ML Overview

GeoComput & ML

2021-05-04 Tue

Course Structure

GeoComputation

- Linux environment
- Geo computational tools : gdal/ogr, pktools, grass, etc.

$\Uparrow$

*Geo***Coding**

$\Downarrow$

GeoModelling

- GeoMath
- GeoStats

Class Survey

Topics	Speakers
Python	web
R	web
TensorFlow	guest ?
unSupervised Learning	LS
Image processing ?	LS ?
ML Optimisation	LS
rivernetwork delinearation	GA + LS
from project discussions	GA + LS

Course Outlook

Dates	Contents	Speaker
0504	projects + ML overview	LS
0506	projects + ML opt.	LS
0511	unsupervised learning	LS
0518	specific topics	LS + GA
0520	ANN	guest
0525	ANN	guest
0527	LSTM	guest
0601	presentation day	
0604	presentation day	

Project Workflow

- Problem Statement : project definition, data collection

Project Workflow

- Problem Statement : project definition, data collection
- Model Construction

Project Workflow

- Problem Statement : project definition, data collection
- Model Construction
  1. Explorative
     Data exploration : missing data, correction, manipulation
     Geocomput tools, math/stats, programming

Project Workflow

- Problem Statement : project definition, data collection
- Model Construction
  1. Explorative
     Data exploration : missing data, correction, manipulation
     Geocomput tools, math/stats, programming
  2. Mathematical
     Modelling knowledge, Programming skills

Project Workflow

- Problem Statement : project definition, data collection
- Model Construction
  1. Explorative
     Data exploration : missing data, correction, manipulation
     Geocomput tools, math/stats, programming
  2. Mathematical
     Modelling knowledge, Programming skills
- Model Analysis
  solution determination

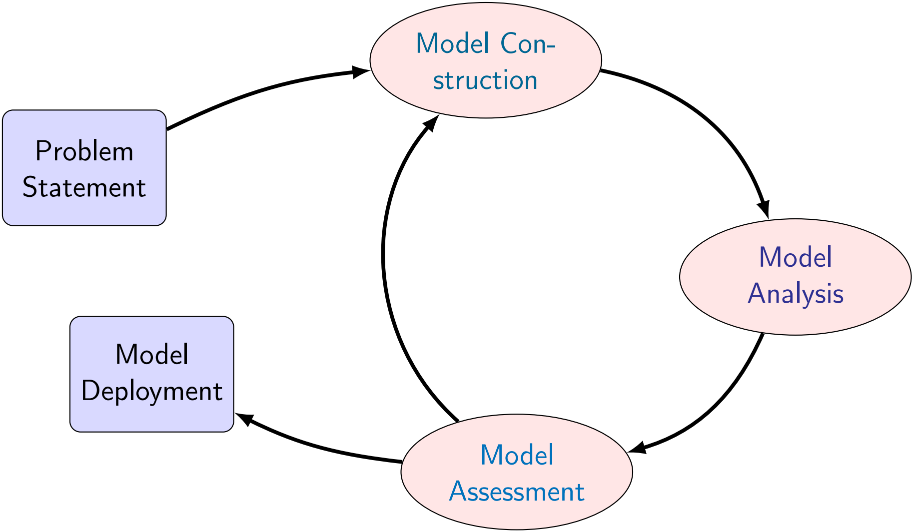
Project Workflow

- Problem Statement : project definition, data collection
- Model Construction
  1. Explorative
     Data exploration : missing data, correction, manipulation
     Geocomput tools, math/stats, programming
  2. Mathematical
     Modelling knowledge, Programming skills
- Model Analysis
  solution determination
- Model Assessment
  fidelity, cost, complexity, flexibility, etc.

Project Workflow

- Problem Statement : project definition, data collection
- Model Construction
  1. Explorative
     Data exploration : missing data, correction, manipulation
     Geocomput tools, math/stats, programming
  2. Mathematical
     Modelling knowledge, Programming skills
- Model Analysis
  solution determination
- Model Assessment
  fidelity, cost, complexity, flexibility, etc.
- Model Deployment
  presentation : map output

