ML Tutorial

LLRF Workshop 2023

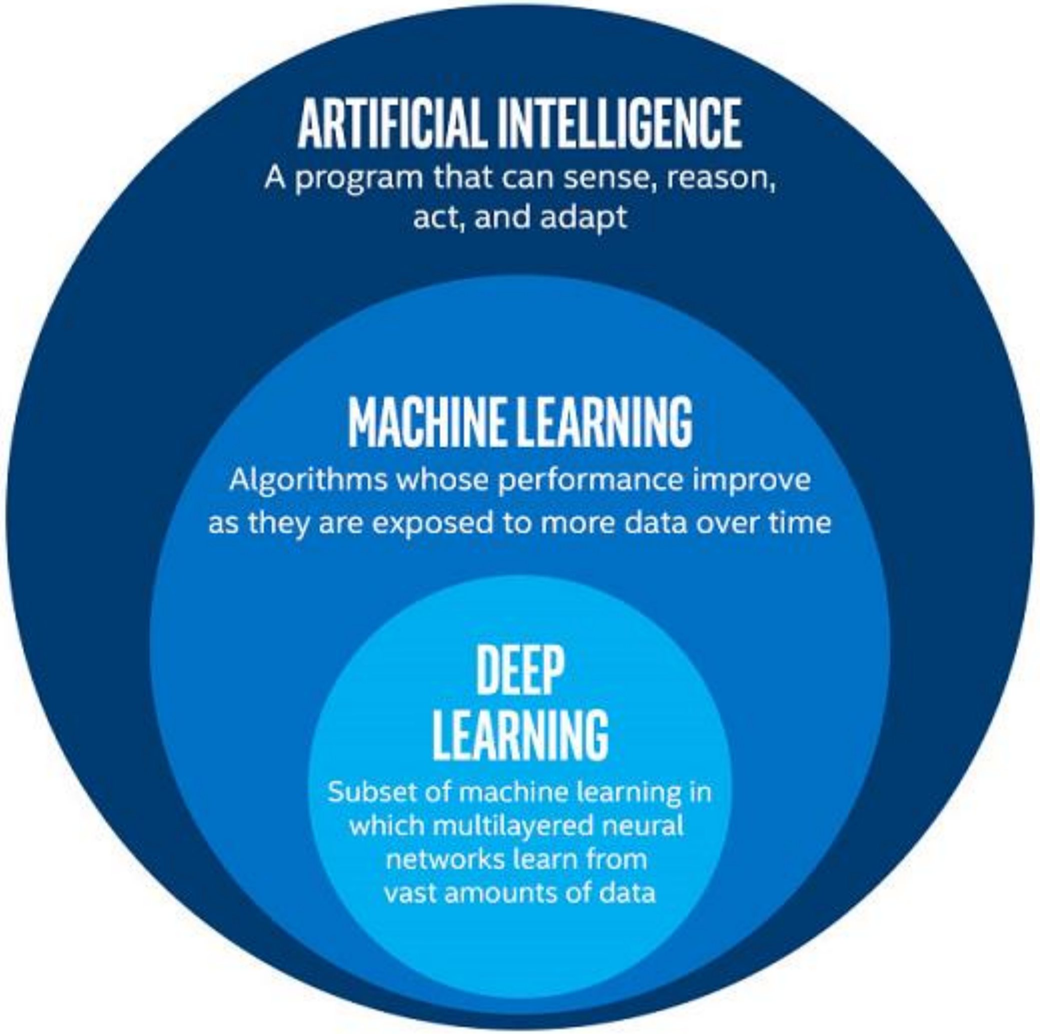
Annika Eichler
24.10.2023

HELMHOLTZ RESEARCH FOR GRAND CHALLENGES

Machine Learning

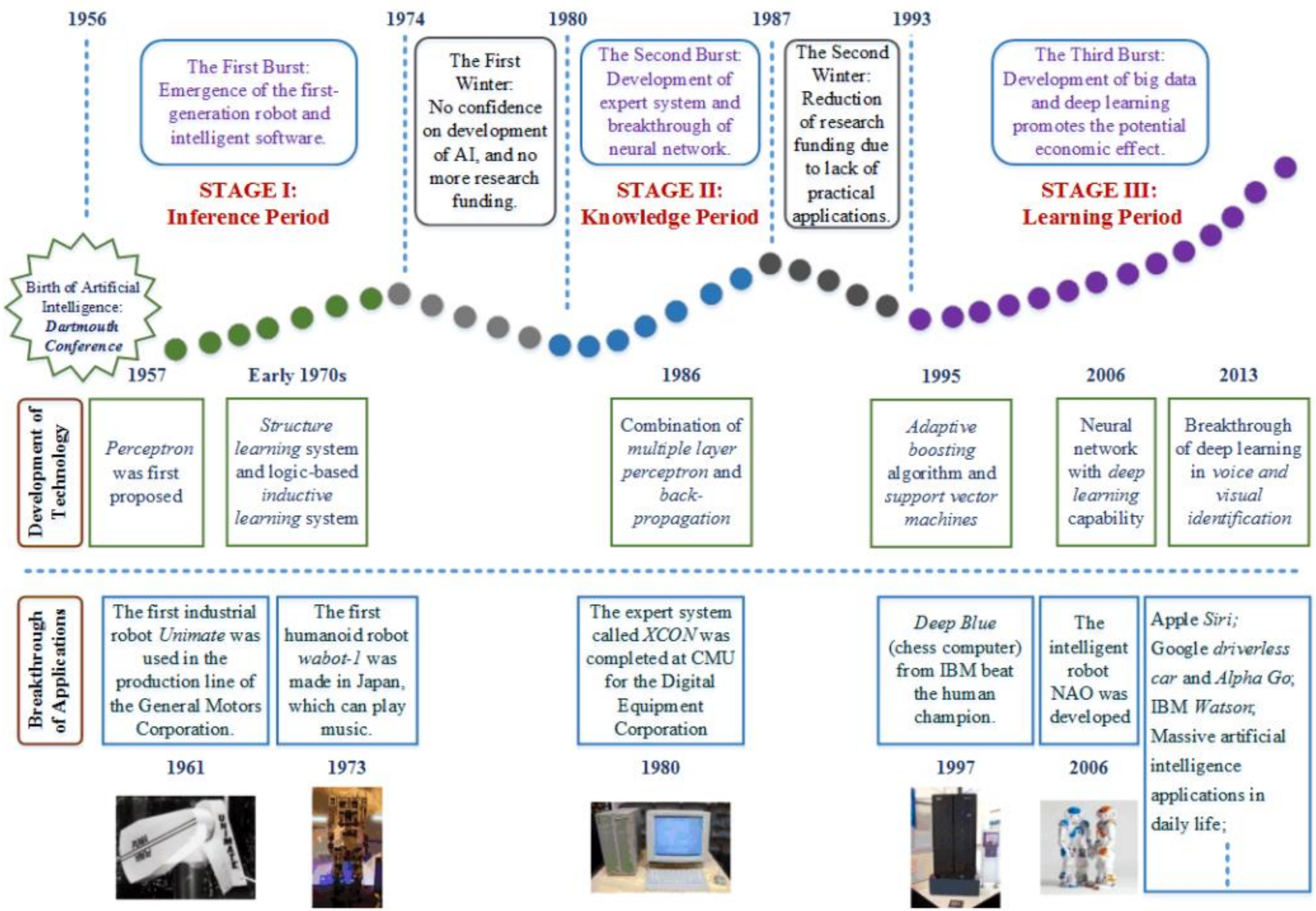
Definition

- Use and development of computer systems that are able to **learn and adapt** without following explicit instructions, by using **algorithms and statistical models** to analyze and draw **inferences from patterns in data**.
- Subfield of **artificial intelligence**



https://www.intel.com/content/www/us/en/developer/topic-technology/artificial-intelligence/overview.html

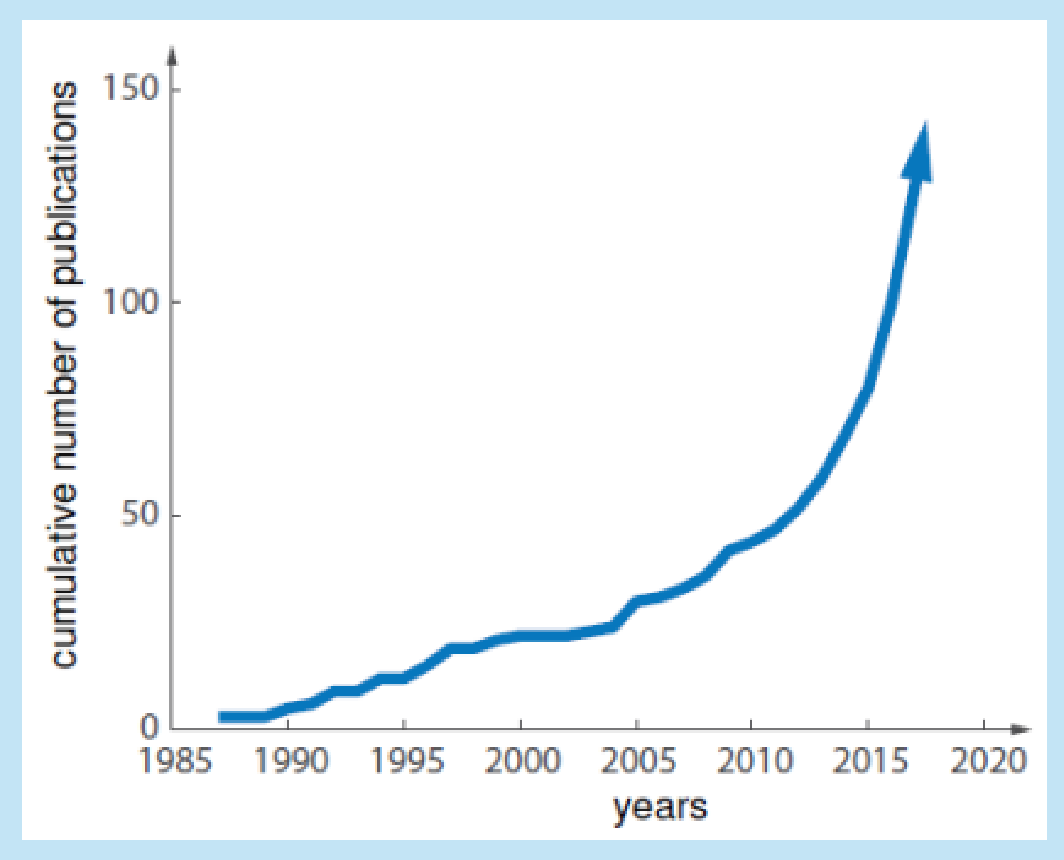
History of AI



[(5) (PDF) State-of-the-Art Mobile Intelligence: Enabling Robots to Move Like Humans by Estimating Mobility with Artificial Intelligence (researchgate.net)]

ML for accelerators

How it developed



Literature review
The analysis of the literature review in the field of AI, machine learning and advanced control methods for the application of particle accelerators is shown. The literature review presented here does not claim to be exhaustive but shows an obvious trend of the exponential increase of interest in the field. [Status: October 2018]

ICFA Workshop on Machine Learning for Charged Particle Accelerators

- 2018, SLAC, US
  Machine Learning Applications for Particle Accelerators | This is the Site Slogan (stanford.edu)
- 2019, PSI, Switzerland
  2nd ICFA Workshop on Machine Learning for Charged Particle Accelerators (February 26, 2019 - March 1, 2019) · Indico (psi.ch)
- 2022, Brookhaven, US
  3rd ICFA Beam Dynamics Mini-Workshop on Machine Learning Applications for Particle Accelerators (bnl.gov)

Seminar series OWLE (The One World charged particle accelerator Colloquium & Seminar Series)

- OWLE Seminar Series - Past ML Seminars (google.com)
- OWLE Seminar Series - Past Colloquiums (google.com)

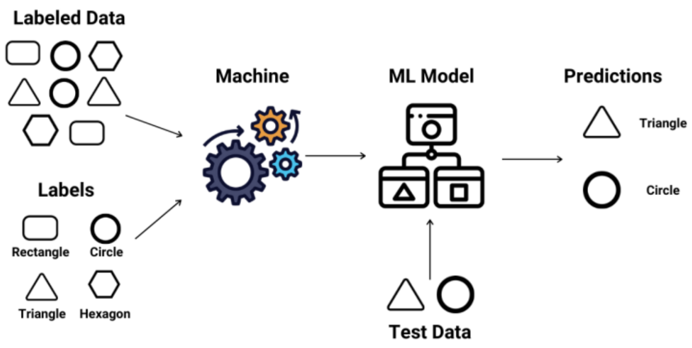
Brief introduction ML

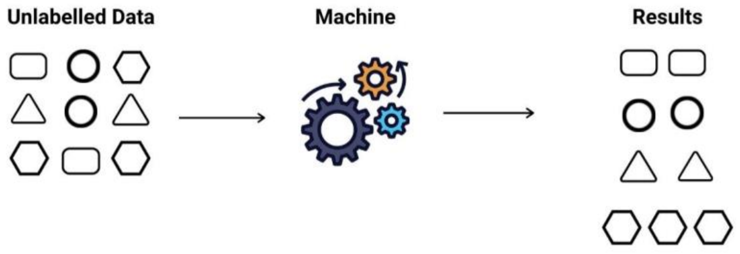
Machine Learning

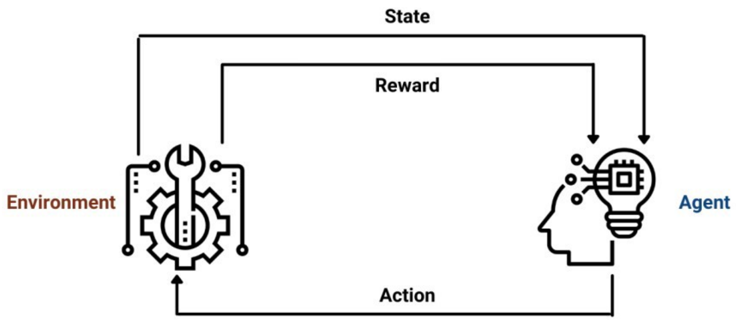
Types and processes

Machine Learning

Supervised Learning Inputs and outputs are known (labeled data)	Unsupervised Learning Inputs are known (unlabeled data)	Reinforcement Learning trial and error (learning from interactions with the environment)







- Labeled or classified data for training
- Apply what has been learned in the past to new data using labeled data to predict future events.

- Neither classified nor labeled data for training
- Infer a function to describe a hidden structure from unlabeled data

- Interacts with its environment by producing actions and discovers feedback errors or rewards
- Trial and error search to automatically determine the ideal behavior within a specific context in order to maximize its performance.

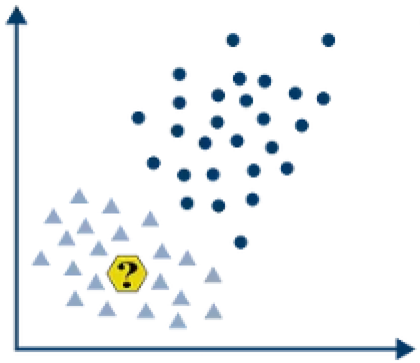
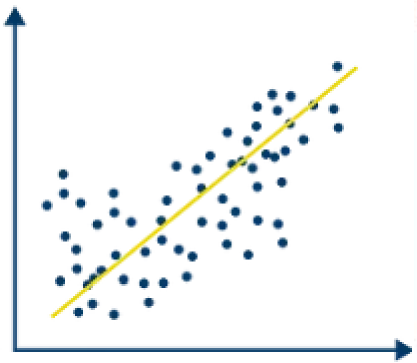
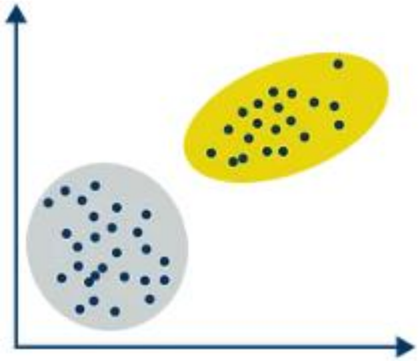
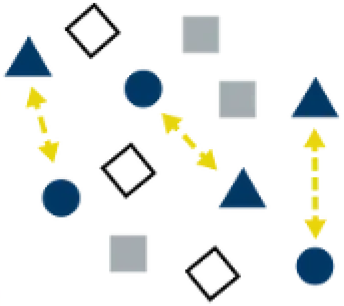
Machine Learning

Types and processes

[7 of the Most Used Regression Algorithms and How to Choose the Right One | by Dominik Polzer | Towards Data Science]

Machine Learning

Supervised Learning Inputs and outputs are known (labeled data)		Unsupervised Learning Inputs are known (unlabeled data)		Reinforcement Learning trial and error (learning from interactions with the environment)
Classification discrete outputs	Regression continuous outputs	Clustering finding relationship among data points	Association finding relationship among features of data points	

Machine Learning

Types and processes

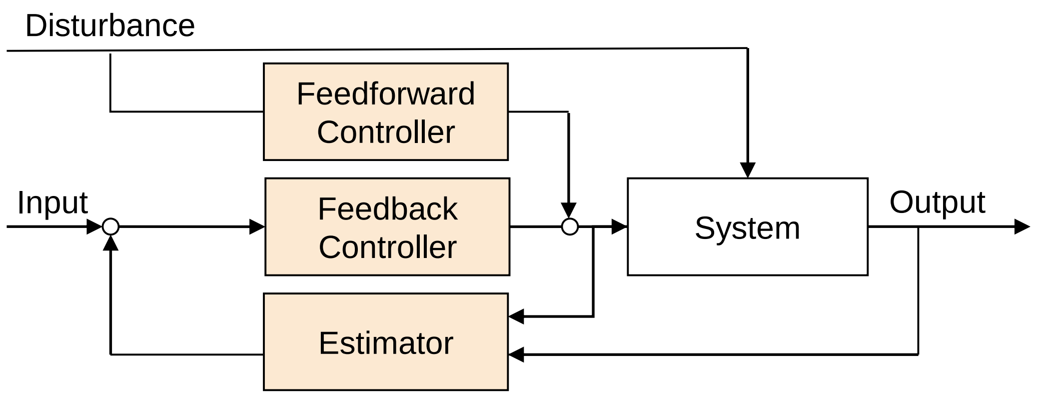
Machine Learning

Supervised Learning Inputs and outputs are known (labeled data)		Unsupervised Learning Inputs are known (unlabeled data)		Reinforcement Learning trial and error (learning from interactions with the environment)
Classification discrete outputs	Regression continuous outputs	Clustering finding relationship among data points	Association finding relationship among features of data points	• Model-free RL • Model-based RL (learn the model/given the model) • Policy optimization • Q-learning
• Linear & logistic regression • Support vector machines • Random forest • Neural networks • Decision trees • Naïve Bayes • Nearest neighbor		• Hierarchical clustering • K-means clustering • PCA • t-SNE • Apriori algorithm for association • Autoencoder • Anomaly detection • Dimensionality reduction • Isolation forest		

Machine Learning vs Control

Basics of control

- Control theory is a branch of Applied Mathematics dealing with the use of **feedback** to influence the behavior of a dynamical **system** in order **to achieve a desired goal**.
- A control algorithm is **a mathematical-logical action specification for the work of a controller**.



Feedback

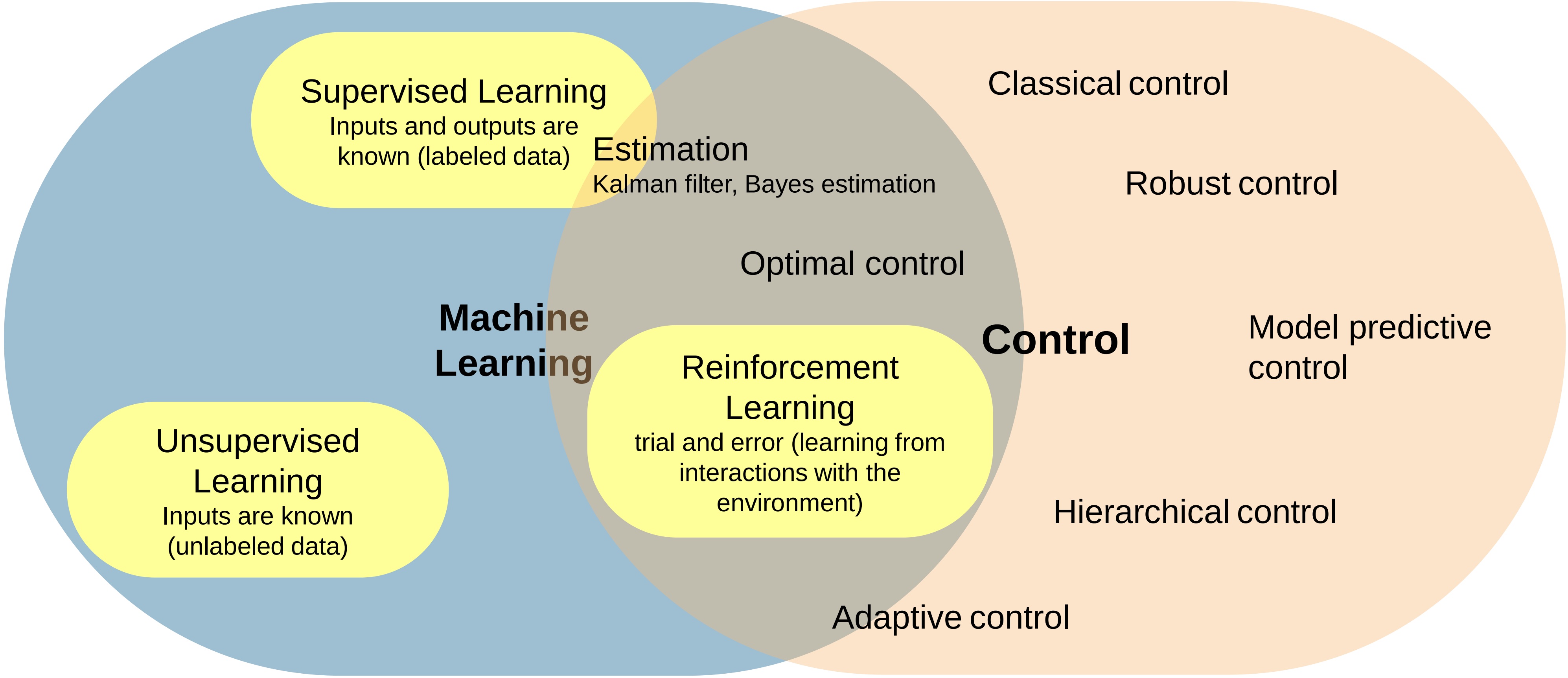
- Corrective action in case of set point deviation.
- Requires minimal knowledge about the system.
- No predictive control action to compensate for the effects of known or measurable disturbances.

