UPPSALA
UNIVERSITET

Formulating flexible probabilistic models

Thomas Schön
Uppsala University
Sweden

IFAC Symp. on System Identification: learning models for decision and control
July 13, 2021 (Online).

Deep learning for classification is handled using standard losses and output representations, but this is **not** *(yet) the case when it comes to regression.*

I have two messages

1. The combined use of **probabilistic models** and **deep learning** is more interesting than we think.

Illustration: Formulating and solving regression problems.

---

2. **Analysis:** An **overparametrized** model has enough degrees of freedom to perfectly fit the training data and **still** they achieve state-of-the-art generalization performance!

We need **new theory** to understand this.

Key lesson from contemporary Machine Learning

Flexible models often give the best predictive 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.

Be careful as flexible models can be deceptive!

Deep learning

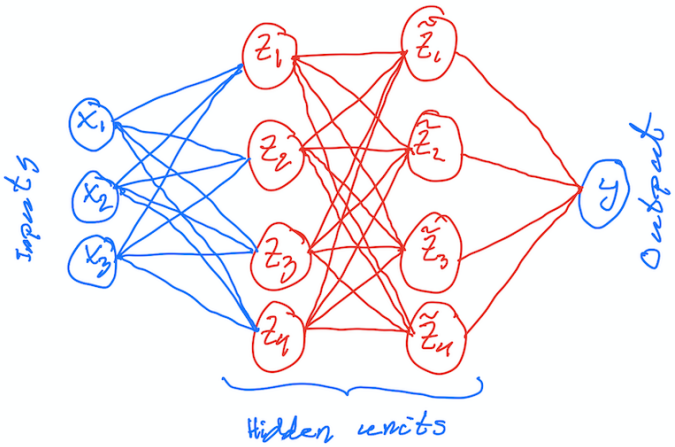
Works **extremely well** on **clearly defined** problems with **lots** of data.

Accounts for the vast majority of all progress and investment in AI/ML.

Has generated massive business potential.

The mathematical model is 70 years old...

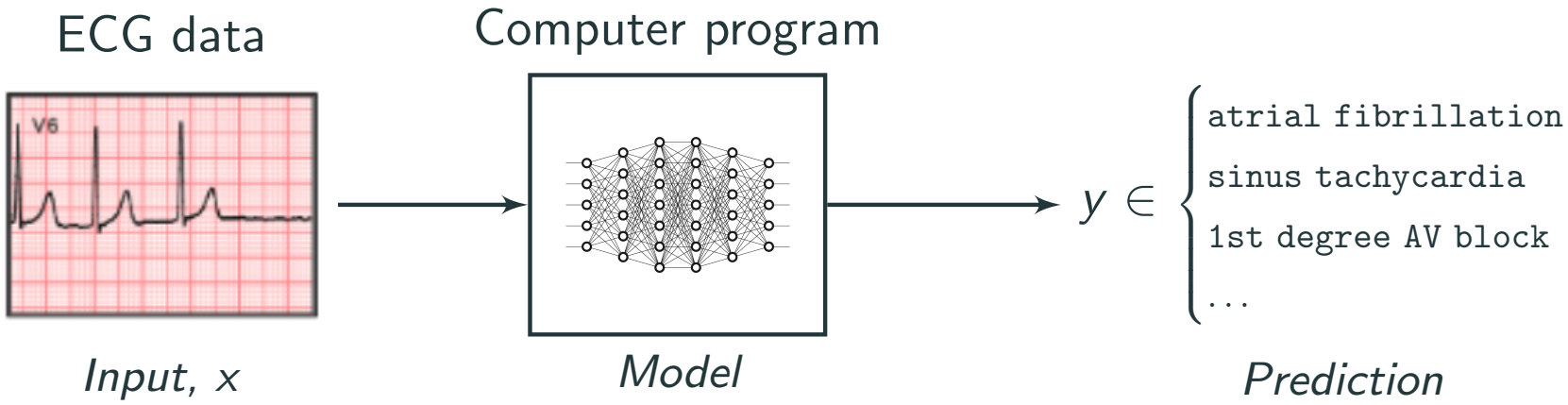
Need for new mathematics to understand why it works.



When representing a function, a deep, hierarchical model can be **exponentially more efficient** than a shallow model.

Flexible models solve relevant problems – an example

Aim: Automatic classification of Electrocardiography (ECG) data.



We are reaching human level performance on specific tasks.

Key difference to classical approach: The model is **not** derived based on our ability to mathematically explain what we see in an ECG.

Instead, a generic model is **automatically learned** based on data.

Antonio H. Ribeiro, Manoel H. Ribeiro, Gabriela M.M. Paixao, Derick M. Oliveira, Paulo R. Gomes, Jessica A. Canazart, Milton P. S. Ferreira, Carl R. Andersson, Peter W. Macfarlane, Wagner Meira Jr., TS, Antonio Luiz P. Ribeiro. **Automatic diagnosis of the 12-lead ECG using a deep neural network**. *Nature Communications*, 11(1760), 2020.

