Lecture 1. Introduction. Probability Theory

COMP90051 Machine Learning

Sem2 2017

Lecturer: Trevor Cohn

Adapted from slides provided by Ben Rubinstein

THE UNIVERSITY OF MELBOURNE

Why Learn Learning?

Motivation

- *"We are drowning in information,*
  but we are starved for knowledge"
  - John Naisbitt, *Megatrends*

- Data = raw information
- Knowledge = patterns or models behind the data

Solution: Machine Learning

- Hypothesis: pre-existing data repositories contain a lot of potentially valuable knowledge
- Mission of learning: find it
- Definition of learning:

(semi-)automatic extraction of **valid**, **novel**, **useful** and **comprehensible** knowledge – in the form of rules, regularities, patterns, constraints or models – from arbitrary sets of data

Applications of ML are Deep and Prevalent

- Online ad selection and placement
- Risk management in finance, insurance, security
- High-frequency trading
- Medical diagnosis
- Mining and natural resources
- Malware analysis
- Drug discovery
- Search engines

  …

Draws on Many Disciplines

- Artificial Intelligence
- Statistics
- Continuous optimisation
- Databases
- Information Retrieval
- Communications/information theory
- Signal Processing
- Computer Science Theory
- Philosophy
- Psychology and neurobiology

…

Job$

Many companies across all industries hire ML experts:

Data Scientist
Analytics Expert
Business Analyst
Statistician
Software Engineer
Researcher
…

About this Subject

(refer to subject outline on github for more information – linked from LMS)

Vital Statistics

Lecturers: *Weeks 1; 9-12*	Trevor Cohn (DMD8., tcohn@unimelb.edu.au) A/Prof & Future Fellow, Computing & Information Systems *Statistical Machine Learning, Natural Language Processing*
Weeks 2-8	Andrey Kan (andrey.kan@unimelb.edu.au) Research Fellow, Walter and Eliza Hall Institute *ML, Computational immunology, Medical image analysis*
Tutors:	Yasmeen George (ygeorge@student.unimelb.edu.au) Nitika Mathur (nmathur@student.unimelb.edu.au) Yuan Li?
Contact:	*Weekly you should attend 2x Lectures, 1x Workshop*
Office Hours	*Thursdays 1-2pm, 6.24 DMD Building*
Website:	https://trevorcohn.github.io/comp90051-2017/

About Me (Trevor)

- PhD 2007 – UMelbourne
- 10 years abroad UK
  - Edinburgh University, in Language group
  - Sheffield University, in Language & Machine learning groups
- Expertise: Basic research in machine learning; Bayesian inference; graphical models; deep learning; applications to structured problems in text (translation, sequence tagging, structured parsing, modelling time series)

Subject Content

- The subject will cover topics from

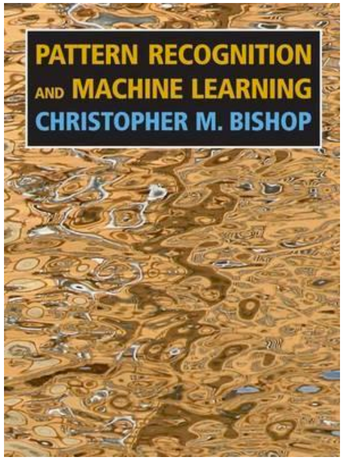
  Foundations of statistical learning, linear models, non-linear bases, kernel approaches, neural networks, Bayesian learning, probabilistic graphical models (Bayes Nets, Markov Random Fields), cluster analysis, dimensionality reduction, regularisation and model selection

- We will gain hands-on experience with all of this via a range of toolkits, workshop pracs, and projects

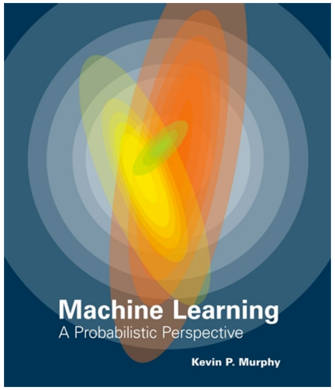
Subject Objectives

- Develop an appreciation for the role of statistical machine learning, both in terms of foundations and applications
- Gain an understanding of a representative selection of ML techniques
- Be able to design, implement and evaluate ML systems
- Become a discerning ML consumer

