Foundation of ML

Week 1

List of contents

Introductory overview of:

- Real-world applications of Machine Learning (ML)
- Definitions of ML
- Types of ML
    - Unsupervised learning
    - Supervised learning
        - Linear models
        - Nonlinear models
- Model assessment and selection

Real-world applications of machine learning

- ML involves “computer algorithms” that learn how to perform different tasks
    - Robotics
    - Board Games
    - Voice Recognition
    - Digit Recognition

Examples - Robotics

- ML is a fundamental part of robotics for enabling robots to perform -
  - household work ranging from cleaning, cooking, reading and scheduling tasks
  - Simultaneous Localization and Mapping (SLAM)
  - walking patterns of humanoid robots
  - finding routes for rescue robots

Examples - Board Games

- one of the oldest applications of ML
- in March 2016, AlphaGo, the board-game-playing AI from Google’s DeepMind played Korean Go Champion Lee Sedol
- AlphaGo won the game 4 points to 1.

Examples - Voice Recognition

- benefited from advances in deep learning
- as well as big data
- Siri uses
  - speech recognizer,
  - natural language processing
  - text-to-speech techniques

Examples - Digit Recognition

- The task of reading in the images of handwritten numbers and letters
- Recognise the digits
  - output the machine-encoded equivalent
- ML methods (SVM and Deep Learning) have hit >99% accuracy for this task

Definitions of ML

- “Field of study that gives computers the ability to learn without being explicitly programmed,” (*Samuel 1959*)
- If not explicitly programmed can it learn to do things? How? Magic?
- “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,” (*Mitchell 1997, p. 2*)

Steps in ML

- How do we learn to perform a task?
  - have access to data from which we can learn (**Data Manipulation**)
  - find patterns or build the model (**Analytics**)
  - finally, evaluate the model and visualise results (**Evaluation and Visualisation**)

Data representation

- Text data
  - Without tools, it’s difficult for humans to analyse and interpret larges volumes of data
  - ML requires data to be described by attributes or parameters prior to learning a model

Without tools, it’s difficult for humans to analyse and interpret larges volumes of data.. Without tools, it’s difficult for humans to analyse and interpret larges volumes of data. ML requires data to be described by attributes or parameters prior to learning a model . Without tools, it’s difficult for humans to analyse and interpret larges volumes of data. ML requires data to be described by attributes or parameters prior to learning a model. ML requires data to be described by attributes or parameters prior to learning a model. Without tools, it’s difficult for humans to analyse and interpret larges volumes of data. ML requires data to be described by attributes or parameters prior to learning a model. Without tools, it’s difficult for humans to analyse and interpret larges volumes of data. ML requires data to be described by attributes or parameters prior to learning a model.

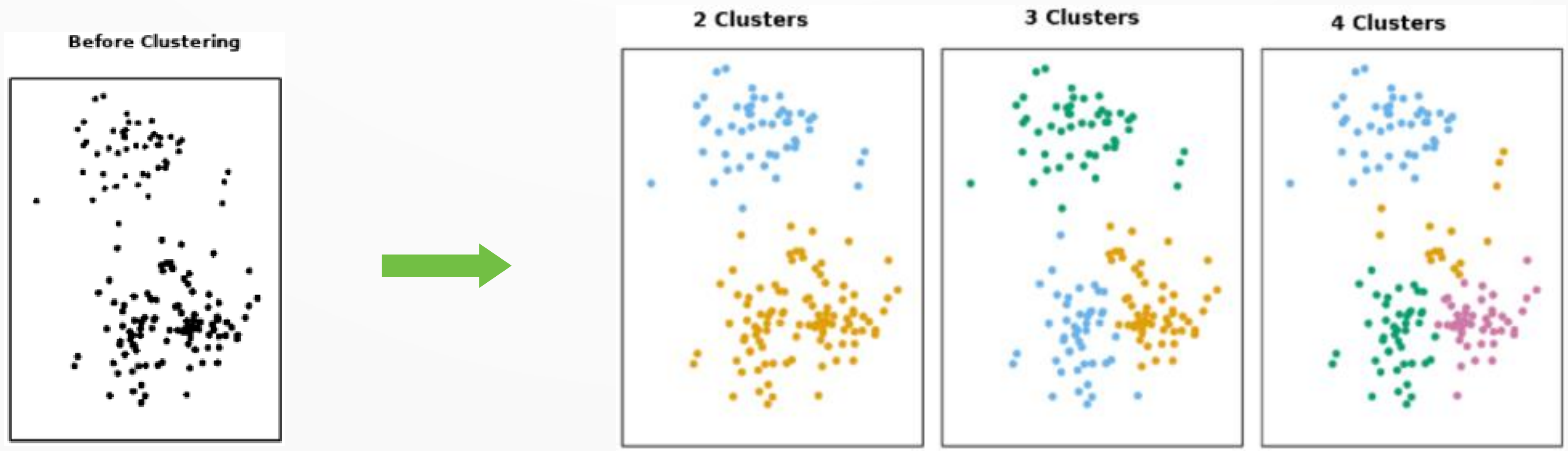
- Number of vowels
- Frequency of a given set of words
- Length of the document
- Set of repetitive words
- Frequency of words
- Number of sentences
- Number of adjectives
- Number of positive and negative words