Reflection on the example

Standard deep learning classification problem formulation.

What about regression problems like the ones in system identification?

Let us make this very concrete

NARX models assume that the output y_t depends on

- past outputs $y_{t-1}, \ldots, y_{t-D_y}$
- and past inputs $u_{t-1}, \ldots, u_{t-D_y}$

Goal: Find $p(y_t \mid x_t)$, where

$$x_t = \{y_{t-1}, \ldots, y_{t-D_y}, u_{t-1}, \ldots, u_{t-D_y}\}.$$

Challenge: How should we choose this predictive distribution?

The straightforward option is to assume a functional form $p_\theta(y_t \mid x_t)$.

Immediately gives rise to at least two questions:

1. How should we parameterize this distribution?
2. How should we learn it from data?

The most common answers to these two questions

1. How should we parameterize this distribution?

   Traditionally we assume an output equation

   $$y_t = f_\theta(x_t) + e_t.$$

   The solution $p_\theta(y_t \mid x_t)$ is **dictated by the assumption** on e_t.

2. How should we learn $p_\theta(y_t \mid x_t)$ from data?

   Traditionally we make use of maximum likelihood.

Ex. Assuming $e_t \sim \mathcal{N}(0, \sigma^2)$, the maximum likelihood problem becomes

$$\widehat{\theta} = \arg\max_\theta \sum_{t=1}^{T} \|y_t - f_\theta(x_t)\|^2$$

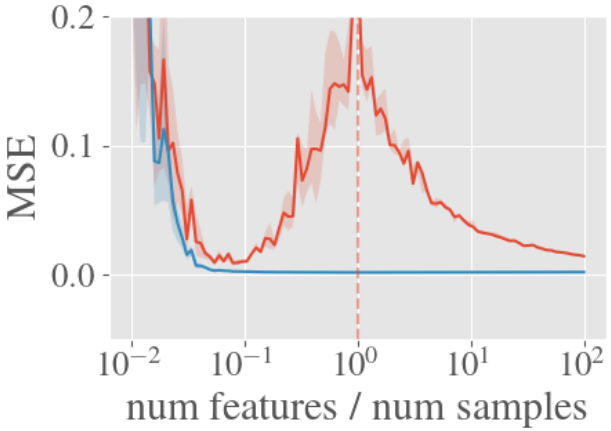
Question for you: Why not use of more flexible model?

A more general question

How should we best formulate and solve regression problems using deep learning?

Outline

1. Introduction and main message
2. **Deep probabilistic regression**
3. Open problems and entry-points



Far from being solved, lots of interesting opportunities for future work!!

Regression using deep neural networks

Supervised regression: learn to predict a continuous output (target) value $y^\star \in \mathcal{Y} = \mathbb{R}^K$ from a corresponding input $x^\star \in \mathcal{X}$, given a training set $\mathcal{D}$ of i.i.d. input-output data

$$\mathcal{D} = \{(x_n, y_n)\}_{n=1}^N, \qquad (x_n, y_n) \sim p(x, y).$$

Deep neural network (DNN): a function $f_\theta : \mathcal{X} \to \mathcal{Y}$, parameterized by $\theta \in \mathbb{R}^P$, that maps an input $x \in \mathcal{X}$ to an output $f_\theta(x) \in \mathcal{Y}$.

Recall that with a probabilistic take on regression, the task is to learn the conditional output density $p_\theta(y_t \mid x_t)$.

Current status

DL for **classification** is handled using standard losses and representations.

When it comes to **regression** the situation is quite different.

In fact, current standard practice involves—implicitly or explicitly—the use of simple unimodal distributions.

Four existing approaches:

1. Direct regression
2. Probabilistic regression
3. Confidence-based regression
4. Regression-by-classification

1. Direct regression (I/II)

Train a DNN $f_\theta : \mathcal{X} \to \mathcal{Y}$ to directly predict the target $y^\star = f_\theta(x^\star)$.

Learn the parameters θ by minimizing a loss function $\ell(f_\theta(x_i), y_i)$, penalizing discrepancy between prediction $f_\theta(x_i)$ and ground truth y_i

$$\hat{\theta} = \arg\min_\theta J(\theta),$$

where

$$J(\theta) = \frac{1}{N} \sum_{i=1}^{N} \ell(f_\theta(x_i), y_i).$$

Common choices for ℓ are the L^2 loss, $\ell(\hat{y}, y) = \|\hat{y} - y\|_2^2$, and the L^1 loss, $\ell(\hat{y}, y) = \|\hat{y} - y\|_1$.

1. Direct regression (II/II)

Minimizing

$$J(\theta) = \frac{1}{N} \sum_{i=1}^{N} \ell(f_\theta(x_i), y_i)$$

then corresponds to minimizing the negative log-likelihood $\sum_{i=1}^{N} - \log p(y_i \mid x_i; \theta)$, **for a specific model** $p(y \mid x; \theta)$ of the conditional target density.

