G51FAI
Fundamentals of AI

Instructor: Siang Yew Chong

Introduction to Machine Learning

Outlines

- Overview
- Machine Learning Framework
  - steps & processes
  - training vs. test set
- Learning Tasks
  - Supervised & unsupervised
- Training Issues
  - generalisation
  - measuring model quality
  - cross-validation
- Datasets & Software

What is Machine Learning?

"Machine learning refers to a system capable of the autonomous acquisition and integration of knowledge"

What is Learning?

"Learning denotes changes in a system that ... enable a system to do the same task … more efficiently the next time"

-- Herbert Simon

"Learning is constructing or modifying representations of what is being experienced"

-- Ryszard Michalski

"A computer program is said to *learn* from *experience E* with respect to some class of *tasks T* and *performance measure P* if its performance at tasks in T, as measured by P, improves with experience E"

-- Tom Mitchell

Machine Learning

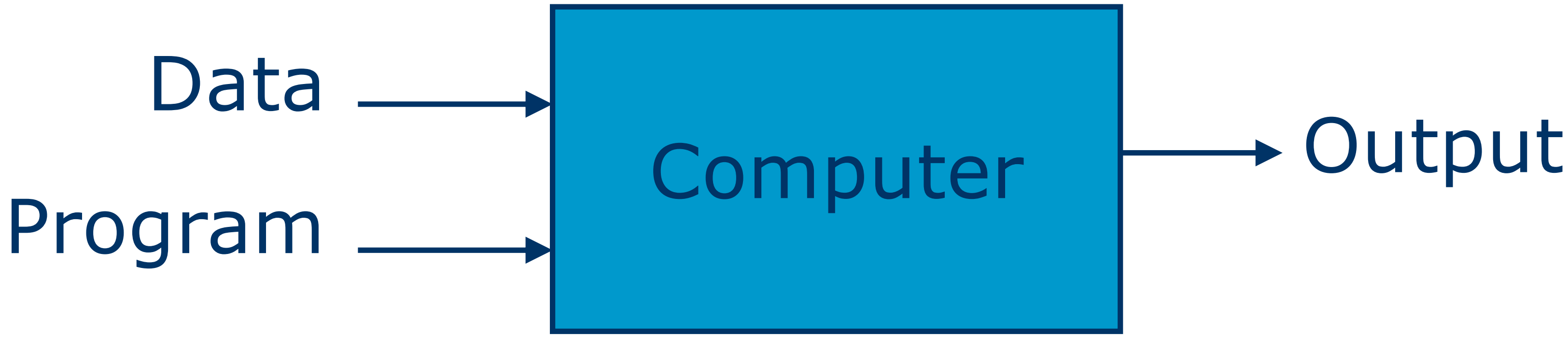
- The world is driven by data
  - Google processes 24 petabytes per day
  - data in 2013 > all data in history
  - Powerful, cheaper computers, cloud storage

- Many applications are hard to program directly but most are “pattern recognition” tasks (e.g. targeted advertising, reading handwriting)

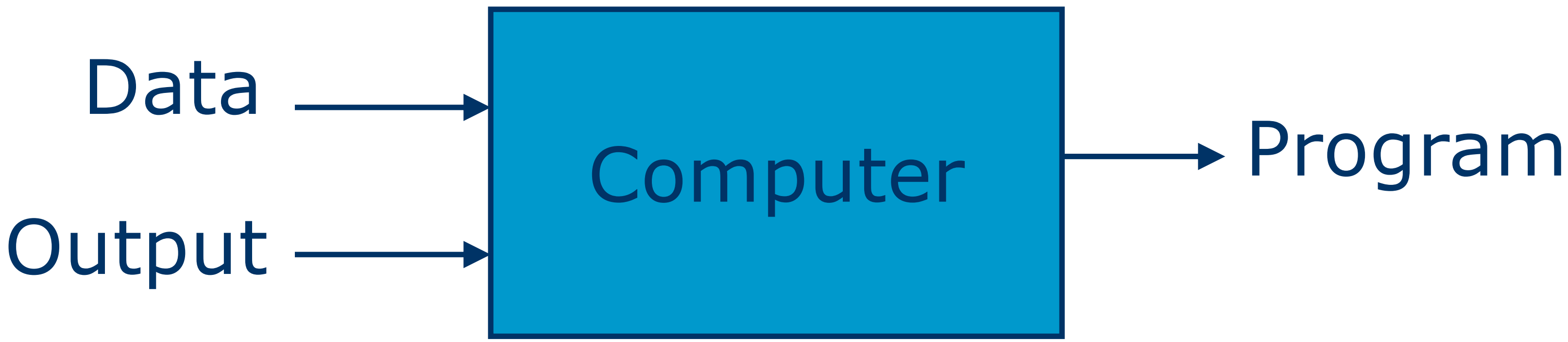
- Machine learning
  - collect lots of “training” data (examples) that specify the correct output for a given set of inputs
  - a machine learning algorithm then takes these examples and produces a program that does the job (*learning from examples)*

Machine Learning vs. Traditional Programming

- *Traditional Programming*



- *Machine Learning*



Tom Mitchell

Machine Learning Problems

- Amount of knowledge might be too large for explicit encoding by humans
- Human expertise may be scarce or very costly
  - navigating on Mars, drug design, astronomic discovery
- Black box human expertise that cannot be explained, and functional relationships cannot be expressed mathematically
  - speech/face recognition, driving a car, flying a plane
  - else we would just code the algorithm
- Rapidly changing phenomena
  - credit scoring, financial modeling, fraud detection
- Need for customisation/personalisation
  - biometrics, movie/book recommendation
- Often only data from measurements are available

How Machine Learning Works

Training

Training set may be noisy, e.g., $(\boldsymbol{x}, (f(\boldsymbol{x}) + \varepsilon))$

Training Images

Image Features → $\mathbf{x}$ → Training → Learned Model $f()$

Training Labels → y → Training

Training: given a *training set* of labeled examples $\{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N)\}$, the function f is estimated by minimising the prediction error on the training set

Testing: apply f to a never seen before *test example* $\mathbf{x}$ and output the predicted value $y = f(\mathbf{x})$