Textbooks

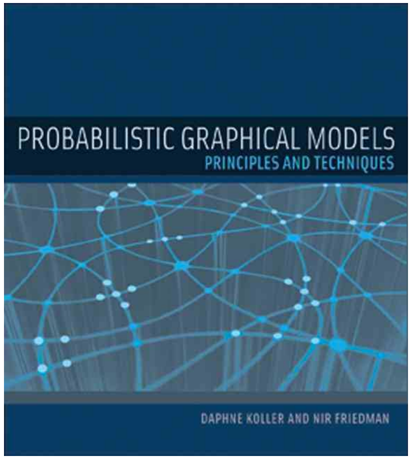


- Primarily references to
  - Bishop (2007) *Pattern Recognition and Machine Learning*
- Other good general references:
  - Murphy (2012) *Machine Learning: A Probabilistic Perspective* [read free ebook using 'ebrary' at http://bit.ly/29SHAQS]
  - Hastie, Tibshirani, Friedman (2001) *The Elements of Statistical Learning: Data Mining, Inference and Prediction* [free at http://www-stat.stanford.edu/~tibs/ElemStatLearn]



Textbooks

- References for PGM component
  - Koller, Friedman (2009) *Probabilistic Graphical Models: Principles and Techniques*



Assumed Knowledge

(Week 2 Workshop revises COMP90049)

- Programming
  - Required: proficiency at programming, ideally in python
  - Ideal: exposure to scientific libraries numpy, scipy, matplotlib etc. (similar in functionality to matlab & aspects of R.)
- Maths
  - Familiarity with formal notation

$$\mathbf{Pr}(\boldsymbol{x}) = \sum_{\boldsymbol{y}} \mathbf{Pr}(\boldsymbol{x}, \boldsymbol{y})$$

  - Familiarity with probability (Bayes rule, marginalisation)
  - Exposure to optimisation (gradient descent)
- ML: decision trees, naïve Bayes, kNN, kMeans

Assessment

- Assessment components
  - Two projects – one released early (w3-4), one late (w7-8); will have ~3 weeks to complete
    - First project fairly structured (20%)
    - Second project includes competition component (30%)
  - Final Exam
- Breakdown
  - 50% Exam
  - 50% Project work
- 50% Hurdle applies to both **exam** and **ongoing assessment**

