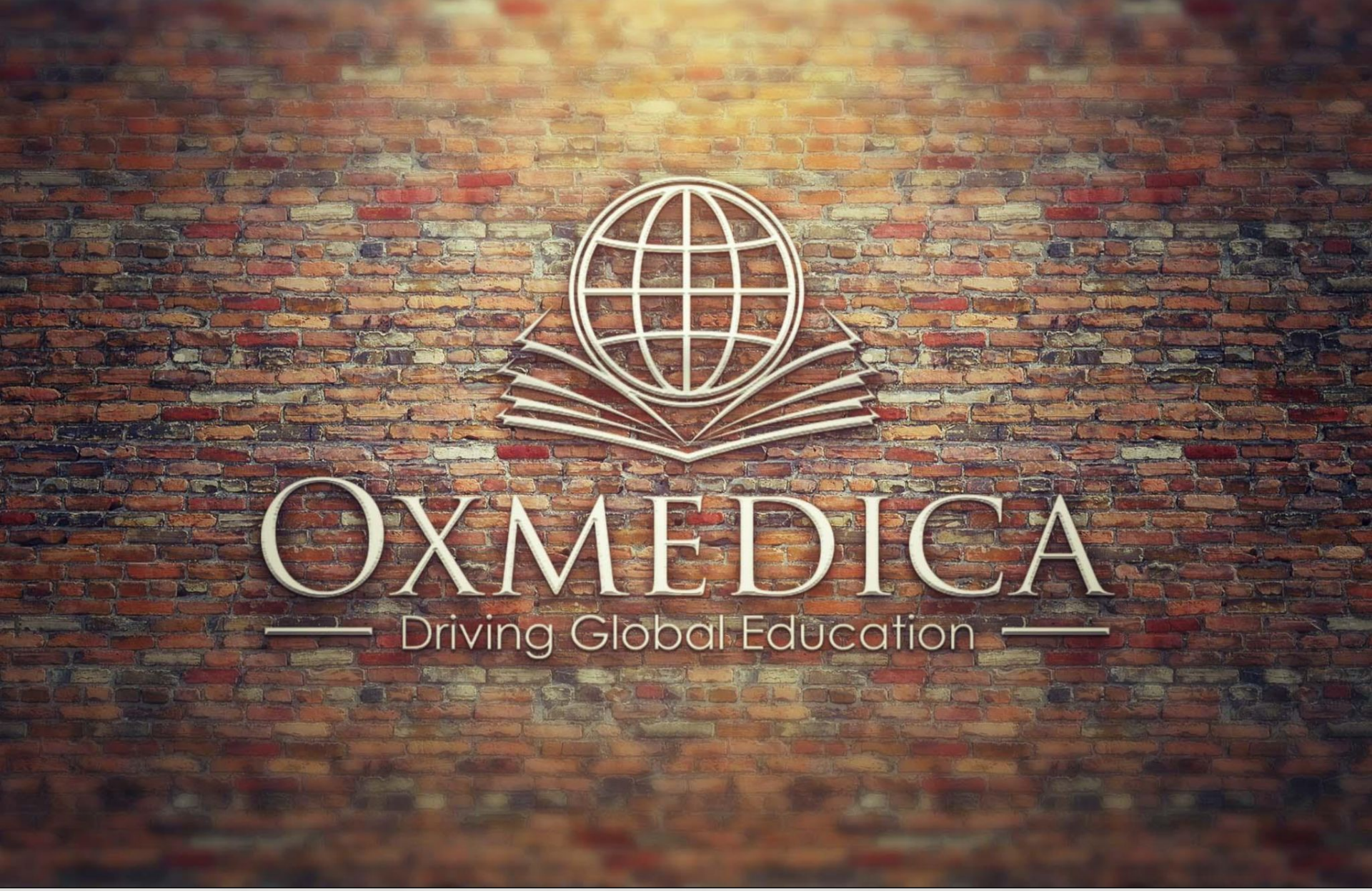


Week 2

Classification Models

Overview

1. Daily recap quiz
2. Classification models vs regression models
3. Tree-based methods
4. Classification and regression trees
5. Random Forest

[Daily Quiz - 15 mins]

Discussion

Classification Models

Two types of problems in supervised learning

Regression Problems: Continuous, numerical prediction problems e.g. house price.

Classification Problems: Predicting a category from a pre-defined and fixed list of categories e.g. sorting images of animals into the categories [‘Dog’, ‘Cat’, ‘Bird’].