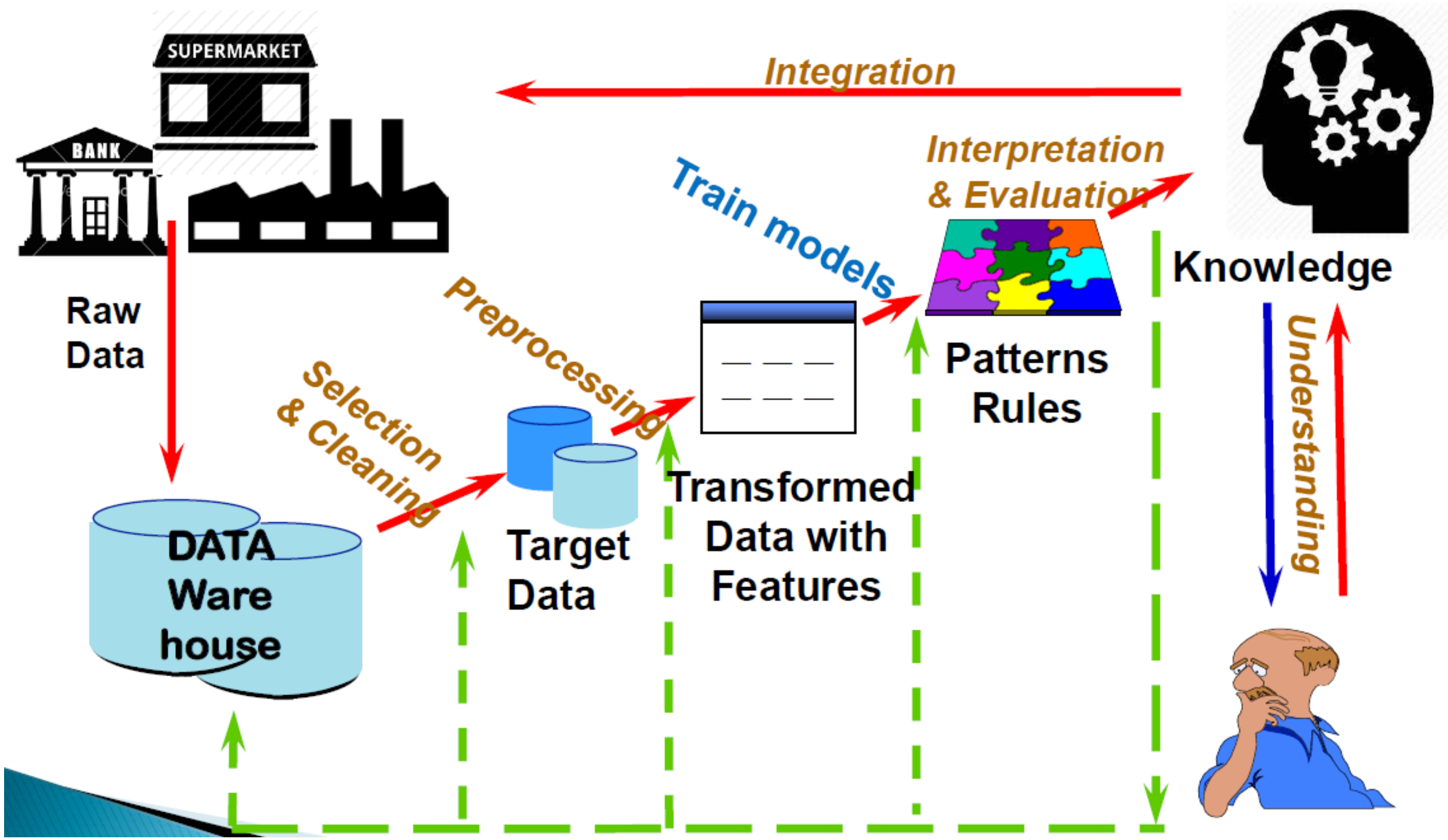
Testing

Test Image

Image Features → $\mathbf{x}$ → Learned Model $f(\mathbf{x})$ → Prediction y

Slide credit: D. Hoiem and L. Lazebnik

Machine Learning Process



Machine Learning Tasks

- **Supervised:** given input samples (**x**) and labeled outputs (y) of a function y = f(**x**), "*learn*" f, and evaluate it on new data
  - ***Classification***: y is discrete (class labels). Learn a decision boundary that separates one class from another
  - ***Regression***: y is continuous, e.g. linear regression. Learn a continuous input-output mapping, also known as "*curve fitting*" and "*function approximation*"

- Examples:
  - is this image a cat, dog, car, house?
  - how would this user score that restaurant?
  - what will be the sales, stock price next year?

Machine Learning Tasks

- **Unsupervised:** given only samples **x** of the data, infers a function f such that y = f(**x**) describes the hidden structure of the unlabeled data - more of an exploratory/descriptive data analysis
  - ***Clustering***: y is discrete. Learn any intrinsic structure that is present in the data
  - ***Dimensional Reduction:*** y is continuous. Discover a lower-dimensional surface on which the data lives

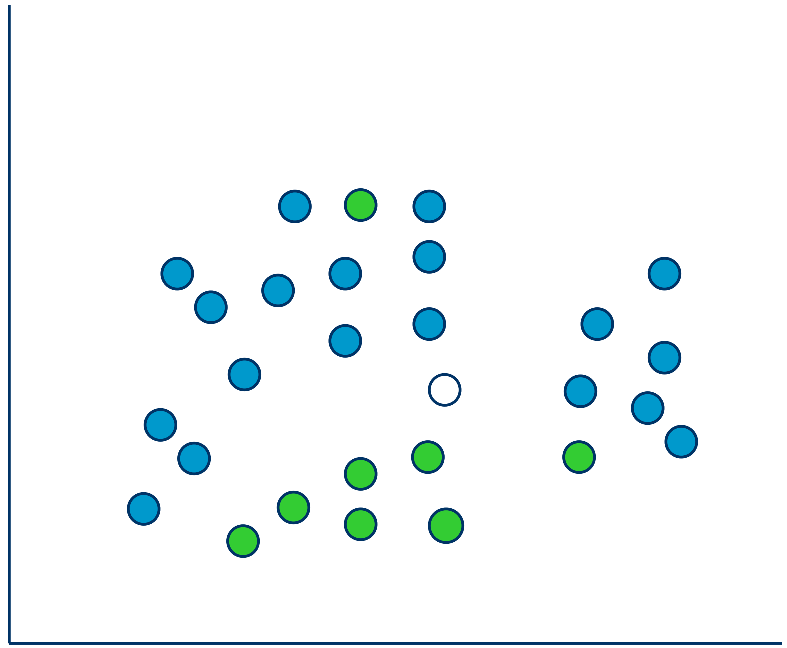
- Examples:
  - cluster some hand-written digit data into 10 classes
  - what are the top 20 topics in Twitter right now?
  - discover interesting relations between variables in large databases

Supervised vs. Unsupervised Learning

Supervised	Un-supervised
y = F(x): function	**y = ?**: no function
D: labeled training set	**D**: unlabeled data set
Learn: G(x): model trained to predict labels of new cases	**Learn:** ?
Goal: $E[(F(x)-G(x))^2] \approx 0$	**Goal:** ?

