Learning in the Physical World

Markus Kaiser
mrksr.de

7 November 2019

Siemens AG, Technical University of Munich

Who am I?

Collaborating with

- Prof. Thomas Runkler
- Clemens Otte
- Stefan Depeweg
- Daniel Hein

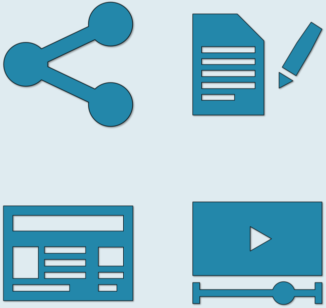
Collaborating with

- Prof. Carl Henrik Ek
- Prof. Neill Campbell
- Ieva Kazlauskaite
- Erik Bodin

Industrial applications are different

Web ML

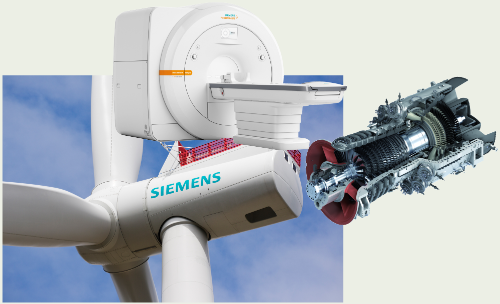
- People generate data
- Focus on the virtual world



Examples: E-commerce, web search, NLP

Industrial ML

- Machines generate data
- Focus on the physical world

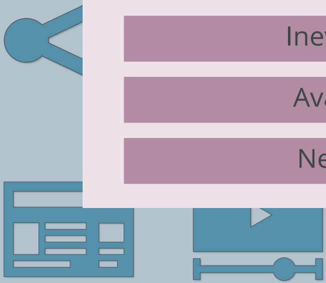


Examples: Machine control, commissioning

Industrial applications are different

Web ML

- People generate data
- Focus on the virtual world



Examples: E-commerce, web search, NLP

Industrial ML

- Machines generate data
- Focus on the physical world

Examples: Machine control, commissioning

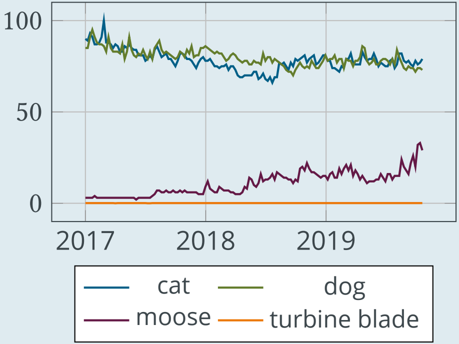
Main differentiators

- Inevitability of uncertainty
- Availability of knowledge
- Need for interpretability

Industrial data is scarce

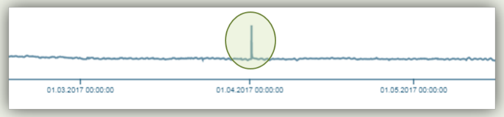
Web ML

- Model what people care about
- Relevant data is abundant



Industrial ML

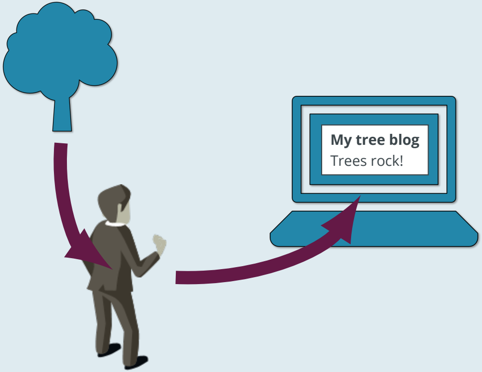
- Machines are designed not to fail
- Most valuable data is never produced



Understanding machine data is hard

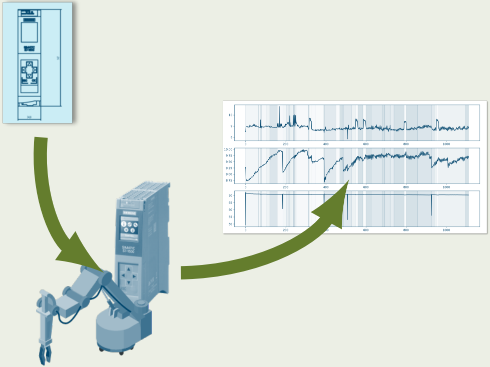
Web ML

- Data is mostly intuitive
- Lack of structure



Industrial ML

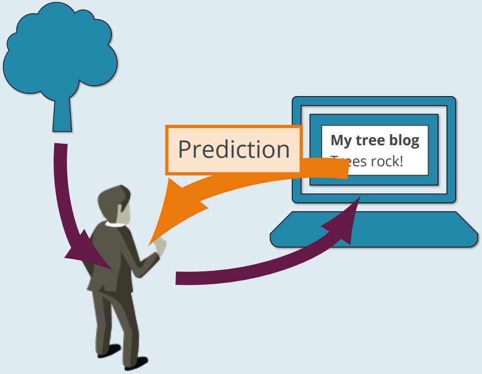
- Data requires domain knowledge
- Knowledge implies expectations



Industry needs interpretability

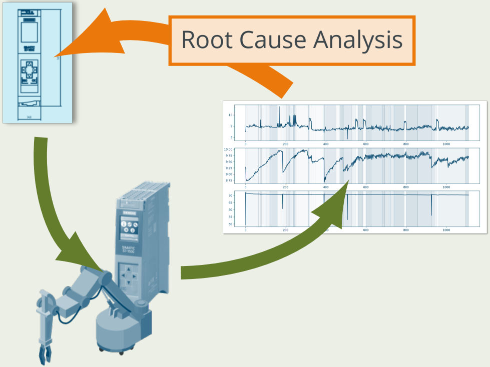
Web ML

- Inform or influence the person
- Understanding is secondary



Industrial ML

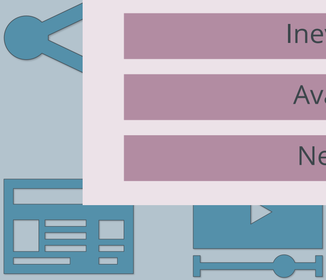
- Create better or safer machines
- Understanding is key