Machine Learning Basics

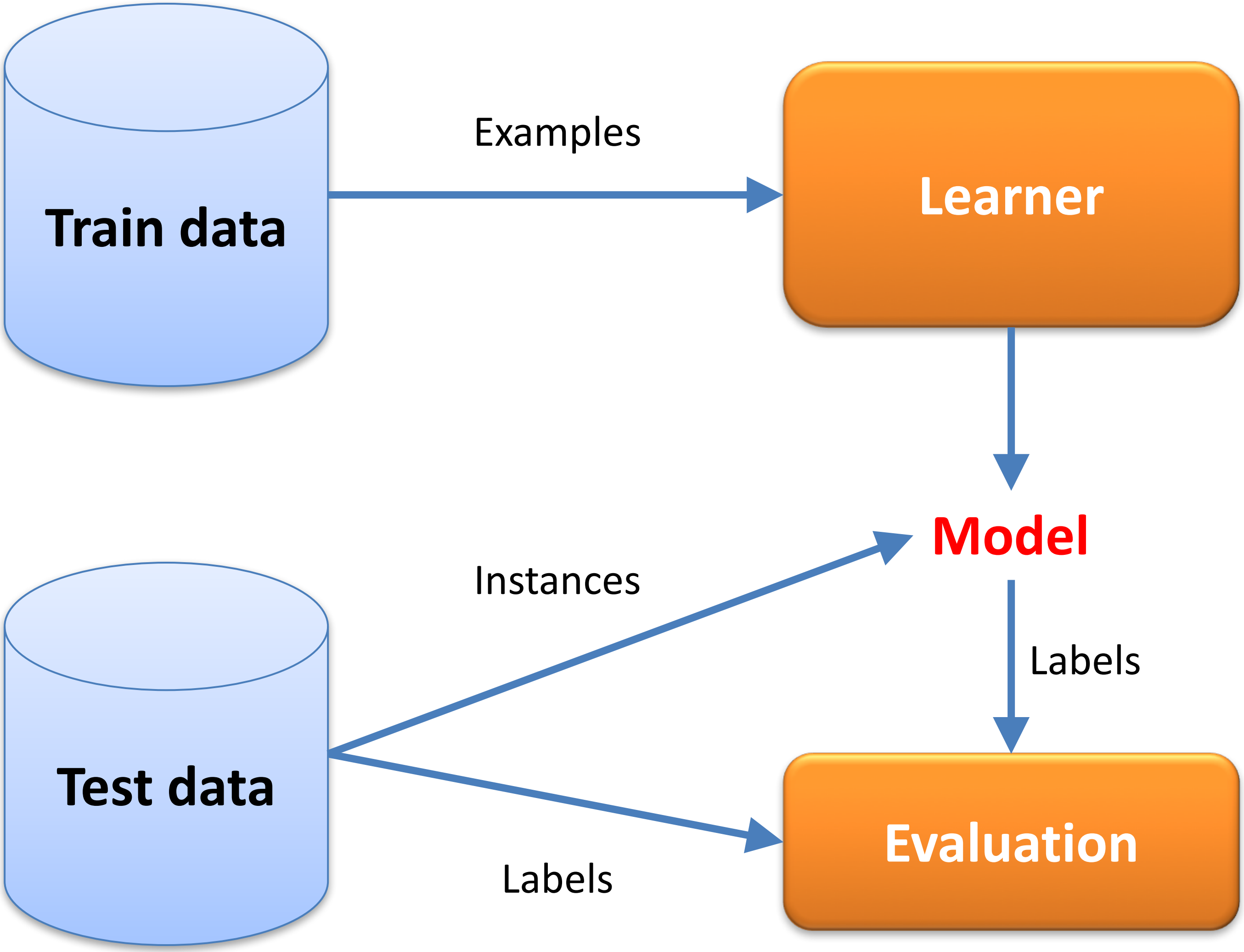
Terminology

- Input to a machine learning system can consist of
  - Instance: measurements about individual entities/objects
    a loan application
  - Attribute (aka Feature, explanatory var.): component of the instances
    the applicant's salary, number of dependents, etc.
  - Label (aka Response, dependent var.): an outcome that is categorical, numeric, etc.
    forfeit vs. paid off
  - Examples: instance coupled with label
    <(100k, 3), "forfeit">
  - Models: discovered relationship between attributes and/or label

Supervised vs Unsupervised Learning

	Data	Model used for
Supervised learning	Labelled	Predict labels on new instances
Unsupervised learning	Unlabelled	Cluster related instances; Project to fewer dimensions; Understand attribute relationships

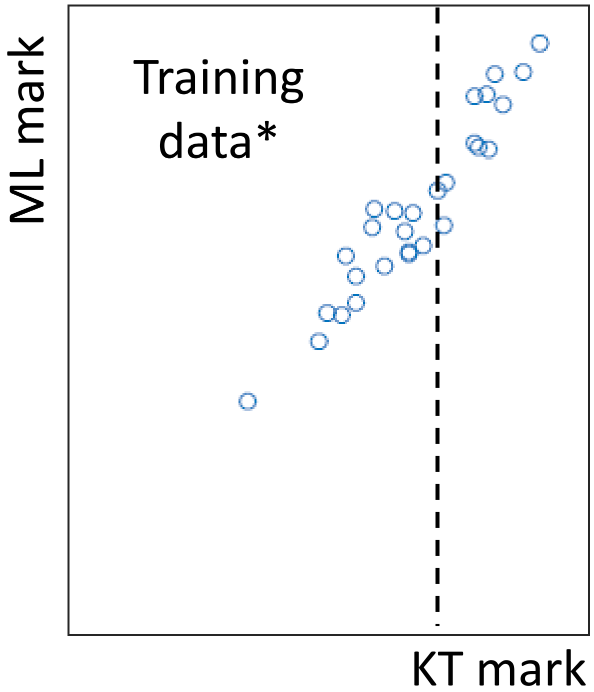
Architecture of a Supervised Learner



Evaluation (Supervised Learners)

- How you measure quality depends on your problem!
- Typical process
  * Pick an evaluation metric comparing label vs prediction
  * Procure an independent, labelled test set
  * “Average” the evaluation metric over the test set
- Example evaluation metrics
  * Accuracy, Contingency table, Precision-Recall, ROC curves
- When data poor, cross-validate