Data representation...

- Image data
  - build a system able to identify if a given image is from outdoors or not
  - needs to be represented in a vector of features
  - consider a image divided into $9 \times 15 = 135$ blocks
  - For these blocks we can compute – Mean, Variance, radiant, other statistics
  - p features per block leads to
    - **135p** features per image
    - for n images, the size of Feature Matrix is **135p×n**

ML Type - Unsupervised learning

- How do you find the underlying structure of a dataset which is unlabelled?
- Popular approaches - **Clustering** (similarity-based)
  - the process of grouping similar points together
  - gives insight into underlying patterns of different groups



ML Type - Unsupervised learning...

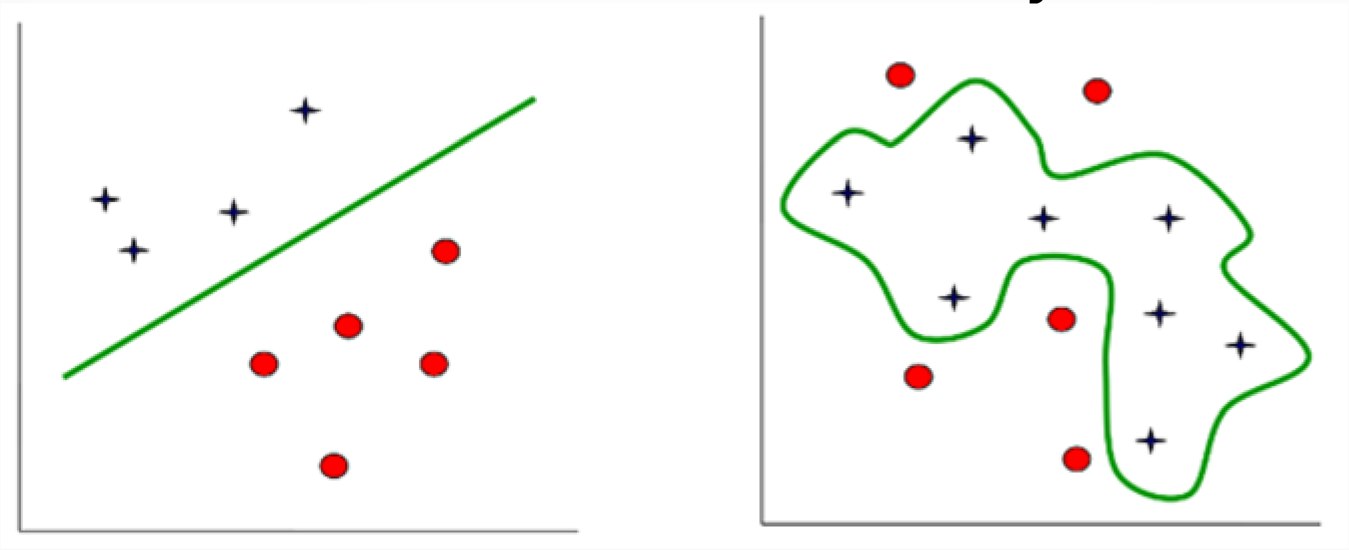
- Common examples -
  - Data understanding and visualization
  - Anomaly detection
  - Information retrieval
  - Data compression (reduction)

