
MACHINE LEARNING

MACHINE LEARNING

- ▶ Machine learning is the study and development of algorithms that can learn from and make predictions on data
- ▶ Machine learning has quite a bit of overlap with statistical modeling, optimization and data mining

MACHINE LEARNING

- ▶ Two broad classes
 - ▶ Unsupervised learning
 - ▶ Supervised learning

MACHINE LEARNING

- ▶ Unsupervised learning is:
 - ▶ having the computer learn general patterns in the data
 - ▶ without focusing on predicting a particular variable
- ▶ The data is unlabeled
- ▶ Typically some form of cluster analysis or partitioning method
- ▶ Meant to identify homogeneous subgroups

MACHINE LEARNING

- ▶ Examples of unsupervised learning
 - ▶ k-means clustering
 - ▶ hierarchical clustering
 - ▶ Self-organizing maps
 - ▶ Mixture models

SUPERVISED LEARNING

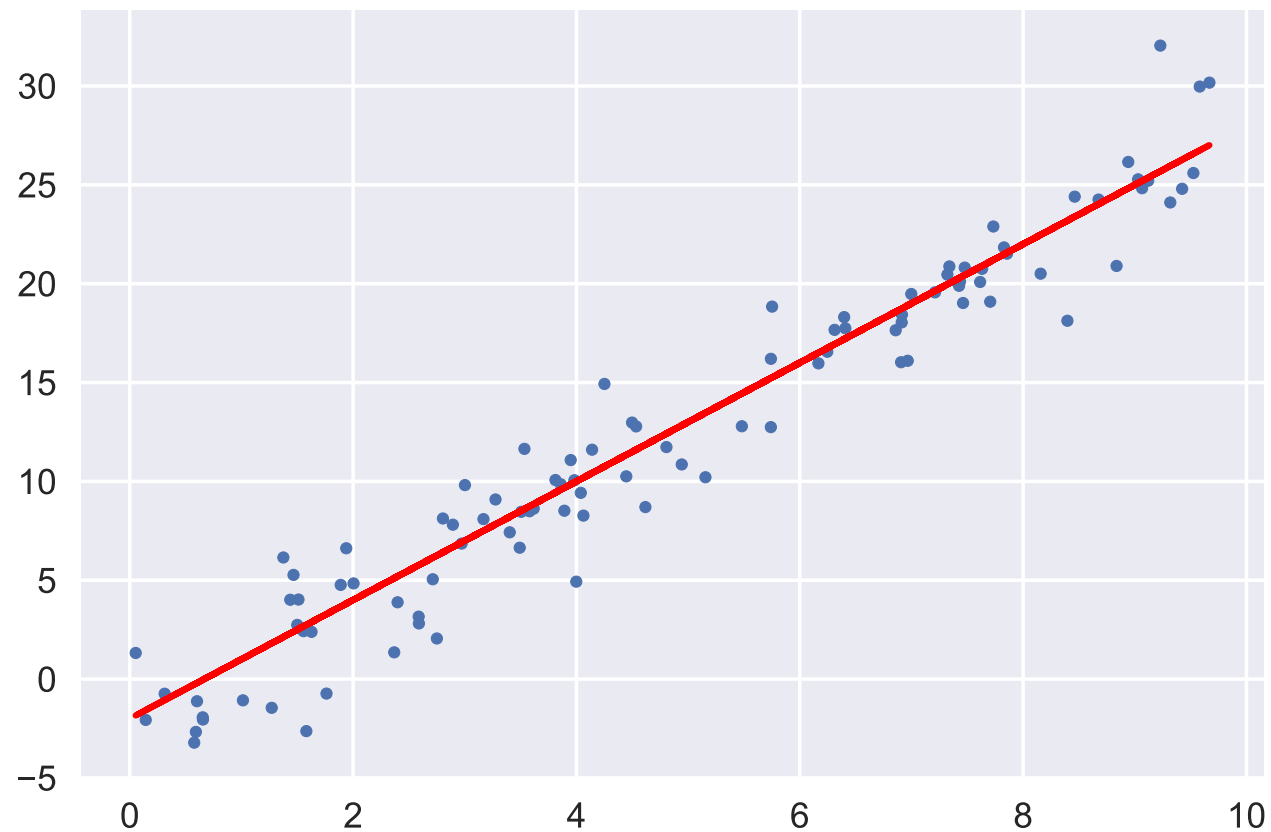
- ▶ One variable is "labelled", in that we know the truth
 - ▶ Class labels (high risk/low risk, tumor/normal)
 - ▶ Continuous labels (income, age)
- ▶ Objective is to try and predict the label from other predictors (called "features" in this literature)