Industrial applications are different

Web ML

- People generate data
- Focus on the virtual world



Examples: E-commerce, web search, NLP

Industrial ML

- Machines generate data
- Focus on the physical world

Examples: Machine control, commissioning

Main differentiators

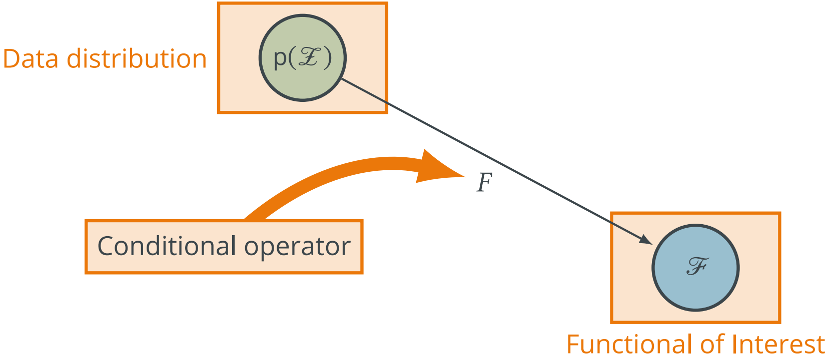
- Inevitability of uncertainty
- Availability of knowledge
- Need for interpretability

What is learning?

Statistical Learning Theory

- Given a **data distribution** $p(\mathscr{Z}) = p(\mathbf{x}, \mathbf{y})$
- **Find** $f \in \mathscr{F}$ such that

$$f(\mathbf{x}) \sim \mathbf{y}$$

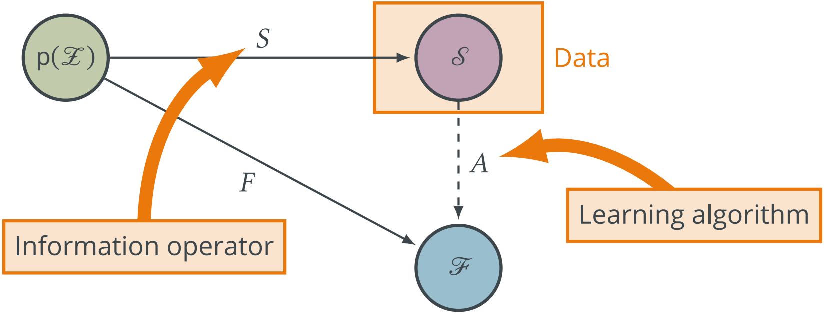


What is learning?

Statistical Learning Theory

- **Problem:** All we have are samples $\{(\mathbf{x}_1, \mathbf{y}_1), \ldots, (\mathbf{x}_N, \mathbf{y}_N)\}$ via S
- **Find a learning algorithm** $A : \mathscr{S} \to \mathscr{F}$ such that

$$A \circ S \approx F$$



A learning algorithm

Risk minimization

- Choose a **loss function** $\ell(f(\mathbf{x}), \mathbf{y})$ to measure closeness
- The **expected risk** R is given by

$$\mathrm{R}(f) := \int \ell(f(\mathbf{x}), \mathbf{y})\, \mathrm{p}(\mathbf{x}, \mathbf{y})\, \mathrm{d}\mathbf{x}\, \mathrm{d}\mathbf{y}$$

- **Learning algorithm:** Choose hypothesis space $\mathscr{H} \subseteq \mathscr{F}$ and use

$$\hat{f} \in \operatorname*{argmin}_{f \in \mathscr{H}} \mathrm{R}(f)$$

- Annoyingly, we cannot evaluate $\mathrm{R}(f)$ exactly!

A usable learning algorithm

Empirical risk minimization

- Approximate the **global** true risk with the **local empirical risk** in the training data

$$\mathrm{R}_{\mathrm{emp}}(f) := \frac{1}{N}\sum_{i=1}^{N} \ell(f(\mathbf{x}_i), \mathbf{y}_i)$$

- **Learning algorithm:** Choose hypothesis space $\mathscr{H} \subseteq \mathscr{F}$ and use

$$\hat{f} \in \operatorname*{argmin}_{f \in \mathscr{H}} \mathrm{R}_{\mathrm{emp}}(f)$$

- **Not trustworthy** in general!
- Need to encode knowledge

How to encode knowledge?

Places to encode knowledge

Information operator S	**Data** selection, feature engineering, data augmentation
Hypothesis space $\mathcal{H}$	Choice of **model**, architecture design
Loss function ℓ	Choice of norm, **regularization**
Optimization min	Choice of **optimizer**, initialization, parameter tuning

How to encode knowledge?

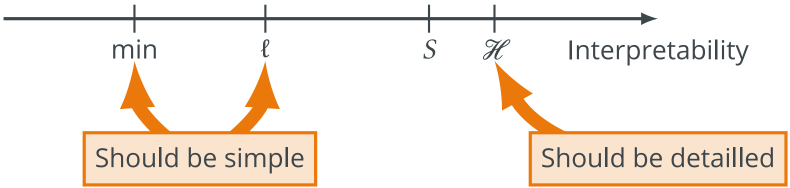
Places to encode knowledge

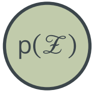
Information operator S	**Data** selection, feature engineering, data augmentation
Hypothesis space $\mathscr{H}$	Choice of **model**, architecture design
Loss function ℓ	Choice of norm, **regularization**
Optimization min	Choice of **optimizer**, initialization, parameter tuning

How to encode knowledge?

Places to encode knowledge

Information operator S	**Data** selection, feature engineering, data augmentation
Hypothesis space $\mathscr{H}$	Choice of **model**, architecture design
Loss function ℓ	Choice of norm, **regularization**
Optimization min	Choice of **optimizer**, initialization, parameter tuning

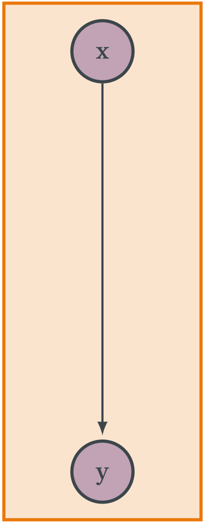




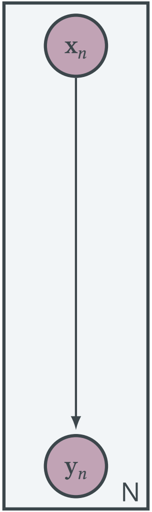
$$p(\mathscr{Z}) = p(\mathbf{x}, \mathbf{y})$$