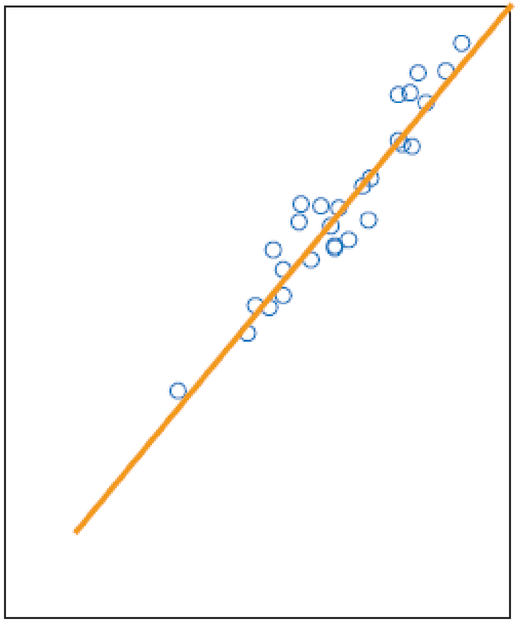
Data is noisy (almost always)



- Example:
  - given mark for Knowledge Technologies (KT)
  - predict mark for Machine Learning (ML)

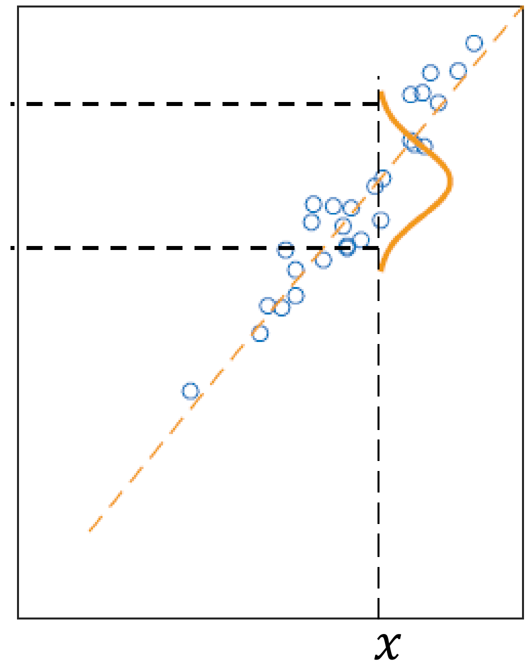
* synthetic data :)

Types of models



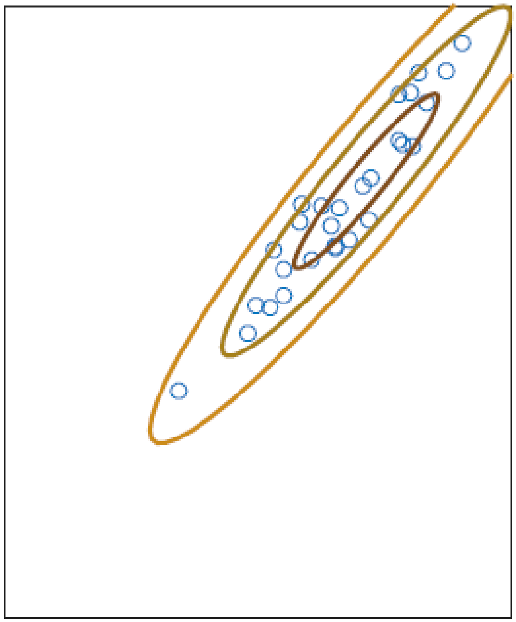
$\hat{y} = f(x)$

KT mark was 95, ML mark is predicted to be 95



$P(y|x)$

KT mark was 95, ML mark is likely to be in (92, 97)



$P(x, y)$

probability of having $(KT = x, ML = y)$

Probability Theory

Brief refresher

Basics of Probability Theory

- A probability space:
  - Set Ω of possible outcomes
  - Set F of events (subsets of outcomes)
  - Probability measure P: $F \rightarrow \mathbf{R}$

- Example: a die roll
  - {1, 2, 3, 4, 5, 6}
  - { φ, {1}, ..., {6}, {1,2}, ..., {5,6}, ..., {1,2,3,4,5,6} }
  - P(φ)=0, P({1})=1/6, P({1,2})=1/3, ...

