UPPSALA
UNIVERSITET

Machine learning – trends and tools

Introducing the field and some of its key concepts

Thomas Schön, Uppsala University

The Löwdin Lectures, Uppsala.

2018-11-16

"Machine learning gives computers the ability to **learn without being explicitly programmed** *for the task at hand."*

“Anyone making confident predictions about anything having to do with the future of artificial intelligence is either kidding you or kidding themselves.”

Andrew McAfee, MIT

What we do in the team — who we are

We automate the extraction of knowledge and understanding from data.

Both basic research **and** applied research (with companies).

Create **probabilistic models** of dynamical systems and their surroundings.

Develop methods to **learn** models from data.

The models can then be used by machines (or humans) to **understand** or **take decisions** about what will happen next.

What is machine learning all about?

Machine learning is about learning, reasoning and acting based on data.

Machine learning gives computers the ability to **learn without being explicitly programmed** for the task at hand.

"It is one of today's most rapidly growing technical fields, lying at the intersection of computer science and statistics, and at the core of artificial intelligence and data science."

Ghahramani, Z. **Probabilistic machine learning and artificial intelligence**. *Nature* 521:452-459, 2015.

Jordan, M. I. and Mitchell, T. M. **Machine Learning: Trends, perspectives and prospects**. *Science*, 349(6245):255-260, 2015.

The four cornerstones

Cornerstone 1 **(Data)** Typically we need lots of it.

Cornerstone 2 **(Mathematical model)** A mathematical model is a compact representation of the data that in precise mathematical form captures the key properties of the underlying situation.

Cornerstone 3 **(Learning algorithm)** Used to compute the unknown variables from the observed data using the model.

Cornerstone 4 **(Decision/Control)** Use the understanding of the current situation to steer it into a desired state.

Mathematical models – representations

The performance of an algorithms typically depends on which representation that is used for the data.

When solving a problem – start by thinking about **which model/representation to use**!

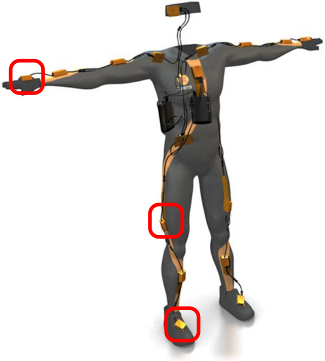
International Conference on Learning Representations (ICLR)
http://www.iclr.cc/

Ex) What is a mathematical model? – a more classic example

Aim: Compute the position and orientation of the different body segments of a person moving around indoors (motion capture).

We need to **learn unknown variables** (positions and orientations) based on **observed data** (accelerations and angular velocities).

We **use a model** to extract knowledge from the observed data.



Ex) What is a mathematical model? – a more classic example

Illustrate the use of three different models:

1. Integration of the observations from the sensors.
2. Add a biomechanical model.
3. Add a world model.

Add ultrawideband measurements for absolute position.

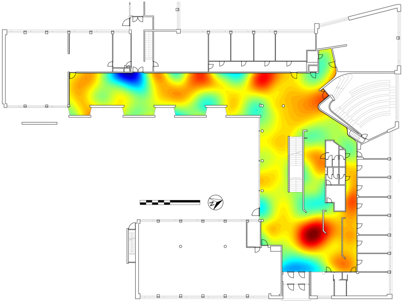
Show movies!

Ex) Ambient magnetic field map

The Earth's magnetic field sets a background for the ambient magnetic field. Deviations make the field vary from point to point.

Aim: Build a map (i.e., a model) of the magnetic environment based on magnetometer measurements.

Solution: Customized Gaussian process that obeys Maxwell's equations.



www.youtube.com/watch?v=enlMiUqPVJo

Arno Solin, Manon Kok, Niklas Wahlström, TS and Simo Särkkä. **Modeling and interpolation of the ambient magnetic field by Gaussian processes**. *IEEE Transactions on Robotics*, 34(4):1112–1127, 2018.

Carl Jidling, Niklas Wahlström, Adrian Wills and TS. **Linearly constrained Gaussian processes**. *Advances in Neural Information Processing Systems (NIPS)*, Long Beach, CA, USA, December, 2017.