Iterative Process



Project Guidance

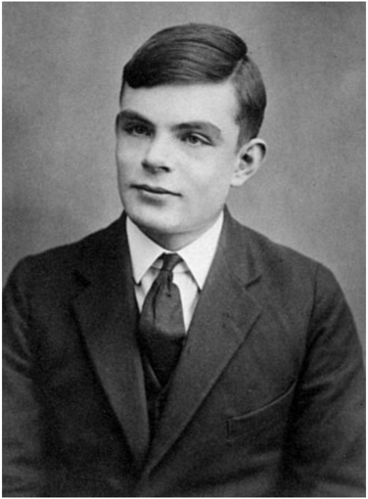
Broad Sense

- Prediction and/or Analytics
- Coding languages

Evaluation

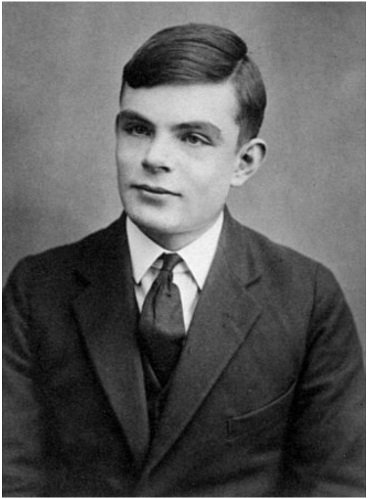
- Clear concepts
- Logic reasoning
- Numerical ability
- Presentability

Machine Thinking



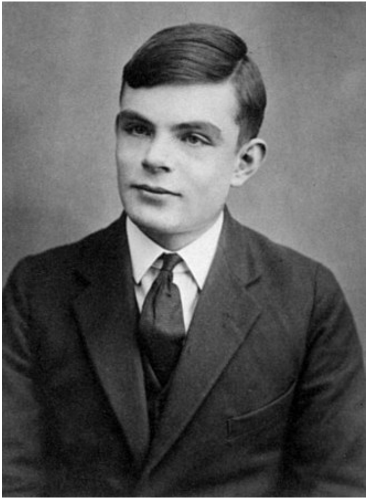
- 1930s : Turing machine : mathematics into recipe

Machine Thinking



- 1930s : Turing machine : mathematics into recipe
- Can machine think ?

Machine Thinking



- 1930s : Turing machine : mathematics into recipe
- Can machine think ?
- Turing test : indistinguishable from human reactions

AI Foundation

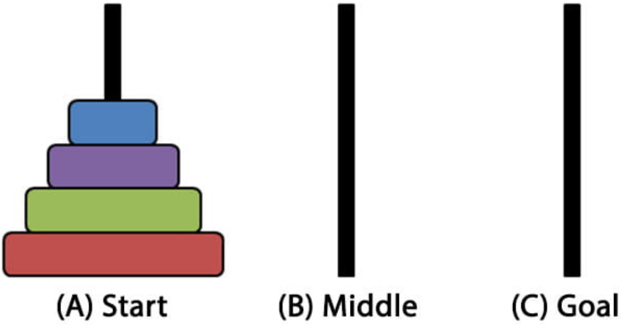
- 1950s : John McCarthy : AI research

AI Foundation

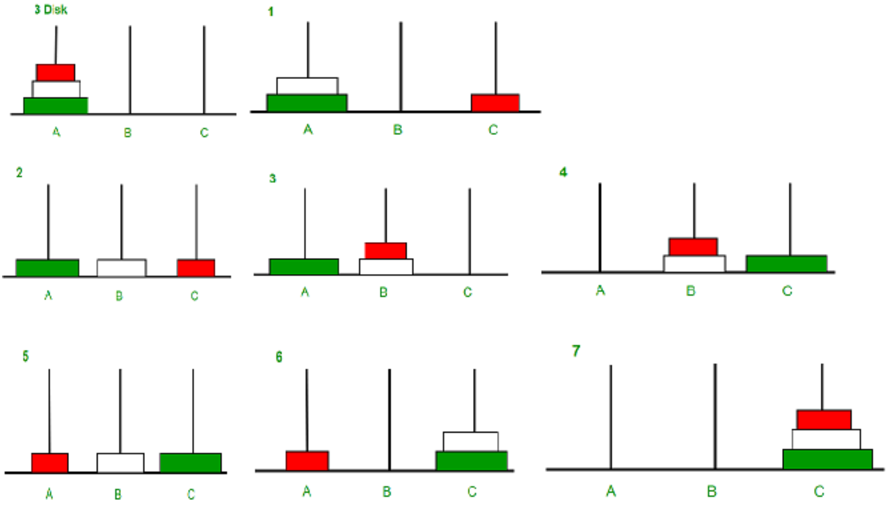
- 1950s : John McCarthy : AI research
- Machine learning : learning from experience

AI Foundation

- 1950s : John McCarthy : AI research
- Machine learning : learning from experience



Optimal Solution



Combinatorial explosion

- average 200 possible moves
- anticipating next four moves
- more than 320 billion combinations

Expert System

- 1970s : capturing human knowledge
- logic-based
- deduction
- logic knots

Agent-based AI

- 1980 : Intelligence as emergent property rising from the interaction w/ the environment

Agent-based AI

- 1980 : Intelligence as emergent property rising from the interaction w/ the environment
- Bayesian theorem : coping w/ uncertainty