Classification

Learn a method for predicting the instance class from pre-labeled (classified) instances



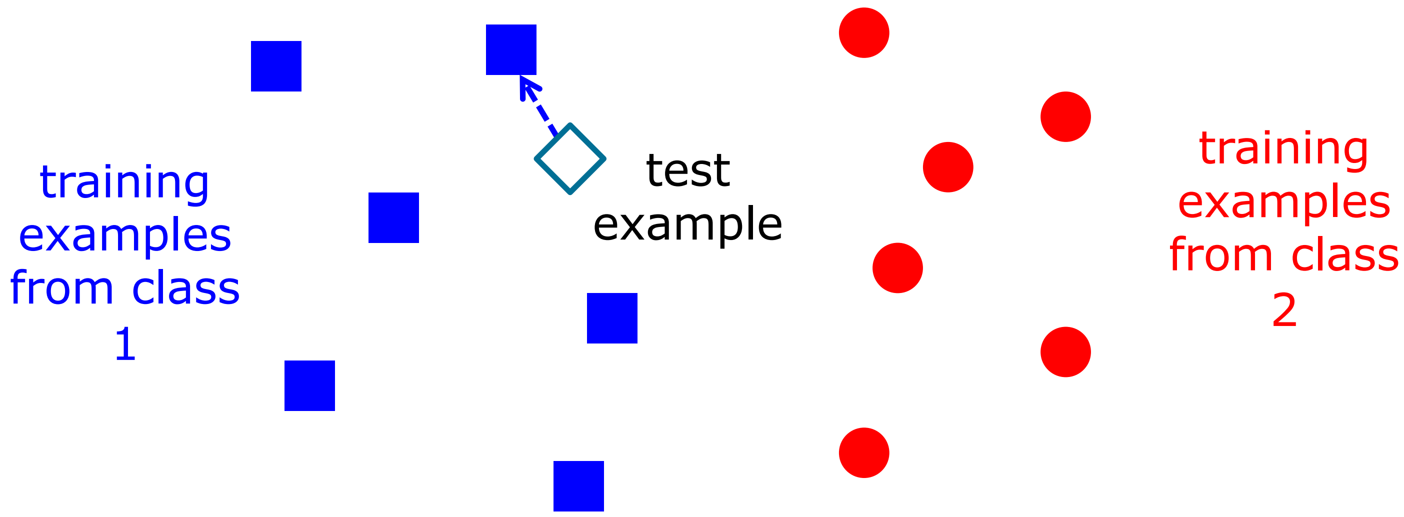
Many approaches:

Nearest Neighbour,
Regression,
Decision Trees,
Bayesian,
Neural Networks,
...

Given a set of points from classes ● ●

what is the class of new point ○ ?

Classifiers: Nearest Neighbour



- The test example **x** will be classified as belonging to the same class as f($\mathbf{x}_1$), i.e. label of the training example nearest to **x**
  - all we need is a distance function for the inputs
  - no training required!
  - also known as instance-based learning

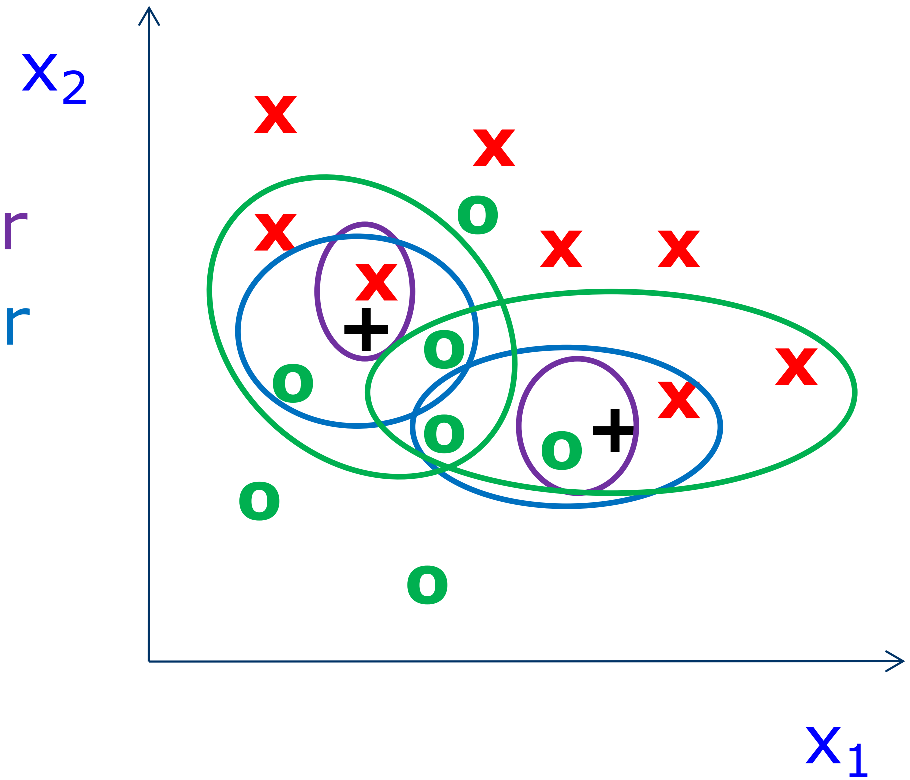
Slide credit: L. Lazebnik

k-Nearest Neighbour (kNN)

1-nearest neighbour

3-nearest neighbour

5-nearest neighbor



Binary-class (o-x)

Query or new test point (+)

k-NN Issues

- *The Data is the Model*
  - no training needed
  - accuracy generally improves with more data
  - matching is simple and fast (and single pass)
  - usually need data in memory, but can be run off disk

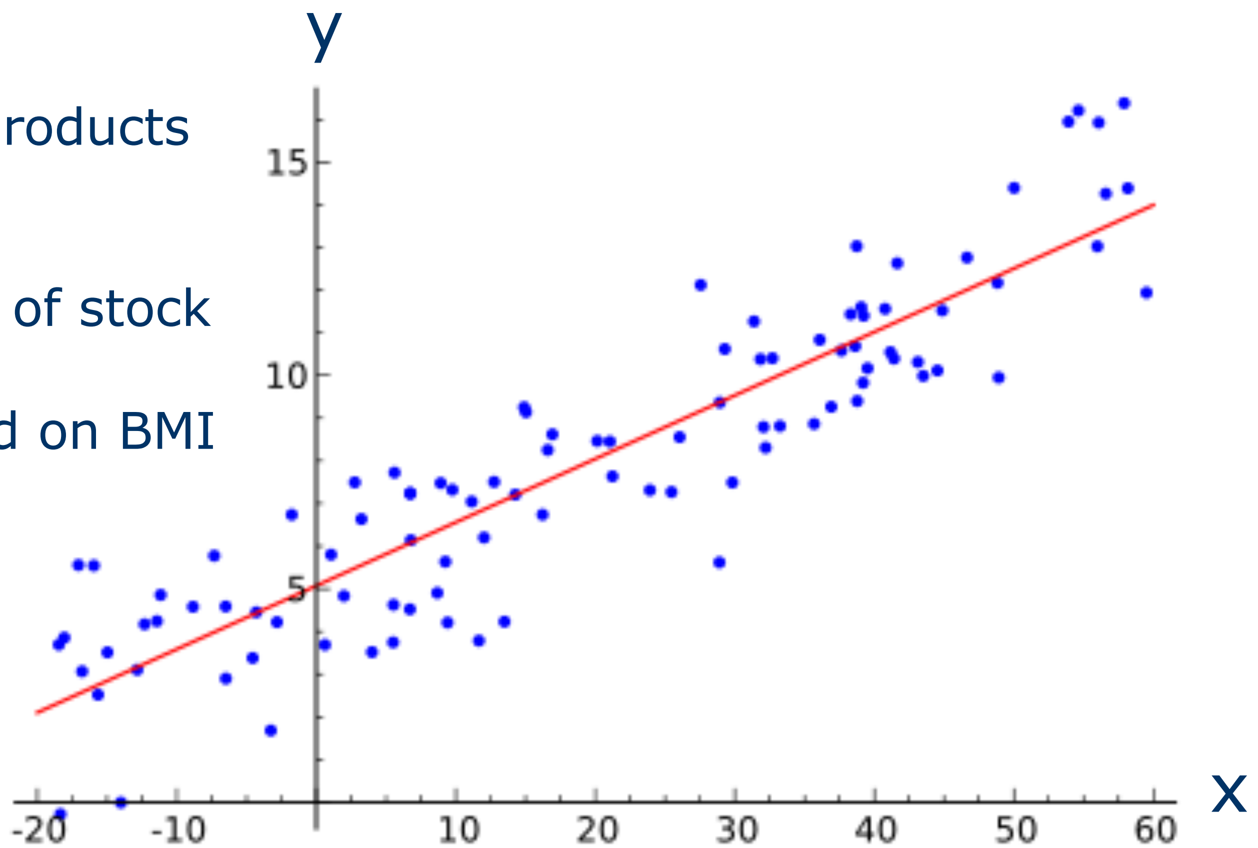
- *Minimal configuration, only* parameter is k (number of neighbours)