Axioms of Probability

1. $P(f) \geq 0$ for every event f in F

2. $P\left(\bigcup_f f\right) = \sum_f P(f)$ for all collections* of pairwise disjoint events

3. $P(\Omega) = 1$

* We won't delve further into advanced probability theory, which starts with measure theory. But to be precise, additivity is over collections of countably-many events.

Random Variables (**r.v.'s**)

- A random variable *X* is a numeric function of outcome $X(\omega) \in \boldsymbol{R}$
- $P(X \in A)$ denotes the probability of the outcome being such that *X* falls in the range *A*

- Example: *X* winnings on $5 bet on even die roll
  * *X* maps 1,3,5 to -5
    X maps 2,4,6 to 5
  * P(*X*=5) = P(*X*=-5) = ½

Discrete vs. Continuous Distributions

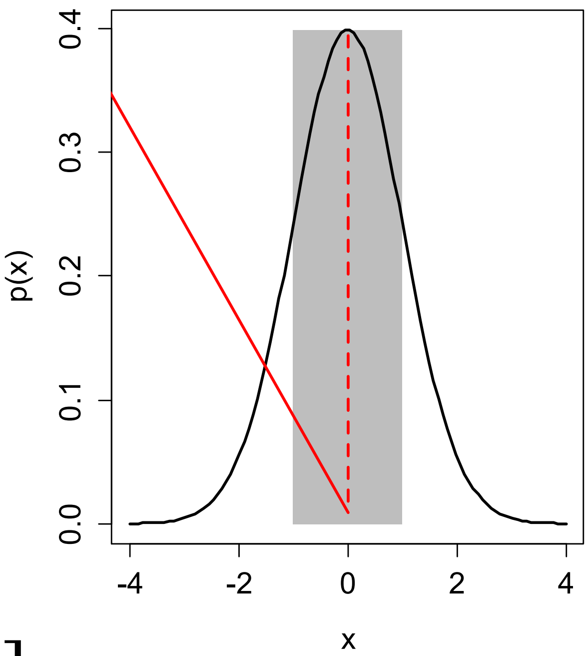
- Discrete distributions
  - Govern r.v. taking discrete values
  - Described by probability mass function p(x) which is P(X=x)
  - $P(X \leq x) = \sum_{a=-\infty}^{x} p(a)$
  - **Examples**: Bernoulli, Binomial, Multinomial, Poisson

- Continuous distributions
  - Govern real-valued r.v.
  - Cannot talk about PMF but rather probability density function p(x)
  - $P(X \leq x) = \int_{-\infty}^{x} p(a)da$
  - **Examples**: Uniform, Normal, Laplace, Gamma, Beta, Dirichlet

Expectation

- Expectation $E[X]$ is the r.v. X's "average" value
  - Discrete: $E[X] = \sum_x x\, P(X = x)$
  - Continuous: $E[X] = \int_x \; x\, p(x)\, dx$
- Properties
  - Linear: $E[aX + b] = aE[X] + b$

    $E[X + Y] = E[X] + E[Y]$
  - Monotone: $X \geq Y \;\Rightarrow\; E[X] \geq E[Y]$
- Variance: $Var(X) = E[(X - E[X])^2]$



Independence and Conditioning

- *X*, *Y* are independent if
  - $P(X \in A, Y \in B) = P(X \in A)P(Y \in B)$
  - Similarly for densities: $p_{X,Y}(x,y) = p_X(x)p_Y(y)$
  - **Intuitively**: knowing value of *Y* reveals nothing about *X*
  - **Algebraically**: the joint on *X*,*Y* factorises!
- Conditional probability
  - $P(A|B) = \frac{P(A \cap B)}{P(B)}$
  - Similarly for densities $p(y|x) = \frac{p(x,y)}{p(x)}$
  - **Intuitively**: probability event *A* will occur given we know event *B* has occurred
  - X,Y independent equiv to $P(Y = y|X = x) = P(Y = y)$

Inverting Conditioning: Bayes' Theorem

Bayes

- In terms of events *A*, *B*
  - $P(A \cap B) = P(A|B)P(B) = P(B|A)P(A)$
  - $P(A|B) = \frac{P(B|A)P(A)}{P(B)}$
- Simple rule that lets us swap conditioning order
- Bayesian statistical inference makes heavy use
  - Marginals: probabilities of individual variables
  - Marginalisation: summing away all but r.v.'s of interest

Summary

- Why study machine learning?
- Machine learning basics
- Review of probability theory