$$p(\mathbf{x}, \mathbf{y}) = p(\mathbf{y} \,|\, \mathbf{x})\, p(\mathbf{x})$$

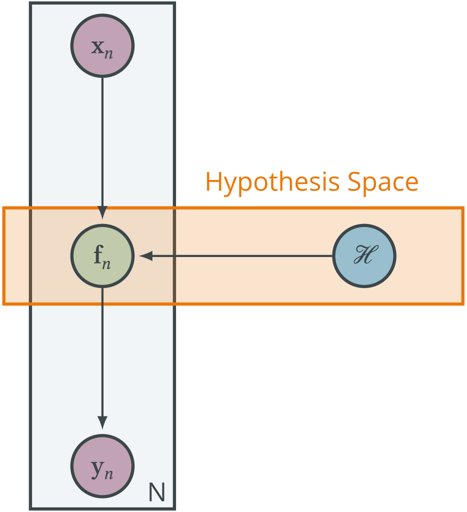
Generating Process



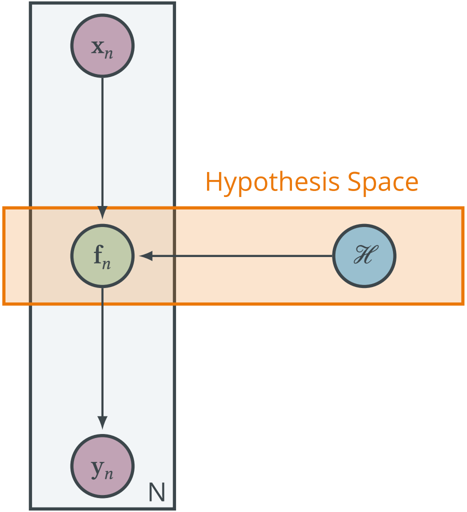
$$\mathrm{p}(\mathbf{x}, \mathbf{y}) \neq \prod_{n=1}^{N} \mathrm{p}(\mathbf{y}_n \mid \mathbf{x}_n)\, \mathrm{p}(\mathbf{x})$$



$$\mathrm{p}(\mathbf{x}, \mathbf{y}) = \mathrm{p}(\mathbf{x}) \cdot \prod_{n=1}^{N} \int \mathrm{p}(\mathbf{y}_n \,|\, \mathbf{f}_n)\, \mathrm{p}(\mathbf{f}_n \,|\, \mathbf{x}_n, \mathscr{H})\, \mathrm{p}(\mathscr{H})\, \mathrm{d}\mathscr{H}$$

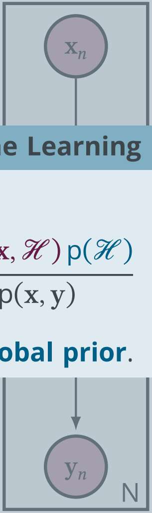


$$p(\mathbf{x}, \mathbf{y}) = p(\mathbf{x}) \cdot \prod_{n=1}^{N} \int p(\mathbf{y}_n \mid \mathbf{f}_n)\, p(\mathbf{f}_n \mid \mathbf{x}_n, \mathcal{H})\, p(\mathcal{H})\, d\mathcal{H}$$



Learning Algorithm: Find $p(\mathcal{H} \mid \mathbf{x}, \mathbf{y})$ that explains the data well.

Bayesian Machine Learning



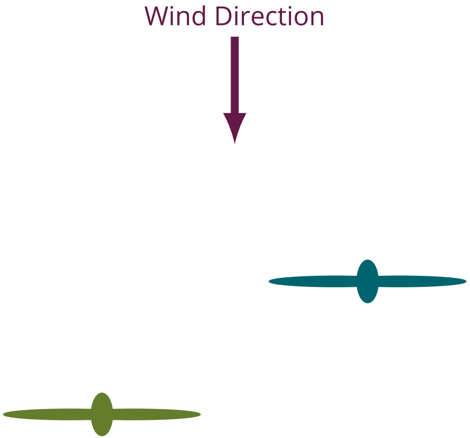
Bayesian Machine Learning

Find a compromise

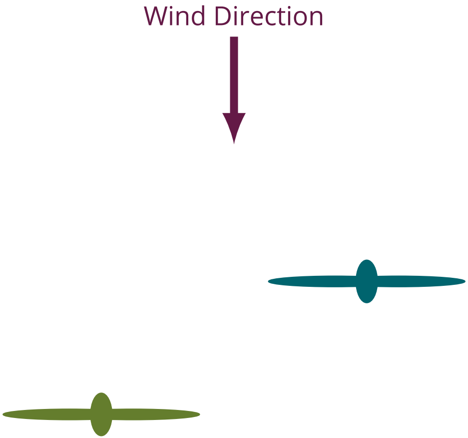
$$p(\mathcal{H} \mid \mathbf{x}, \mathbf{y}) = \frac{p(\mathbf{y} \mid \mathbf{x}, \mathcal{H}) \, p(\mathcal{H})}{p(\mathbf{x}, \mathbf{y})}$$

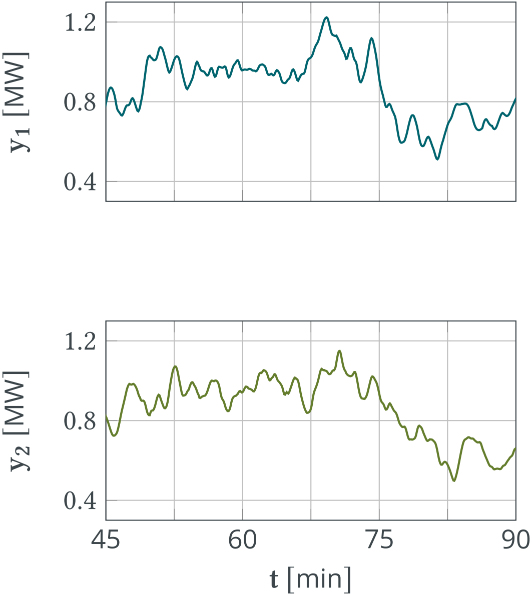
between the **local likelihood** and **global prior**.