- Two other choices are important:
  - weighting of neighbours (e.g. inverse distance)
  - similarity metric

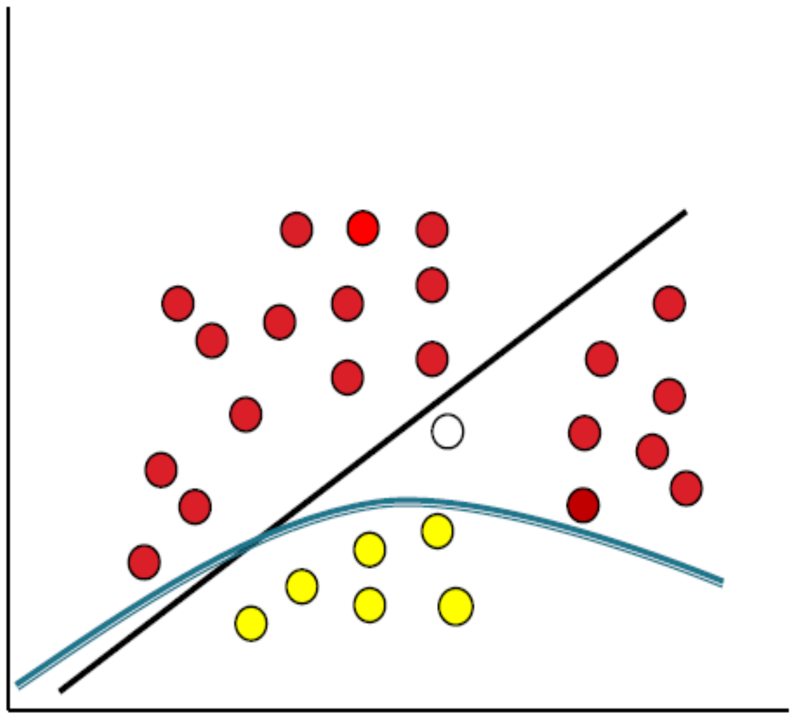
Regression

- To find the best line (linear function y=f(x)) to explain the data
  - assuming a linear or nonlinear model of dependency

❖ predict sales of new products based on advertising expenditure
❖ time series prediction of stock market indices
❖ estimate weight based on BMI

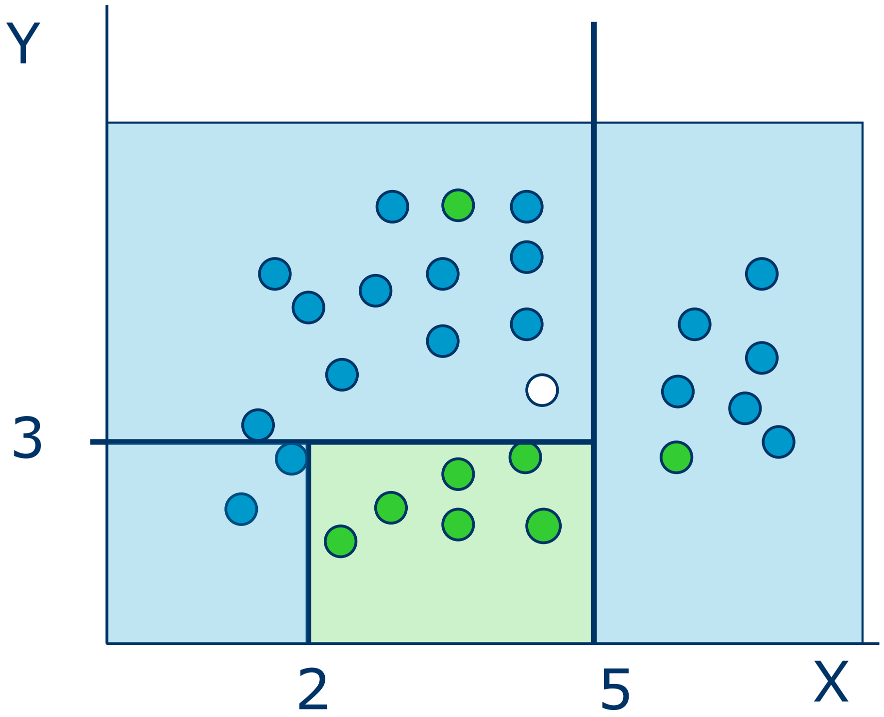


Regression



- Linear Regression
  - $w_0 + w_1 x + w_2 y >= 0$
- Regression computes w_i from data to minimise squared error to ‘fit’ the data
- Not flexible enough