SUPERVISED LEARNING

- ▶ If labels are discrete -> Classification
- ▶ If labels are continuous -> Regression
- ▶ This distinction is somewhat arbitrary
- ▶ We'll see later that you can do better with regression even if the problem is for "classification"

SUPERVISED LEARNING

- ▶ We already know one example of supervised learning
 - ▶ Linear regression



STATISTICS VS MACHINE LEARNING

- ▶ How do we learn to fit a linear regression?
- ▶ We optimize the sum of squared deviations

$$\min_{\alpha, \beta} \sum_{i=1}^n (y_i - \alpha - \beta x_i)^2$$

- ▶ This is a global solution, i.e. we want that line that fits best on average

STATISTICS VS MACHINE LEARNING

- ▶ Machine learning takes a different approach
- ▶ It wants a model that, once trained on a data set, will accurately predict on other data sets
- ▶ This is an important distinction

STATISTICS VS MACHINE LEARNING

- ▶ A model that is fit really well to the data (R^2 is large)
 - ▶ may actually overfit the data
 - ▶ may not work well on new data
- ▶ So our perspective has to change here.

STATISTICS VS MACHINE LEARNING

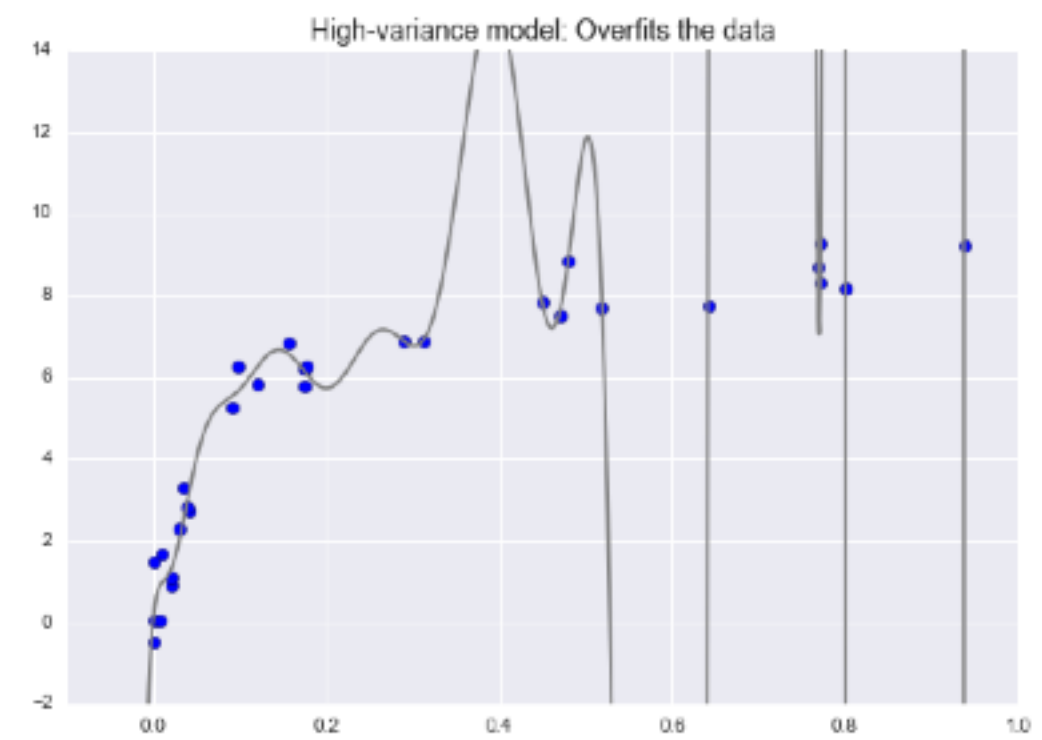
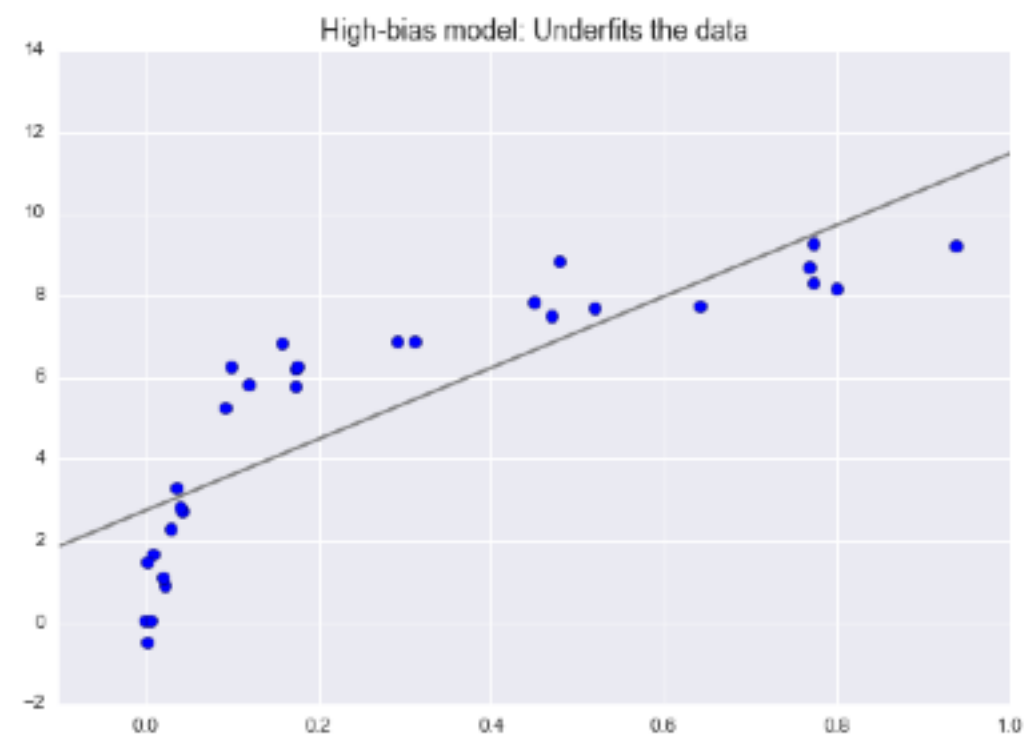
- ▶ The standard way to fit machine learning methods is:
 - ▶ Split your data set into a training set and a test set
 - ▶ Train your data on the training set
 - ▶ See how well it performs (predictively) on the test set
 - ▶ Take that model which does well on the test set