Machine Learning AI

\+ Other images of dogs

Dog

Photo source

Photo source

Photo source

Not Dog

Photo source

Dog ✅

Photo source

Not Dog ✅

Photo source

Not Dog ✅

Classical Artificial Intelligence (Not machine learning)

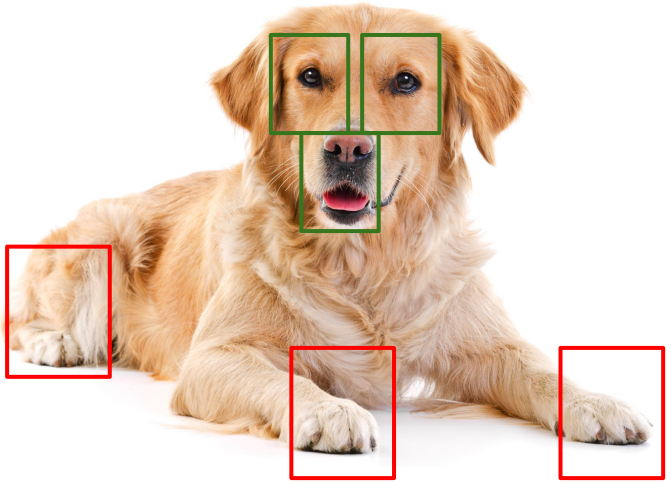


Photo source

Photo source

Dog ✅

Photo source

Not Dog ✅

Photo source

Dog ❌

Examples?

A primer on mathematically representing binary and categorical features

[Aside]

[Aside] binary and categorical features

Binary Features:

Features that can take on two values e.g. Does a student wear glasses? It can be either ‘Yes’ or ‘No’

These are encoded as 0s and 1s

Categorical Features:

Features that can take on a number of predefined values from a fixed list e.g. What is a student’s favourite colour from [‘Red’, ‘Green’, ‘Blue’]

Different ways of handling these features: Label encoding or One-hot encoding

How do categorical targets look mathematically….?

See whiteboard

Classification Models

Main types of model

1. Logistic regression (might cover if time)
2. Tree-based methods: Classification trees
3. Tree-based methods: Random Forest
4. Tree-based methods: Boosting (won't cover)
5. Support vector machines (won't cover)
6. Neural networks (will cover)