The model – learning relationship

The problem of learning (estimating) a model based on data leads to computational challenges, both

- **Integration:** e.g. the HD integrals arising during marg. (averaging over all possible parameter values z):

$$p(D) = \int p(D \mid z) p(z) \mathrm{d}z.$$

- **Optimization:** e.g. when extracting point estimates, for example by maximizing the posterior or the likelihood

$$\widehat{z} = \arg\max_{z} \, p(D \mid z)$$

Typically impossible to compute exactly, use approximate methods

- Monte Carlo (MC), Markov chain MC (MCMC), and sequential MC (SMC).
- Variational inference (VI).
- Stochastic optimization.

Key lesson from contemporary Machine Learning

Flexible models often give the best performance.

How can we build and work with these flexible models?

1. Models that use a large (but fixed) number of parameters.
   (**parametric**, ex. deep learning)

   LeCun, Y., Bengio, Y., and Hinton, G. **Deep learning**, *Nature*, Vol 521, 436–444, 2015.

2. Models that use more parameters as we get access to more data.
   (**non-parametric**, ex. Gaussian process)

   Ghahramani, Z. **Bayesian nonparametrics and the probabilistic approach to modeling**. *Phil. Trans. R. Soc. A* 371, 2013.

   Ghahramani, Z. **Probabilistic machine learning and artificial intelligence**. *Nature* 521:452-459, 2015.

Outline

1. What is machine learning?
2. A closer look at the models
3. **Two concrete examples of flexible models**
    a) **Deep learning**
    b) **Gaussian processes**
4. A few other trends/tools
5. WASP
6. Conclusion

Machine learning gives computers the ability to **learn without being explicitly programmed** for the task at hand.

Deep learning – what is it?

The mathematical model has been around for 70 years, but over the last $5-7$ years there has been a **revolution**. Key reasons:

1. Very large datasets
2. Better and faster computers
3. Enormous industrial interest (e.g. Google, Facebook, MS)
4. Some methodological breakthroughs

The underlying model is a big mathematical function with **multiple layers of abstraction**, commonly with millions of parameters.

The parameter values are **automatically** determined based on a large amount of training data.

Constructing an NN for regression

A **neural network (NN)** is a hierarchical nonlinear function $y = g_\theta(x)$ from an input variable x to an output variable y parameterized by θ.

Linear regression models the relationship between a continuous output variable y and an input variable x,

$$y = \sum_{i=1}^{n} \theta_i x_i + \theta_0 + \varepsilon = \theta^\mathsf{T} x + \varepsilon,$$

where θ is the parameters composed by the "weights" θ_i and the offset ("bias") term θ_0,

$$\theta = \begin{pmatrix} \theta_0 & \theta_1 & \theta_2 & \cdots & \theta_n \end{pmatrix}^\mathsf{T},$$

$$x = \begin{pmatrix} 1 & x_1 & x_2 & \cdots & x_n \end{pmatrix}^\mathsf{T}.$$

Generalized linear regression and NNs

We can generalize this by introducing nonlinear transformations of the predictor $\theta^\mathsf{T} x$,

$$y = f(\theta^\mathsf{T} x).$$

We can think of the neural network as a **sequential construction** of several generalized linear regressions.

Deep neural networks

Let the computer **learn from experience** and understand the situation in terms of a **hierarchy of concepts**, where each concepts is defined in terms of its relation to simpler concepts.

If we draw a graph showing these concepts of top of each other, the graph is **deep**, hence the name deep learning.

It is accomplished by using **multiple levels of representation**. Each level transforms the representation at the previous level into a new and more abstract representation,

$$z^{(l+1)} = f\left(\Theta^{(l+1)} z^{(l)} + \theta_0^{(l+1)}\right),$$

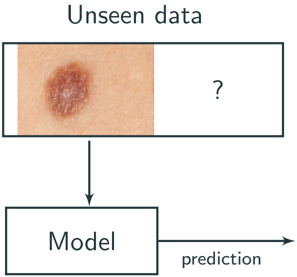
starting from the input (raw data) $z^{(0)} = x$.