Feedforward

- Measures disturbances variables and take corrective proactively.
- Disturbance variables must be measured on-line
- Approximate process model should be available.

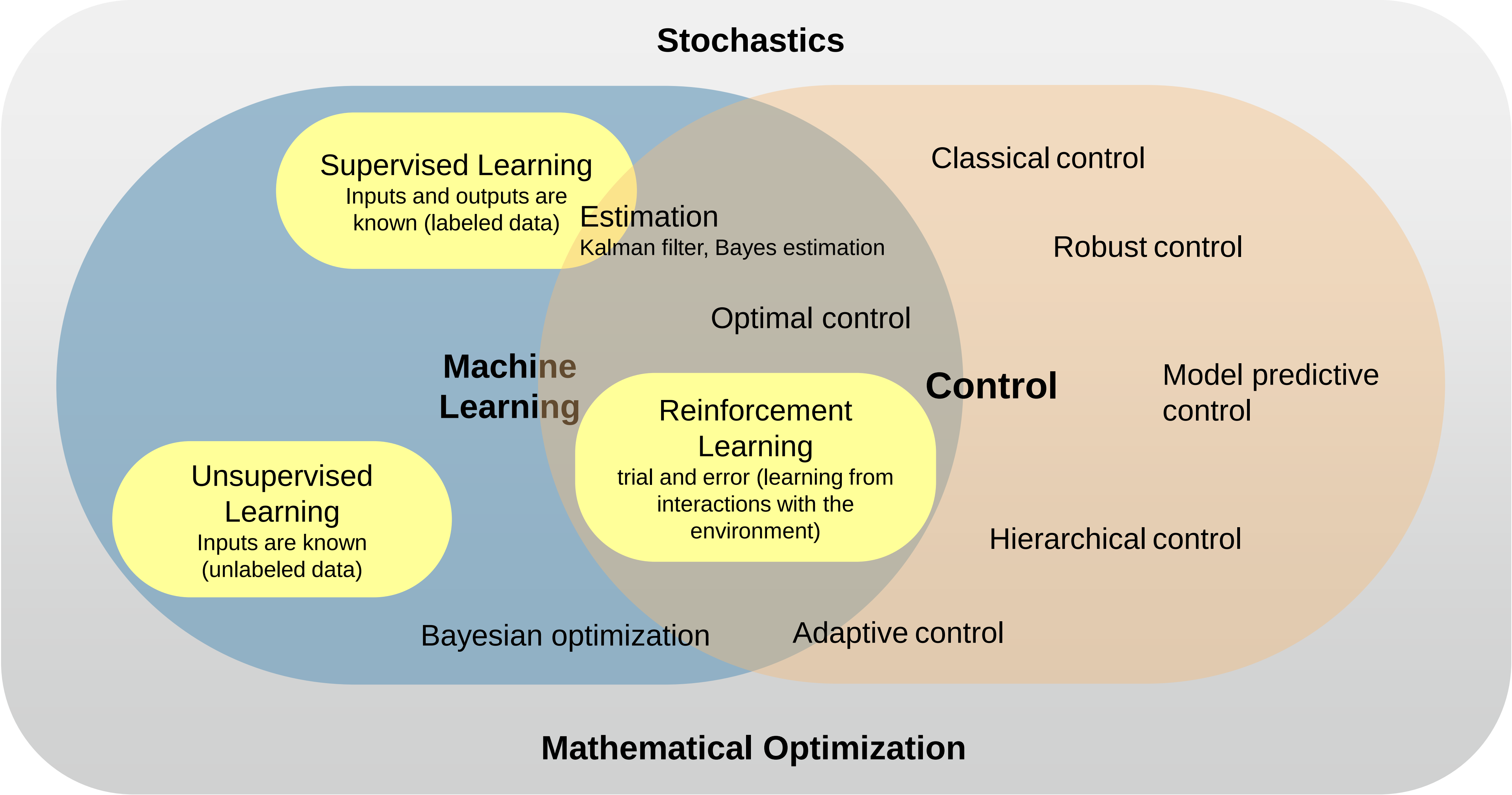
Machine Learning vs Control

They are not so different



Machine Learning vs Control

Optimization and Stochastics



ML for controlling accelerators

ML for Accelerators

What are the most important fields?

Data analysis

- Understanding physics
- Find new correlations of parameters
- Identify relevant data channels

→ New physical insight

Estimating and predicting

Surrogate models

→ Models for online control and optimization, and for accelerator design

Virtual diagnostics

→ Additional, nondestructive, (online) information

Fault diagnosis

- Predict & prevent failures
- Protect the system
- Identify poor conditions
- Find the root cause of errors encountered

→ Improve the availability/ reliability of machine operation

Tuning and control

- Exploit data to retrieve desired machine settings
- Push the way of operation
- Optimize performance

→ Better performance for users

[1811.03172] Opportunities in Machine Learning for Particle Accelerators (arxiv.org) 2018

ML for Accelerators

What are the most important fields?

Data analysis

- Understanding physics
- Find new correlations of parameters
- Identify relevant data channels

→ New physical insight

Estimating and predicting

Surrogate models

→ Models for online control and optimization, and for accelerator design

Virtual diagnostics

→ Additional, nondestructive, (online) information

Fault diagnosis

- Predict & prevent failures
- Protect the system
- Identify poor conditions
- Find the root cause of errors encountered

→ Improve the availability/ reliability of machine operation

Tuning and control

- Exploit data to retrieve desired machine settings
- Push the way of operation
- Optimize performance

→ Better performance for users

For modeling and validation

For data-based approaches (developing and testing)

[1811.03172] Opportunities in Machine Learning for Particle Accelerators (arxiv.org) 2018

ML for Accelerators

What are the most important fields?

Data analysis

- Understanding physics
- Find new correlations of parameters
- Identify relevant data channels

→ New physical insight

Estimating and predicting

Surrogate models

→ Models for online control and optimization, and for accelerator design

Virtual diagnostics

→ Additional, nondestructive, (online) information

Fault diagnosis

- Predict & prevent failures
- Protect the system
- Identify poor conditions
- Find the root cause of errors encountered

→ Improve the availability/ reliability of machine operation

Tuning and control

- Exploit data to retrieve desired machine settings
- Push the way of operation
- Optimize performance

→ Better performance for users

For model-based approaches
For developing and testing

[1811.03172] Opportunities in Machine Learning for Particle Accelerators (arxiv.org) 2018

ML for Accelerators

What are the most important fields?

Data analysis	Estimating and predicting	Fault diagnosis	Tuning and control
• Understanding physics • Find new correlations of parameters • Identify relevant data channels → New physical insight	Surrogate models → Models for online control and optimization, and for accelerator design Virtual diagnostics → Additional, nondestructive, (online) information	• Predict & prevent failures • Protect the system • Identify poor conditions • Find the root cause of errors encountered → Improve the availability/ reliability of machine operation	• Exploit data to retrieve desired machine settings • Push the way of operation • Optimize performance → Better performance for users
• Unsupervised Learning	• Supervised Learning	• Supervised Learning • Unsupervised Learning • (Statistics/Control)	• Reinforcement Learning • Optimization • (Control)

[1811.03172] Opportunities in Machine Learning for Particle Accelerators (arxiv.org) 2018

Algorithms …

Machine Learning

Types and processes

Machine Learning

Supervised Learning Inputs and outputs are known (labeled data)		Unsupervised Learning Inputs are known (unlabeled data)		Reinforcement Learning trial and error (learning from interactions with the environment)
Classification discrete outputs	Regression continuous outputs	Clustering finding relationship among data points	Association finding relationship among features of data points	• Model-free RL • Model-based RL (learn the model/given the model) • Policy optimization • Q-learning
• Linear & logistic regression • Support vector machines • Random forest • Neural networks • Decision trees • Naïve Bayes • Nearest neighbor		• Hierarchical clustering • K-means clustering • PCA • t-SNE • Apriori algorithm for association • Autoencoder • Anomaly detection • Dimensionality reduction • Isolation forest		

Bayesian Optimization

Linear Regression

Fit a linear function to the data points (x_i, y_i)

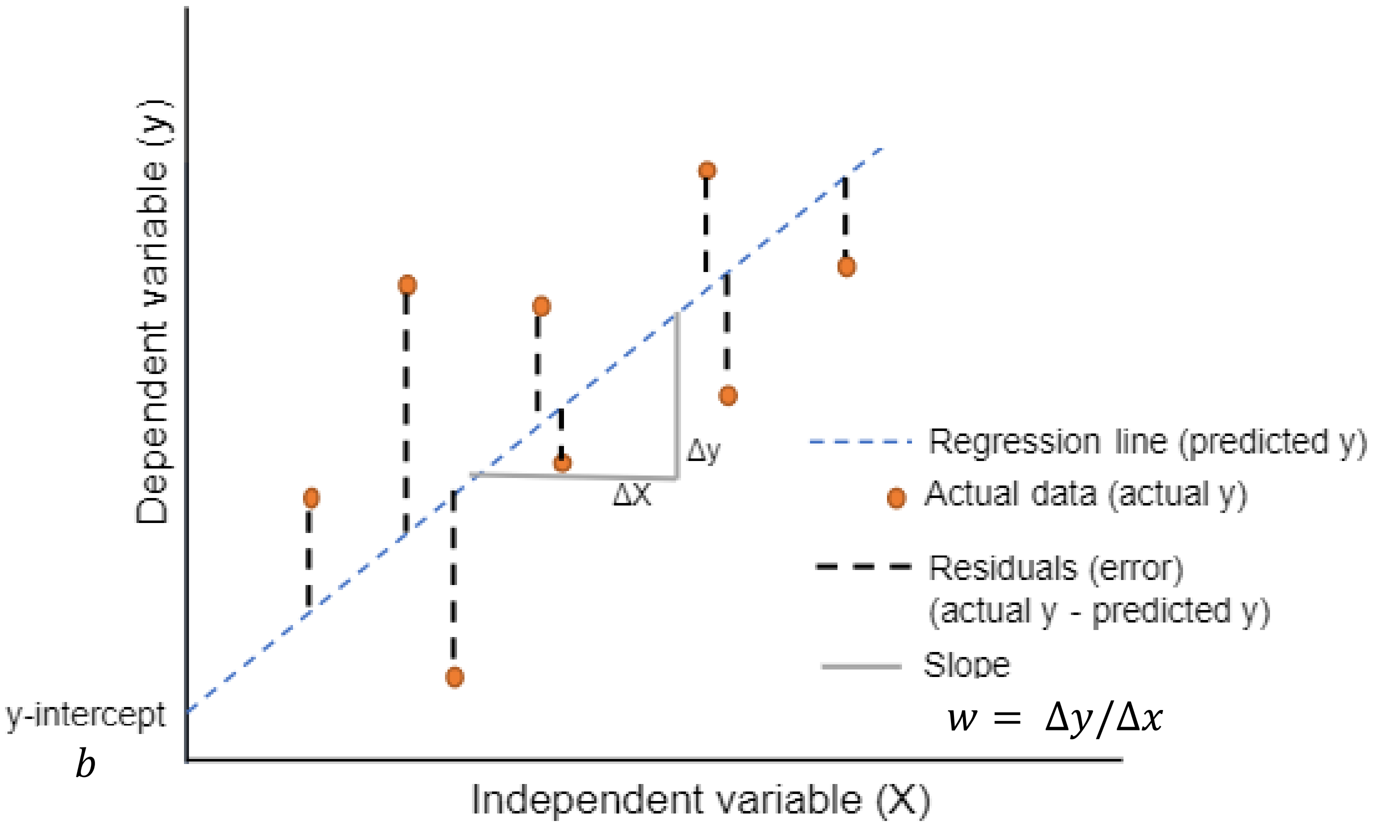
x: independent variables (inputs)

y: dependent variables (outputs)

Function model: $h(x) = b + wx$

b : bias

w : weight



Find b and w such $r_i = h(x_i) - y_i$ is as small as possible for all i

$$\min_{b,w} \frac{1}{2} \sum_{i=1}^{n} (h(x_i) - y_i)^2)$$

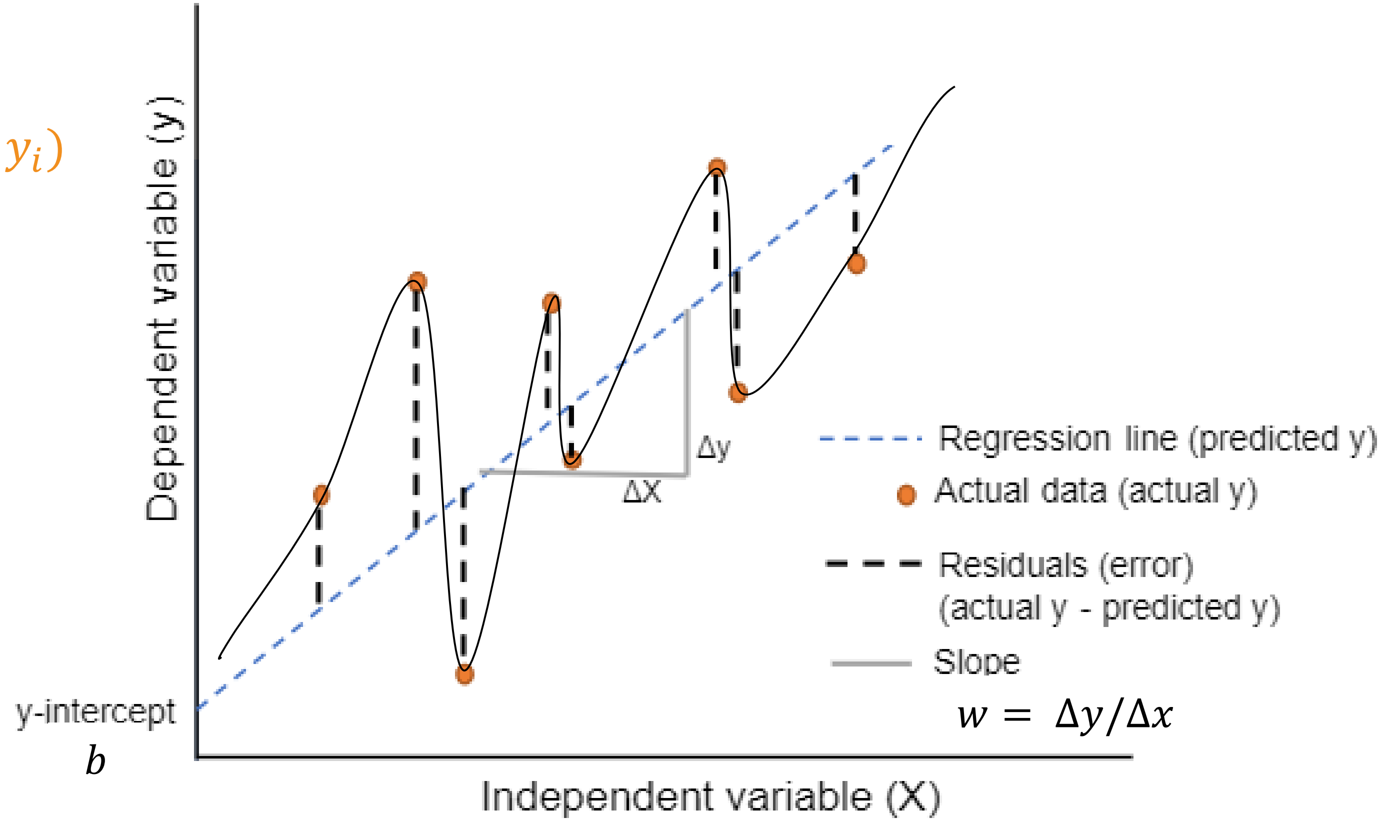
Loss (cost) function
$J(b, w)$

Nonlinear Regression

Fit a nonlinear function to the data points (x_i, y_i)

x: independent variables (inputs)

y: dependent variables (outputs)



Function model: $h(x) = b + w\phi(x)$

b : bias

w : weight

Find b and w such $r_i = h(x_i) - y_i$ is as small as possible for all i

$$\min_{b,w} \left(\frac{1}{2} \sum_{i=1}^{n} (h(x_i) - y_i)^2 \right)$$

Loss (cost) function

$J(b, w)$

Example

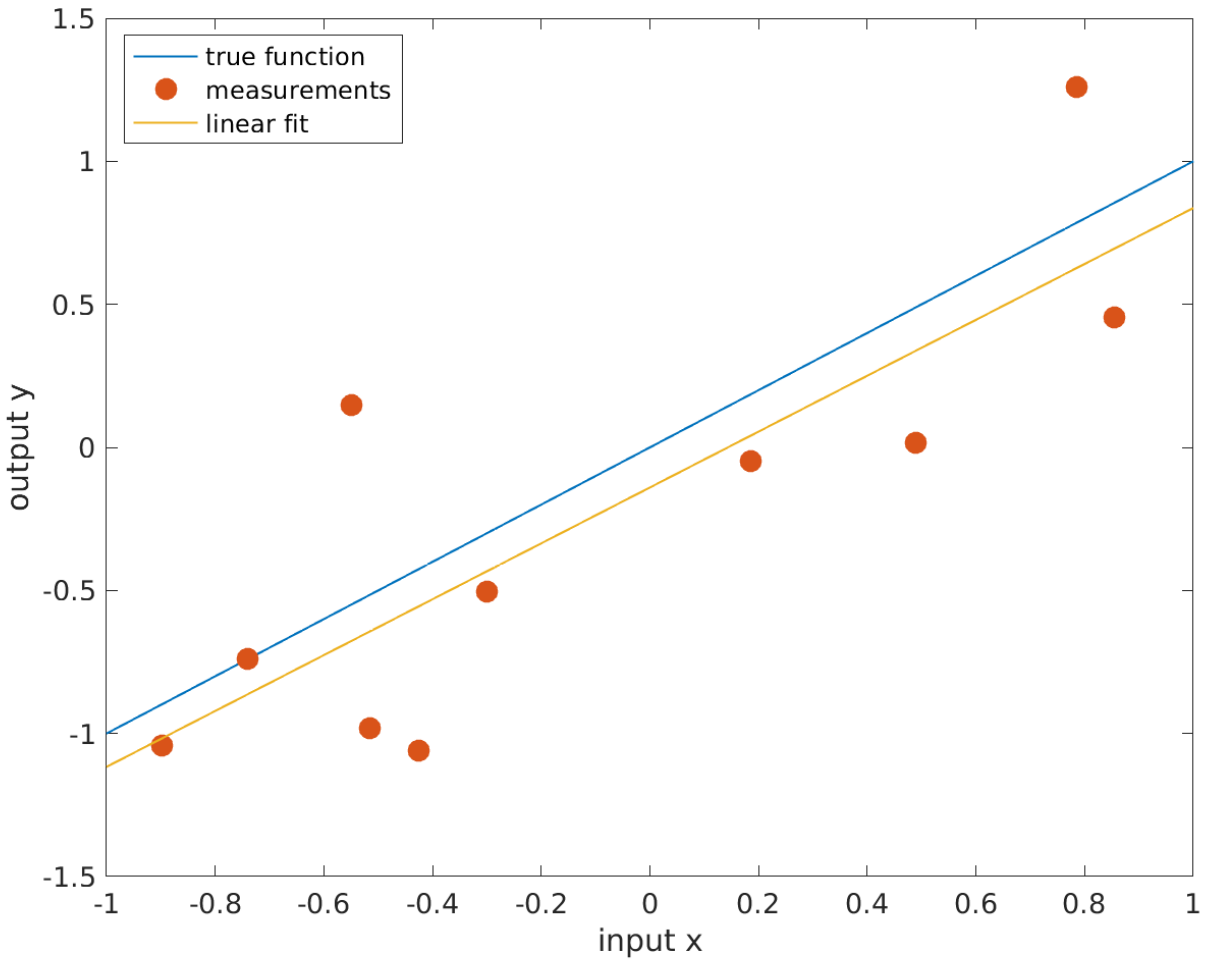
Linear regression

- One dimensional function: $h(x) = wx + b$: $\mathbb{R} \rightarrow \mathbb{R}$
- 10 measurements available:
  - Inputs: $x_1, x_2, \dots, x_{10}$
  - Outputs: $y_1, y_2, \dots, y_{10}$
- Loss function: $\min_{b,w} \frac{1}{2} \sum_{i=1}^{10} (h(x_i) - y_i)^2)$
- Taking the derivative leads to the linear least square problem

$$[y_1\ y_2\ \dots\ y_{10}] = [w \quad b] \begin{bmatrix} x_1 & x_2 & \dots & x_{10} \\ 1 & 1 & \dots & 1 \end{bmatrix}$$