Case study: Wind Propagation

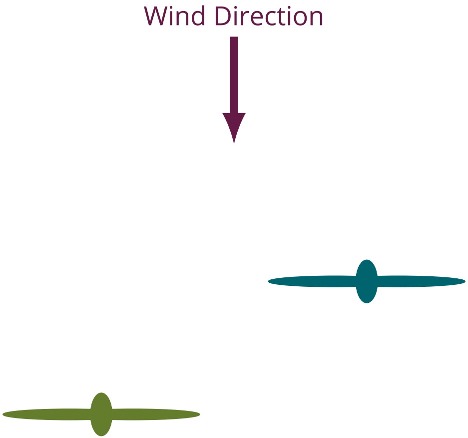


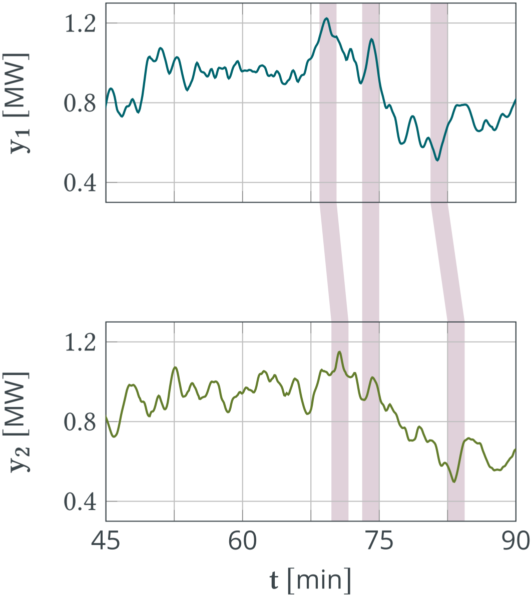
Case study: Wind Propagation



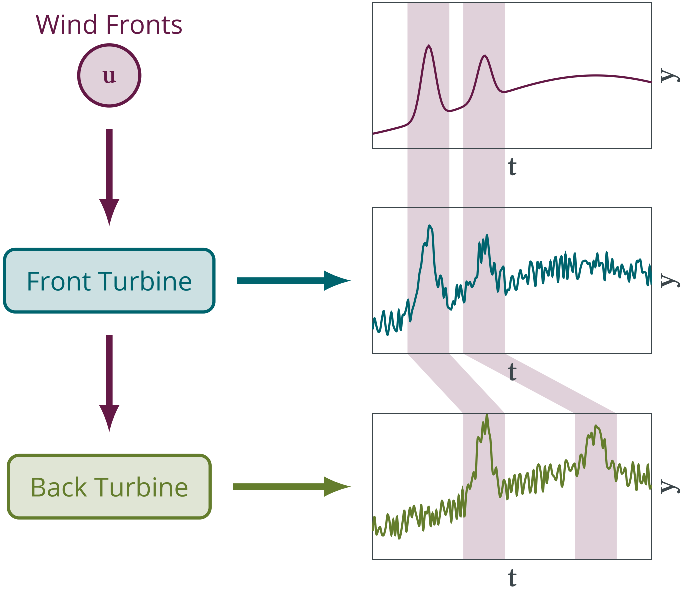


Case study: Wind Propagation

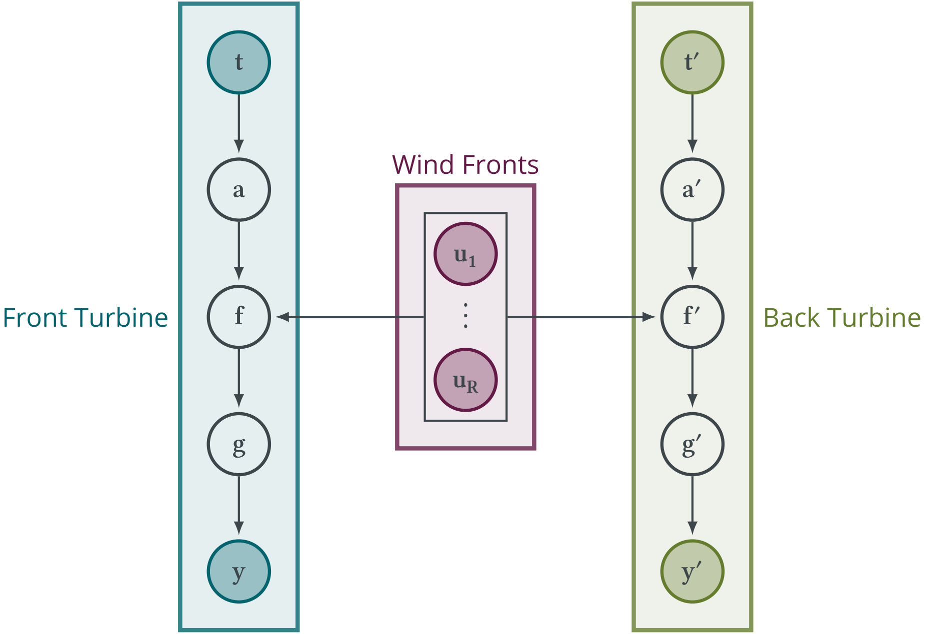




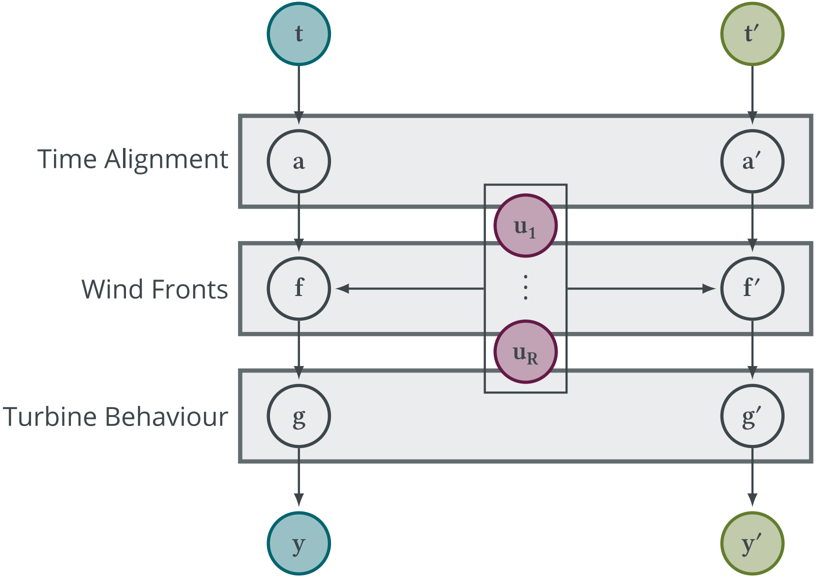
Generative process



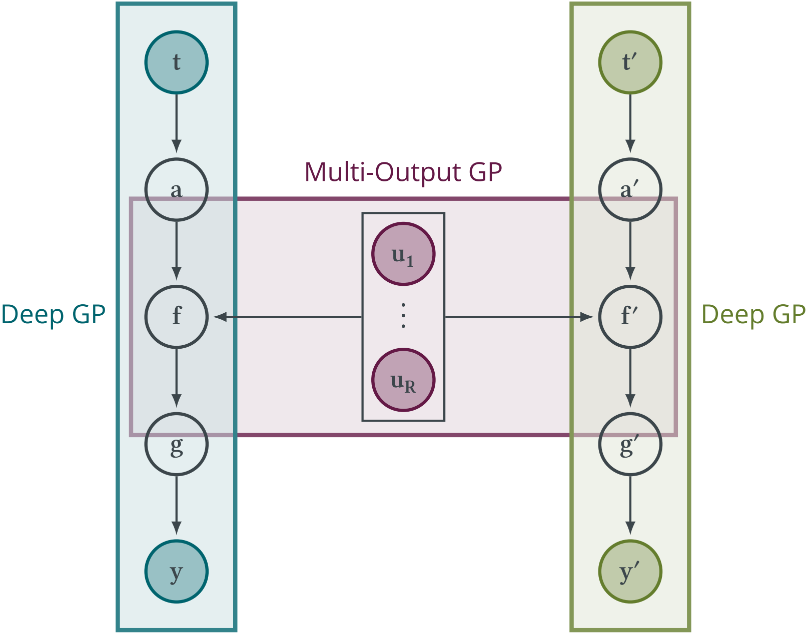
A Bayesian graphical model



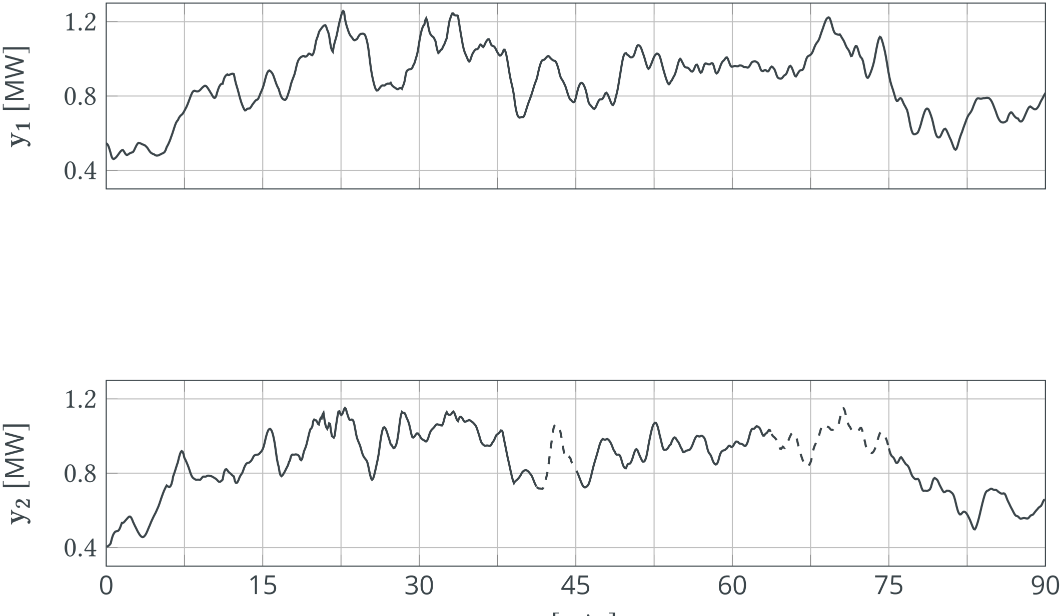