Key aspect: The layers are **not** designed by human engineers, they are generated from (typically lots of) data using a learning procedure and lots of computations.

Deep learning – example (skin cancer)

Start from a mathematical model trained on 1.28 million images (**transfer learning**). Make minor modifications of it, specializing to present situation.

Learn new model parameters using 129 450 clinical images ($\sim$ 100 times more images than any previous study).


Unseen data

?

Model

prediction

The results are on par with professional dermatologists on specific tasks. Still, far from being clinically useful, but at least they give us “valid reasons to remain cautiously optimistic” as someone said.

Andre Esteva, A., Kuprel, B., Novoa, R. A., Ko, J., Swetter, S. M., Blau, H. M. and Thrun, S. **Dermatologist-level classification of skin cancer with deep neural networks**. *Nature*, 542, 115–118, February, 2017.

The Gaussian process is a model for nonlinear functions

Q: Why is the Gaussian process used everywhere?

It is a **non-parametric** and **probabilistic** model for nonlinear functions.

- **Non-parametric** means that it does not rely on any particular parametric functional form to be postulated.
- **Probabilistic** means that it takes uncertainty into account in every aspect of the model.

An abstract idea

In probabilistic (Bayesian) linear regression

$$y_t = \underbrace{\beta^\mathsf{T}\mathbf{x}_t}_{f(\mathbf{x}_t)} + e_t, \qquad e_t \sim \mathcal{N}(0, \sigma^2),$$

we place a prior on β, e.g. $\beta \sim \mathcal{N}(0, \alpha^2 I)$.

(Abstract) idea: What if we instead place a prior directly on the function $f(\cdot)$

$$f \sim p(f)$$

and look for $p(f \mid y_{1:T})$ rather than $p(\beta \mid y_{1:T})$?!

One concrete construction

Well, one (arguably simple) idea on how we can reason probabilistically about an unknown function f is by assuming that $f(x)$ and $f(x')$ are jointly Gaussian distributed

$$\begin{pmatrix} f(x) \\ f(x') \end{pmatrix} \sim \mathcal{N}\left(m, K\right).$$

If we accept the above idea we can without conceptual problems generalize to any *arbitrary* finite set of input values $\{x_1, x_2, \ldots, x_T\}$.

$$\begin{pmatrix} f(x_1) \\ \vdots \\ f(x_T) \end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix} m(x_1) \\ \vdots \\ m(x_N) \end{pmatrix}, \begin{pmatrix} k(x_1, x_1) & \ldots & k(x_1, x_T) \\ \vdots & \ddots & \vdots \\ k(x_T, x_1) & \ldots & k(x_T, x_T) \end{pmatrix} \right)$$

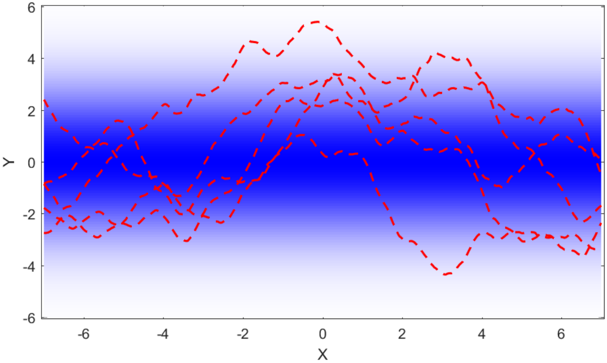
Definition

Definition: (Gaussian Process, GP) A GP is a (potentially infinite) collection of random variables such that any finite subset of it is jointly distributed according to a Gaussian.

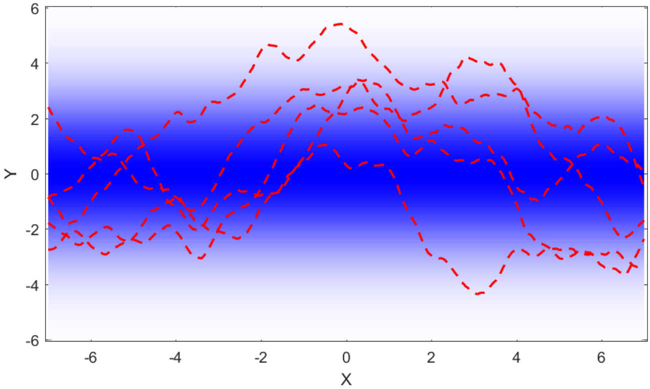
We now have a prior!

$$f \sim \mathcal{GP}(m, k)$$

The GP is a **generative** model so let us first sample from the prior.