Ex: The L^2 loss corresponds to a fixed-variance Gaussian model:

$$p(y \mid x; \theta) = \mathcal{N}(y; f_\theta(x), \sigma^2).$$

2. Probabilistic regression (I/II)

Why not explicitly employ this probabilistic perspective and try to create **more flexible** models $p(y \mid x; \theta)$ of the conditional target density $p(y \mid x)$?

One idea is to restrict the parametric model to unimodal distributions such as Gaussian or Laplace.

Probabilistic regression: train a DNN $f_\theta : \mathcal{X} \to \mathcal{Y}$ to predict the parameters ϕ of a certain family of probability distributions $p(y; \phi)$, then model $p(y \mid x)$ with

$$p(y \mid x; \theta) = p(y; \phi(x)), \qquad \phi(x) = f_\theta(x).$$

The parameters θ are learned by minimizing $\sum_{i=1}^{N} -\log p(y_i \mid x_i; \theta)$.

2. Probabilistic regression (II/II)

Ex: A general 1D Gaussian model can be realized as:

$$p(y \mid x; \theta) = \mathcal{N}\big(y; \mu_\theta(x), \sigma_\theta^2(x)\big),$$

where the DNN is trained to output

$$f_\theta(x) = \begin{pmatrix} \mu_\theta(x) & \log \sigma_\theta^2(x) \end{pmatrix}^\mathsf{T} \in \mathbb{R}^2.$$

The negative log-likelihood $\sum_{i=1}^N -\log p(y_i \mid x_i; \theta)$ then corresponds to

$$J(\theta) = \frac{1}{N} \sum_{i=1}^N \frac{(y_i - \mu_\theta(x_i))^2}{\sigma_\theta^2(x_i)} + \log \sigma_\theta^2(x_i).$$

3. Confidence-based regression

The quest for improved regression accuracy has also led to the development of more specialized methods.

Confidence-based regression: train a DNN $f_\theta : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ to predict a scalar confidence value $f_\theta(x, y)$ that can be maximized over y to predict the output

$$y^\star = \arg\max_y f_\theta(x^\star, y)$$

Key to this approach is that $f_\theta(x, y)$ depends on **both** the input x and the target y.

The parameters θ are learned by generating **pseudo** ground truth confidence values $c(x_i, y_i, y)$, and minimizing a loss function $\ell\big(f_\theta(x_i, y), c(x_i, y_i, y)\big)$.

4. Regression-by-classification

Discretize the output space $\mathcal{Y}$ into a finite set of C classes and use standard classification techniques...

High-level description of our idea

Confidence-based regression give impressive results, but:

- it requires important (and tricky) task-dependent design choices (e.g. how to generate the pseudo ground truth labels)
- and usually lacks a clear probabilistic interpretation.

Probabilistic regression is straightforward and generally applicable, but:

- it can usually not compete in terms of regression accuracy.

Our construction **combines the benefits** of these two approaches while **removing the problems** above.

Background – Energy-based model (EBM)

An **energy-based model (EBM)** specifies a probability density

$$p(x;\theta) = \frac{e^{f_\theta(x)}}{Z(\theta)}, \qquad Z(\theta) = \int e^{f_\theta(x)} dx,$$

explicitly parameterized by the scalar function $f_\theta(x)$.

By defining $f_\theta(x)$ using a **deep neural network**, $p(x;\theta)$ becomes expressive enough to learn practically any density from observed data.

LeCun, Y., Chopra, S., Hadsell, R. Ranzato, M and Huang, F. J. **A tutorial on energy-based learning**. In *Predicting structured data*, 2006.

Teh, Y. W., Welling, M., Osindero, S. and Hinton, G. E. **Energy-based models for sparse overcomplete representations**. *Journal of Machine Learning Research*, 4:1235–1260, 2003.

Bengio, Y., Ducharme, R., Vincent, P. and Jauvin, C. **A neural probabilistic language model**. *Journal of machine learning research*, 3:1137–1155, 2003.

Hinton, G., Osindero, S., Welling, M. and Teh, Y-W. **Unsupervised discovery of nonlinear structure using contrastive backpropagation**. *Cognitive science*, 30(4):725–731, 2006.

Mnih, A. and Hinton, G. **Learning nonlinear constraints with contrastive backpropagation**. In *Proceedings of the IEEE International Joint Conference on Neural Networks*, 2005.

Osadchy, M., Miller, M. L. and LeCun, Y. **Synergistic face detection and pose estimation with energy-based models**. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2005.

Background – Energy-based models (EBM)

The EBM allows for the full predictive power of the DNN to be exploited, enabling us to learn

- multimodal and
- asymmetric densities

directly from data.