THE VARIANCE-BIAS TRADE-OFF

- ▶ Some models are too simple
 - ▶ They under-fit the data
 - ▶ They do equally well on training and test data
 - ▶ These are *high bias* models

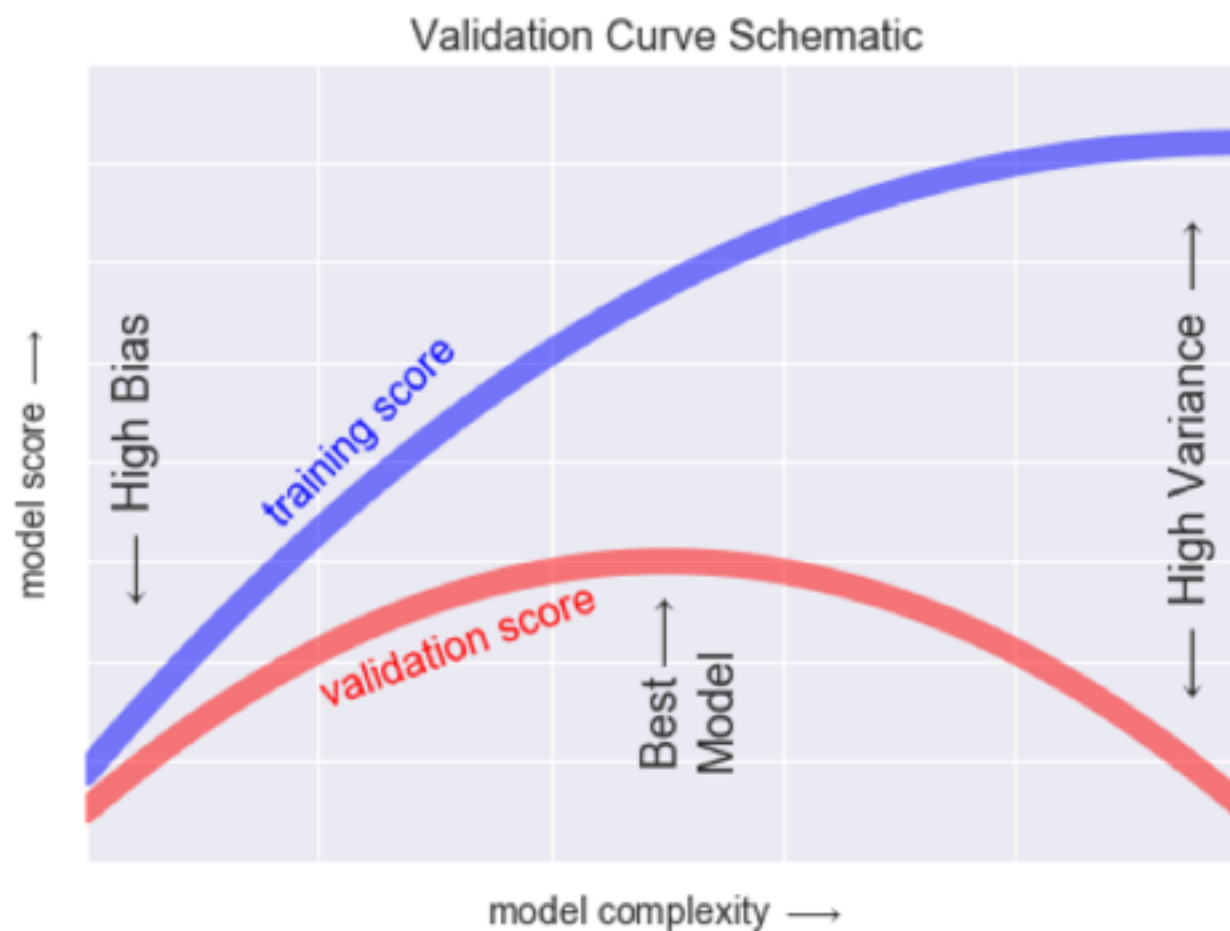
THE VARIANCE-BIAS TRADEOFF

- ▶ Some models are too complex
 - ▶ They fit the data too well (overfitting)
 - ▶ Think interpolation
 - ▶ However they do very poorly on new data
 - ▶ These are termed *high variance* models



THE VARIANCE-BIAS TRADE-OFF

- ▶ We want a model that is low-variance, low-bias



MODEL PERFORMANCE

- ▶ Need a metric to see how well a model works
- ▶ Can't rely on only one test data