- Solution

$$[w \quad b] = [y_1\ y_2\ \dots\ y_{10}] / \begin{bmatrix} x_1 & x_2 & \dots & x_{10} \\ 1 & 1 & \dots & 1 \end{bmatrix}$$



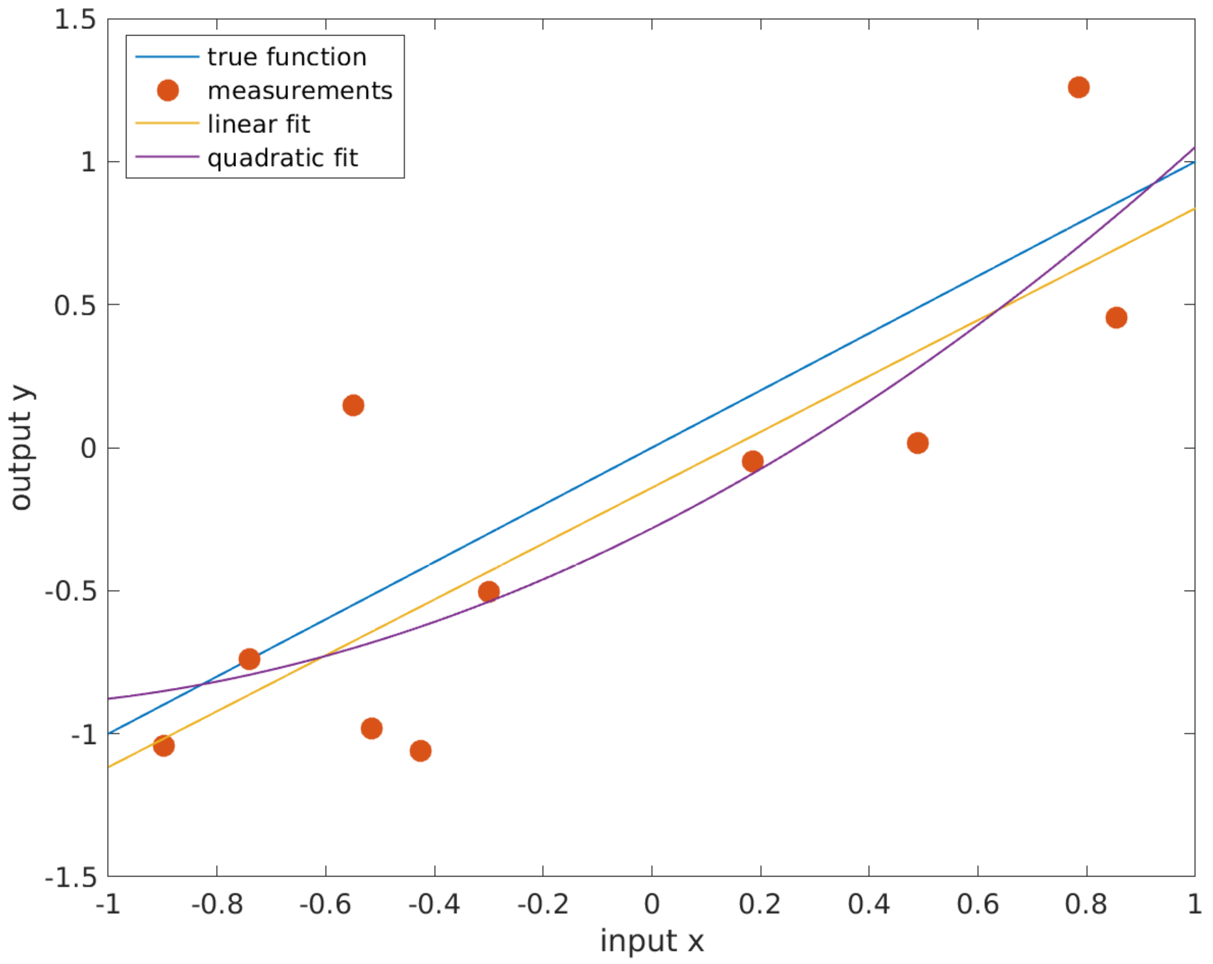
Example

Quadratic regression

- One dimensional function:
  $h(x) = w_1\, x^2 + w_2\, x + b\colon\ \mathbb{R} \rightarrow \mathbb{R}$
- 10 measurements available:
  - Inputs: $x_1\,, x_2, \ldots, x_{10}$
  - Outputs: $y_1\,, y_2, \ldots, y_{10}$
- Loss function: $\min_{b,w}\ \frac{1}{2}\sum_{i=1}^{10}(h(x_i) - y_i)^2)$
- Taking the derivative leads to the linear least square problem

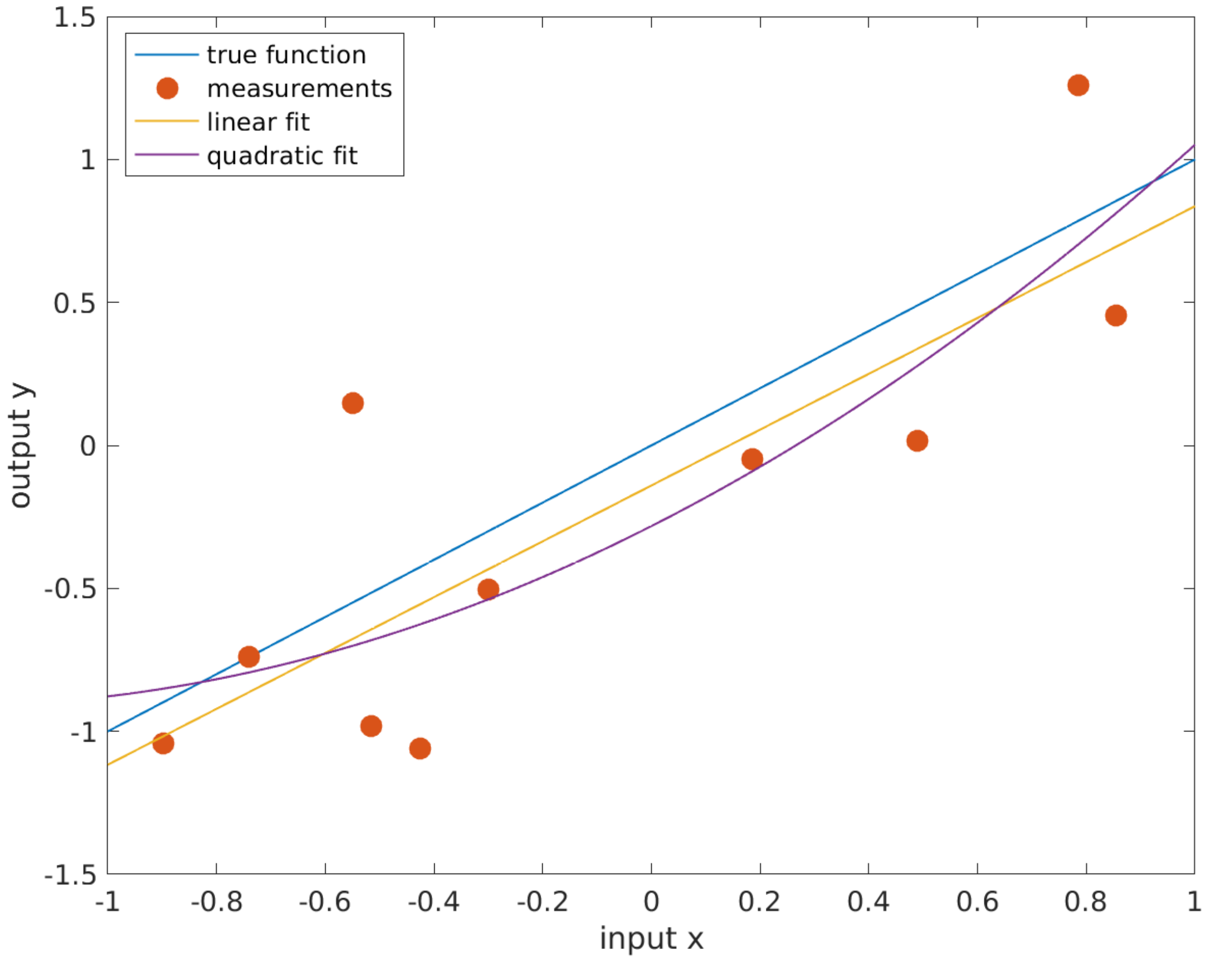
$$[y_1\, y_2 \ldots\, y_{10}\,] = \begin{bmatrix} w_1 & w_2 & b \end{bmatrix} \begin{bmatrix} x_1^2 & x_2^2 & \ldots & x_{10}^2 \\ x_1 & x_2 & \ldots & x_{10} \\ 1 & 1 & \ldots & 1 \end{bmatrix}$$

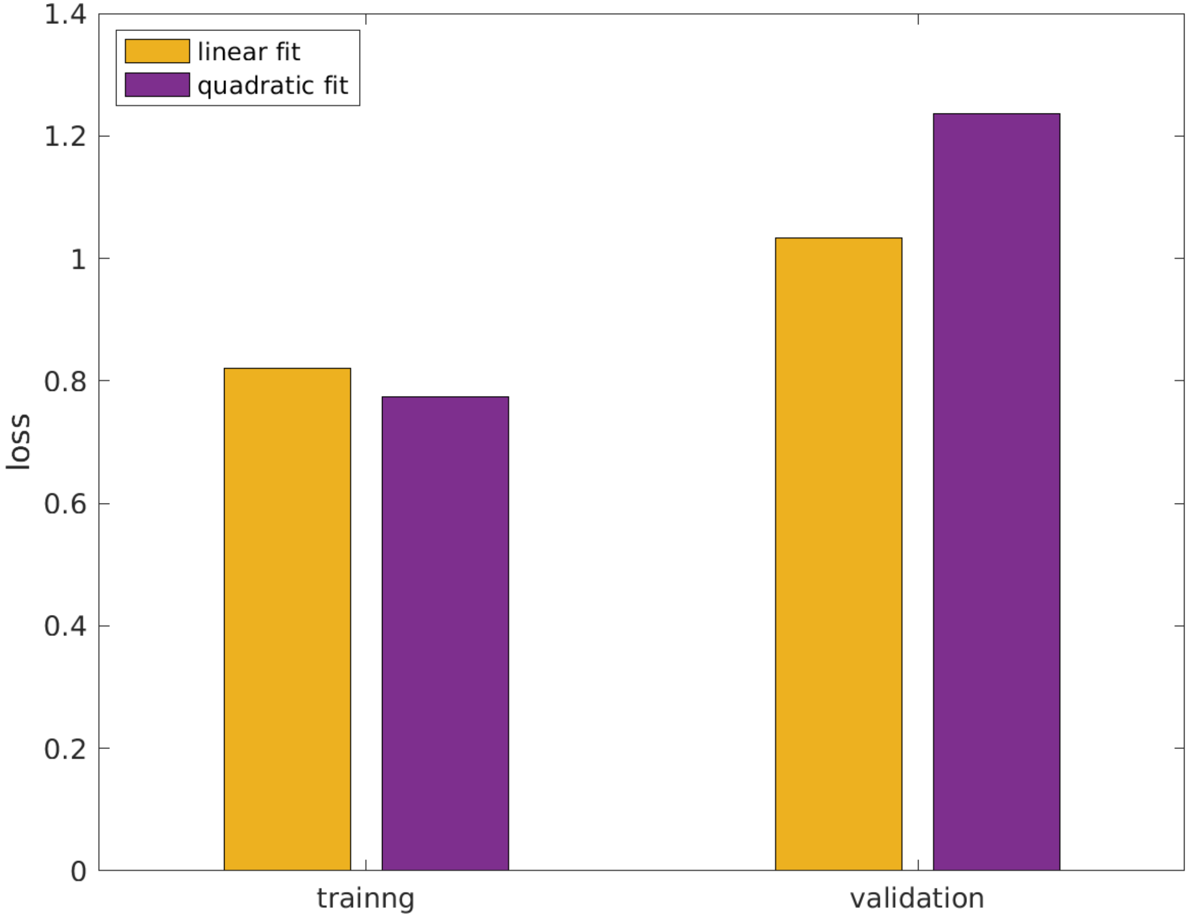
- Solution: $\begin{bmatrix} w_1 & w_2 & b \end{bmatrix} = [y_1\, y_2 \ldots\, y_{10}\,] / \begin{bmatrix} x_1^2 & x_2^2 & \ldots & x_{10}^2 \\ x_1 & x_2 & \ldots & x_{10} \\ 1 & 1 & \ldots & 1 \end{bmatrix}$



Example

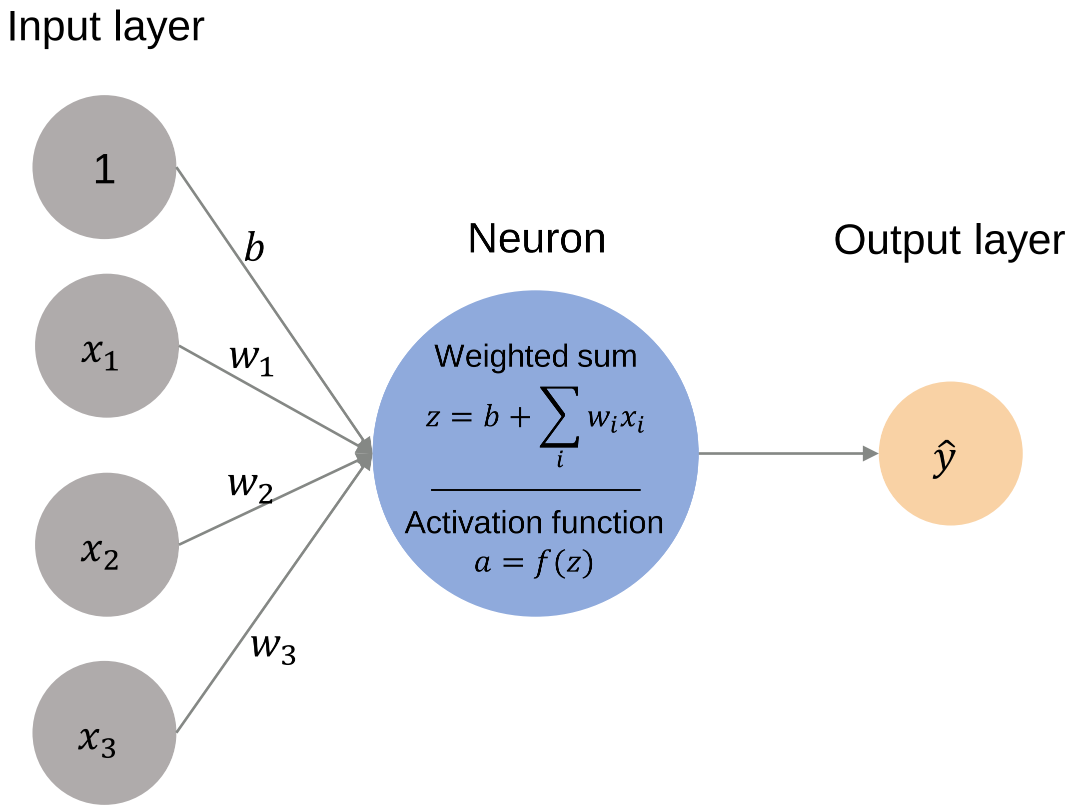
Comparison so far

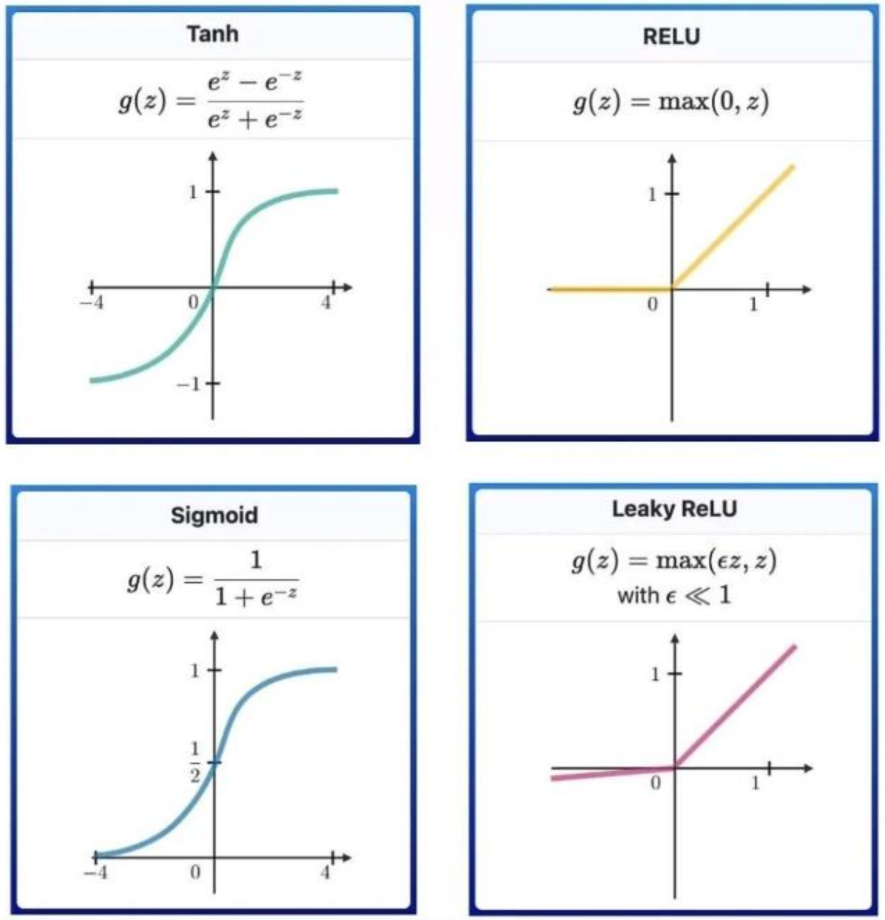




The Most Simple Neural networks

The Perceptron

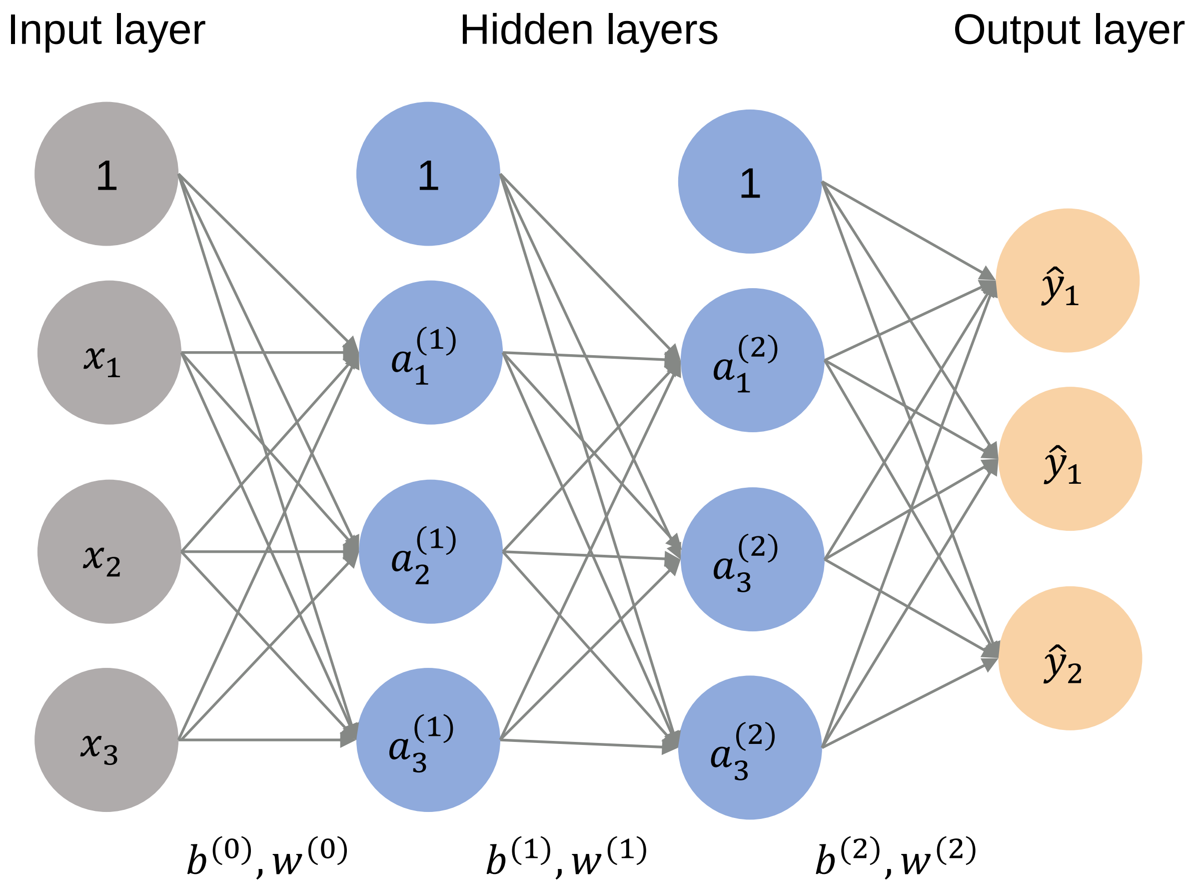




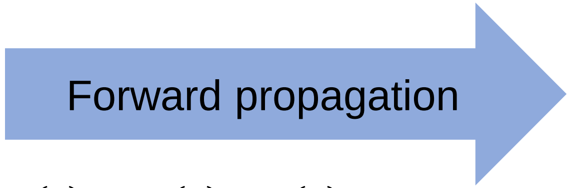
Most used activation functions in Neural Networks - AI ML - Artificial Intelligence and Machine Learning (webyes.com.br)