A Bayesian graphical model



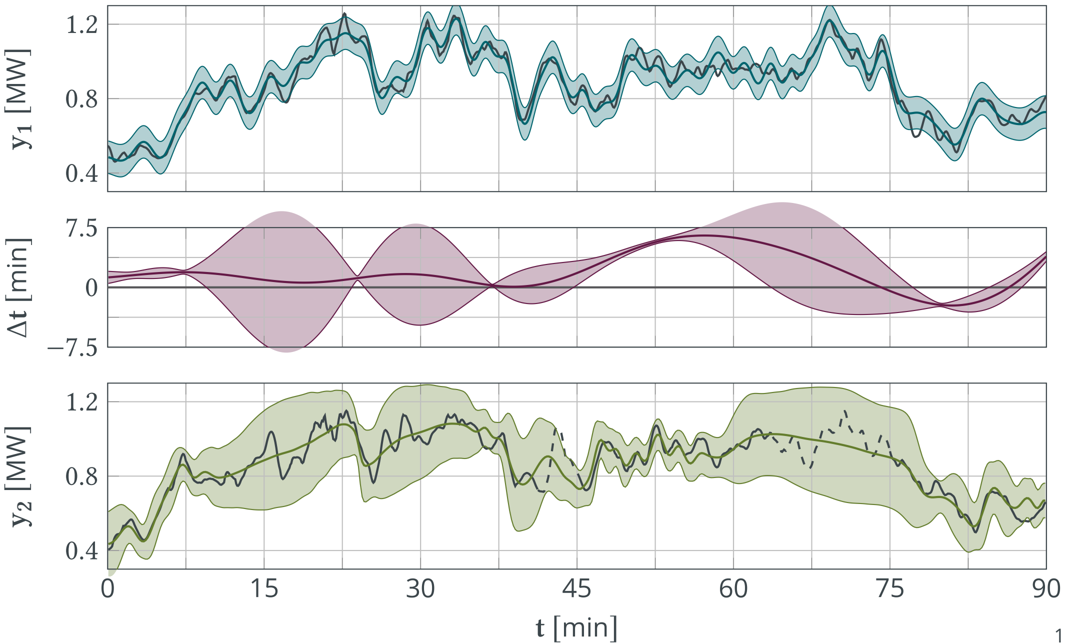
A Bayesian graphical model



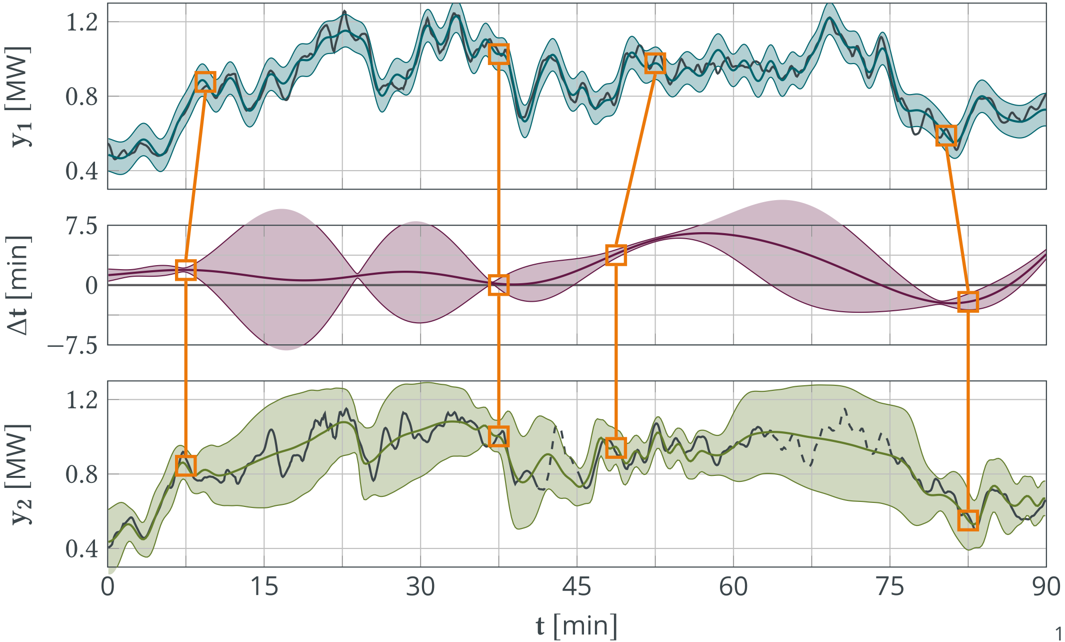
Posterior: Uncertain time alignment



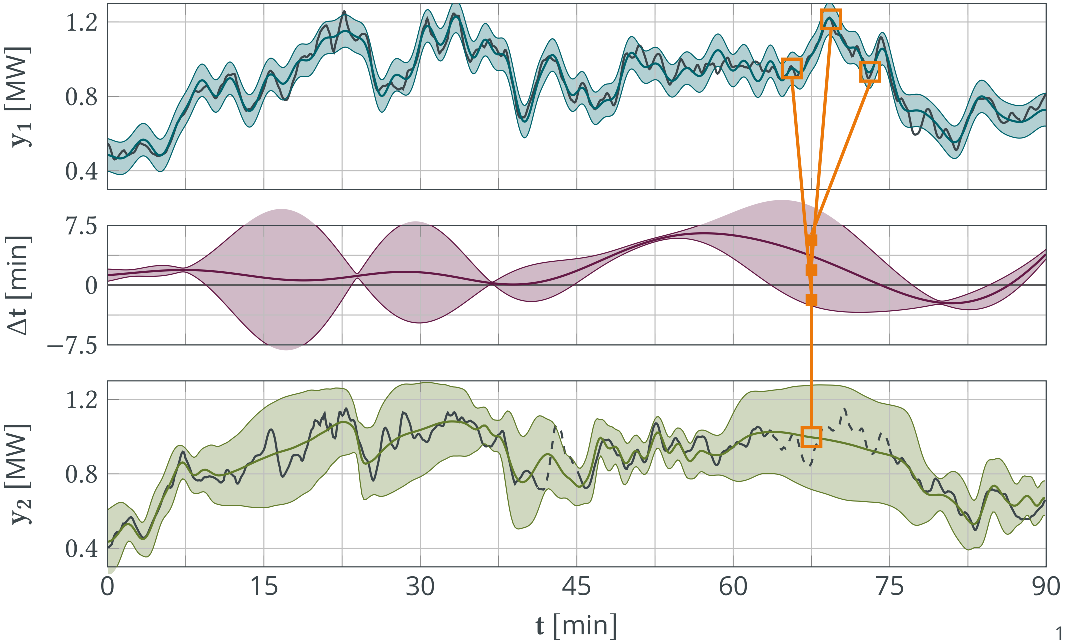
Posterior: Uncertain time alignment



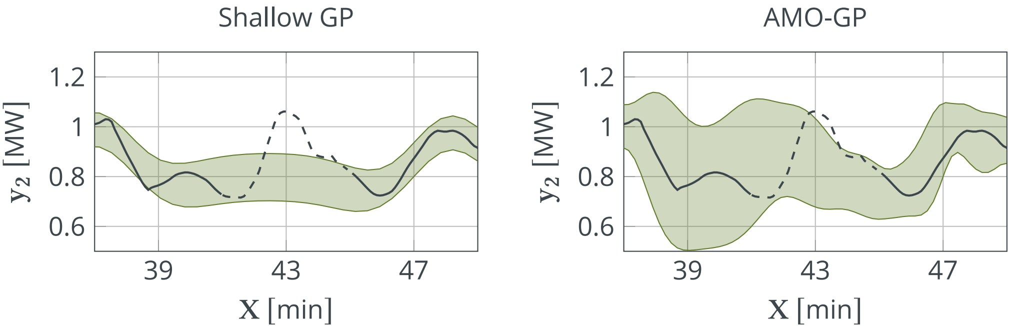
Posterior: Uncertain time alignment



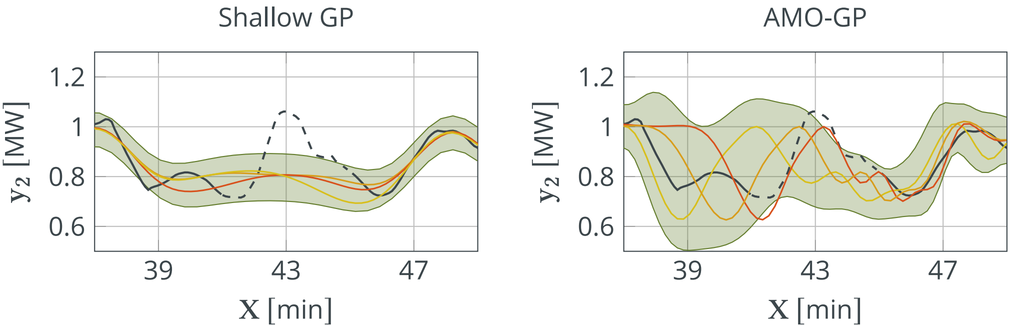
Posterior: Uncertain time alignment



Comparing samples



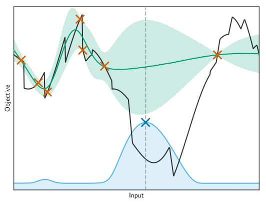
Comparing samples



Ongoing Research

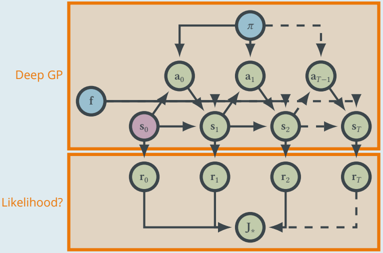
Bayesian Optimization[1]

- Reinterpretation of surrogates in BO
- Model global instead of local structure



Reinforcement Learning

- RL as a generative model
- Inference over optimal policy

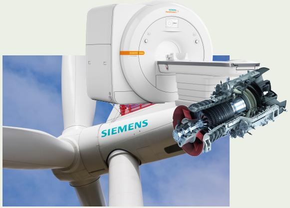


[1]Bodin et al. 2019.