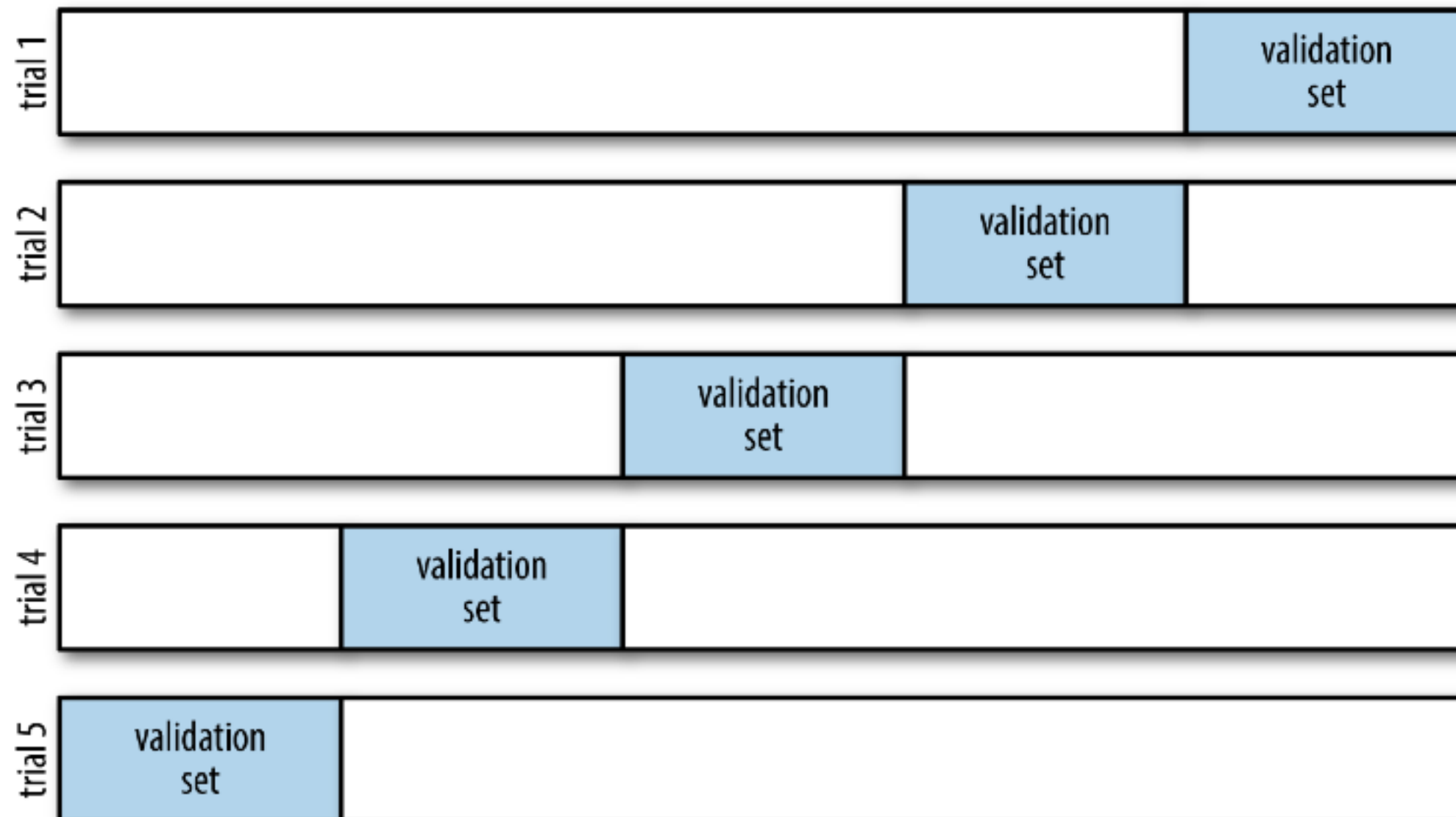
MODEL PERFORMANCE

- ▶ Classification
 - ▶ Accuracy (misclassification)
- ▶ Regression
 - ▶ Mean square error

MODEL PERFORMANCE

- ▶ Take multiple random splits to generate training and test data
- ▶ See overall performance across different splits
- ▶ Idea behind cross-validation

CROSS-VALIDATION



SUPERVISED LEARNING METHODS

- ▶ k-nearest neighbors
- ▶ Decision trees
- ▶ Ensemble methods
 - ▶ Random forests
 - ▶ Boosted trees
- ▶ Support Vector Machines

SUPERVISED LEARNING METHODS

- ▶ These methods often look at local estimates rather than global estimates

K-NEAREST NEIGHBORS

- ▶ For each data point, find its k nearest neighbors in the predictor space
 - ▶ Decide on a distance metric
- ▶ The prediction at that data point is the
 - ▶ average of the labels (regression)
 - ▶ the most prevalent of the labels (classification)
 - ▶ observed in those neighbors