Neural network

Multi layer network



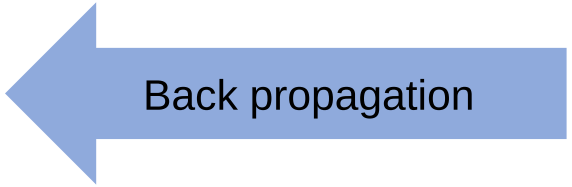
Example: $a_2^{(1)} = f_2^{(0)}\left(b_2^{(0)} + \sum_i w_{i,2}^{(0)} x_i\right)$
$a^{(1)} = f^{(0)}(W^{(0)}X)$



$$a^{(1)} = f^{(0)}(W^{(0)}X)$$

$$a^{(2)} = f^{(1)}(W^{(1)}X)$$

$$\hat{y} = f^{(2)}\left(W^{(2)}a^{(2)}\right)$$



Find the weights that minimize the loss

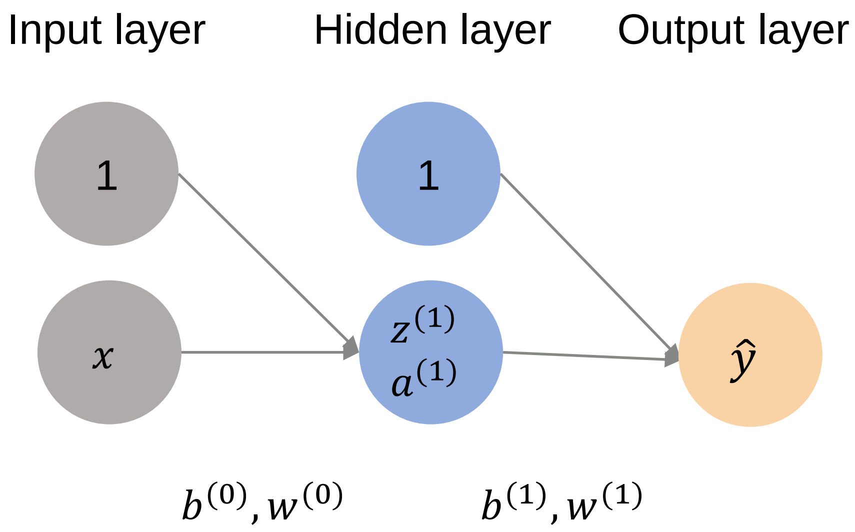
$$\min_W J(W)$$

With any gradient-based optimizers:

Calculate gradients: $\frac{\partial J(W)}{\partial W^{(2)}}, \frac{\partial J(W)}{\partial W^{(1)}} = \frac{\partial J(W)}{\partial W^{(2)}} \frac{\partial a^{(2)}}{\partial W^{(1)}}, \ldots$

Example continued

Neural network



Activation function: $f(z) = \sigma(z) = \frac{1}{1+e^{-z}}$

$$\frac{\partial f}{\partial z} = \sigma(z)(1 - \sigma(z))$$

Loss function: $J = \frac{1}{2}\sum_{i=1}^{10}(\hat{y}_i - y_i)^2)$

Forward propagation

$$z^{(1)} = b^{(0)} + w^{(0)} x$$

$$a^{(1)} = \frac{1}{1+e^{-z^{(1)}}}$$

$$\hat{y} = b^{(1)} + w^{(1)}\, a^{(1)}$$

Back propagation

Calculate gradients: $\frac{\partial J}{\partial \hat{y}} = \hat{y}_i - y_i$

$$\frac{\partial J}{\partial w^{(1)}} = \frac{\partial J}{\partial \hat{y}} \frac{\partial \hat{y}}{\partial w^{(1)}} = (\hat{y}_i - y_i)'\, a^{(1)}$$

$$\frac{\partial J}{\partial b^{(1)}} = \frac{\partial J}{\partial \hat{y}} \frac{\partial \hat{y}}{\partial b^{(1)}} = (\hat{y}_i - y_i)'\, 1$$

$$\frac{\partial J}{\partial w^{(0)}} = \frac{\partial J}{\partial \hat{y}} \frac{\partial \hat{y}}{\partial a^{(1)}} \frac{\partial a^{(1)}}{\partial z^{(1)}} \frac{\partial z^{(1)}}{\partial w^{(0)}} = (\hat{y}_i - y_i)'\, w^{(1)} \frac{\partial f}{\partial z^{(1)}} x$$

$$\frac{\partial J}{\partial b^{(0)}} = \frac{\partial J}{\partial \hat{y}} \frac{\partial \hat{y}}{\partial a^{(1)}} \frac{\partial a^{(1)}}{\partial z^{(1)}} \frac{\partial z^{(1)}}{\partial b^{(0)}} = (\hat{y}_i - y_i)'\, w^{(1)} \frac{\partial f}{\partial z^{(1)}} 1$$

Update

$$w^{(1)} \leftarrow w^{(1)} - \eta \frac{\partial J}{\partial w^{(1)}}$$

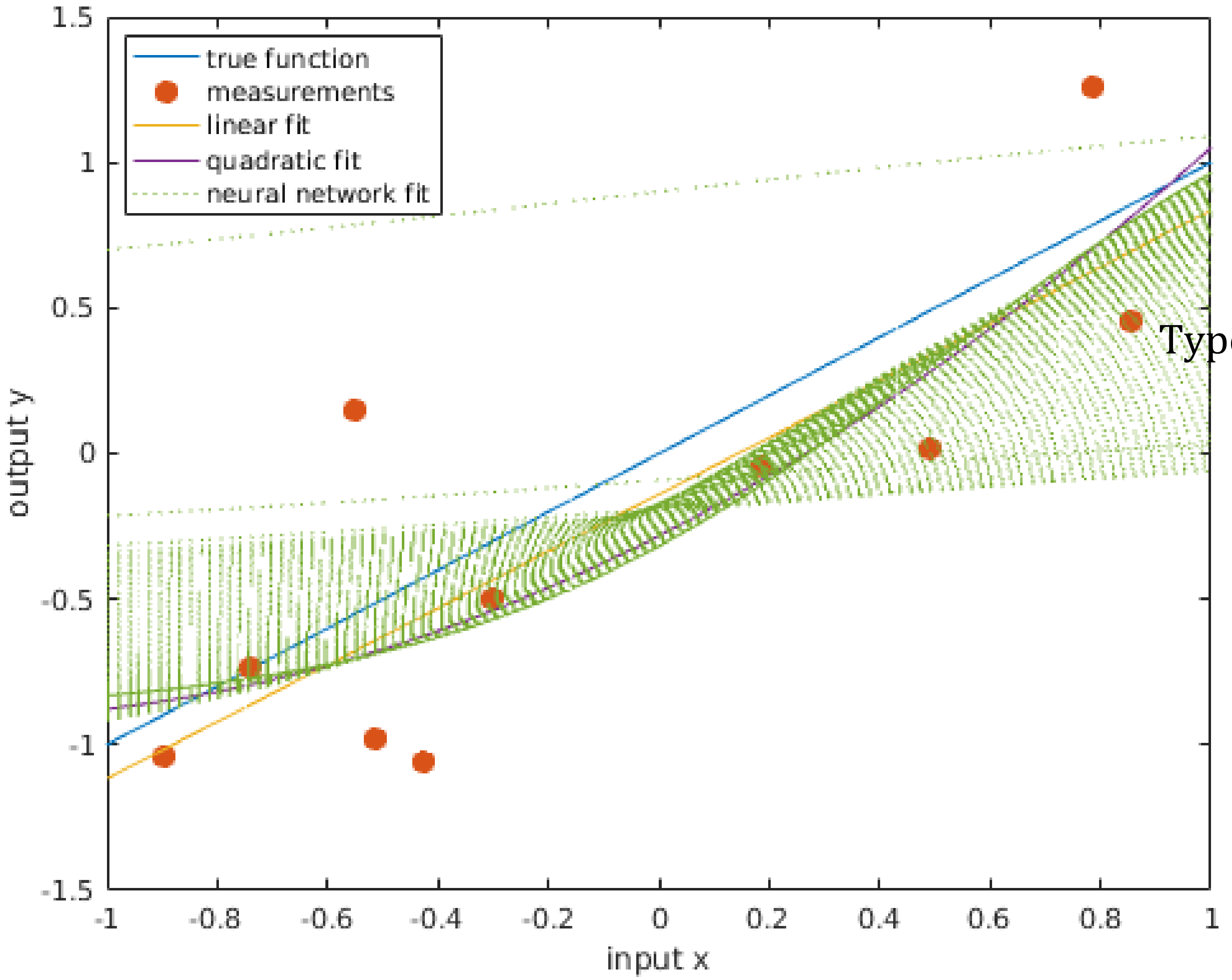
$$b^{(1)} \leftarrow b^{(1)} - \eta \frac{\partial J}{\partial b^{(1)}}$$

$$w^{(0)} \leftarrow w^{(0)} - \eta \frac{\partial J}{\partial w^{(0)}}$$

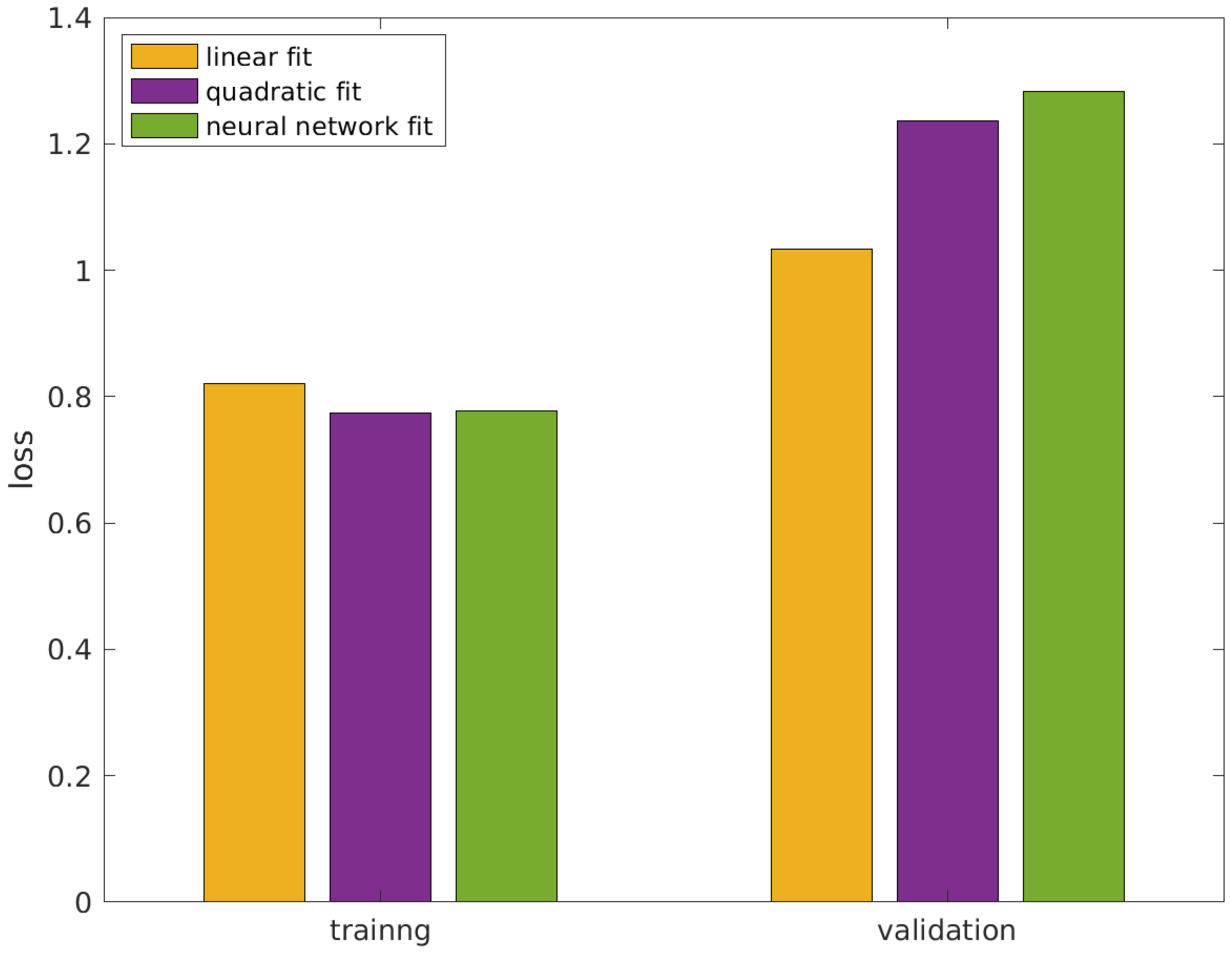
$$b^{(0)} \leftarrow b^{(0)} - \eta \frac{\partial J}{\partial b^{(0)}}$$

Example

Neural network



Type



Techniques for handling underfitting and overfitting in Machine Learning | by Manpreet Singh Minhas | Towards Data Science

Underfitting and Overfitting

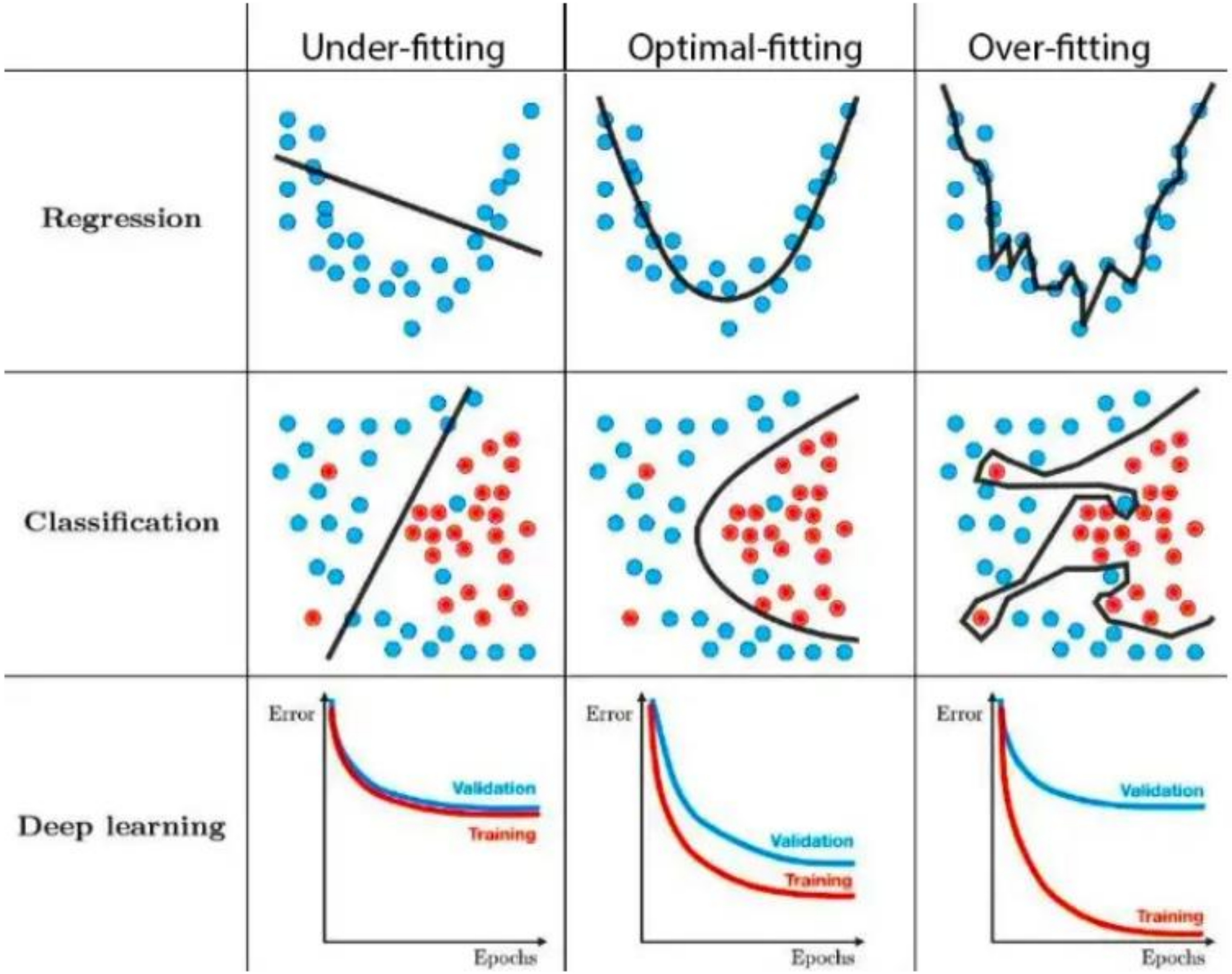
How to deal with it

Underfitting

- Increase model complexity
- Train for more epochs

Overfitting

- Get more data
- Data augmentation
- Early stopping
- Regularization (L1, L2)
- Dropout
- DropConnect



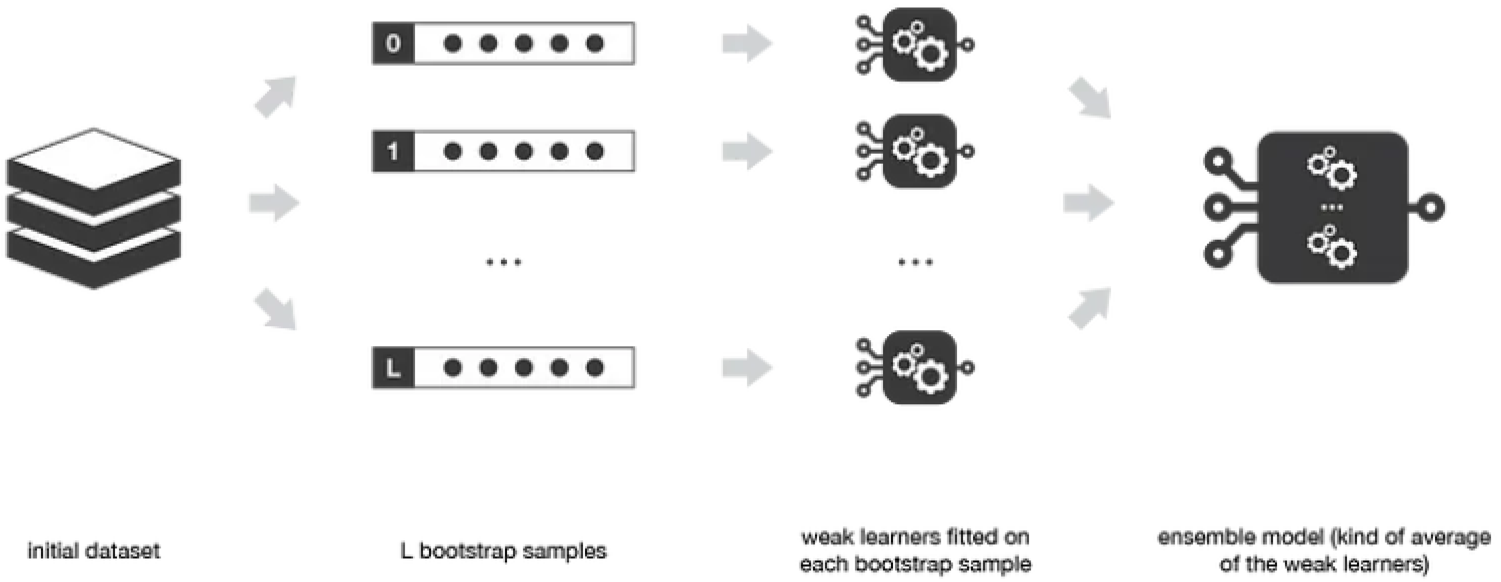
Neural network

Uncertainty quantification methods

Getting uncertainty to a neural network prediction → Robust prediction

Ensemble methods

- Random parameters initialization:
  - multiple networks are trained with different initial conditions
  - Mean and variance are the mean and variance of the predictions
- Bagging
  - Bootstrapping
  - Neural network for each bootstrap samples
- …