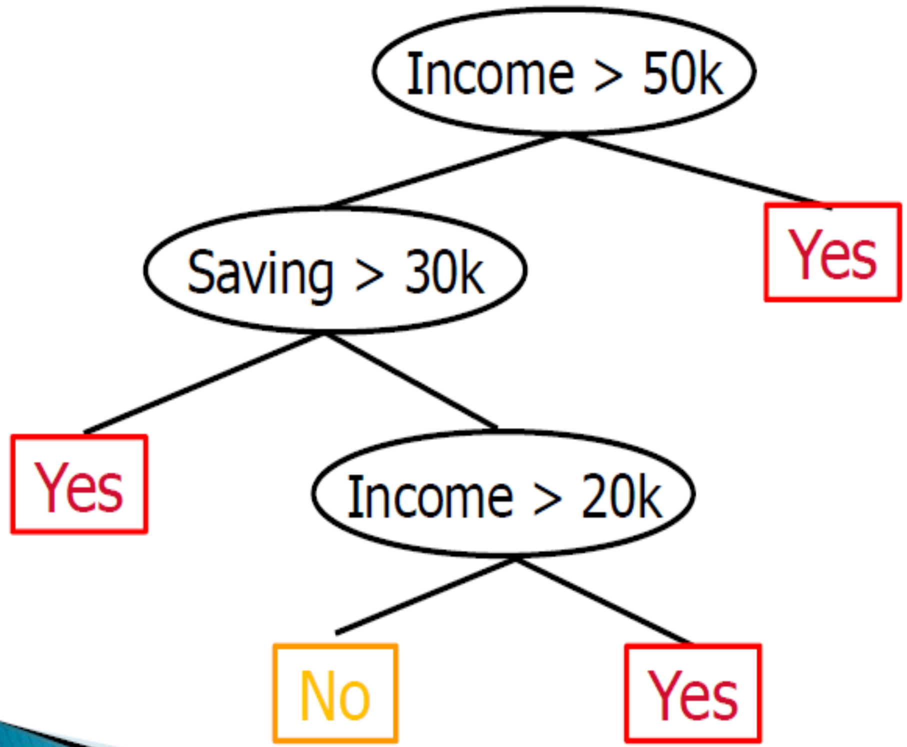
Classification: Decision Trees

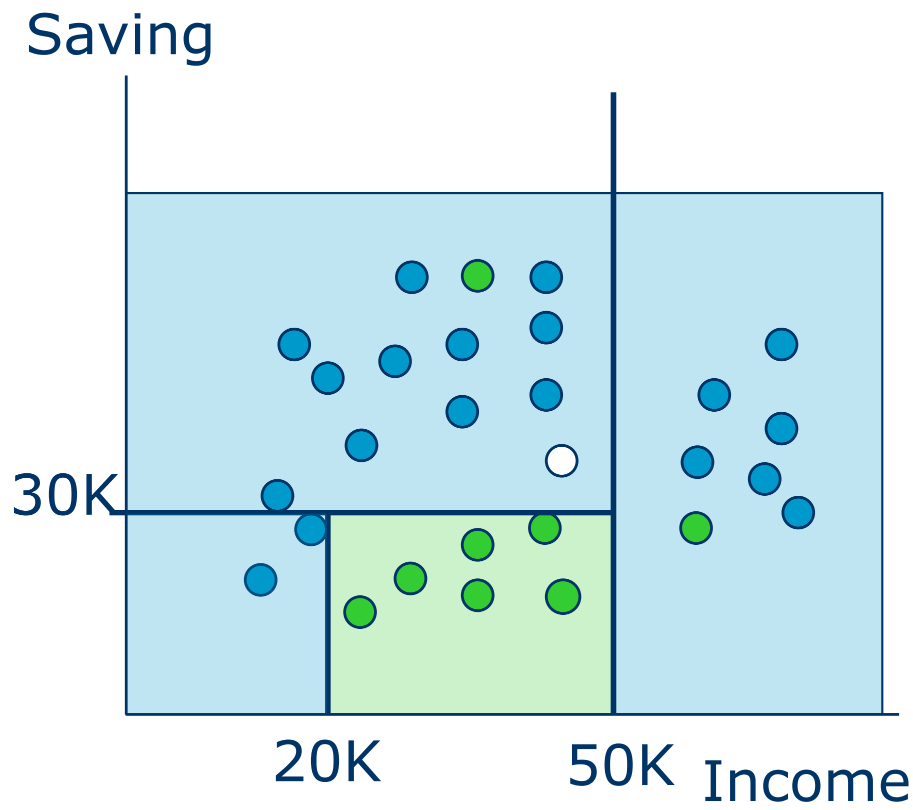


if X > 5 then blue
else if Y > 3 then blue
else if X > 2 then green
else blue

Classification: Decision Trees

- **Internal node:** decision rule on one or more attributes
- **Leaf node:** a predicted class label

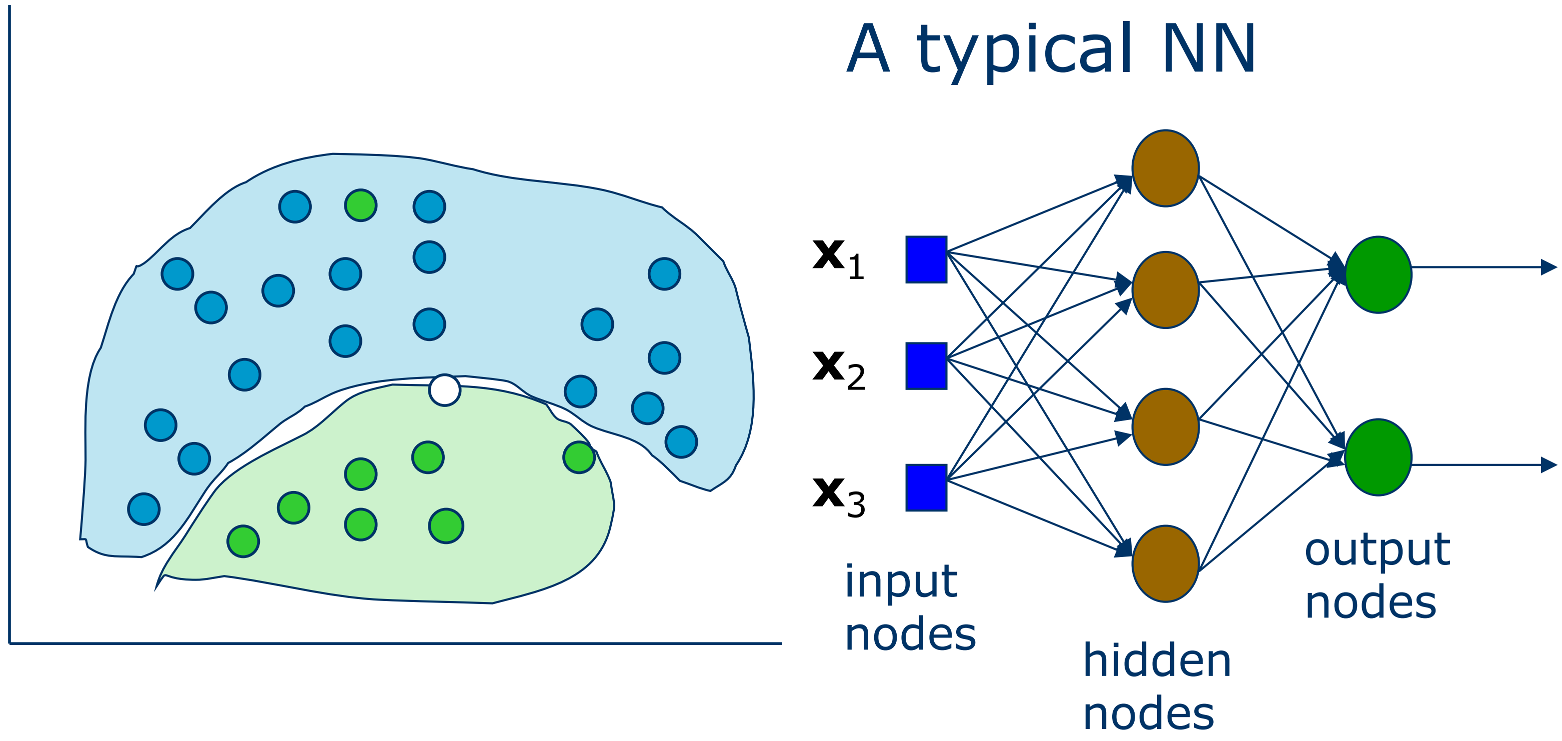




Classification: Decision Trees

Pros	Cons
Reasonable training time	Simple decision boundaries
Can handle large number of attributes	Problems with lots of missing data
Easy to implement	Cannot handle complicated relationship between
Easy to interpret	