K-NEAREST NEIGHBORS

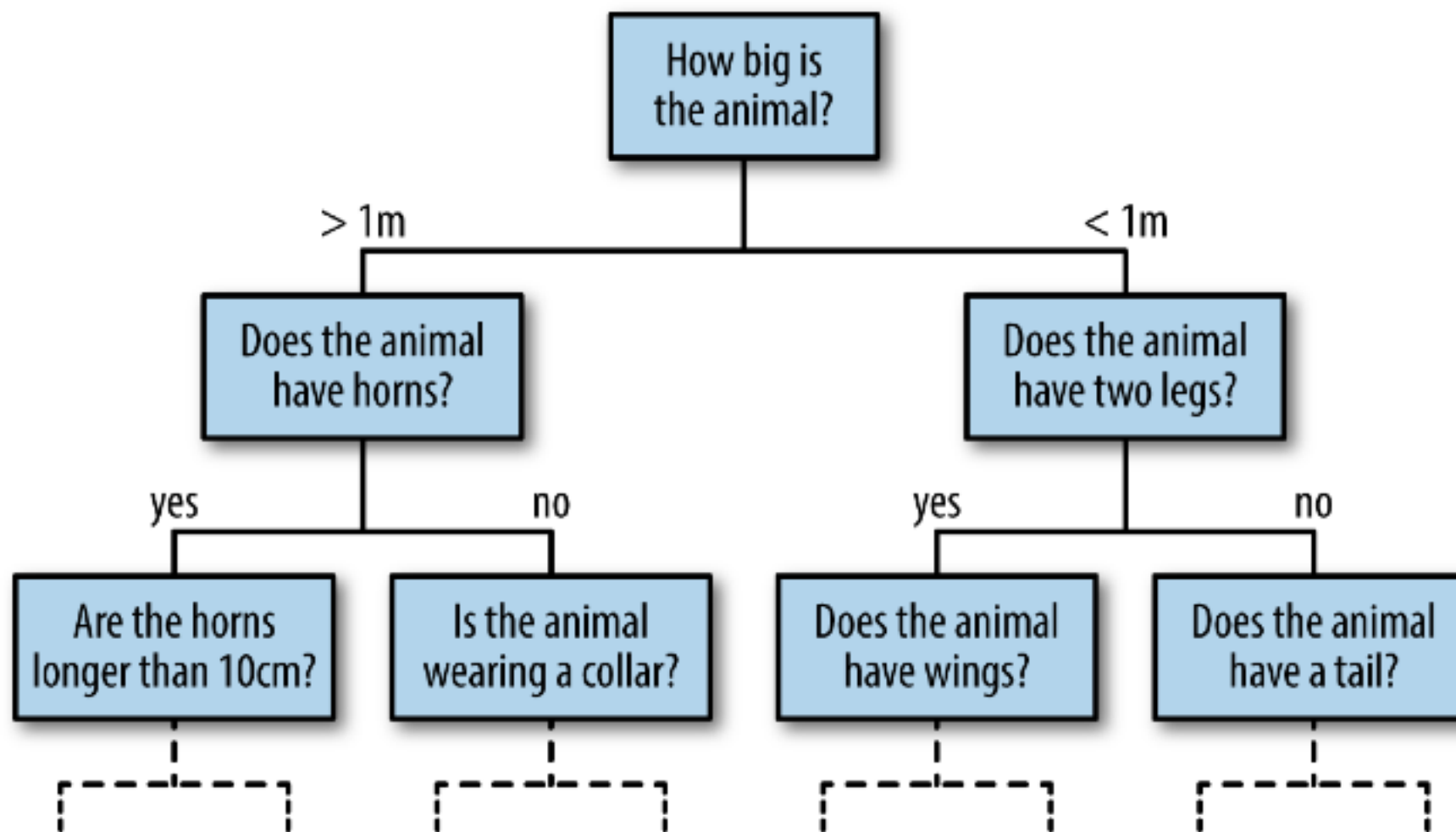
- ▶ This is a very local solution
- ▶ Let's see how to approach this problem.

K-NEAREST NEIGHBORS

- ▶ This is a very greedy algorithm
- ▶ With even moderate data this is untenable
- ▶ There are some advanced methods to try to get around this
- ▶ Not very practical

GOING FORWARD

- ▶ We'll be concentrating on decision trees



GOING FORWARD

- ▶ The other main methods we will look at
 - ▶ Random Forests
 - ▶ Boosting
 - ▶ XGBoost
- ▶ have decision trees as the base learner

THE SCIKIT-LEARN PACKAGE

- ▶ This is the workhorse for machine learning
- ▶ We will use this extensively in fitting machine learning models to data

THE WAY FORWARD

- ▶ Introduce a machine learning model
- ▶ Create a heuristic idea about how the model works
- ▶ Implement the heuristic idea in Python
- ▶ See how it is implemented in scikit-learn
- ▶ Investigate how to optimize the model
 - ▶ Tuning parameters
- ▶ Examples

DATA

- ▶ I'll be bringing toy datasets to class
- ▶ If you want, you can bring your own data
- ▶ We'll have time each day for exercises and laboratory work
- ▶ You can work on your own datasets if you like
- ▶ We'll learn together.