GP regression – illustration



Outline

1. What is machine learning?
2. A closer look at the models
3. Two concrete examples of flexible models
   a) Deep learning
   b) Gaussian processes
4. **A few other trends/tools**
5. WASP
6. Conclusion

Machine learning gives computers the ability to **learn without being explicitly programmed** for the task at hand.

Probabilistic numerics

Reinterpreting numerical tasks (e.g., linear algebra, int., opt. and solving differential equations) as probabilistic inference problems.

A numerical method **estimates** a certain **latent** property **given** the result of computations, i.e. computation is inference.

Ex: Basic alg. that are equivalent to Gaussian MAP inference

- Conjugate Gradients for linear algebra
- BFGS etc. for nonlinear optimization
- Gaussian Quadrature rules for Integration
- Runge-Kutta solvers for ODEs

Hennig, P., Osborne, M., Girolami, M. **Probabilistic numerics and uncertainty in computations**. *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 471(2179), 2015.

Probabilistic programming

Probabilistic programming makes use of computer programs to represent probabilistic models.

Probabilistic programming lies on the interesting intersection of

1. Programming languages: Compilers and semantics.
2. Machine learning: Algorithms and applications.
3. Statistics: Inference and theory.

Creates a clear **separation** between the model and the inference methods, encouraging model based thinking. Automate inference!

http://probabilistic-programming.org/

Markov chain Monte Carlo (MCMC)

Represent distributions using a large number of samples.

Markov chain Monte Carlo (MCMC) methods are used to sample from a probability distribution by **simulating a Markov chain** that has the desired distribution as its stationary distribution.

Used to compute numerical approximations of intractable integrals.

Constructive algorithms:

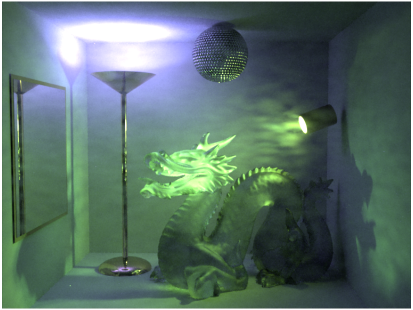
1. The **Metropolis Hastings** sampler
2. The **Gibbs** sampler

Andrieu, C. De Freitas, N. Doucet, A. and Jordan, M. I. **An Introduction to MCMC for Machine Learning**, *Machine Learning*, 50(1): 5-43, 2003

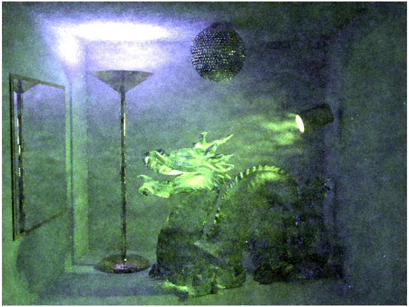
“Visualizing” a Markov chain

Built a Markov chain to sample light paths connecting the sensor with light sources in the scene.

Results using equal time rendering



Our method that builds on MLT



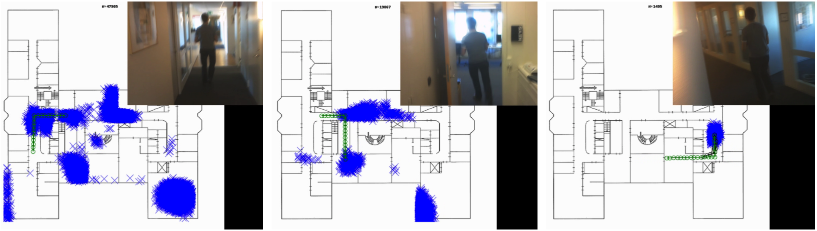
Metropolis light transport (MLT)

Joel Kronander, TS and Jonas Unger. **Pseudo-marginal Metropolis light transport**. In *Proceedings of SIGGRAPH ASIA Technical Briefs*, Kobe, Japan, November, 2015.