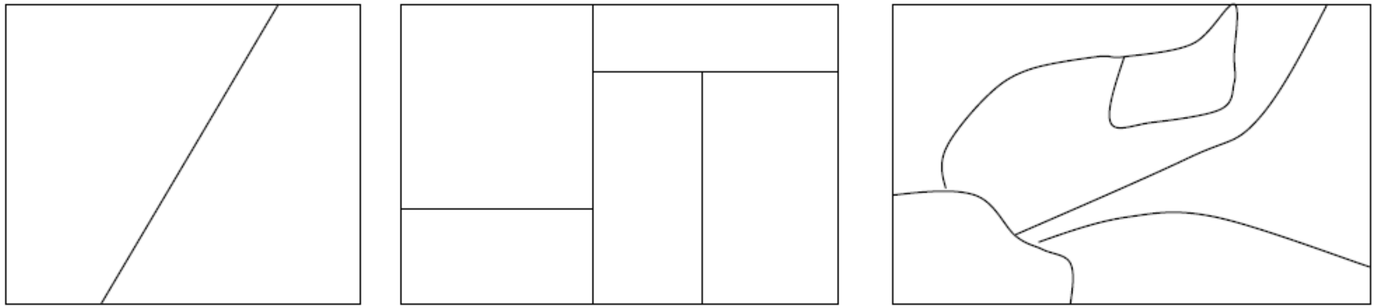
Classification: Neural Networks



Classification: Neural Networks

- Useful for learning complex data like speech, image and handwriting recognition

Decision boundaries:



Linear regression **Decision tree** **Neural network**

- ❖ Regression: use of linear or any other polynomial
- ❖ Decision Trees: divide decision space into piecewise regions
- ❖ Neural Networks: partition by nonlinear boundaries

Classification: Neural Networks

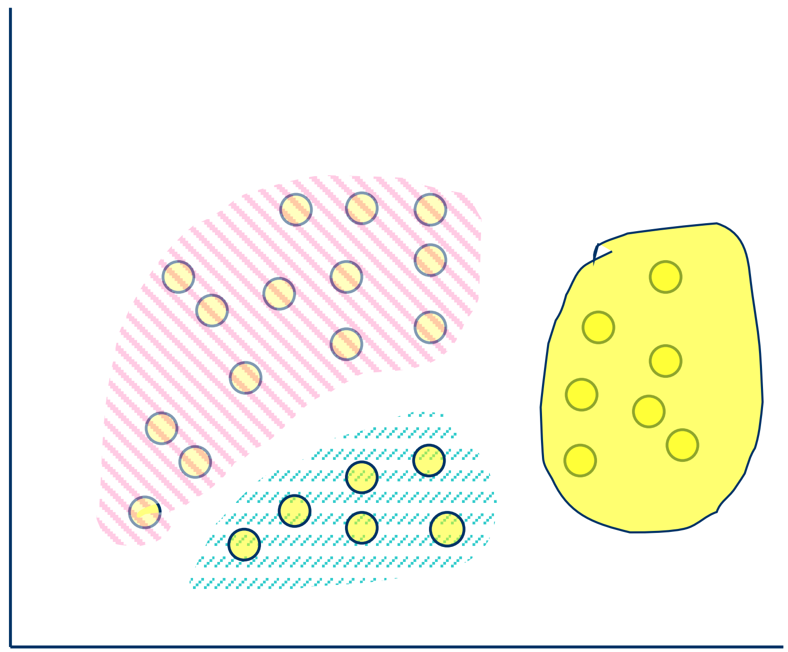
Pros	Cons
Can learn more complicated class boundaries	Hard to implement: trial and error for choosing parameters and network structure
Can be more accurate	Slow training time
Can handle large number of features	Can over-fit the data: find patterns in random noise
	Hard to interpret

Classification: Applications

- Banking: loan/credit card approval
  - predict good customers based on old customers
- Fraud detection: financial transactions
  - use historical data to build models of fraudulent behavior and use data mining to help identify similar instances
- Customer relationship management (CRM)
  - Which of my customers are likely the most loyal
  - Which are most likely to leave for a competitor?
  - Identify likely responders to sales promotions

Clustering

- What we have
  - a set of un-labeled data points, each with a set of attributes
  - a similarity measure
- What we need
  - find "natural" partitioning of data, or groups of similar/close items



- Key: measure of similarity between instances
  - Euclidean or Manhattan distance
  - Hamming distance
  - other problem specific measures

Clustering Applications

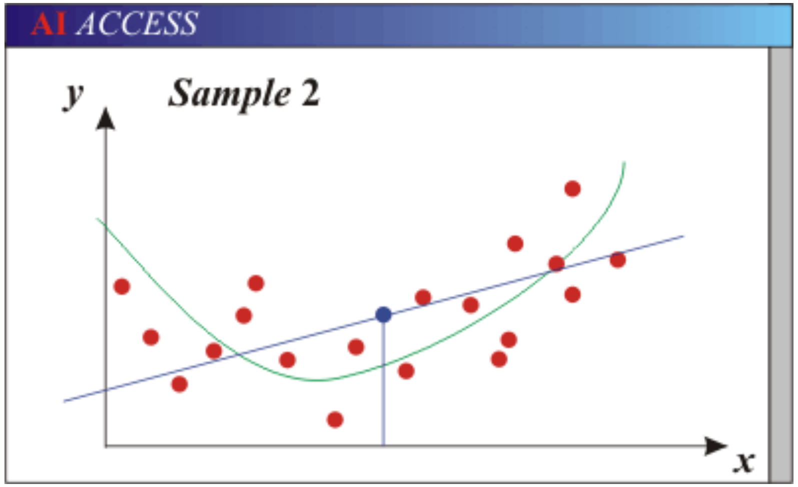
- Market Segmentation
  - Goal: divide a market into distinct subsets of customers, any subset may be a target market
  - Approach:
    - ❖ collect different attributes of customers, based on their related information (lifestyle etc.)
    - ❖ find clusters of similar customers
    - ❖ evaluate buying patterns in the same cluster vs. those from other clusters
- Supermarket Shelf Management
  - Goal: identify items bought together by customers
  - Approach:
    - ❖ process data collected with barcode scanners
    - ❖ find dependencies among items
  - A classic rule:
    - ❖ if a customer buys diaper & milk, then he is very likely to buy beer
    - ❖ friday afternoon, men between 25 and 35 years-old use to buy both products …
    - ❖ don’t be surprised if six-packs next to diapers!