Quantile regression

- Predict multiple quantiles with tilted loss function
- A separate network is trained for each quantile

Ensemble methods: bagging, boosting and stacking | by Joseph Rocca | Towards Data Science

Autoencoder

Neural networks continued

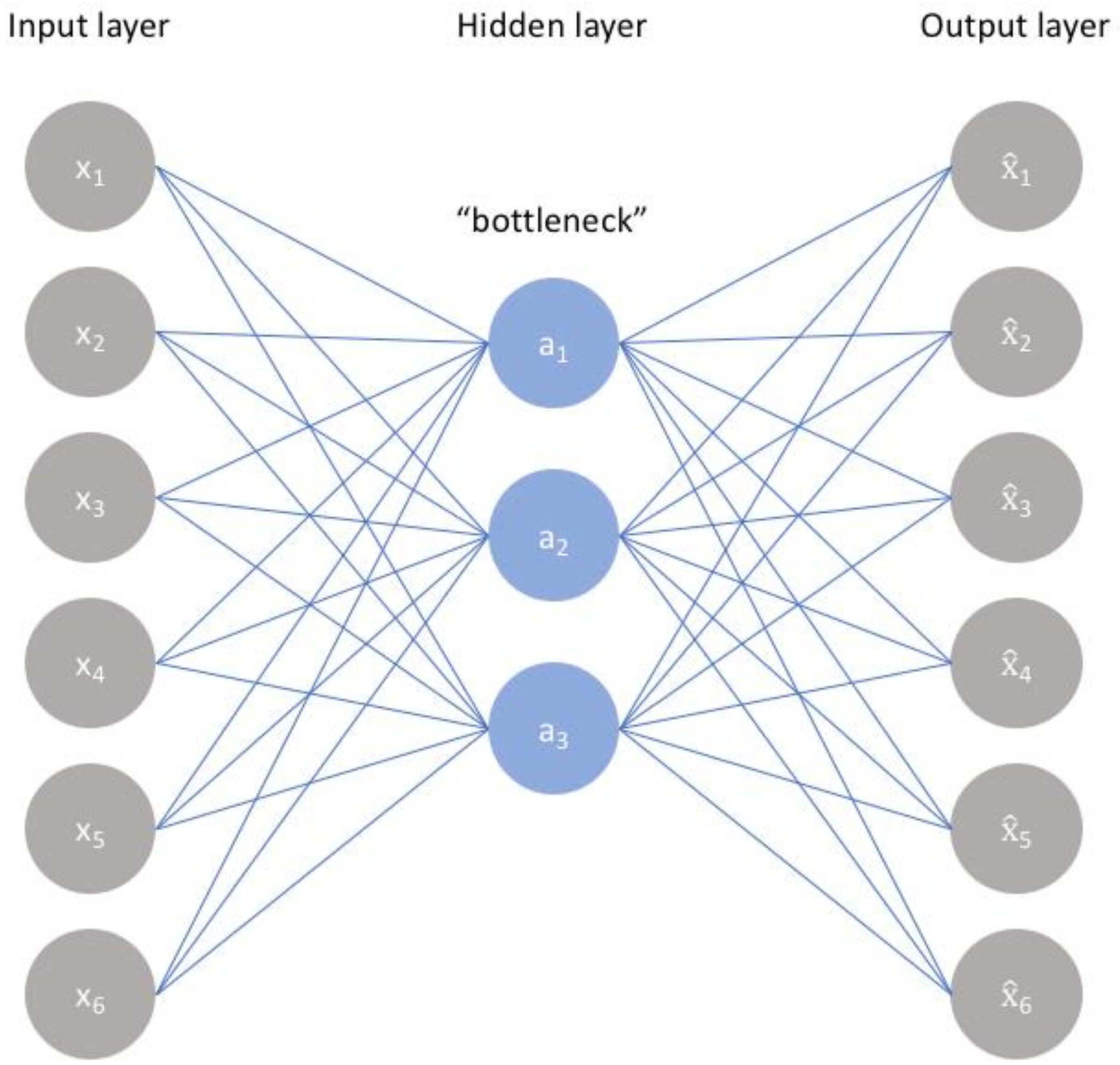
Unsupervised learning

→ Network tries to learn the original input

→ Loss function: $L(x, \hat{x})$

Distance between estimates and input

→ Bottleneck constraints the amount of information that can go through



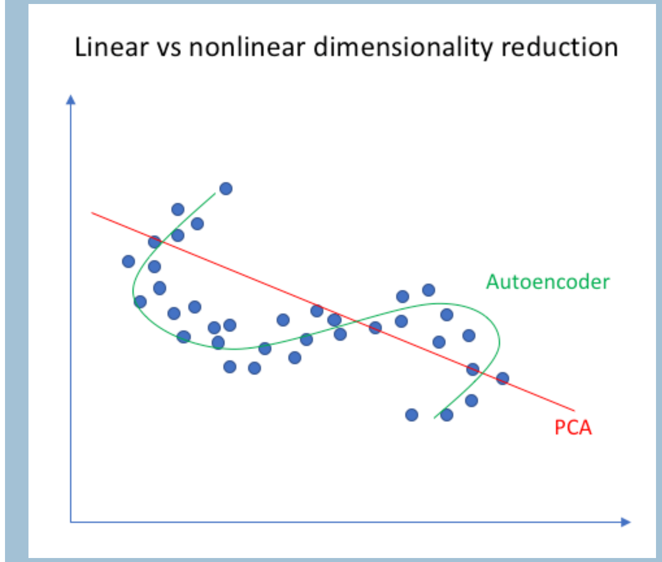
Introduction to autoencoders. (jeremyjordan.me)

Applications of Autoencoder

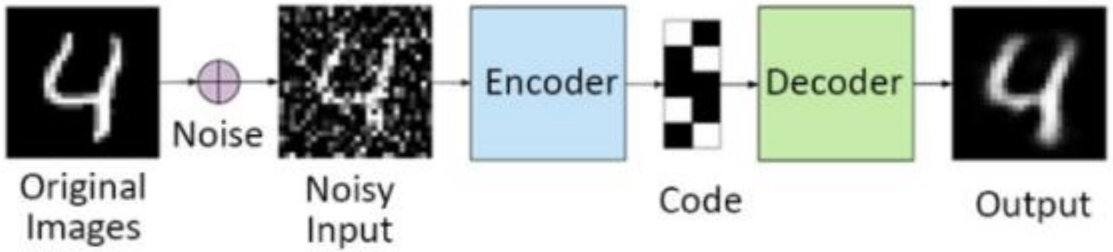
What they are used for

- Dimensionality reduction
- Denoising
  - Add random noise to the input data
  - Train the autoencoder to recover the original, nonperturbed signal
- Fault detection
- Data analysis
  - (beta-)variational Autoencoder

(5) (PDF) Fully Convolutional Variational Autoencoder For Feature Extraction Of Fire Detection System (researchgate.net)



If linear network, same results as PCA



Autoencoders | Main Components and Architecture of Autoencoder (educba.com)

(Beta)-Variational Autoencoder

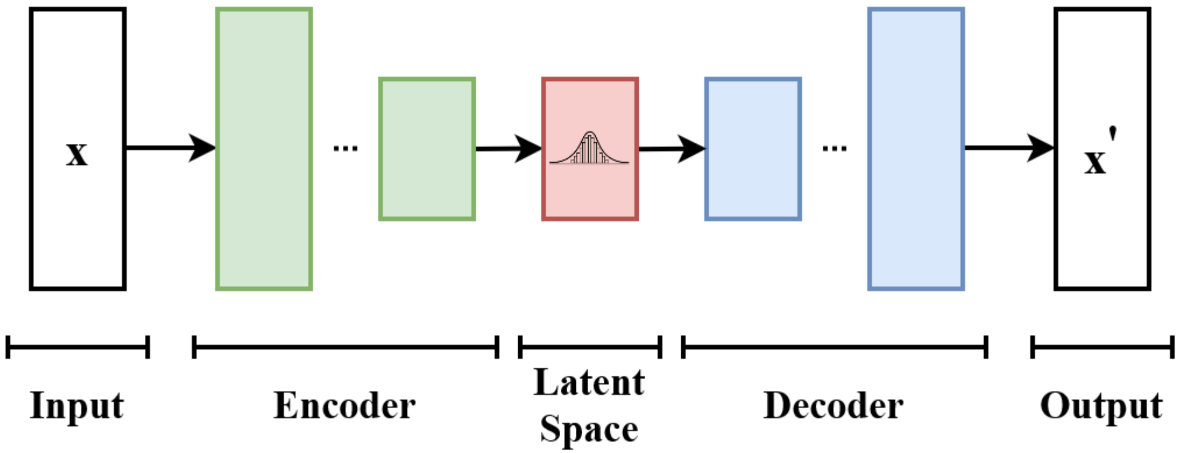
What does it do?

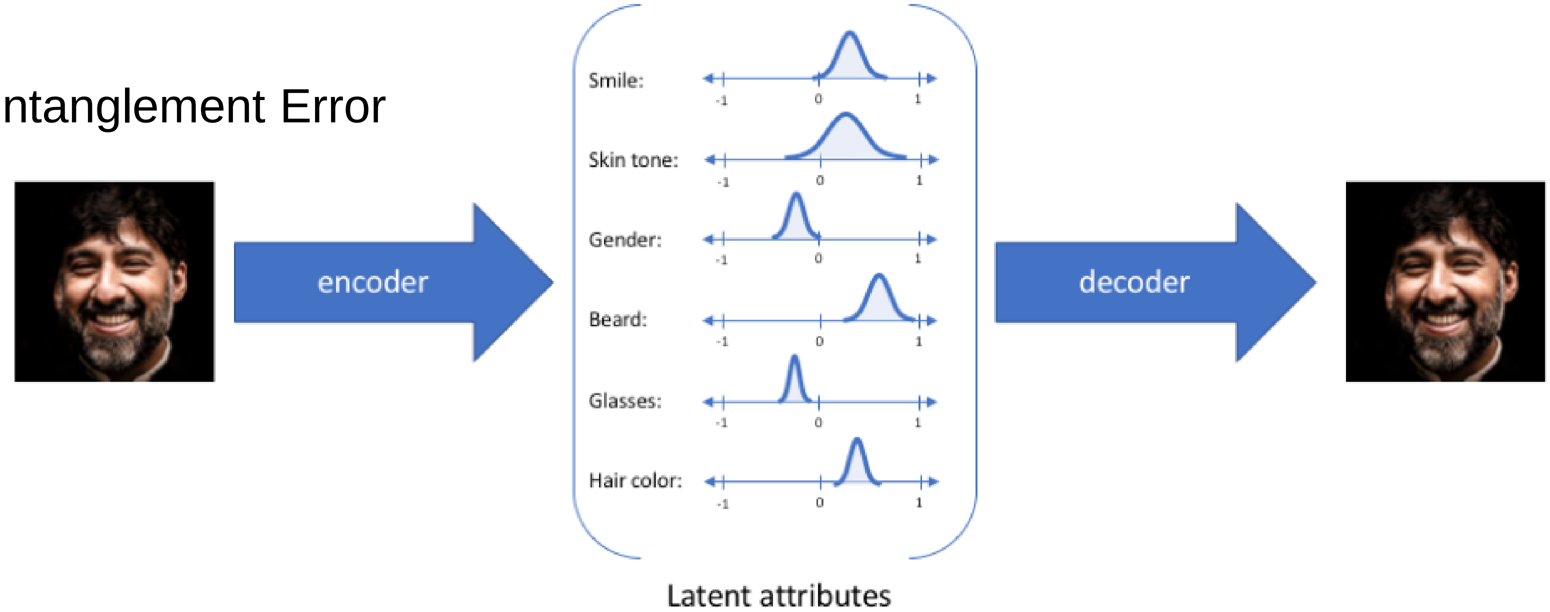
Variational autoencoder (VAE)

- Instead of mapping the input into a fixed vector, we want to map it into a **distribution** in the latent space

Beta-variational autoencoder

- Get **disentangled** latent space variables
- Include deviation from a Gaussian normal distribution in loss
- Loss = Reconstruction Error + β Disentanglement Error





Courtesy: Arne Grünhagen

Fault Diagnosis

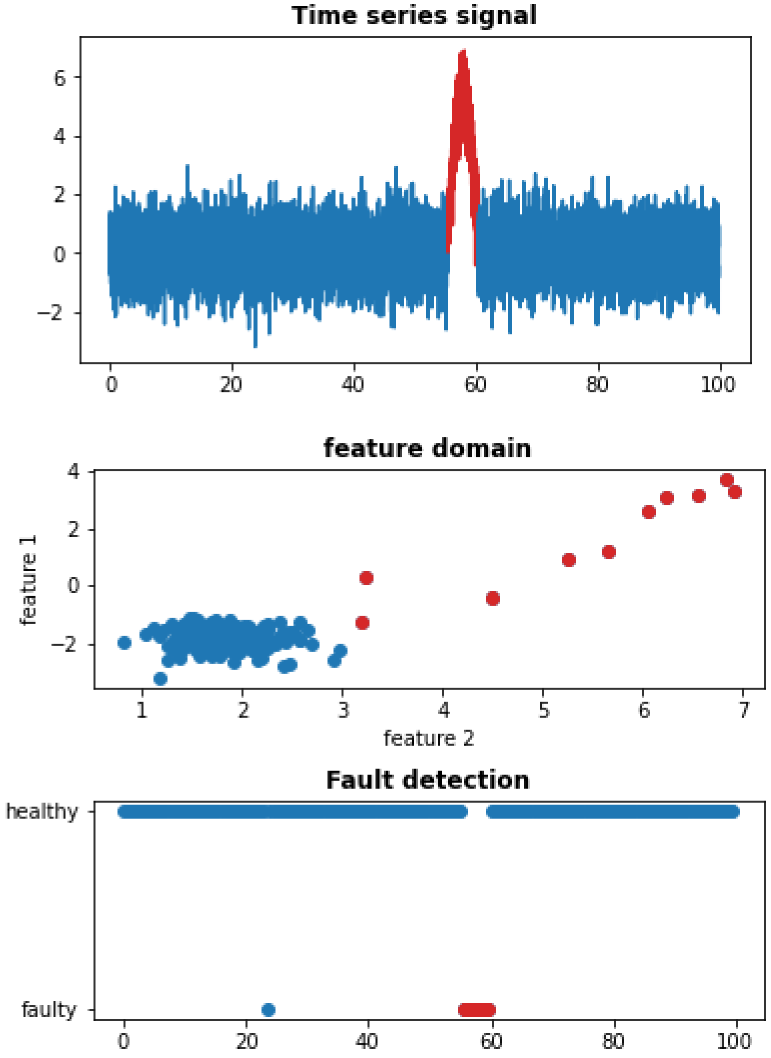
Using Machine Learning

Feature Extraction

- Dimensionality reduction of several signals
- Required for time-series data
- Automated time series feature extraction using the python package *tsfresh* [1]
- Feature extraction based on Neural network autoencoders [2]

Clustering and Outlier Detection

- Clustering algorithms aim to group data samples into classes with similar elements (faulty and healthy)
- Outlier detection algorithms aim to identify rare items or events that differ significantly from the rest of the dataset



[1] Christ M, Braun N, Neuffer J, Kempa-Liehr AW. Time Series FeatuRe Extraction on basis of Scalable Hypothesis tests (tsfresh – A Python packagAe). Neurocomputing. 2018;307:72-7

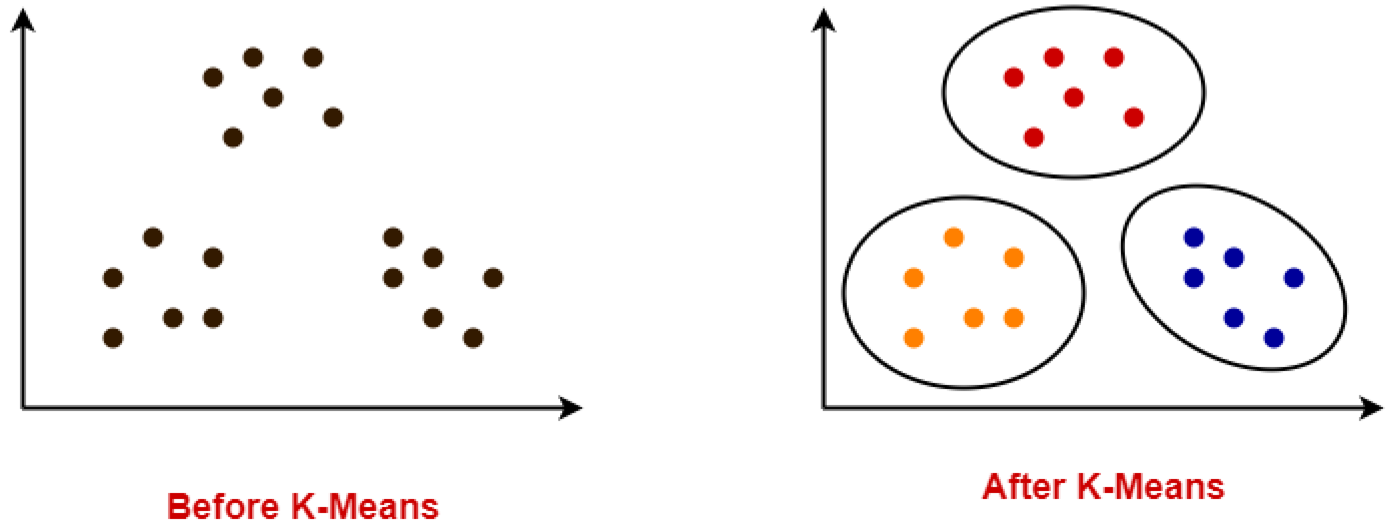
[2] Meng, Q., Catchpoole, D., Skillicom, D., & Kennedy, P. J. (2017, May). Relational autoencoder for feature extraction. In *2017 International joint conference on neural networks (IJCNN)* (pp. 364-371). IEEE.

K-means Clustering

Dividing the entire data into clusters based on patterns

Goal:

partition the observations into k sets such that the distance between cluster centroids points within a cluster is minimized (within-cluster sum of squares)



- Unsupervised iterative clustering technique
- Partitions the given data set into k distinct clusters
- Each point belongs to the cluster with nearest mean

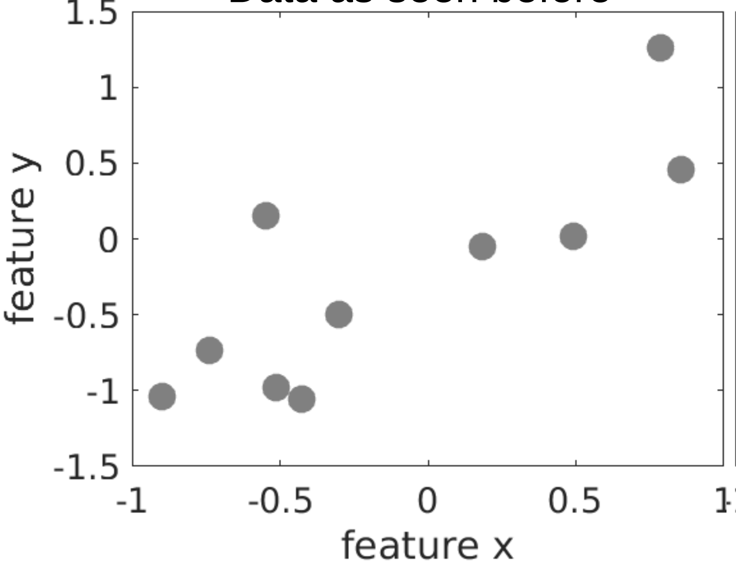
Disadvantages

- k needs to be predefined in advance (alternative: DBSCAN, hierarchical clustering)
- No handling of noise, outliers
- Only convex clusters