ML Type - Supervised learning

- "Learn a function (model) from data to relate the inputs with outputs."
- In supervised learning, the training data includes output information (labels/targets)
- **Target function**: $f: X \rightarrow Y$
- **Examples**: It is in the form of (x,y), denoted as $(x1,y1)$, ..., (xn,yn)
- **Hypothesis** $g: X \rightarrow Y$ such that $g(x)=f(x)$
  - x = set of attribute values
  - y = discrete label (*classification*), real valued number (*regression*)

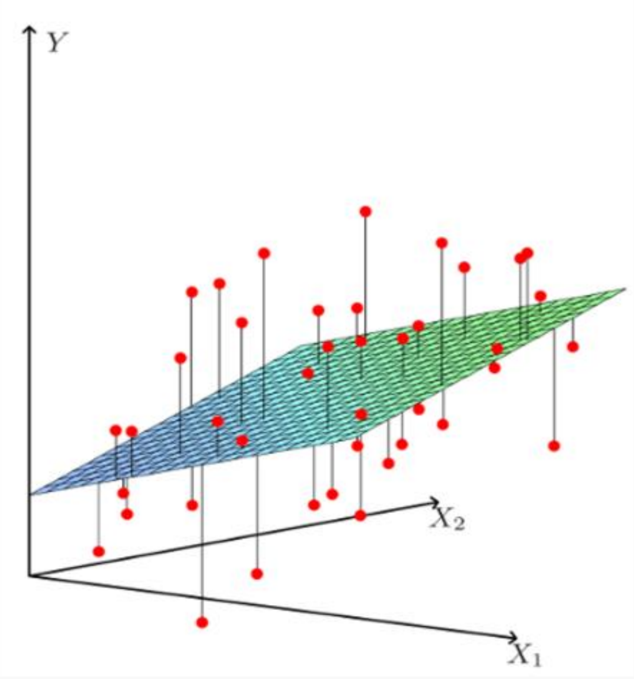
ML Type - Supervised learning..

- Classification problem
  - with two classes, decision boundaries are a hyper-surface that partitions data space into two sets
  - each of these sets represents one of the classes
  - linear vs non-linear decision boundary



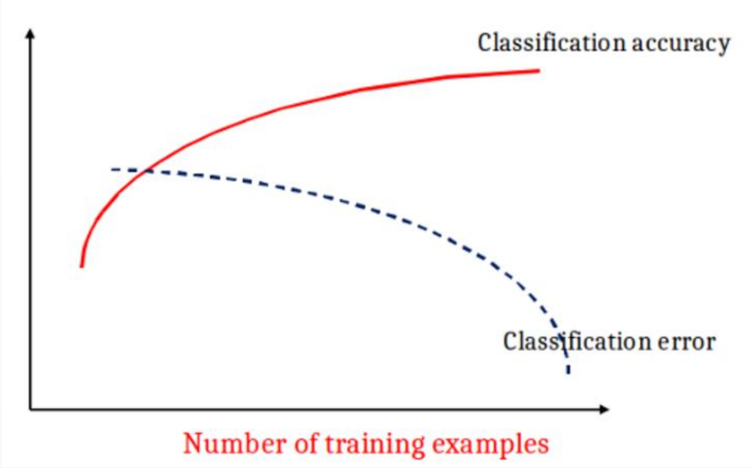
ML Type - Supervised learning...

- Regression problem
  - to examine the relationship between response variables and one or more predictor variables
  - examination can result in a hyperplane, representing the regression analysis
  - regression problem in 2 dimensions



Model assessment and adjustment

- Model evaluation
  - to determine if it will do a perfect job of predicting the labels on new and future test data
    - randomly split examples into a training set and test set
    - use training set to learn a model
    - evaluate the model using test set and a measurement (such as accuracy of prediction)
    - repeat for different random splits and average results
    - more training data, more accuracy



Model assessment and adjustment...

- Model selection
  - how to find the BEST model (hypothesis)?
  - There are often many knobs (parameters and hyper-parameters) that we can use to vary its fitness to the data
  - effective ways in which people approach this problem
    - look at averaged evaluation score on many random test sets
    - cross-validation (train using one set and test on the other, rotate them) etc.
  - be aware of *Over-fitting*

Summary

Data

Model

Decision

Thank You.