Sequential Monte Carlo (SMC)

SMC provide approximate solutions to **integration** problems where there is a **sequential structure** present.

Important example where we have a sequential structure present is in dynamical systems, where we call SMC **particle filtering**.



A. Doucet and A. M. Johansen. **A Tutorial on Particle filtering and smoothing: Fiteen years later**. In *The Oxford Handbook of Nonlinear Filtering*, D. Crisan and B. Rozovsky (eds.). Oxford University Press, 2011.

TS, Fredrik Lindsten, Johan Dahlin, Johan Wågberg, Christian A. Naesseth, Andreas Svensson and Liang Dai. **Sequential Monte Carlo methods for system identification**. In *Proceedings of the 17th IFAC Symposium on System Identification (SYSID)*, Beijing, China, October 2015.

Variational inference

Variational inference provides an approximation to the posterior distribution by **assuming that it has a certain functional form** that contain unknown parameters.

These unknown parameters are found using **optimization**, where some distance measure is minimized.

Variational inference methods are used to approximate intractable integrals and are thus an alternative to MCMC.

Ex. Variational Bayes (VB) and expectation propagation (EP).

Blei, D. Kucukelbir, A. and McAuliffe, J. **Variational inference: A review for statisticians**, *Journal of the American Statistical Association*, 112(518):859–877, 2017.

A few other trends/tools (too brief)

1. **Probabilistic numerics** – Reinterpreting numerical tasks as probabilistic inference problems.
2. **Probabilistic programming** – Makes use of computer programs to represent probabilistic models.
3. Two strategies to approximate intractable target distributions
   - **Markov chain Monte Carlo (MCMC)** – Construct a Markov chain with the traget distribution as its stationary distribution. Then, we sample from the chain to eventually collect independent samples from the stationary distribution.
   - **Variational inference** – Assume a family of distributions and find the member (using optimization) of that family which is closest to the target distribution.

WASP

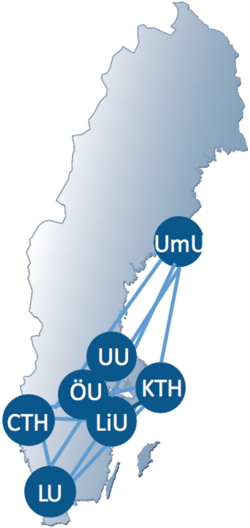
WASP | WALLENBERG AI, AUTONOMOUS SYSTEMS AND SOFTWARE PROGRAM

Sweden's largest individual research program ever.

Vision: Excellent research and competence in **artificial intelligence**, **autonomous systems** and **software** for the benefit of Swedish industry.

wasp-sweden.org

WASP – participating universities



Research on artificial intelligence and autonomous systems acting in collaboration with humans, adapting to their environment through sensors, information and knowledge.

Software is the main enabler in these systems, and is an integrated research theme of the program.

WASP – Strategic instruments

Designed to achieve leverage, renewal, and expansion.

- A research program aiming for disruptive developments.
- An international recruitment program. Brain-gain to establish new research areas, and to reinforce existing strengths in Sweden.
  - Plan to recruit more than **50 new professors** on different levels
  - 18 in original WASP donation
  - 14+14=28 in WASP-AI donation
  - 5 Wallenberg Distinguished Chair in AI
- A national graduate school with close interaction with Swedish industry with the aim to raise the knowledge level in Sweden.
  - Produce at least **300 new PhDs**, with at least 80 of those being industrial PhD students.
- Platforms for research and demonstration in collaboration.

Conclusion

Machine learning gives computers the ability to **learn without being explicitly programmed** for the task at hand.

Uncertainty is a key concept!

The best predictive performance is currently obtained from **highly flexible** learning systems, e.g.

1. Deep learning
2. Gaussian processes

WASP allows Sweden to **move faster** in the area of Machine Learning.