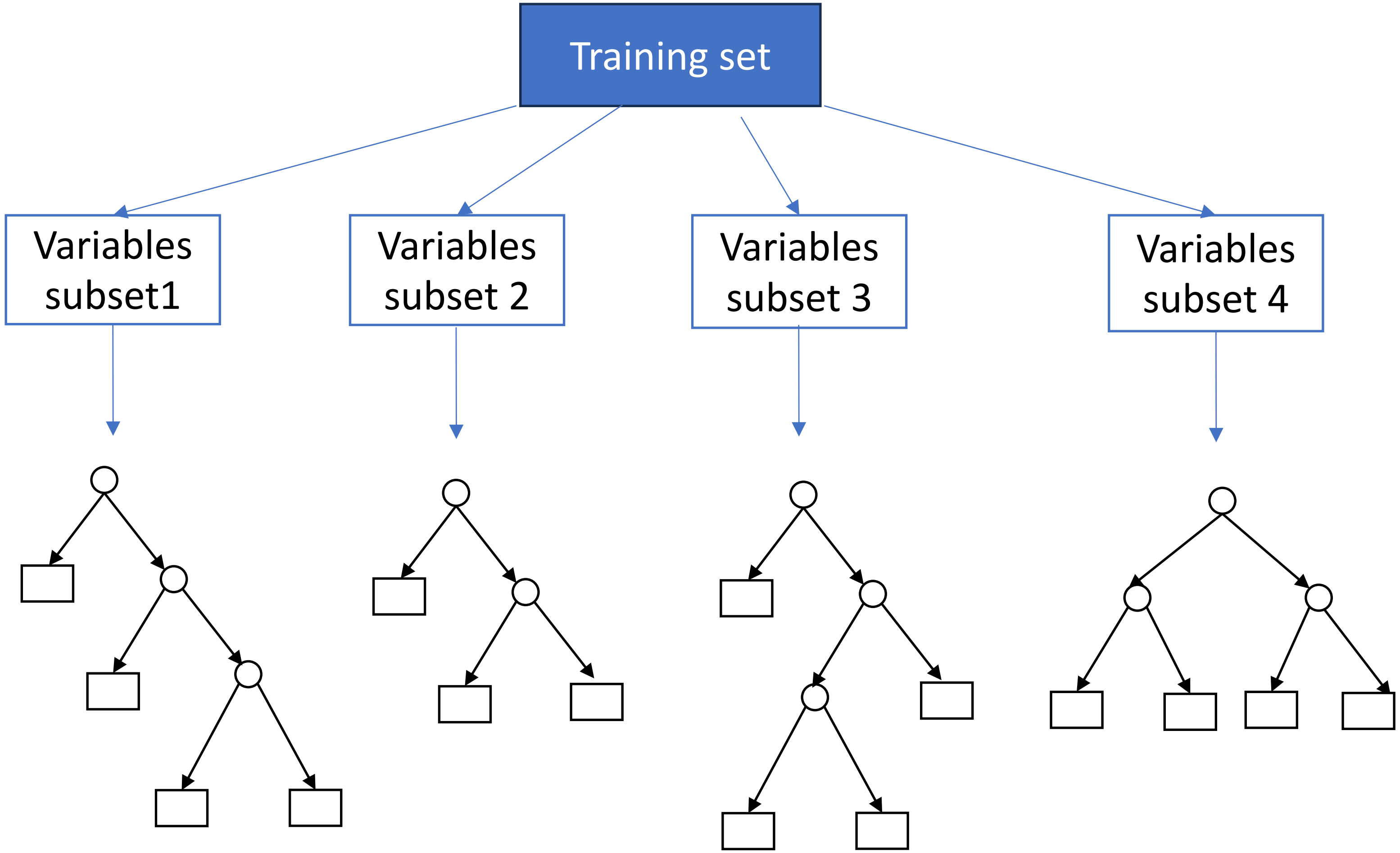



Uncorrelatedness is critical for random forests models

**Method 1:** Bootstrapping (Bagging)

**Prediction: Class with the most votes**

# Building different models




Uncorrelatedness is critical for random forests models

**Method 1:** Bootstrapping (Bagging)

**Method 2:** Feature randomness

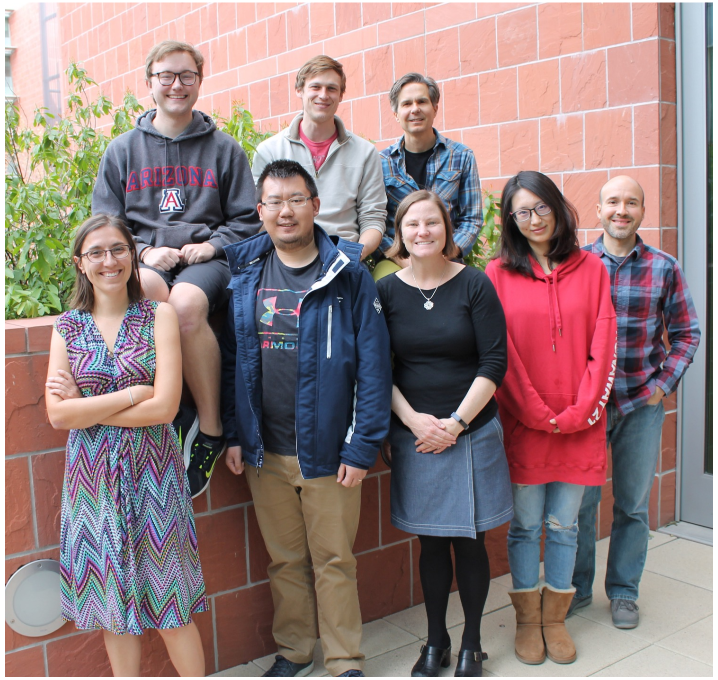
**Prediction: Class with the most votes**

# Let’s recap

- Random forests are an ensemble of decision trees all voting for the same prediction tasks

- Random forests can be composed of thousands of small decision trees, all very simple to train

- Methods such as bagging or variable randomness ensure that the trees are different enough and learn different visions of the reality




**Hurwitz lab**

**Salonen lab**

**My contact:**
alise.ponsero@helsinki.fi

**LinkedIn:**
alise-ponsero-843b953b