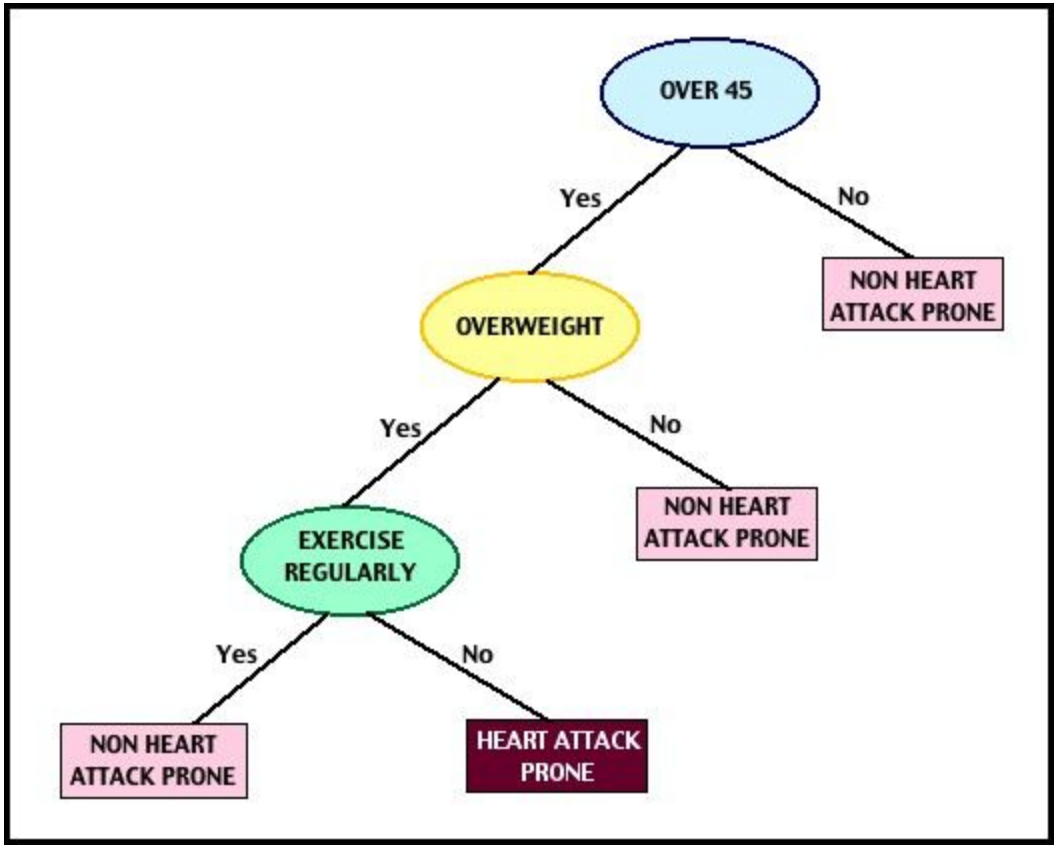
Main types of model

1. Logistic regression (might cover if time)
2. **Tree-based methods: Classification trees**
3. Tree-based methods: Random Forest
4. Tree-based methods: Boosting (won't cover)
5. Support vector machines (won't cover)
6. Neural networks (will cover)

Decision Trees

Source

A simple tree

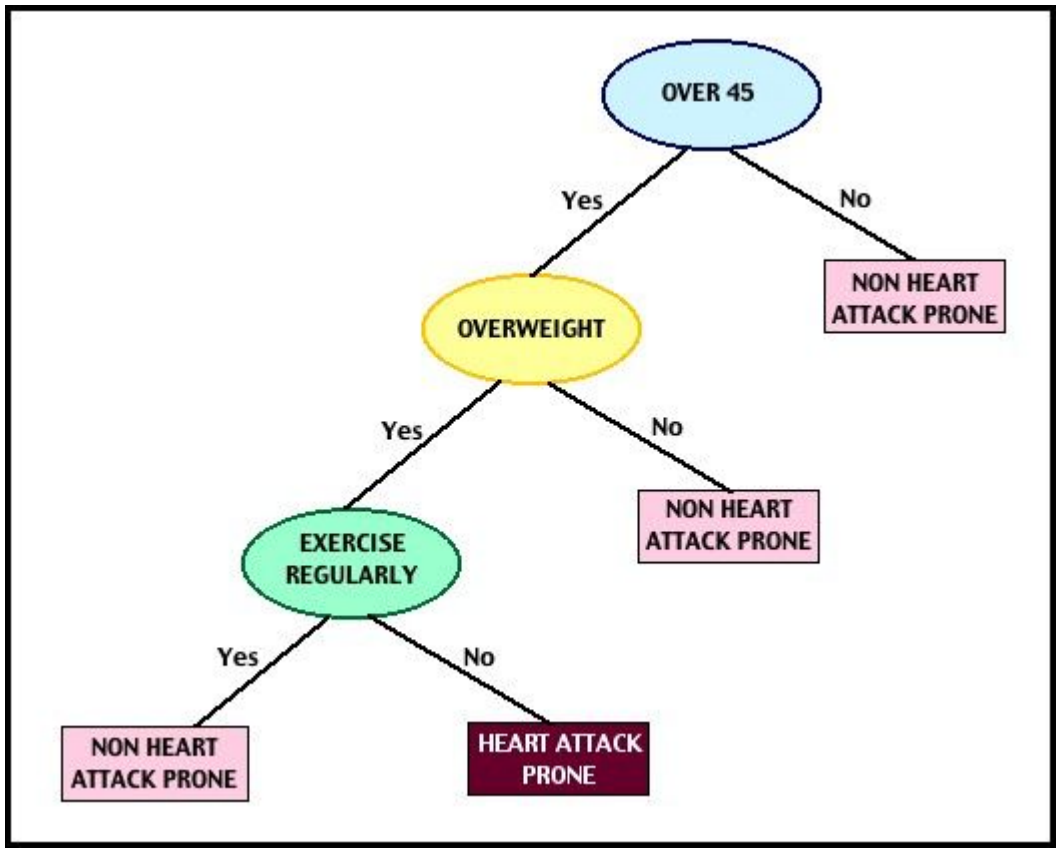


Source

Example: The wages dataset

See whiteboard

Naming different parts of the tree!



