UPPSALA
UNIVERSITET

Deep probabilistic regression

Thomas Schön
Uppsala University
Sweden

One World Signal Processing Seminars, Online
October 13, 2021.

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

Joint work

Fredrik Gustafsson (PhD student, UU)

Martin Danelljan (post-doc, ETH)

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.

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.

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.

Main message

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

Illustration: Formulating and solving regression problems.

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)

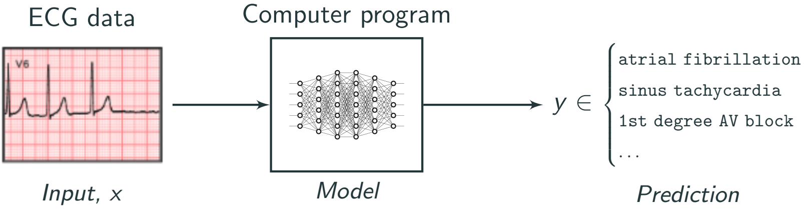
   Ghahramani, Z. **Bayesian nonparametrics and the probabilistic approach to modeling**. *Phil. Trans. R. Soc. A* 371, 2013.

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

Be careful as flexible models can be deceptive!

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?

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 $p_\theta(y_t \mid x_t)$?

   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.

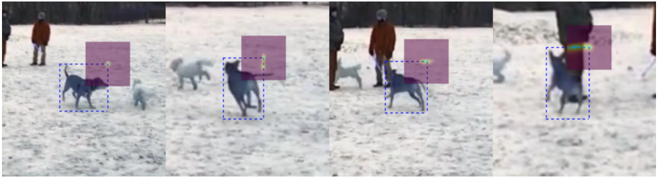
Ex. Assuming $e_t \sim \mathcal{N}(0, \sigma^2)$, the maximum likelihood problem becomes

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

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?



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 $p_\theta(y_t \,|\, x_t)$.

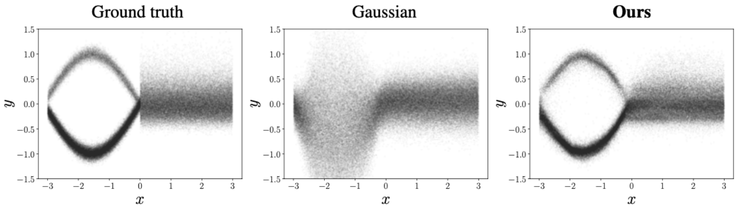
Intuitive preview of our construction

A general regression method with a **clear probabilistic interpretation**.

With a probabilistic take on regression, the task is to learn the conditional target density $p(y \mid x_t)$.

We create and train an energy-based model (EBM) of the conditional target density $p(y \mid x_t)$, allowing for **highly flexible** target densities to be learned directly from data.

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



Ground truth

Gaussian

Ours

Aim and outline

Aim: Create an awareness of how we can use deep neural networks for regression and show that energy-based models are useful in this context.

1. Intuitive preview
2. **Regression using deep neural networks**
3. Energy-based models
4. Our construction
5. Experiments

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 $p_\theta(y \mid x)$ 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

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

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:

$$p_{\theta}(y \mid x) = \mathcal{N}\big(y; \mu_{\theta}(x),\, \sigma^2_{\theta}(x)\big),$$

where the DNN is trained to output

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

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

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

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 output 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:

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

Probabilistic regression is straightforward and generally applicable, but:

1. 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 models (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 $p(y \mid x; \theta)$.

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

The most straightforward method 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 $Z(x,\theta)$:

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

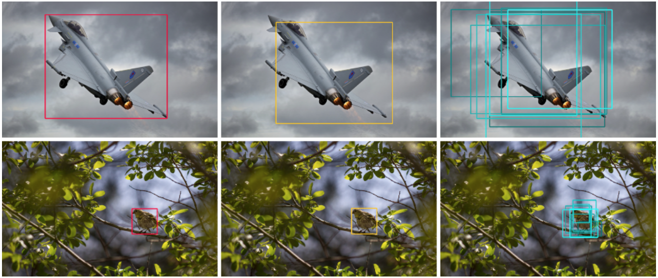
Use a Gaussian mixture (centered around the measurements) as proposal.

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 $p(y \mid x^\star; \theta)$

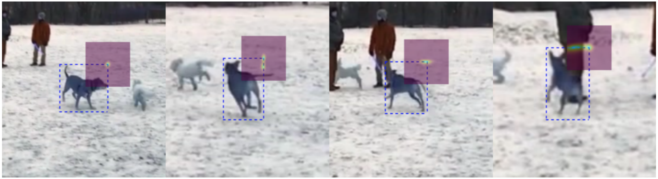
$$y^\star = \arg\max_{y} \; p(y \mid x^\star; \theta) = \arg\max_{y} \; f_\theta(x^\star, y).$$

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

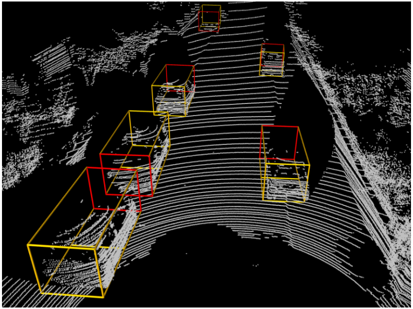
Task (visual tracking): Estimate a bounding box of a target object in every frame of a video. The target object is defined by a given box in the first video frame.



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), estimate their size and position in the 3D world.

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

The **combination** of **probabilistic models** and **deep neural networks** is very exciting and promising.

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.

Current developments

Problem: the proposal distribution $q(y \mid x)$ has to be designed manually.

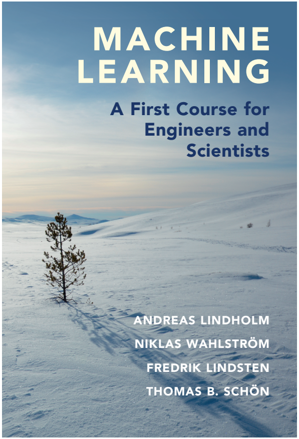
We have just been able to **automate** the learning of suitable proposals.

Result: Improved result with less tuning.

Should appear on arXiv next week.

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.



The book is freely available:

http://smlbook.org/

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

Conclusion

Aim: Create an awareness of how we can use deep neural networks for regression and show that energy-based models are useful in this context.

- Introduced an EBM for regression using DNNs
- The construction is generally applicable
- Solved the training problem using
  - Importance sampling
  - Generalized noise contrastive esimation
- State-of-the-art performance on challenging regression problems using images and laser point clouds.