The **cost** of the flexibility is that the normalization constant

$$Z(\theta) = \int e^{f_\theta(x)} dx$$

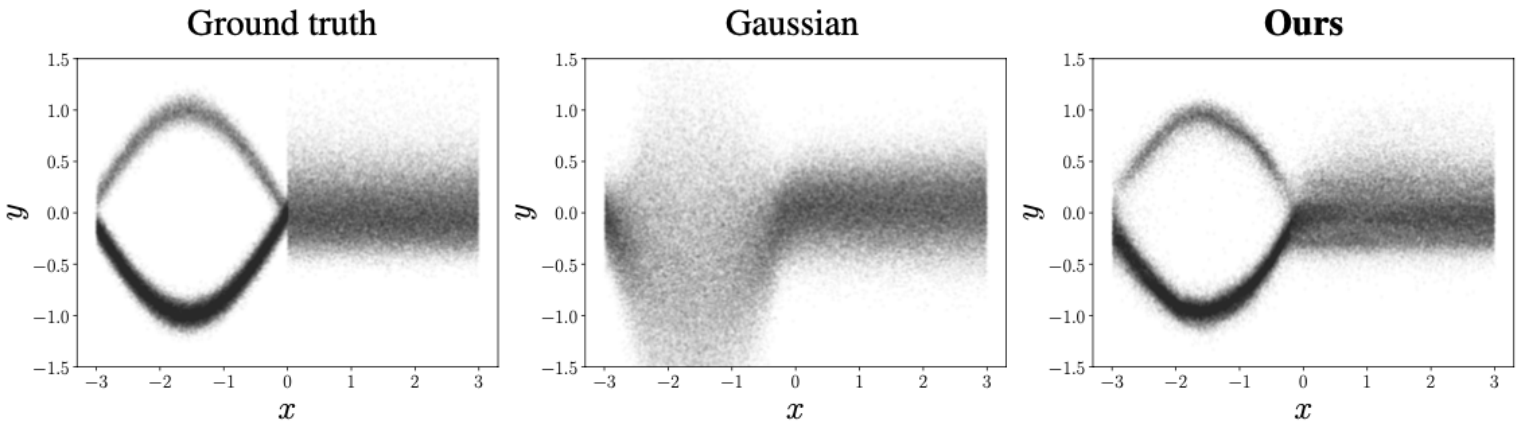
is intractable, which complicates

- evaluating $p(y \mid x; \theta)$ and
- sampling from $p(y \mid x; \theta)$.

Intuitive preview the EBM construction

An energy-based model of the conditional output density $p(y \mid x)$, allowing for **highly flexible** models to be learned directly from data.

1D toy illustration showing that we can learn multi-modal and asymmetric distributions, i.e. the model is **flexible**.



New construction that use EBMs for regression

A general regression method with a **clear probabilistic interpretation** in the sense that we learn a model $p(y \mid x, \theta)$ **without** requiring it to belong to a particular family of distributions.

Let the DNN be a function $f_\theta : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ that maps an input-output pair $\{x_i, y_i\}$ to a scalar value $f_\theta(x_i, y_i) \in \mathbb{R}$.

Define the resulting (flexible) probabilistic model as a conditional EBM,

$$p(y \mid x, \theta) = \frac{e^{f_\theta(x,y)}}{Z(x,\theta)}, \qquad Z(x,\theta) = \int e^{f_\theta(x,\tilde{y})} d\tilde{y}.$$

Training

The DNN $f_\theta(x, y)$ specifying the conditional EBM can be trained using methods for fitting a density $p(y \mid x; \theta)$ to observed data $\{(x_n, y_n)\}_{n=1}^N$.

Most straightforward is to minimize the negative log-likelihood

$$\begin{aligned}\mathcal{L}(\theta) &= -\sum_{i=1}^{N} \log p(y_i \mid x_i; \theta) \\ &= \sum_{i=1}^{N} \log \underbrace{\left(\int e^{f_\theta(x_i, \tilde{y})} d\tilde{y} \right)}_{Z(x_i, \theta)} - f_\theta(x_i, y_i).\end{aligned}$$

Challenge: Requires the normalization constant to be evaluated (the integral is intractable)...

Solution 1 – maximum likelihood using importance sampling

$$p(y \mid x, \theta) = \frac{e^{f_\theta(x,y)}}{Z(x,\theta)}, \qquad Z(x,\theta) = \int e^{f_\theta(x,\tilde{y})} d\tilde{y}$$

The parameters θ are learned by minimizing $\sum_{n=1}^{N} -\log p(y_n \mid x_n; \theta)$.

Use importance sampling to evaluate $Z(x, \theta)$:

$$\begin{aligned}
-\log p(y_i \mid x_i; \theta) &= \log \left(\int e^{f_\theta(x_i,y)} dy \right) - f_\theta(x_i, y_i) \\
&= \log \left(\int \frac{e^{f_\theta(x_i,y)}}{q(y)} q(y) dy \right) - f_\theta(x_i, y_i) \\
&\approx \log \left(\frac{1}{M} \sum_{k=1}^{M} \frac{e^{f_\theta(x_i,y^{(k)})}}{q(y^{(k)})} \right) - f_\theta(x_i, y_i), \quad y^{(k)} \sim q(y).
\end{aligned}$$

Use a Gaussian mixture (centered around the measurements) as proposal.