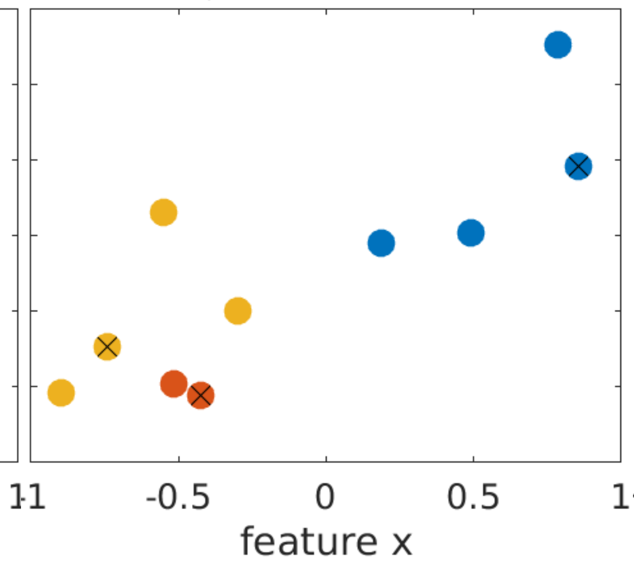
K Means Clustering | Gate Vidyalay

Example

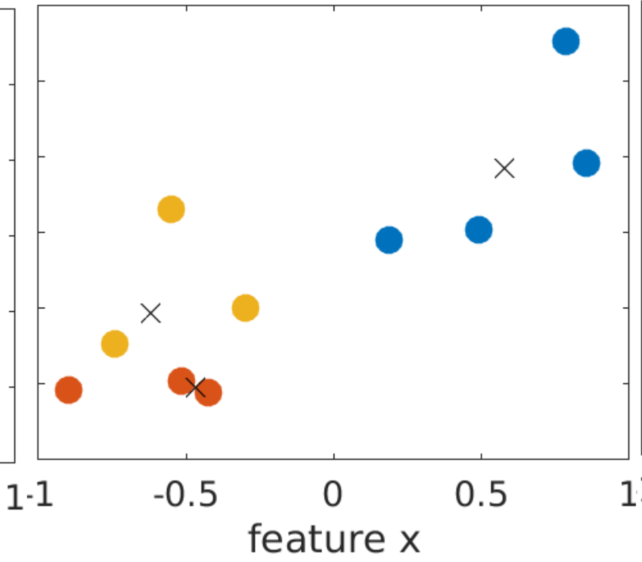
Data as seen before



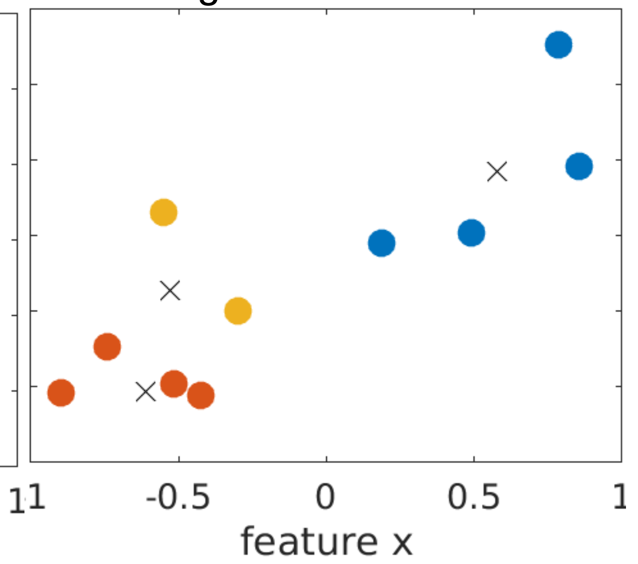
1. Assigning to clusters: according to smallest distance



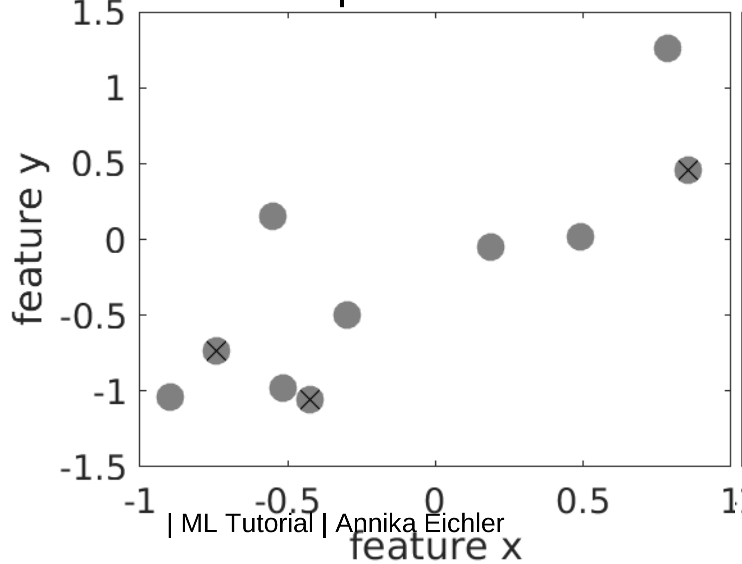
2. Assigning to clusters: according to smallest distance



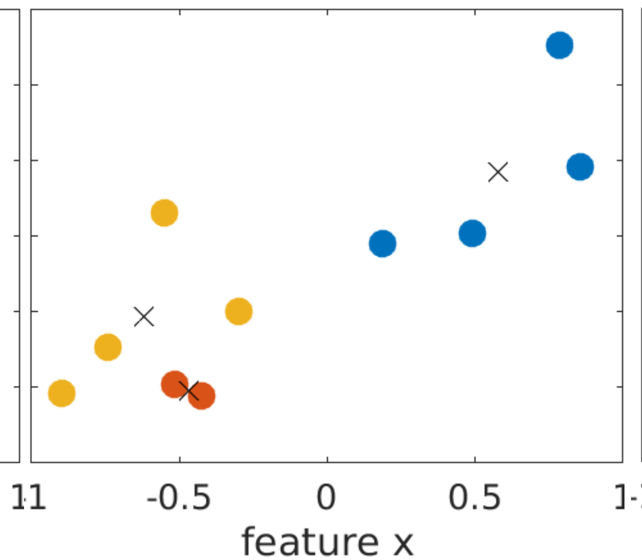
3. Assigning to clusters: according to smallest distance



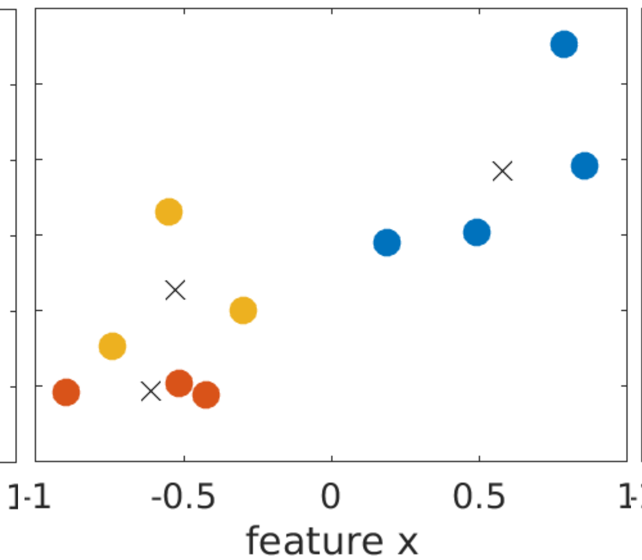
Initialization: choose randomly k data samples as centers



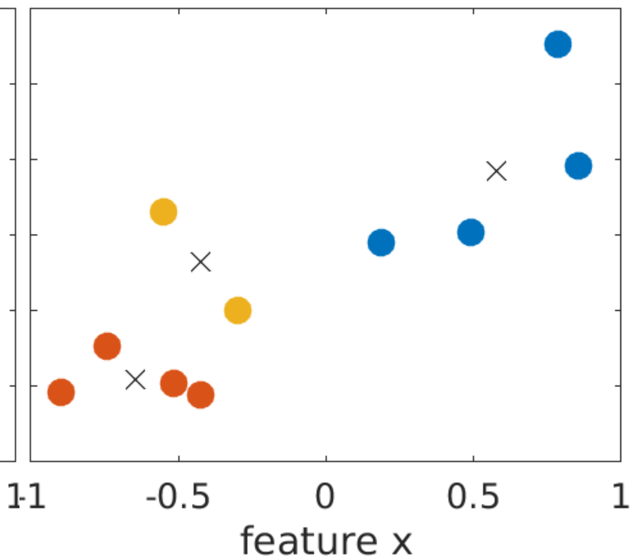
1. Updating centers: mean value of data in cluster



2. Updating centers: mean value of data in cluster



3. Updating centers: mean value of data in cluster



Isolation Forests

Outlier detection

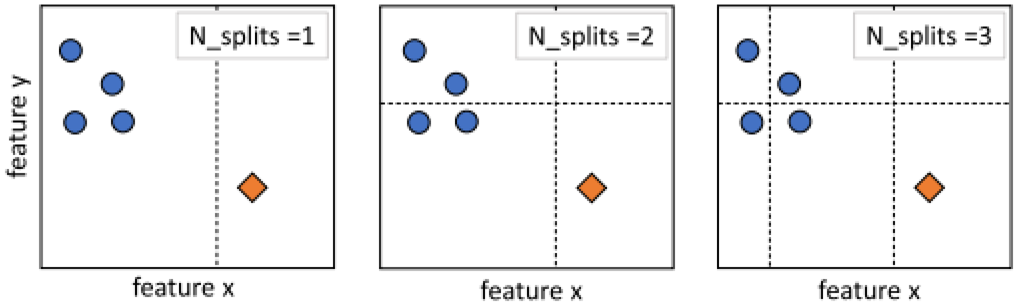
Idea: anomalies are the data points that are "few and different"

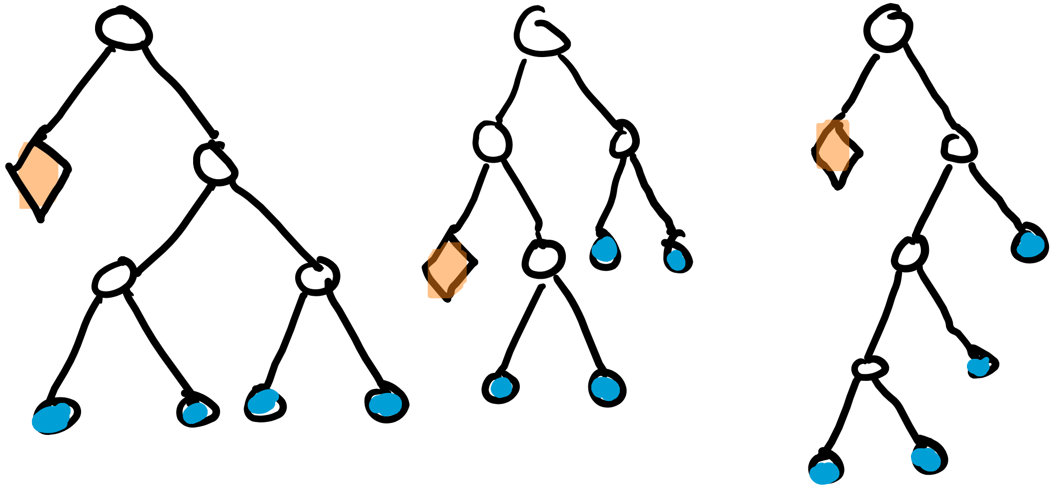
Isolation tree:

- splits the data space using lines that are orthogonal to the origin
- Counts the number of splits needed for isolation, represents the path length from root to leaf in decision tree → fewer splits → higher anomaly score

Isolation forest:

- Ensemble method using multiple isolation trees
- Averaging the path length over the multiple trees
- Requires the expected proportion of outliers (contamination factor)





(2) Point-Denoise: Unsupervised outlier detection for 3D point clouds enhancement | Request PDF (researchgate.net)

Isolation Forests

Outlier detection

Pathlength: $h(x)$

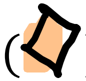
Average path length: $E(h(x))$

Normalization factor: $c(n) = 2H(n-1) - 2\frac{n-1}{n}$

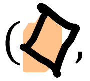
Harmonic number

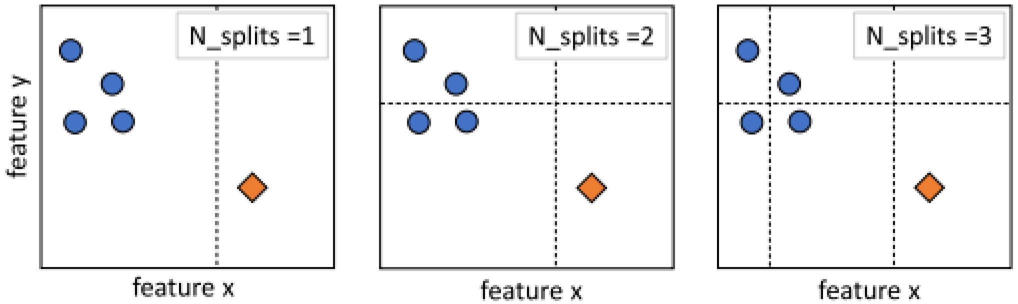
Anomaly score: $s(x,n) = 2^{-\frac{E(h(x))}{c(n)}}$

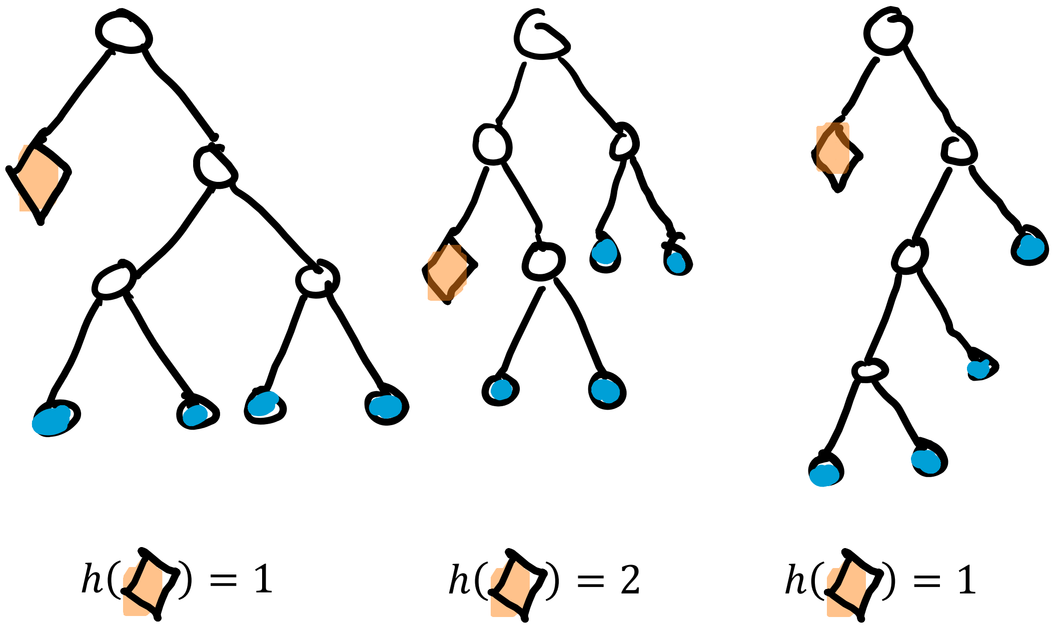
Example:

Average path length: $E\big(h($$)\big) = \frac{4}{3}$

Normalization factor: $c(5) = 2.97$

Anomaly score: $s($$, 5) = 2^{-\frac{E(h(\cdot))}{c(5)}} = 0.73$ 





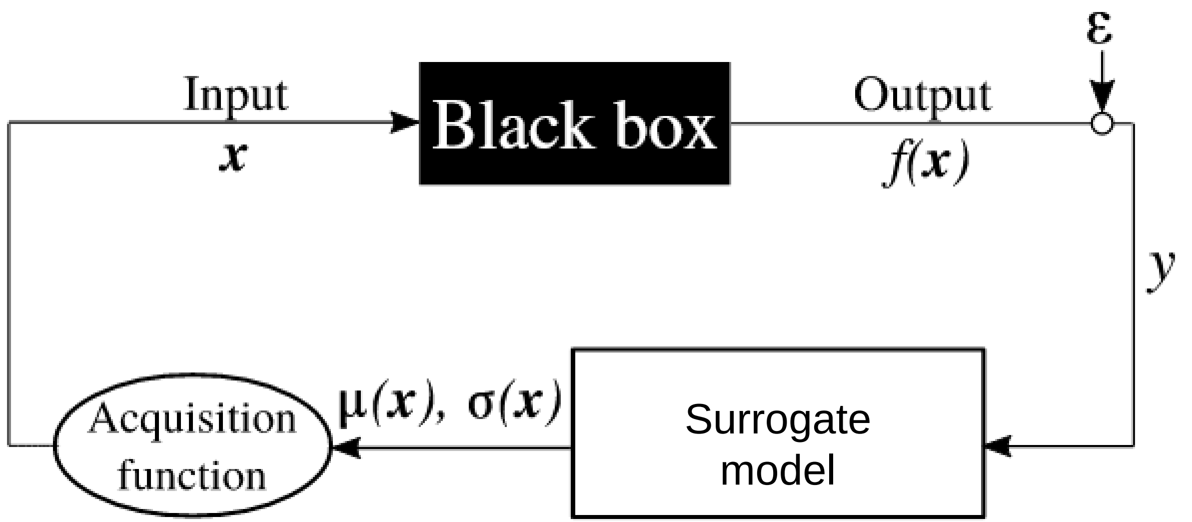
Bayesian Optimization

What is Bayesian optimization

Sequential global optimization algorithms to solve

$$\max_x f(x)$$

by building a surrogate model.



Black box function $f(x)$

- Unknown function, no derivative information
- Noisy samples can be drawn
- Samples are expansive

Surrogate model

- Probabilistic model: Gaussian processes
- $f(x) \sim GP(\mu(x), k(x, x'))$ → Updated via Bayes Theorem
  mean and covariance

Acquisition function $\alpha(x)$

- Determine the next query point as $\underset{x}{\operatorname{argmax}}\, \alpha(x)$
- Exploration vs. exploitation
- Often used
  - Probability of improvement (PI)
  - Expected improvement (EI)
  - Upper confidence bound (UCB)

Bayesian Optimization

Covariance function (kernel)

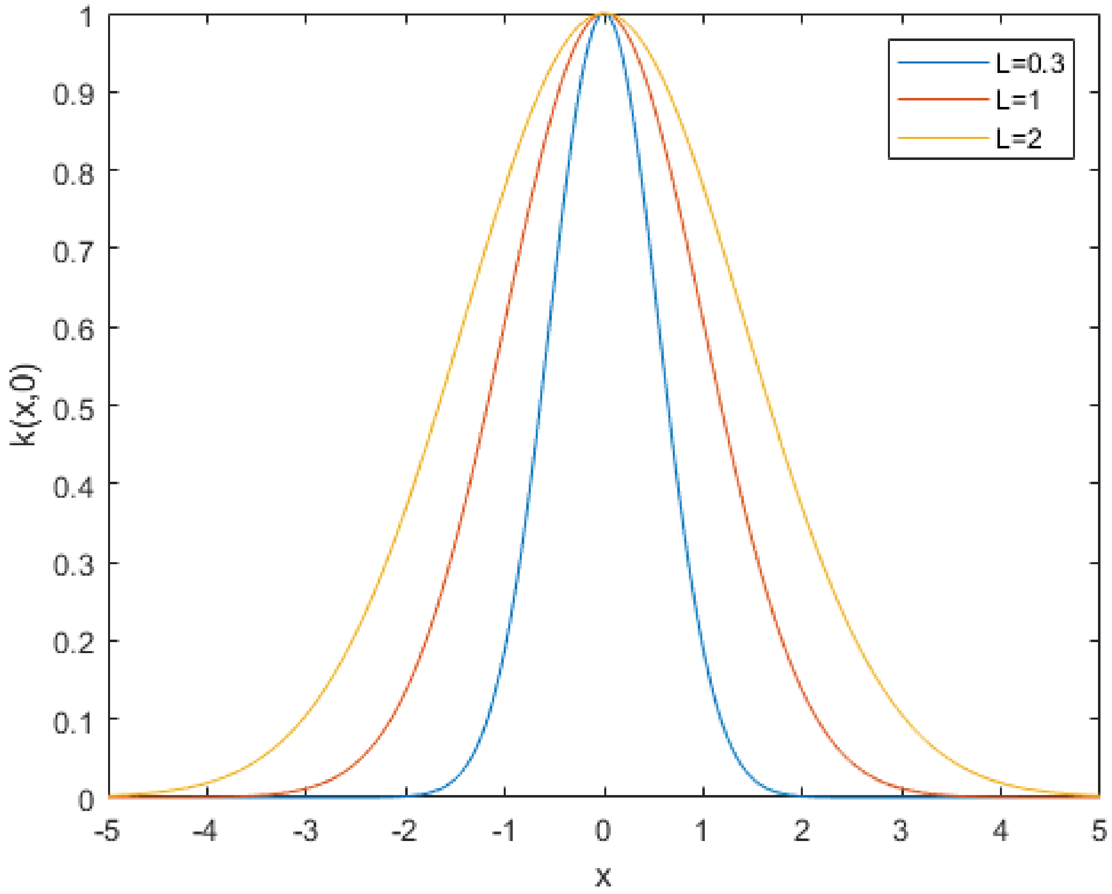
How similar are two data points (x, x')?