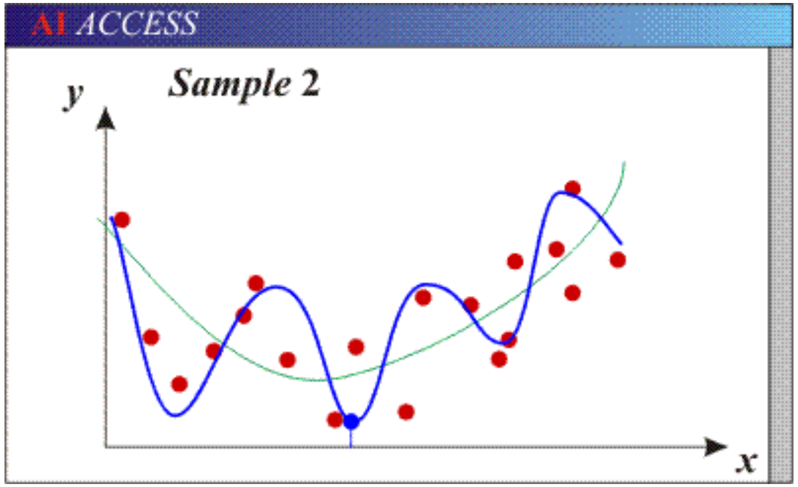
Generalisation

- How well does a learned model generalise from the data it was trained on to a new test set?
- Components of generalisation error
  - inherent: unavoidable
  - **bias:** how much the average model over all training sets differ from the true model?
    - error due to inaccurate assumptions/simplifications made by the model
  - **variance:** how much models estimated from different training sets differ from each other
- **Underfitting:** model is too "simple" to represent all the relevant class characteristics
  - high bias and low variance
  - high training error and high test error
- **Overfitting:** model is too "complex" and fits irrelevant characteristics (noise) in the data
  - low bias and high variance
  - low training error and high test error

Slide credit: L. Lazebnik

Bias-Variance Trade-off

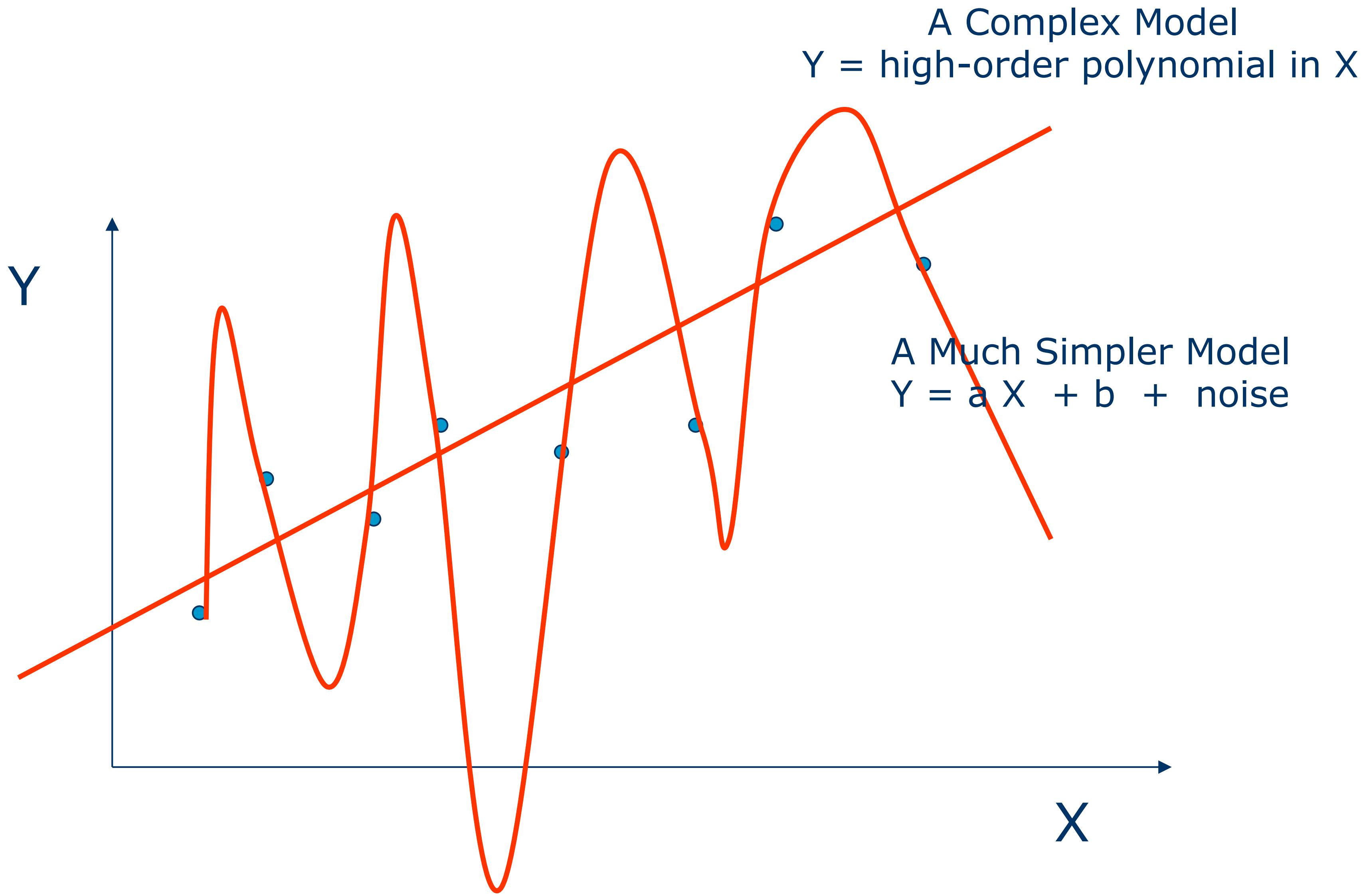




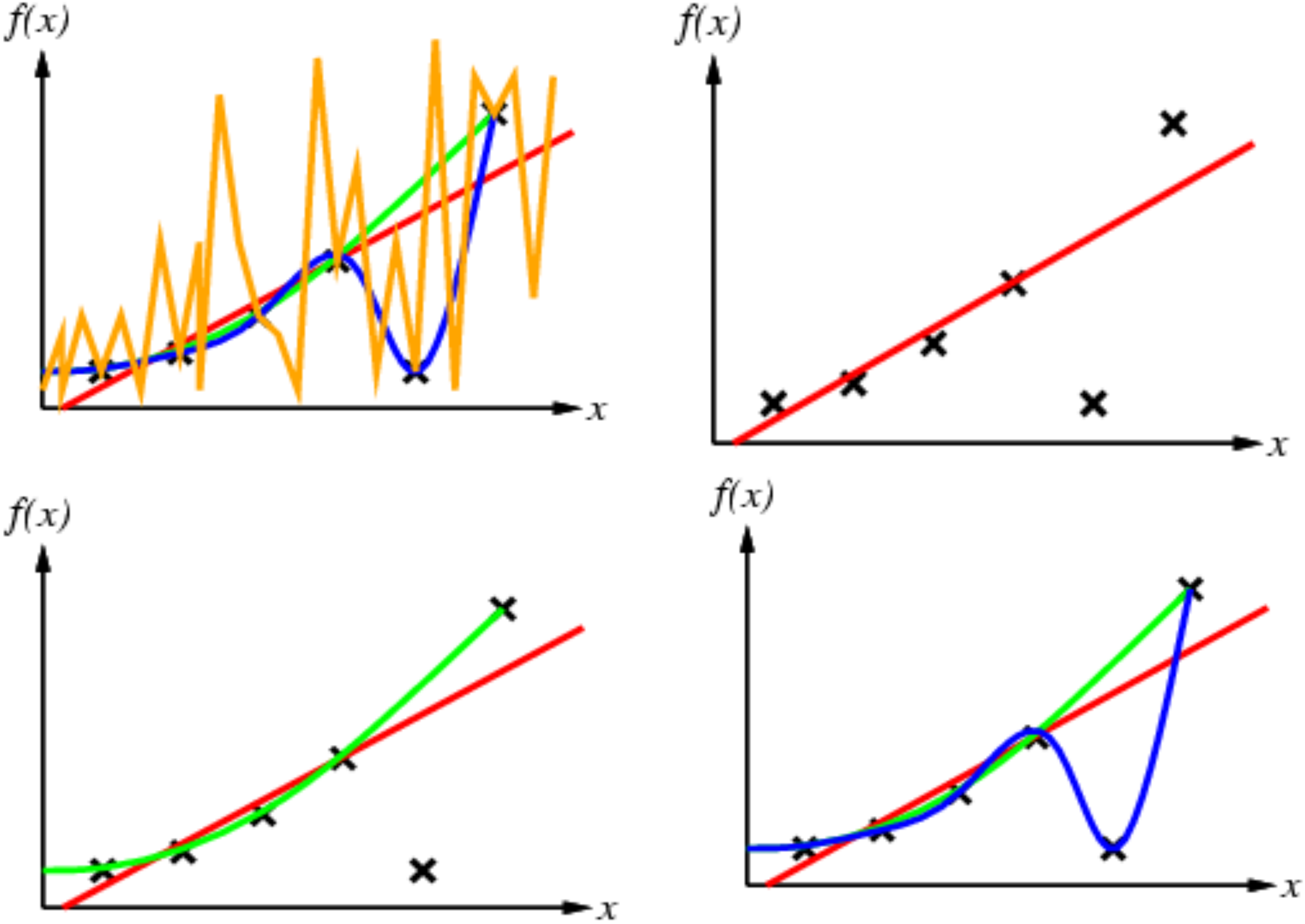
- Models with too few parameters are inaccurate because of a large bias (not enough flexibility)
- Models with too many parameters are inaccurate because of a large variance (too much sensitivity to the sample)