Solution 2 – Noise Contrastive Estimation (NCE)

Noise contrastive estimation is a parameter estimation method, avoiding calculation of the normalization constant and its derivatives.

Michael Gutmann and Aapo Hyvärinen. **Noise-contrastive estimation: A new estimation principle for unnormalized statistical models**. In *Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 297–304, 2010.

NCE entails **learning to discriminate** between observed data examples and samples drawn from a noise distribution.

Gustafsson, Fredrik K and Danelljan, Martin and Timofte, Radu and TS, **How to train your energy-based model for regression**, *Proceedings of the British Machine Vision Conference (BMVC)*, September, 2020.

Prediction at test time

Train a DNN $f_\theta : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ to predict $f_\theta(x, y)$ and model $p(y \mid x)$ with

$$p(y \mid x, \theta) = \frac{e^{f_\theta(x,y)}}{Z(x,\theta)}, \qquad Z(x,\theta) = \int e^{f_\theta(x,\tilde{y})} d\tilde{y}.$$

The parameters θ are learned by minimizing $\sum_{i=1}^{N} -\log p(y_i \mid x_i; \theta)$.

Given a test input $x^\star$, we predict the target $y^\star$ by maximizing $p(y \mid x^\star; \theta)$

$$y^\star = \arg\max_y \, p(y \mid x^\star; \theta) = \arg\max_y \, f_\theta(x^\star, y).$$

By designing the DNN f_θ to be differentiable w.r.t. targets y, the gradient $\nabla_y f_\theta(x^\star, y)$ can be efficiently evaluated using auto-differentiation.

Use gradient ascent to find a local maximum of $f_\theta(x^\star, y)$, starting from an initial estimate $\hat{y}$.

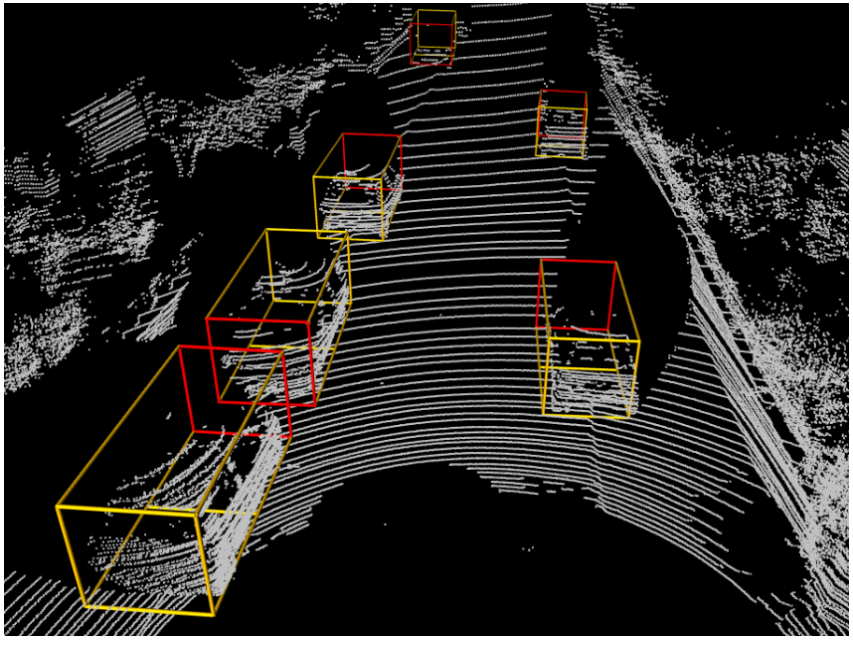
Experiments – Visual tracking

Task (visual tracking): Estimate a bounding box of a target object in every frame of a video. The target object is defined by a given box in the first video frame.

Show Movie!

Gustafsson, Fredrik K and Danelljan, Martin and Bhat, Goutam and TS, **Energy-based models for deep probabilistic regression**, in *Proceedings of the European Conference on Computer Vision (ECCV)*. August, 2020.

Experiments – 3D object detection from laser data



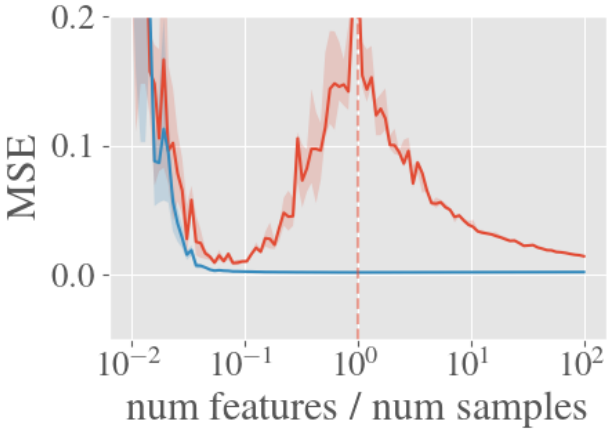
Task: Detect objects from sensor data (here laser) and estimate their size and position in the 3D world.