Summary

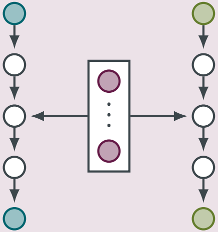
Industrial learning problems

- Inevitability of uncertainty
- Availability of knowledge
- Need for interpretability



Bayesian machine learning

- Inherent handling of uncertainties
- A system to formulate hypotheses
- Good fit for industrial problems

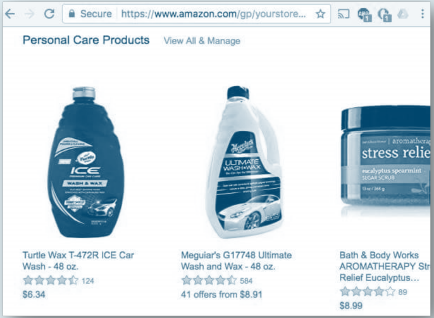


- Bodin, Erik et al. (Nov. 2, 2019). "Modulating Surrogates for Bayesian Optimization". In: arXiv: `1906.11152 [cs, stat]`.
- Breiman, Leo (2001). "Statistical Modeling: The Two Cultures (with Comments and a Rejoinder by the Author)". In: *Statistical science* 16.3, pp. 199–231.
- Kaiser, Markus et al. (2018). "Bayesian Alignments of Warped Multi-Output Gaussian Processes". In: *Advances in Neural Information Processing Systems 31*. Ed. by S. Bengio et al. Curran Associates, Inc., pp. 6995–7004.
- Murphy, Kevin P. (Aug. 24, 2012). *Machine Learning: A Probabilistic Perspective*. MIT Press. 1098 pp. ISBN: 978-0-262-01802-9.
- Trevor Hastie, Robert Tibshirani, and Jerome Friedman (Nov. 10, 2013). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer Science & Business Media. ISBN: 978-0-387-21606-5.

Exploration can be expensive

Web ML

- Exploration is cheap…
- …as are mistakes



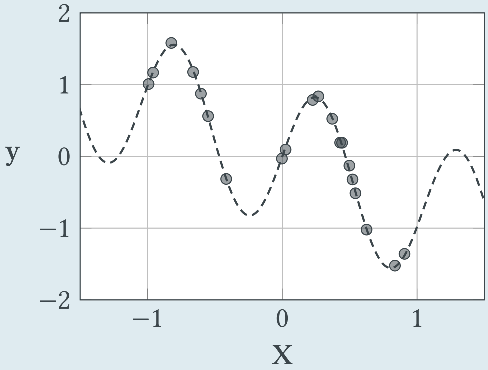
Industrial ML

- Exploration can be expensive…
- …and safety-critical

Models are known to be imperfect

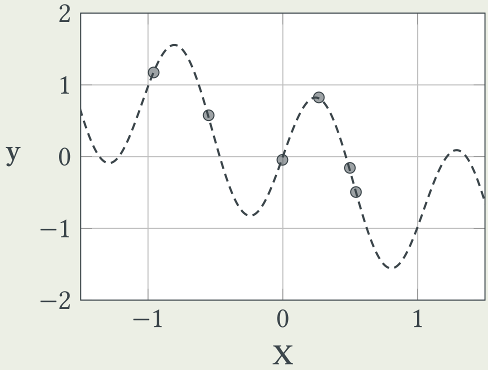
Web ML

- When in doubt, collect more data
- Uncertainties not so important



Industrial ML

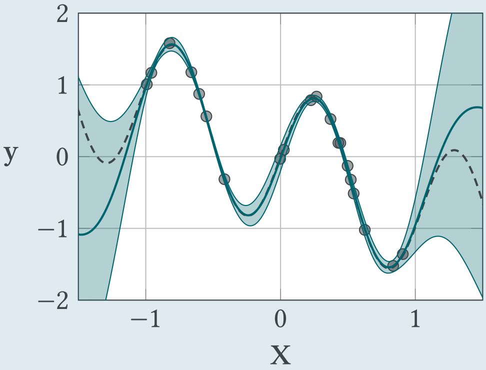
- Need to make do with given data
- Uncertainties are critical



Models are known to be imperfect

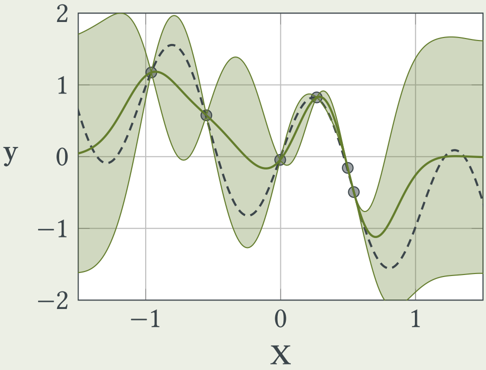
Web ML

- When in doubt, collect more data
- Uncertainties not so important



Industrial ML

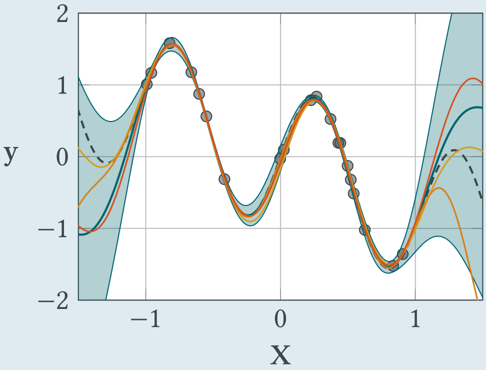
- Need to make do with given data
- Uncertainties are critical