Agent-based AI

- 1980 : Intelligence as emergent property rising from the interaction w/ the environment
- Bayesian theorem : coping w/ uncertainty

```
100 out of 10,000 women at age forty who participate in routine
screening have breast cancer. 80 of every 100 women with breast
cancer will get a positive mammography. 950 out of 9,900 women
without breast cancer will also get a positive mammography. If
10,000 women in this age group undergo a routine screening, about
what fraction of women with positive mammographies will actually
have breast cancer?
```

Agent-based AI

- 1980 : Intelligence as emergent property rising from the interaction w/ the environment
- Bayesian theorem : coping w/ uncertainty

```
100 out of 10,000 women at age forty who participate in routine
screening have breast cancer. 80 of every 100 women with breast
cancer will get a positive mammography. 950 out of 9,900 women
without breast cancer will also get a positive mammography. If
10,000 women in this age group undergo a routine screening, about
what fraction of women with positive mammographies will actually
have breast cancer?
```

$$
\begin{aligned}
P(C|+) &= \frac{P(+|C)P(C)}{P(+|C)P(C) + P(+|C^c)P(C^c)} \\
&= \frac{0.8 \times 0.1}{0.8 \times 0.1 + 0.96 \times 0.9} = 48\%
\end{aligned}
$$

Deep Learning

- ML : generating output w/o a recipe

Deep Learning

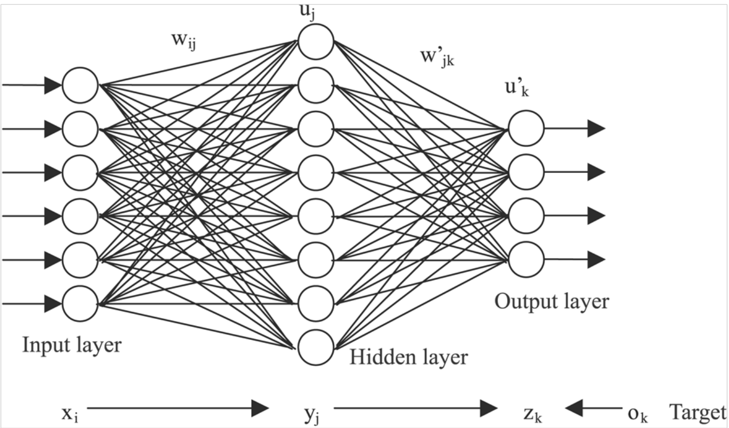
- ML : generating output w/o a recipe
- Categories : supervised + unsupervised learning

Deep Learning

- ML : generating output w/o a recipe
- Categories : supervised + unsupervised learning
- Neural nets

Deep Learning

- ML : generating output w/o a recipe
- Categories : supervised + unsupervised learning
- Neural nets



Well-posed Problem

Definition

A computer program is said to learn from experience E w.r.t. some tasks T and performance measure P, if its performance at tasks T as measured by P improves with experience E.

`For example, a computer programs learns to play Go game might improves its performance as measured by its ability to win, through the experience of playing against itself.`

ML Basics

- generalisation : perform well on previously unseen data

ML Basics

- generalisation : perform well on previously unseen data
- test error : $\mathcal{L}(\hat{f}(\boldsymbol{X} - \boldsymbol{Y}))$

ML Basics

- generalisation : perform well on previously unseen data
- test error : $\mathcal{L}(\hat{f}(\boldsymbol{X} - \boldsymbol{Y}))$
- capacity : ability to fit a wide range of functions

ML Basics

- generalisation : perform well on previously unseen data
- test error : $\mathcal{L}(\hat{f}(\boldsymbol{X} - \boldsymbol{Y}))$
- capacity : ability to fit a wide range of functions
- underfitting vs. overfitting

ML Basics

- generalisation : perform well on previously unseen data
- test error : $\mathcal{L}(\hat{f}(\boldsymbol{X} - \boldsymbol{Y}))$
- capacity : ability to fit a wide range of functions
- underfitting vs. overfitting
- non-free lunch theorem : no ML algorithm universally better than another

ML Basics

- generalisation : perform well on previously unseen data
- test error : $\mathcal{L}(\hat{f}(\boldsymbol{X} - \boldsymbol{Y}))$
- capacity : ability to fit a wide range of functions
- underfitting vs. overfitting
- non-free lunch theorem : no ML algorithm universally better than another
- goal : seeking ML algorithms with good performance on the data generating distribution of concern