Key perception task for self-driving vehicles and autonomous robots.

Fredrik K. Gustafsson, Martin Danelljan, and TS. **Accurate 3D object detection using energy-based models**. *Workshop on Autonomous Driving (WAD) at the conference on Computer Vision and Pattern Recognition (CVPR)*, Online, 2021.

Outline

1. Introduction and main message
2. Deep probabilistic regression
3. **Open problems and entry-points**



Far from being solved, lots of interesting opportunities for future work!!

Experiments – NARX models

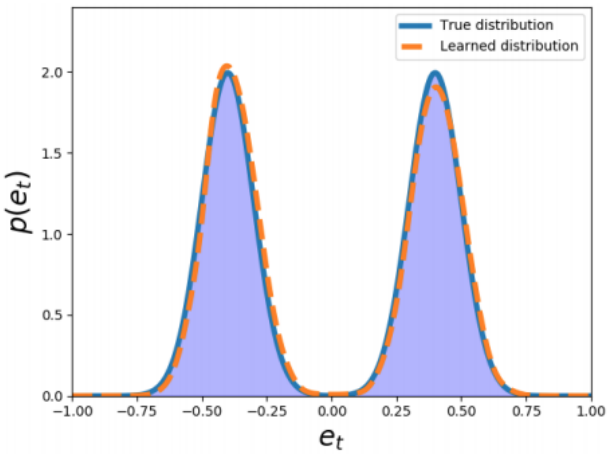
Remark: EBMs are not necessarily the best way to solve this problem...

Toy AR(1) example:

$$y_t = 0.95 y_{t-1} + e_t,$$

where

$$e_t \sim \frac{1}{2}\mathcal{N}(0.4, 0.1^2) + \frac{1}{2}\mathcal{N}(-0.4, 0.1^2).$$



Always test on the simplest simulated example **and real data**.

See Fredrik's talk on this afternoon

Johannes Hendriks, Fredrik K. Gustafsson, Antonio H. Ribeiro, Adrian Wills and TS. **Deep energy-based NARX models**. In *Proceedings of the 19th IFAC Symposium on System Identification (SYSID)*, Online, July, 2021.

Constructing a deep state space model (SSM)

Why?

1. The deep neural networks opens up for **flexibility**.
2. The SSM offers a **natural latent variable representation**.
3. The stochastic nature of the SSM allows for representing **uncertainty**.

Some entry points in case you are interested:

Girin, L., Leglaive, S., Bie, X., Diard, J., Hueber, T. and Alameda-Pineda, X. **Dynamical variational autoencoders: a comprehensive review**. *arXiv:2008.12595*, 2020.

Fraccaro, M., Kamronn, S., Ulrich Paquet, U. and Winther, O. **A disentangled recognition and nonlinear dynamics model for unsupervised learning**. In *Neural Information Processing Systems (NeurIPS)*, 2017.

Bayer, J. and Osendorfer, C. **Learning stochastic recurrent networks**. *arXiv:1411.7610*, 2014.

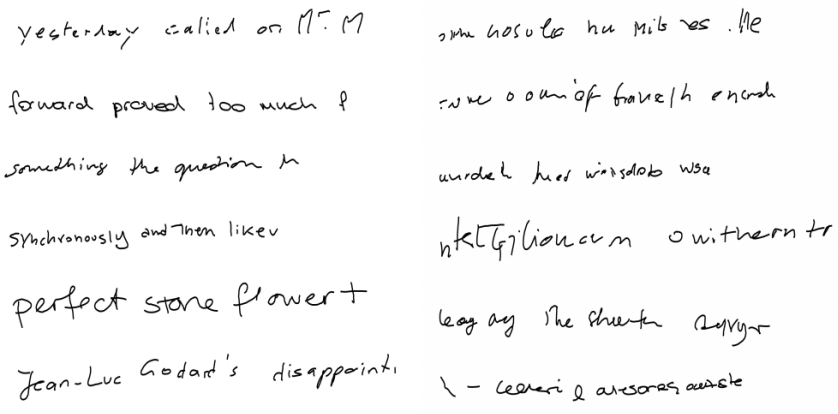
Chung, J., Kastner, K., Dinh, L., Goel, K., Courville, A.C., and Bengio, Y. **A recurrent latent variable model for sequential data**. In *Neural Information Processing Systems (NeurIPS)*, 2015.