Slide credit: D. Hoiem

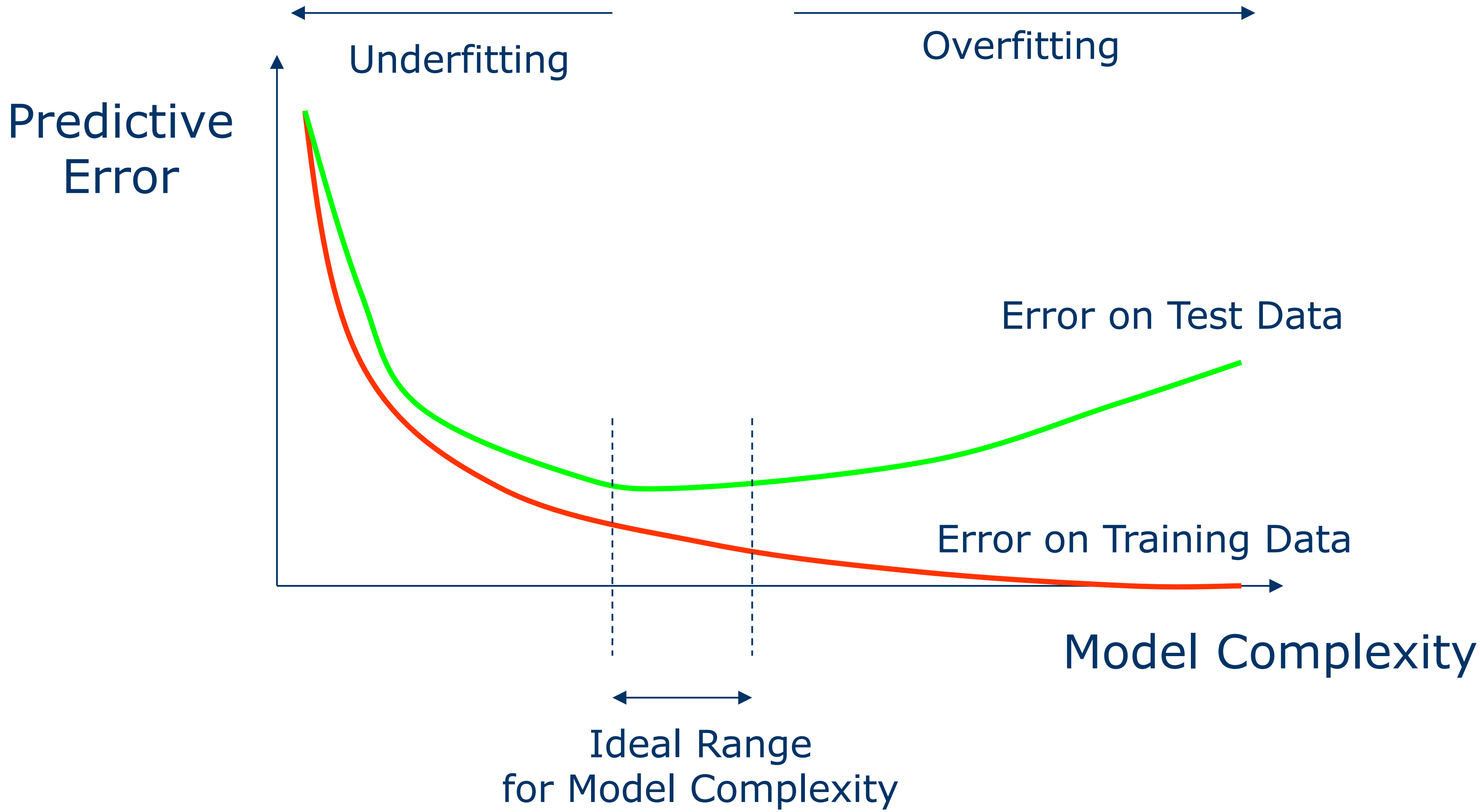
Overfitting & Underfitting



Overfitting & Underfitting



How Overfitting Affects Prediction



The Holdout Method

- Randomly split examples into *training set U* and *test set V*
- Use training set to learn a hypothesis *H*
- Measure % of *V* correctly classified by *H*
- The hold-out method splits the data into training data and test data (e.g 90/10 split)

- *Repeated holdout method* repeats the process with different subsamples
  - in each iteration, a certain proportion is randomly selected for training
  - the error rates on the different iterations are averaged to yield an overall error rate

The v-fold Cross-Validation Method

- v-fold Cross-Validation (e.g., v=10)
  - randomly partition our full data set into v disjoint subsets
  - simplest approach is each subset is roughly of size n/v, n = total number of data points
  - subsets are labelled i = 1,2,3,…,v
  - standard approach
    - for i = 1:v
      - ✓ train on the other of (v-1) subsets
      - ✓ Acc(i) = accuracy on held-out subset i
    - end
    - Cross-Validation-Accuracy = 1/v Σ_i Acc(i)
  - choose the method with the highest cross-validation accuracy
  - can also do “leave-one-out” where v = n

Datasets and Software

- UCI Machine Learning Repository
- KDnugget
- Datasets for DM at University of Edinburgh

- Training

Summary

- Overview
- Machine Learning Framework
  - steps & processes
  - training vs. test set
- Learning Tasks
  - Supervised & unsupervised
- Training Issues
  - generalisation
  - measuring model quality
  - cross-validation
- Datasets & Software

Acknowledgements

Most of the lecture slides are adapted from the same module taught in Nottingham UK

by

Dr. Rong Qu

and

other slides which are credited individually