Models are known to be imperfect

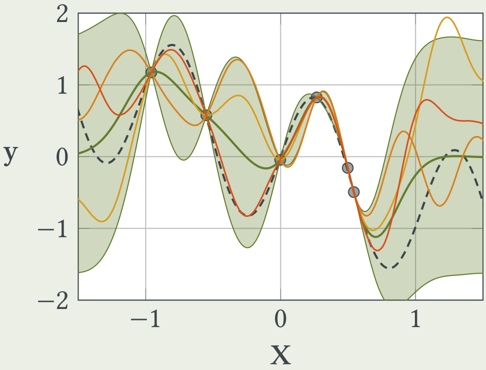
Web ML

- When in doubt, collect more data
- Uncertainties not so important



Industrial ML

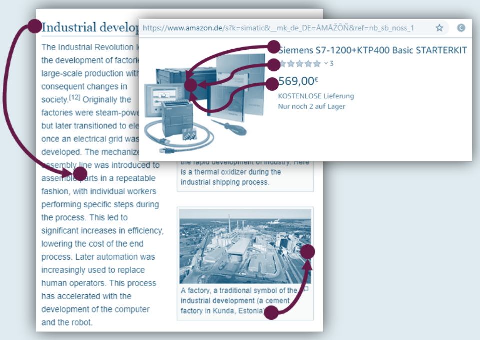
- Need to make do with given data
- Uncertainties are critical



Experts are needed to label data

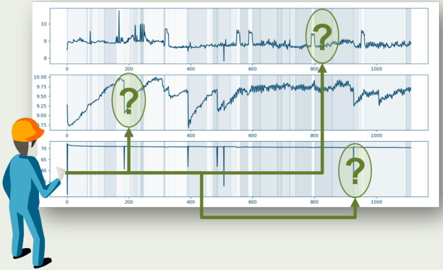
Web ML

- Contextual data available
- Annotation can be automated



Industrial ML

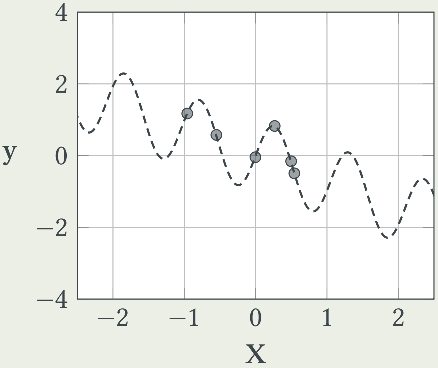
- Labels require domain experts
- Explicit and expensive

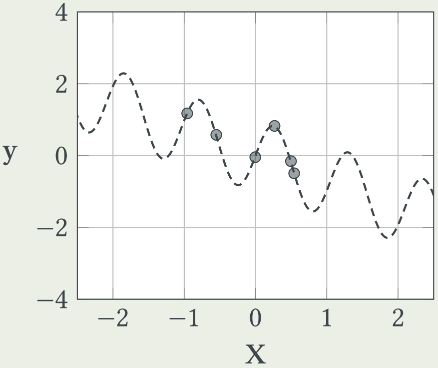


Generalization is necessary

Industrial ML

As data is scarce, **experts** need to tell us how to **generalize aggressively**.

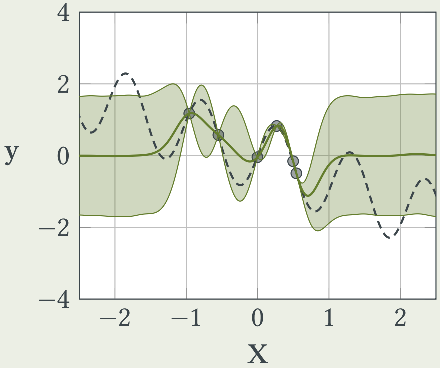


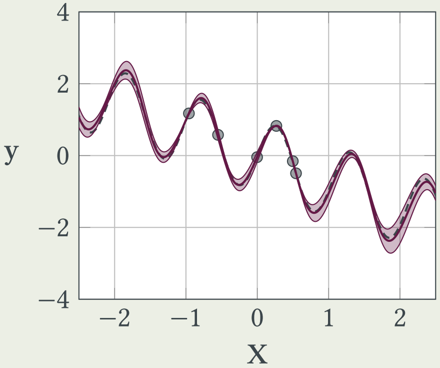


Generalization is necessary

Industrial ML

As data is scarce, **experts** need to tell us how to **generalize aggressively**.

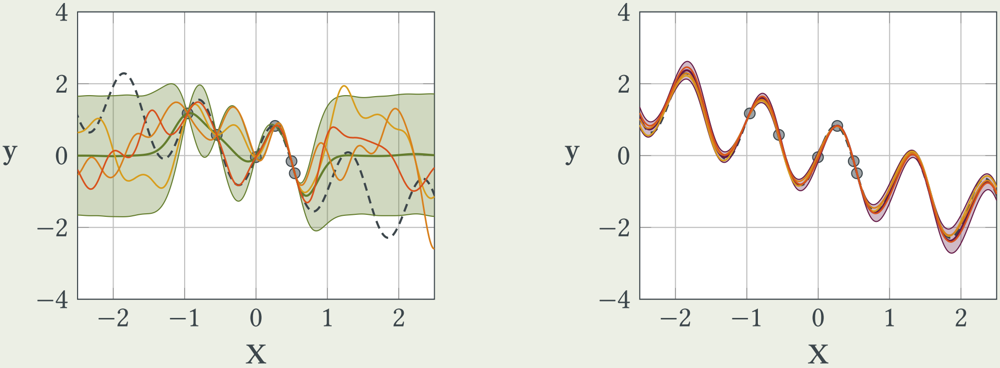




Generalization is necessary

Industrial ML

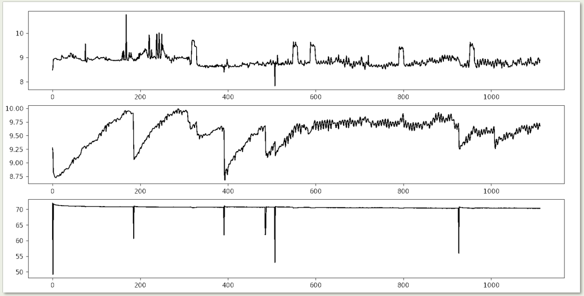
As data is scarce, **experts** need to tell us how to **generalize aggressively**.



Communication with experts

Industrial ML

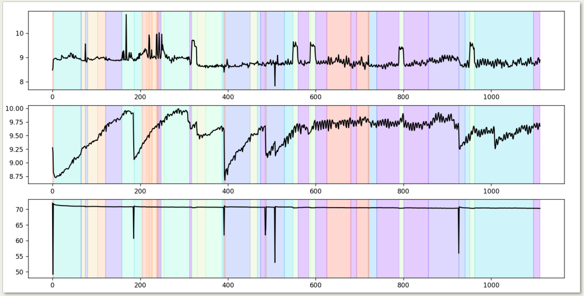
We need a **common language** with experts and **automate the boring things**.



Communication with experts

Industrial ML

We need a **common language** with experts and **automate the boring things**.



- Machine has different states
- State changes are rare
- Expert chooses relevant data