See Daniel's talk on Thursday

Daniel Gedon, Niklas Wahlström, TS and Lennart Ljung. **Deep state space models for nonlinear system identification**. In *Proceedings of the 19th IFAC Symposium on System Identification (SYSID)*, Online, July, 2021.

Not all systems have exponentially decaying memory...

Long memory applications: handwritten text and speech.



(a) Dataset samples (b) Generated samples

Hierarchical structured data

Combines an autoregressive model with a hierarchical VAE.

Simulated model output.

See Carl's talk on Thursday

Carl Andersson, Niklas Wahlström and TS. **Learning deep autoregressive models for hierarchical data**. In *Proceedings of the 19th IFAC Symposium on System Identification (SYSID)*, Online, July, 2021.

Data to control (too brief...)

When you build a model you should have its intended use in mind.

Inspired by stochastic MPC we formulate (using a learned SSM)

$$\mathbf{u}_t{}^* = \arg\min_{\mathbf{u}_t} \int V_t(x_t, u_t, \theta, \mathbf{u}_t, \mathbf{w}_t)\, p(x_t, \theta, \mathbf{w}_t | y_{1:t}, u_{1:t})\, \mathrm{d}x_t\, \mathrm{d}\theta\, \mathrm{d}\mathbf{w}_t$$

$$\text{s.t.} \quad c_u(\mathbf{u}_t) \succeq 0,$$

$$\mathbb{P}\left(c_x^j(x_t, u_t, \theta, \mathbf{u}_t, \mathbf{w}_t) \geq 0\right) \geq 1 - \epsilon, \quad j = 1, \ldots, n_{cx}.$$

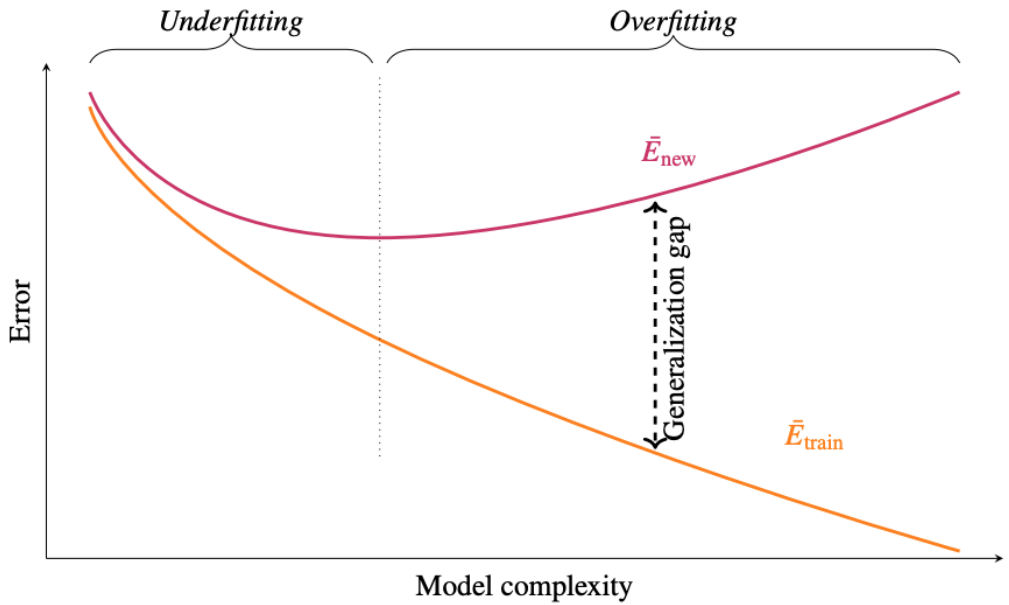
$\mathbf{u}_t \triangleq u_{t+1:t+N+1}, \quad \mathbf{w}_t \triangleq w_{t+1:t+N+1}.$

The cost function $V_t(\cdot)$ can include both

1. control performance (**exploitation**)
2. and identification (**exploration**) related goals.

Johannes Hendriks, James Holdsworth, Adrian Wills, TS and Brett Ninness. **Data to controller for nonlinear systems: an approximate solution.** *IEEE Control Systems Letters (L-CSS)*, 2021. (accepted for publication)

Classical statistics tells use to use less flexible models



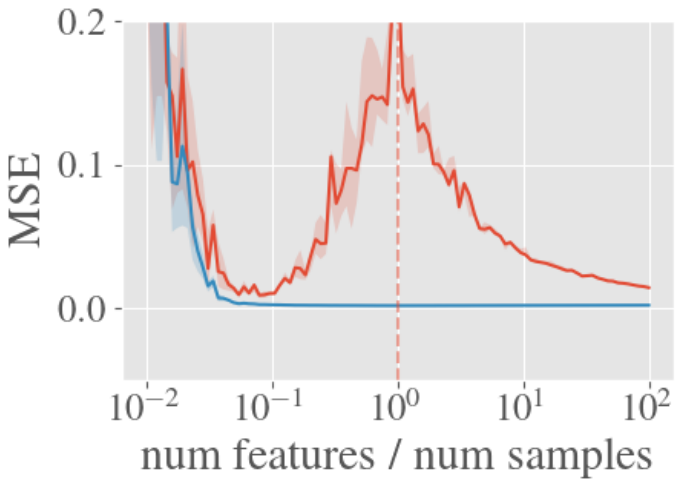
The traditional bias-variance trade-off formalize this principle in probabilistic terms.