Covariance function

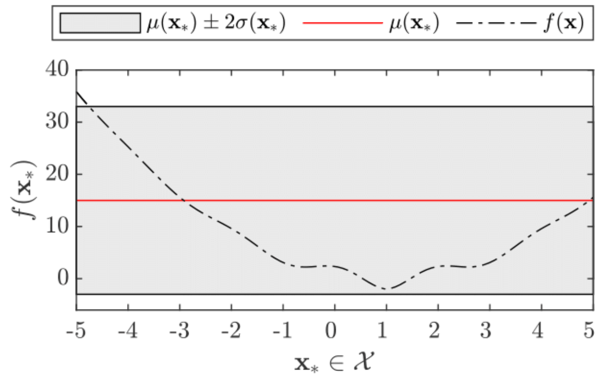
$$k(x, x') = \sigma k_{RBF}(x, x') + \sigma_n$$

Radial basis function
(squared exponential kernel)

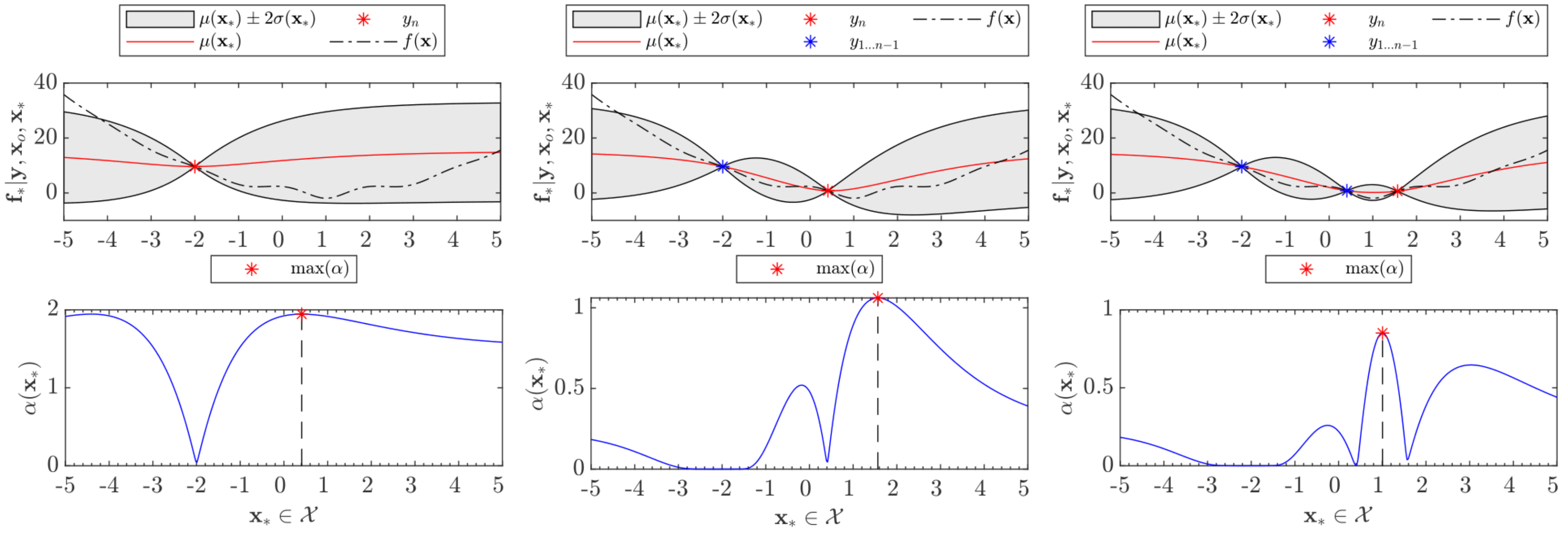
$$k_{RBF}(x, x') = e^{-\frac{(x-x')^2}{2L}}$$

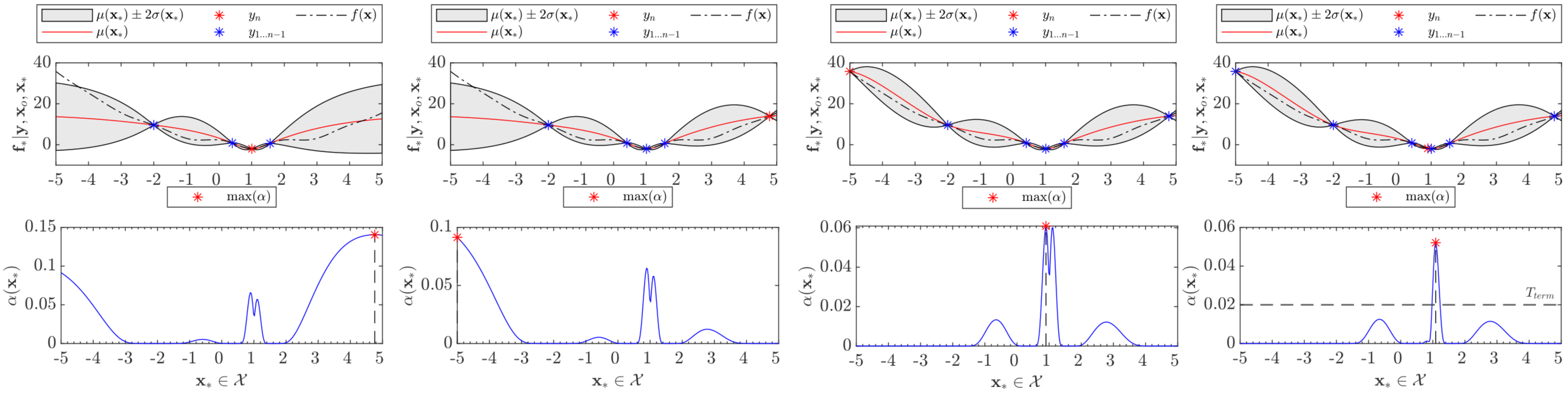


Bayesian Optimization – an illustrative Example



- Expected improvement
- RBF kernel





Reinforcement Learning

Training a control agent by trial and error

Reinforcement Learning (RL)

A machine learning approach where a software *agent* learns iteratively a *policy* to act on an *environment* based on observations in order to solve a given task by maximizing a cumulative *reward*.

Notation

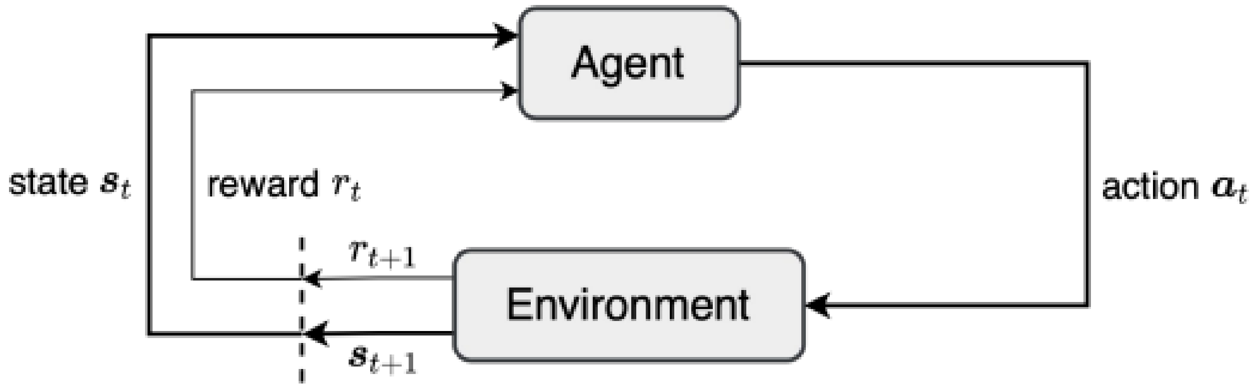
State s_t (observation o_t)

Based on the observation of state s_t the agent following the policy π chooses an action a_t

$\pi(s_t) = a_t$

The action is applied the environment: Transitions from state $s_t \rightarrow s_{t+1}$

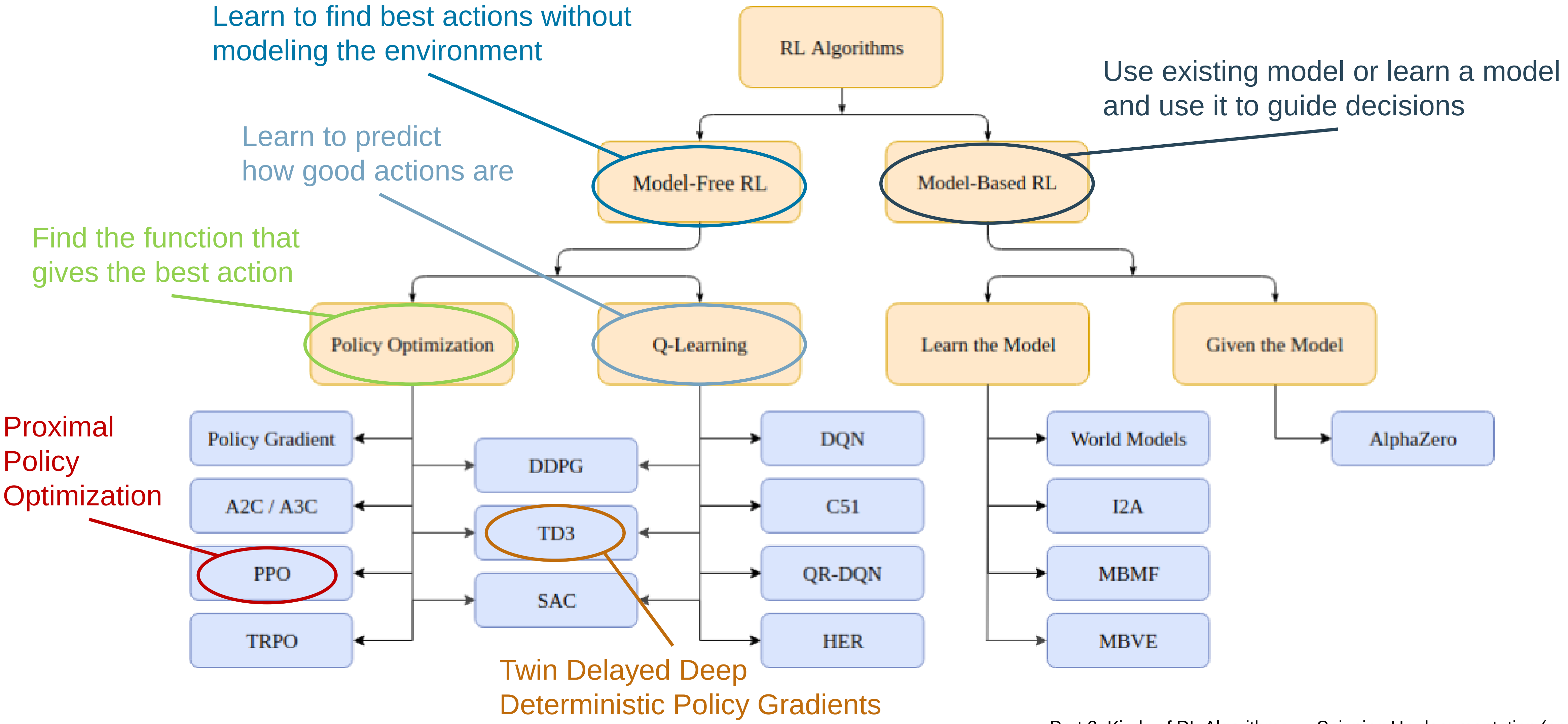
Reward $r_t = r(s_t, a_t)$



Learning: The policy is updated in order to maximize the cumulative reward over all successive steps (episode).

Reinforcement Learning Algorithms

A Taxonomy



Part 2: Kinds of RL Algorithms — Spinning Up documentation (openai.com)

TD3: Twin Delayed Deep Deterministic Policy Gradients

How does it work?

- TD3 is an **off-policy algorithm** (each update can use data collected at any point during training, regardless of how the agent was choosing to explore the environment when the data was obtained).
- TD3 can only be used for environments with **continuous action spaces**.

Key features compared to predecessor (DDPG)

1. Using a pair of critic networks

2. Delayed updates of the actor

3. Action noise regularization

TD3: Learning To Run With AI. Learn to build one of the most powerful… | by Donal Byrne | Towards Data Science

… and applications

ML for Accelerators

What are the most important fields?

Data analysis

- Understanding physics
- Find new correlations of parameters
- Identify relevant data channels

→ New physical insight

- Unsupervised Learning

Estimating and predicting

Surrogate models

→ Models for online control and optimization, and for accelerator design

Virtual diagnostics

→ Additional, nondestructive, (online) information

- Supervised Learning

Fault diagnosis

- Predict & prevent failures
- Protect the system
- Identify poor conditions
- Find the root cause of errors encountered

→ Improve the availability/ reliability of machine operation

- Supervised Learning
- Unsupervised Learning
- (Statistics/Control)

Tuning and control

- Exploit data to retrieve desired machine settings
- Push the way of operation
- Optimize performance

→ Better performance for users

- Reinforcement Learning
- Optimization
- (Control)

β-VAEs for Online X-ray Pulse Profile Reconstruction

Without having ever seen the X-ray profiles

Data

Longitudinal phase space images with different amount of lasing

Train a β-VAE

Get lasing as disentangled tuning knob in the latent space

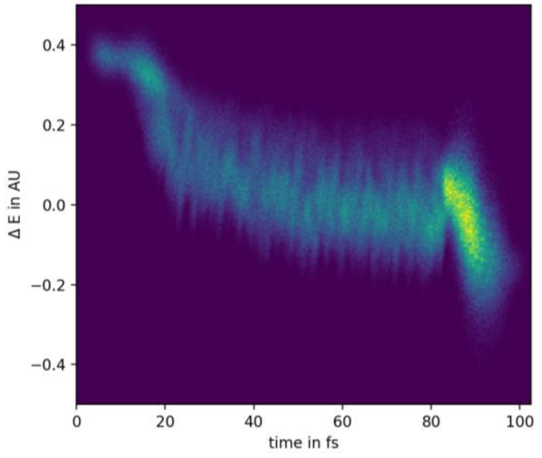
X-ray Pulse Profile Reconstruction

- Given a longitudinal phase space measurement with lasing
- Reconstruct the corresponding phase space without lasing
- Take the difference and reconstruct the X-ray power profile

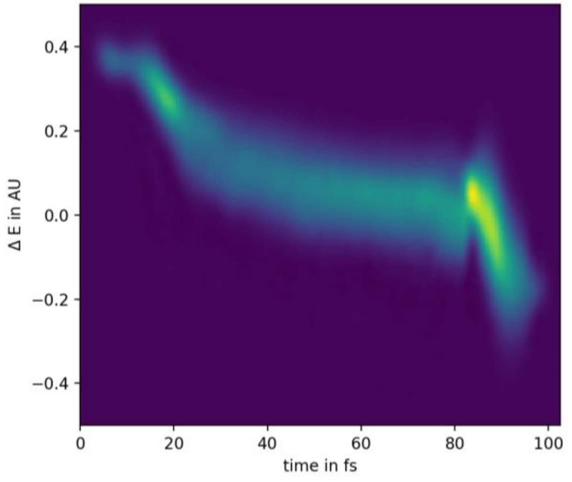
Advantage

No X-ray power profiles are needed for training

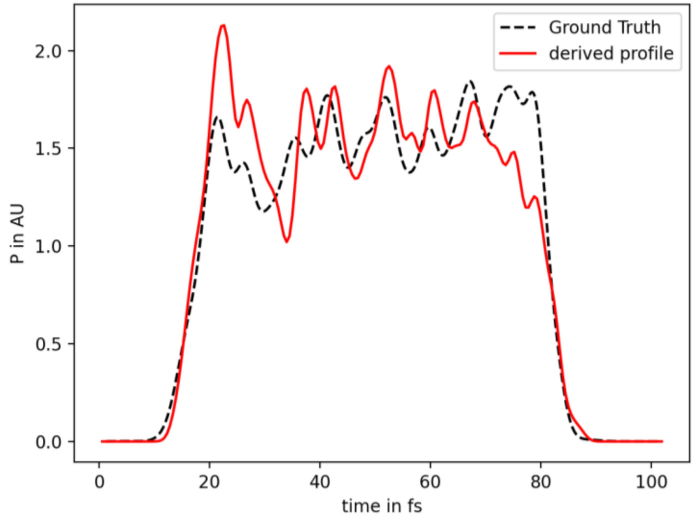
Simulated longitudinal phase space lasing on



Predicted longitudinal phase space lasing off



Reconstructed X-ray power profile



G. Goetzke, et. al., “**AI Methods for an improved evaluation of FEL diagnosic data**”, Science@FEL

ML for Accelerators

What are the most important fields?

Data analysis

- Understanding physics
- Find new correlations of parameters
- Identify relevant data channels

→ New physical insight

- Unsupervised Learning

Estimating and predicting

Surrogate models
→ Models for online control and optimization, and for accelerator design

Virtual diagnostics
→ Additional, nondestructive, (online) information

- Supervised Learning

Fault diagnosis

- Predict & prevent failures
- Protect the system
- Identify poor conditions
- Find the root cause of errors encountered

→ Improve the availability/ reliability of machine operation

- Supervised Learning
- Unsupervised Learning
- (Statistics/Control)

Tuning and control

- Exploit data to retrieve desired machine settings
- Push the way of operation
- Optimize performance

→ Better performance for users

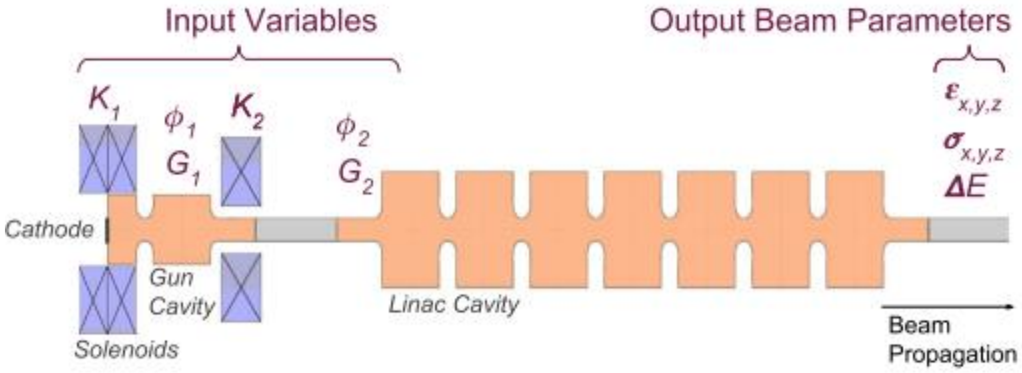
- Reinforcement Learning
- Optimization
- (Control)

[1811.03172] Opportunities in Machine Learning for Particle Accelerators (arxiv.org) 2018

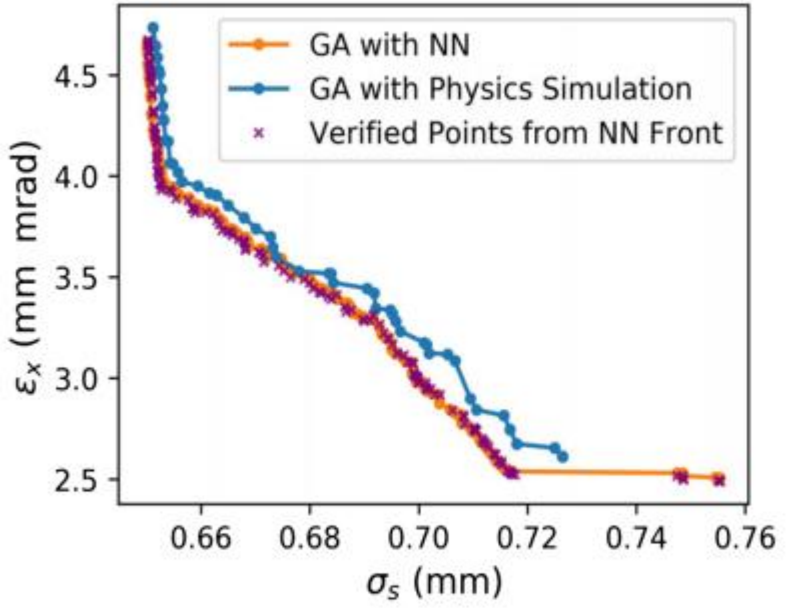
Surrogate modeling for Optimization

ML for orders of magnitude speedup in multiobjective optimization of particle accelerator systems

- Argonne Wakefield Accelerator Facility (AWA)
  - 5 input parameters
  - 7 output parameters



- Goal:
  - Optimize the beam parameters with respect to the inputs
- Approach
  - First train a neural network model on physics simulation
  - Optimize on neural network model with a genetic algorithm (GA)



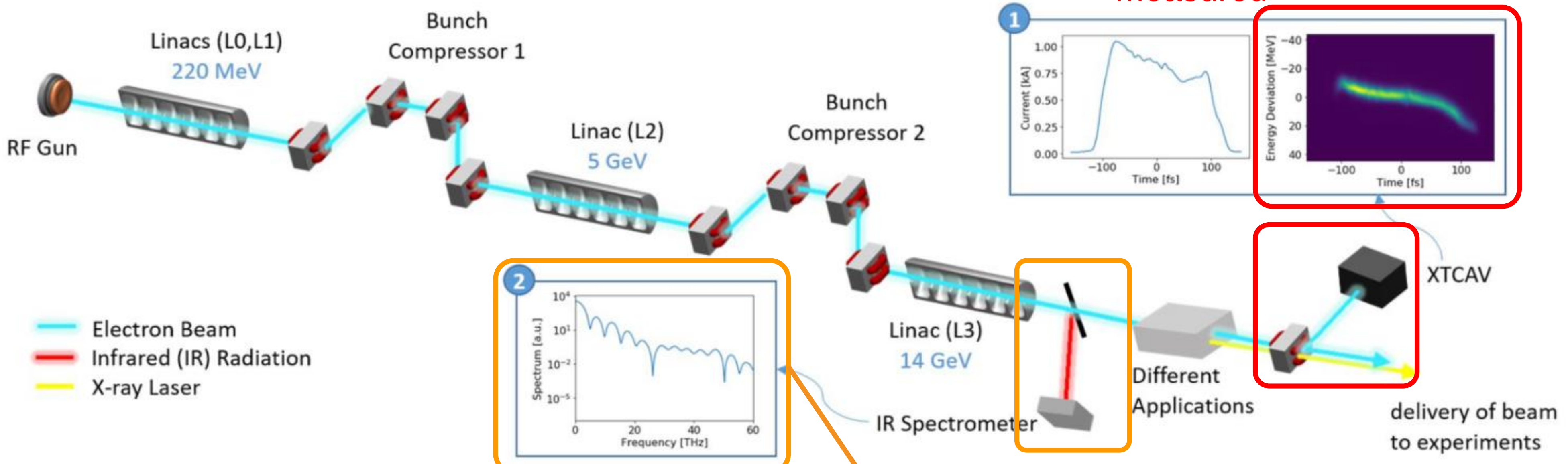
Method	Calculation	Core-hours	Wall time (hours)
Physics simulation	GA on OPAL	95 000	36
ML-based	Generate training data	660	0.33
	Train NN	0.17	0.17
	GA on NN	0.03	0.03
	Speedup—training included	144×	109×
	Speedup—training excluded	3×10^6×	1200×

A. Edelen et al., Machine learning for orders of magnitude speedup in multiobjective optimization of particle accelerator systems, Phys. Rev. Accel. Beams 23 (2020).