ML Basics

- generalisation : perform well on previously unseen data
- test error : $\mathcal{L}(\hat{f}(\boldsymbol{X} - \boldsymbol{Y}))$
- capacity : ability to fit a wide range of functions
- underfitting vs. overfitting
- non-free lunch theorem : no ML algorithm universally better than another
- goal : seeking ML algorithms with good performance on the data generating distribution of concern
- hyperparameters tuning

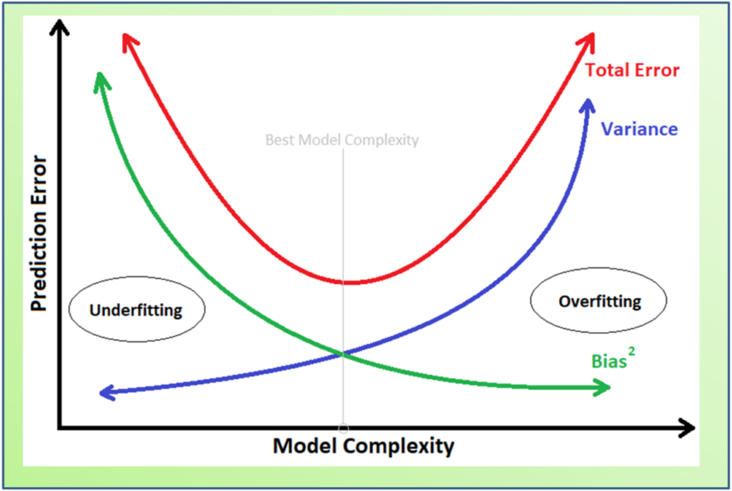
Bias vs. Variance

Given $y = f(x) + \epsilon$

$$E[y] = E[f + \epsilon] = E[f] = f$$

$$\begin{aligned}
E[(y - \hat{f})^2] &= E[(f + \epsilon - \hat{f})^2] \\
&= E[(f - E[\hat{f}] + \epsilon - \hat{f} + E[\hat{f}])^2] \\
&= E[(f - E[\hat{f}])^2] + E[\epsilon]^2 + E[(E[\hat{f}] - \hat{f})^2] - 2E[(E[\hat{f}] - \hat{f})(f - E[\hat{f}])] \\
&= E[(f - E[\hat{f}])^2] + E[\epsilon]^2 + E[(E[\hat{f}] - \hat{f})^2] - 2(E[\hat{f}]f - E[\hat{f}]f + E[\hat{f}]E[\hat{f}] - E[\hat{f}]E[\hat{f}]) \\
&= Bias(\hat{f}^2) + Var[\hat{f}] + \sigma^2
\end{aligned}$$

Bias vs. Variance



MLE

Let $p_{MDL}(\boldsymbol{x};\boldsymbol{\theta})$ be a parametric family of probability distribution over the same space indexed by $\boldsymbol{\theta}$.

$$\boldsymbol{\theta}_{ML} = arg\,\underset{\boldsymbol{\theta}}{max} \sum log(p_{MDL}(\boldsymbol{x};\boldsymbol{\theta}))$$

$$\boldsymbol{\theta}_{ML} = arg\,\underset{\boldsymbol{\theta}}{max}\, \mathbb{E}[x \sim \hat{p}_{DAT}] log(p_{MDL}(\boldsymbol{x};\boldsymbol{\theta}))$$

Bayes Statistics

$\boldsymbol{\theta}$ as a prior distribution : $p(\boldsymbol{\theta})$

$$p(\boldsymbol{\theta}|x^{(1)} \dots x^{(m)}) = \frac{p(x^{(1)}, \dots, x^{(m)}|\boldsymbol{\theta})p(\boldsymbol{\theta})}{p(x^{(1)}, \dots, x^{(m)})}$$

References

- A. Turing. Mind (1950) 59, 433
- D. Wolpert. Neual Comput. (1996) 8, 1341
- D. Wolpert, W. MacReady. IEEE Trans. Evol. Comput. (1997) 1, 67
- M. Wooldridge. A Brief History of Artificial Intelligence (2021)
- N. Nilson. The quest for artificial intelligence (2010)
- T. Mitchell. Machine Learning (1997)
- M. Bishop. Pattern Recognition and Machine Learning (2006)
- I. Goodfellow. Deep Learning (2016)
- https://en.wikipedia.org/wiki/Alan_Turing
- https://en.wikipedia.org/wiki/Bias
- https://www.data-stats.com/bias-variance-tradeoff/
- https://www.geeksforgeeks.org/java-program-for-tower-of-hanoi/
- https://en.wikipedia.org/wiki/John_McCarthy_(computer_scientist)
- https://www.stemlittleexplorers.com/en/make-and-solve-tower-of-hanoi/
- https://en.wikipedia.org/wiki/Go_(game)
- https://www.readthesequences.com/An-Intuitive-Explanation-Of-Bayess-Theorem