Towards a theory of overparametrized (=highly flexible) models

Message 2 (Analysis): DNNs are often **overparametrized**, with enough degrees of freedom to perfectly fit the training data

and still they achieve state-of-the-art generalization performance!

Understanding this requires **new theory**.



History:

Belkin, M., Hsu, D., Ma, S., and Mandal, S. (2019). **Reconciling modern machine-learning practice and the classical bias–variance trade-off**. *Proceedings of the National Academy of Sciences*, 116(32), 15849–15854.

Hastie, T., Montanari, A., Rosset, S., and Tibshirani, R.J.(2019). **Surprises in High-Dimensional Ridgeless Least Squares Interpolation**. *arXiv:1903.08560*.

Bartlett, P.L., Long, P.M., Lugosi, G., and Tsigler, A.(2020). **Benign overfitting in linear regression**. *Proceedings of the National Academy of Sciences*, 117(48):30063–30070.

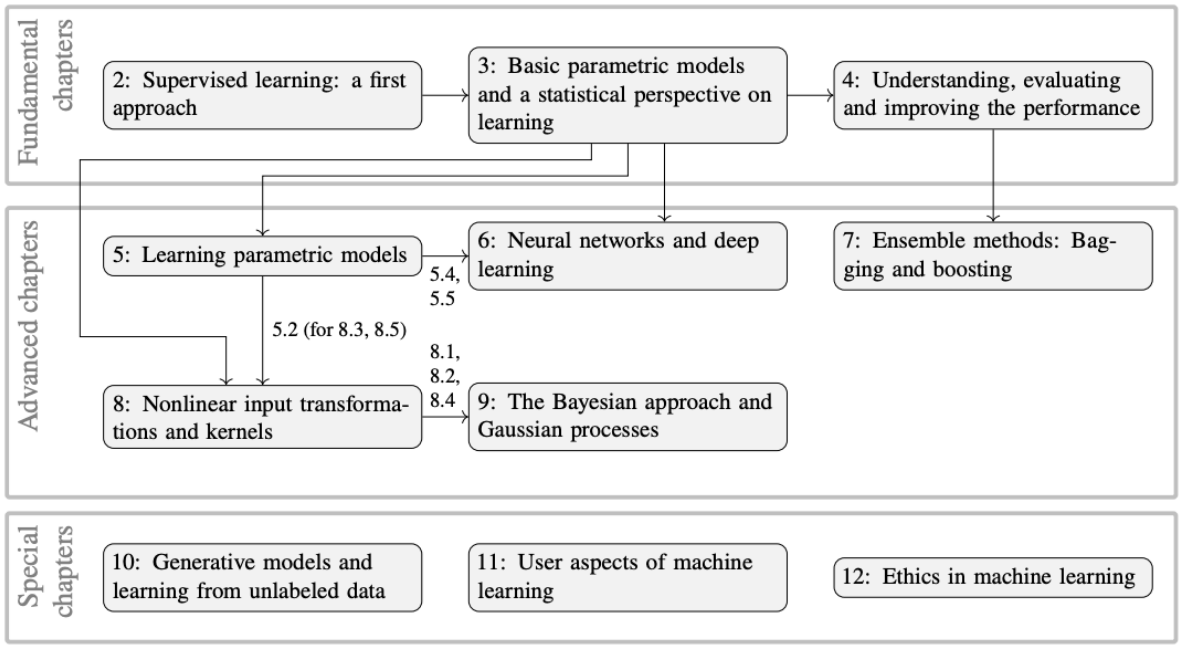
See Antonio's talk this afternoon

Antonio H. Ribeiro, Johannes Hendriks, Adrian Wills and TS. **Beyond Occam's razor in system identification: double-descent when modeling dynamics**. In *Proceedings of the 19th IFAC Symposium on System Identification (SYSID)*, Online, July, 2021.

Education – new book and associated course

Andreas Lindholm, Niklas Wahlström, Fredrik Lindsten, and TS. **Machine Learning – a first course for engineers and scientists**. *Cambridge University Press*, 2021.

http://smlbook.org/



All material for a popular first ML course is available if you are interested.

Conclusions

Message 1: The **combination** of **probabilistic models** and **deep neural networks** is exciting and promising.

Message 2: We need **new theory** to understand the state-of-the-art generalization performance achieved by deep learning.

What I have presented here is work done by a **fantastic team**.

Remember to talk to people who work on **different problems** with **different tools**!! (Visit other fields!)