- Root
- Branch
- Leaves

Source

Getting predictions

Training Time

1. Build your tree using the training data [more on this later]
2. Work out the most common category/class at each leaf
3. The most common category is the leaf prediction

Test Time [New unseen examples]

1. Pass a new example through the tree to get to a leaf
2. Predict the class of that leaf

Exercise in pairs: 5 mins

Classical Artificial Intelligence Task

1. Think of a classification problem that you would like to implement. Why is this an interesting problem?

2. Pick 5 features that might be relevant for helping us predict the categories.

3. Build a hypothetical tree with different splits. What might you leaf prediction look like

From classical AI to machine learning

How to algorithmically construct classification trees
Recursive binary splitting (RBS)

1. Start with no tree
2. Consider different possible splits for the first node
3. Choose the split the best is best for some metric e.g. loss or ‘information gain’
4. Repeat steps 2-3 *greedily* for all future nodes
5. Stop when you reach a predefined stopping criteria or leaf has 1 training example in it

Recursive binary splitting (RBS)

Greedy Algorithms?

Recursive binary splitting (RBS)

What do we mean by a metric?
[Non-trivial]

- **Estimate of Positive Correctness (true positives - false positives)**
- **Gini impurity**
- **Information gain**
- **Cross entropy**

Read https://www.datacamp.com/tutorial/decision-trees-R# for more info…

Exercise in pairs: 5 mins

Pros and cons of decision trees methods?

Strengths and weaknesses

✅ Easy

✅ Interpretable / explainable

✅ Very fast

❌ Very sensitive to examples / features

❌ Generally individual trees are poor predictors

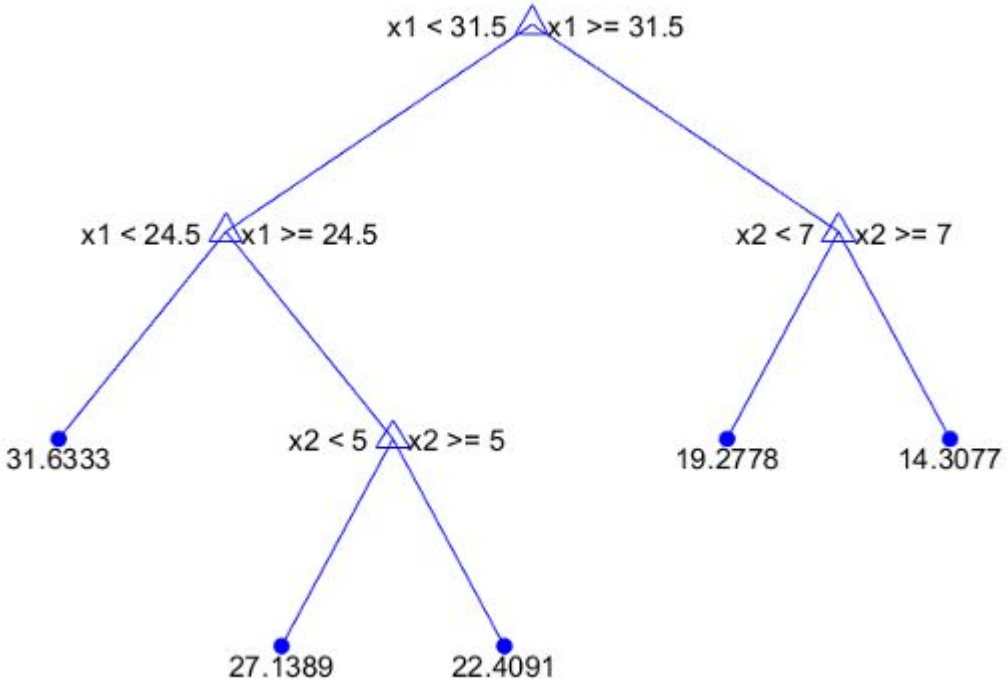
A simple extension to regression problems

How might we extend this to regression problems?

Discussion / whiteboard

How might we extend this to regression problems?

Work out target mean of each leaf rather than most popular class



Random Forest

Source

From individual trees…

… to whole forests

Source

The general idea

Discussion / whiteboard

1. Artificially create ***B*** datasets from your 1 dataset
2. Pick a subset of the features for each tree
3. Train many individual trees
4. Aggregate the results with majority rule or mean average

Questions

1. How do you make the ***B*** artificial datasets?
2. How do you pick the subset of features?
3. How much do you grow each tree?

Strengths and limitations

✅ Final model has lower variance than individual trees

✅ Better predictors

❌ Not very interpretable / explainable!

❌ Can take a long time to train

Recap questions

1. What is the difference between regression and classification problems?
2. What are tree-based models?
3. Explain how classification trees work
4. What makes the recursive binary splitting algorithm **greedy**?
5. How do you extend decision trees to regression problems?
6. What is random forest?