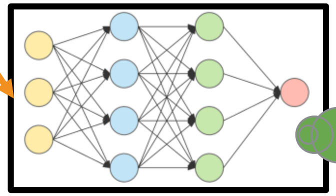
Virtual Diagnostics

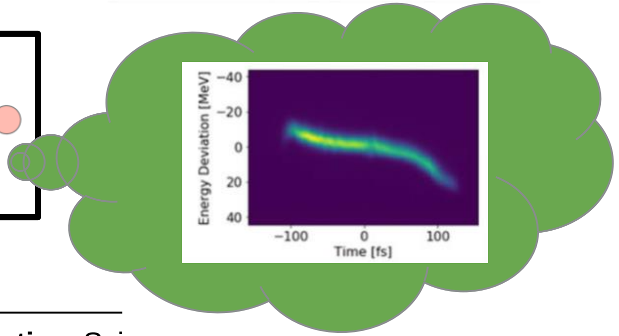
ML models infer measurements that cannot be measured

Invasive measurement blocks beam delivery -> cannot be measured



Non-invasive measurement of other diagnostic



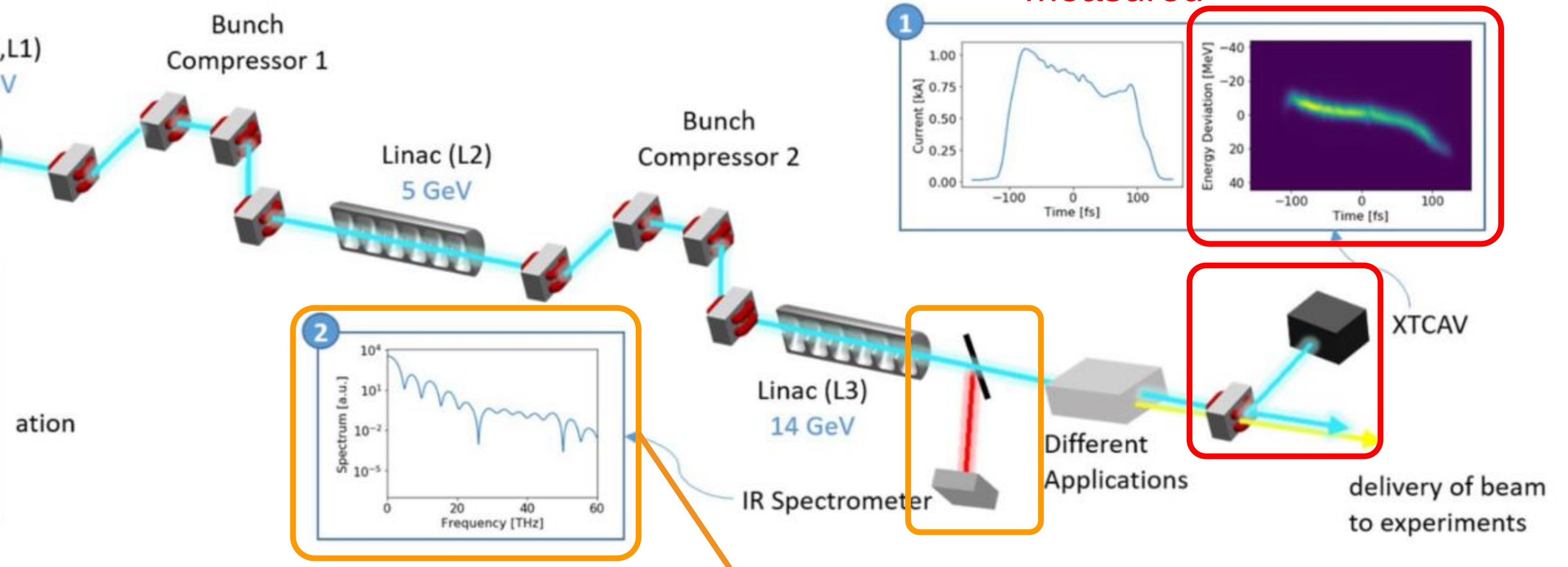


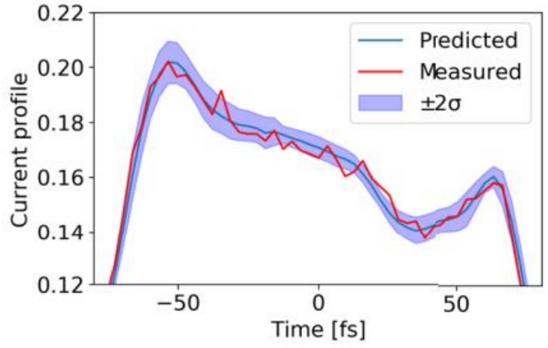
Hanuka, A. et al., **Accurate and confident prediction of electron beam longitudinal properties using spectral virtual diagnostics**, Sci Rep 11, 2945 (2021).
Emma, C. et al., **Machine learning-based longitudinal phase space prediction of particle accelerators**, Phys. Rev. Accel. Beams **21**, 112802, 2018
Zhu, J. et al., **Mixed Diagnostics for Longitudinal Properties of Electron Bunches in a Free-Electron Laser**, Front. Phys., 22 July 2022

Virtual Diagnostics

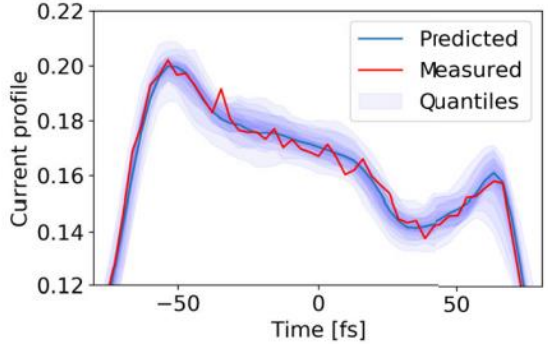
Including Robustness

Invasive measurement blocks beam delivery -> cannot be measured



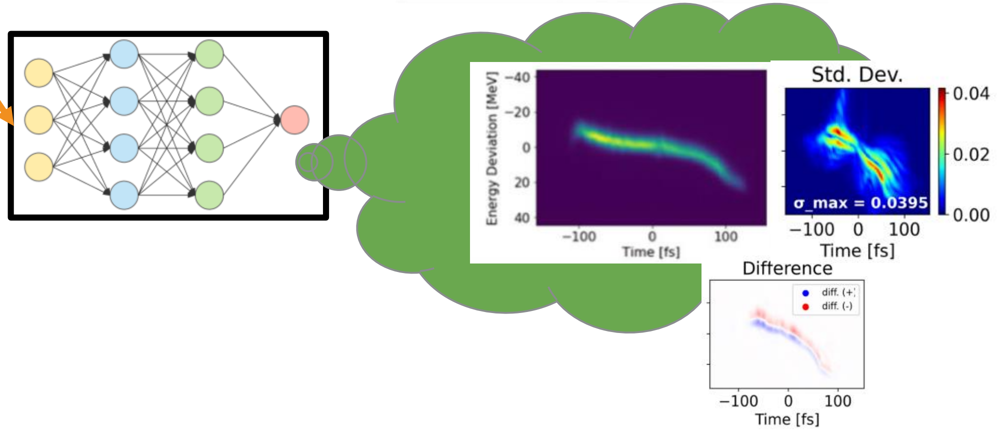


(a) Bagging ensemble



(b) Quantile regression

Non-invasive measurement of other diagnostic



Owen Convery et al., Uncertainty quantification for virtual diagnostic of particle accelerators, Phys. Rev. Accel. Beams 24 (2021).

Further Applications Estimation and Prediction

For particle accelerator

- Andreas Adelmann, On Nonintrusive Uncertainty Quantification and Surrogate Model Construction in Particle Accelerator Modeling, SIAM/ASA Journal on Uncertainty Quantification 7 (2019).
- Leander Grech, Gianluca Valentino, and Diogo Alves, A Machine Learning Approach for the Tune Estimation in the LHC, Information 12 (2021).
- A. Hanuka et al., Accurate and confident prediction of electron beam longitudinal properties using spectral virtual diagnostics, Scientific Reports 11 (2021).
- Owen Convery et al., Uncertainty quantification for virtual diagnostic of particle accelerators, Phys. Rev. Accel. Beams 24 (2021).
- A. L. Edelen et al., Neural Networks for Modeling and Control of Particle Accelerators, IEEE Transactions on Nuclear Science 63 (2016).
- Auralee Edelen et al., Machine learning for orders of magnitude speedup in multiobjective optimization of particle accelerator systems, Phys. Rev. Accel. Beams 23 (2020).
- C. Emma et al., Machine learning-based longitudinal phase space prediction of particle accelerators, Phys. Rev. Accel. Beams 21 (2018).
- C. Xu et al, Surrogate Modelling of the FLUTE Low-Energy Section, Proc. 13th International Particle Accelerator Conference, Bangkok, Thailand (2022).
- J. Zhu, et al., Mixed Diagnostics for Longitudinal Properties of Electron Bunches in a Free-Electron Laser, Front. Phys., 22 July 2022.
- J Zhu et. al., High-Fidelity Prediction of Megapixel Longitudinal Phase-Space Images of Electron Beams Using Encoder-Decoder Neural Networks, *Physical Review Applied* 16 (2), 024005 (2021).
- A. Ivanov and I. Agapov, “Physics-based deep neural networks for beam dynamics in charged particle accelerators”, Physical Review Accelerators and Beams 23, 07461 (2020).
- *M. Kirchen et al., “Optimal beam loading in a laser-plasma accelerator“ PRL 126, 174801 (2021)*

ML for Accelerators

What are the most important fields?

Data analysis

- Understanding physics
- Find new correlations of parameters
- Identify relevant data channels

→ New physical insight

- Unsupervised Learning

Estimating and predicting

Surrogate models

→ Models for online control and optimization, and for accelerator design

Virtual diagnostics

→ Additional, nondestructive, (online) information

- Supervised Learning

Fault diagnosis

- Predict & prevent failures
- Protect the system
- Identify poor conditions
- Find the root cause of errors encountered

→ Improve the availability/ reliability of machine operation

- Supervised Learning
- Unsupervised Learning
- (Statistics/Control)

Tuning and control

- Exploit data to retrieve desired machine settings
- Push the way of operation
- Optimize performance

→ Better performance for users

- Reinforcement Learning
- Optimization
- (Control)

[1811.03172] Opportunities in Machine Learning for Particle Accelerators (arxiv.org) 2018

Quench Detection using Clustering

Together with “model-based” feature extraction

- Quench detection at the European XFEL
- Pulsed-mode operation
- 808 nine-cell SRF cavities
- NO labels
- Features: Generalized likelihood ratio for residual from the physical model
- Clustering using k-means

J. Branlard et. al., “Superconducting cavity quench detection and prevention for the European XFEL,” 16th International Conference on RF Superconductivity, 2013.

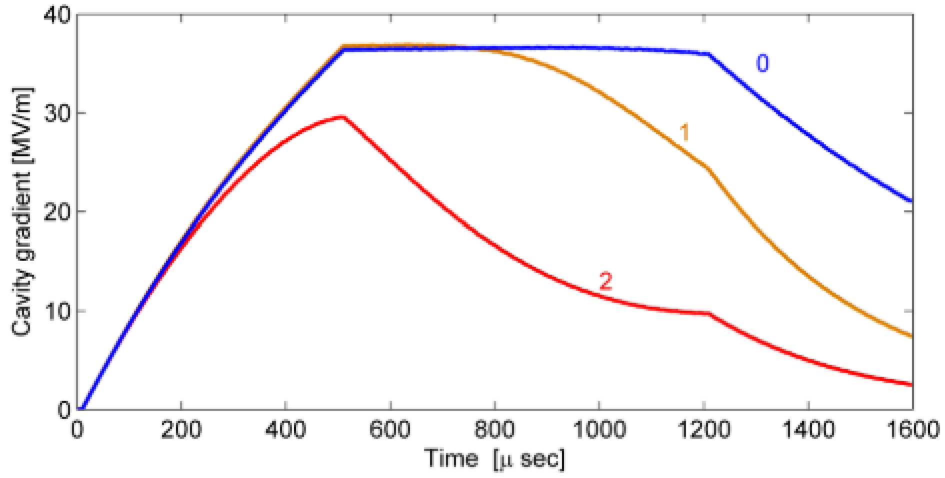
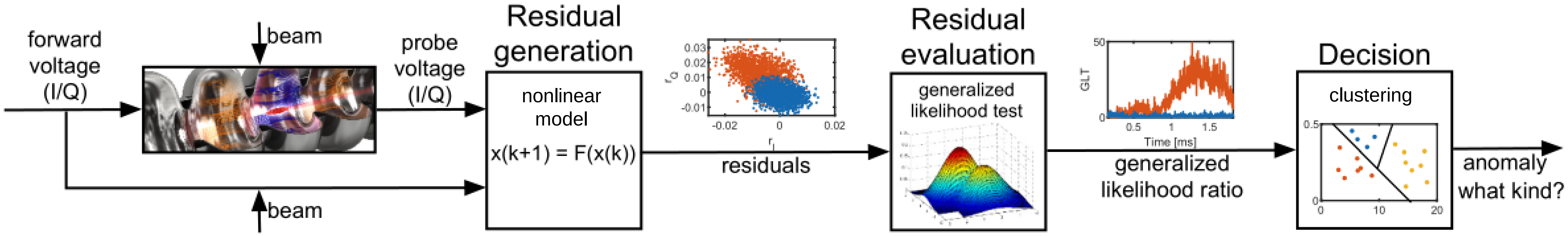


Table 1: Accuracy of QDS and GLR

	TP	TN	FP	FN	a
QDS	55	56	10	3	89.5%
GLR	55	65	1	3	96.8%

J Branlard, A Eichler, J Timm, N Walker, “Machine Learning Assisted Cavity Quench Identification at the European XFEL”, LINAC’22, 2022



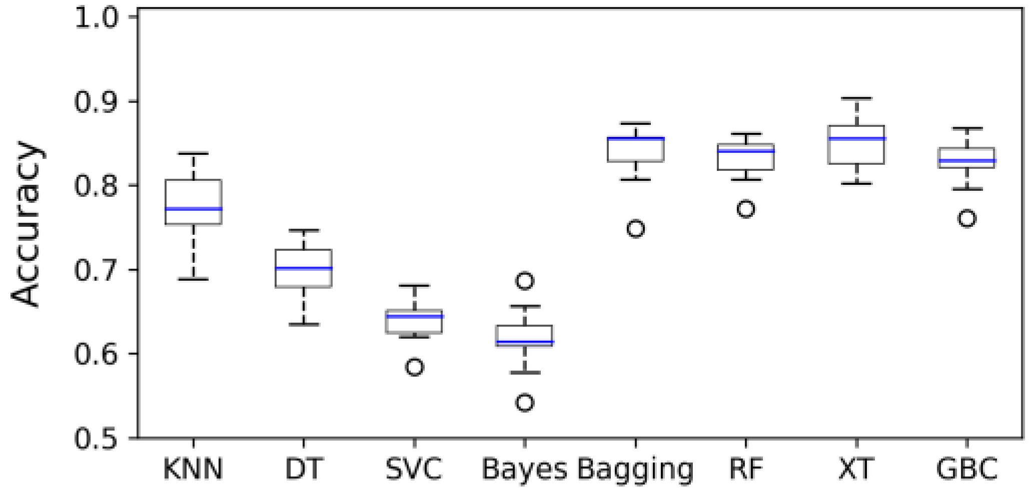
A Eichler, J Branlard, JHK Timm, “Anomaly detection at the European X-ray Free Electron Laser using a parity-space-based method”, Physical Review Accelerators and Beams 26 (1), 012801 (2023)

Quench Detection using Classification

Having labels

- Quench detection at Continuous Electron Beam Accelerator Facility (CEBAF) at Jefferson Lab
- Continuous-mode operation
- 88 seven-cell SRF cavities
- Labels for several thousand fault events (different fault classes)
- Features: parameters of fitting a AR(5) model to
  - accelerating gradient in the cavity
  - rf drive voltage
  - forward power
  - relative phase between the rf voltage applied to the cavity and the electric field minus an offset phase
- Different classification approaches have been tried

Classification results

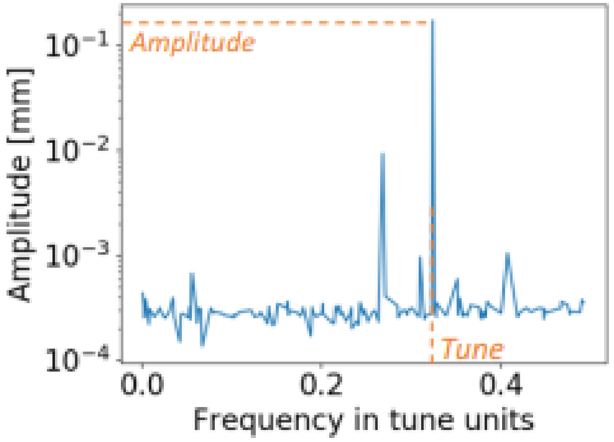


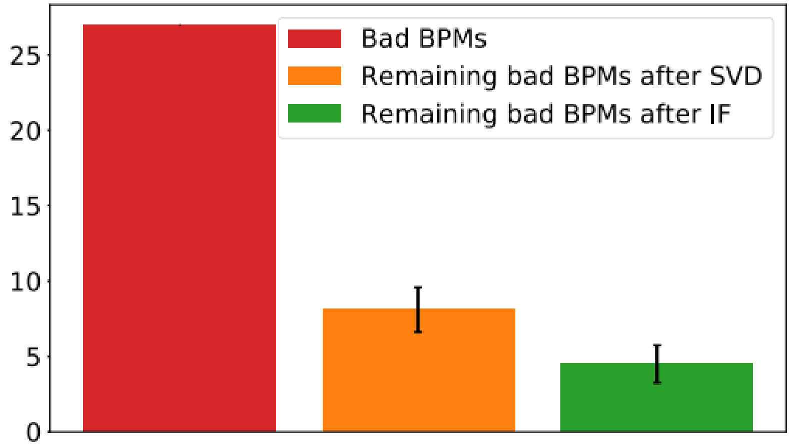
Chris Tennant et al., Superconducting radio-frequency cavity fault classification using machine learning at Jefferson Laboratory, Phys. Rev. Accel. Beams 23 (2020).

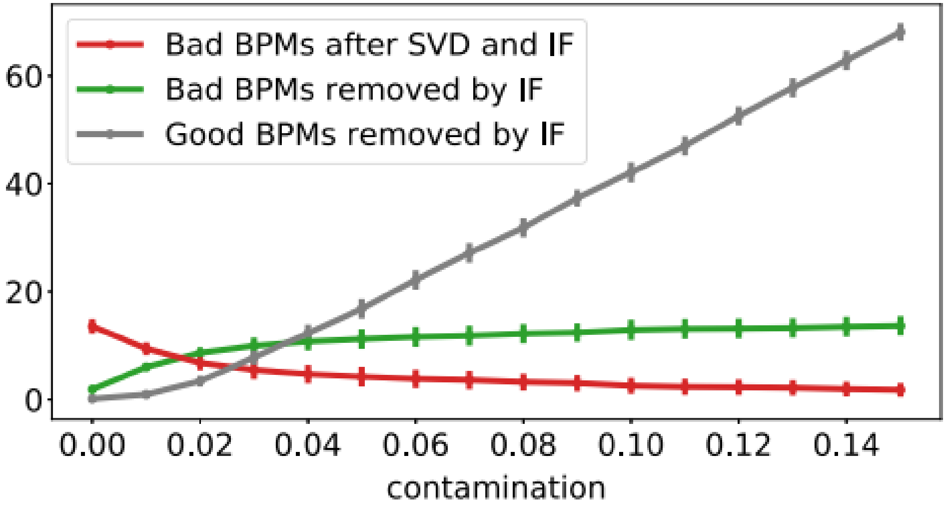
Detection of faulty beam position monitors at LHC

With Isolation Forests

- At LHC: 523 BPMs per plane and per beam
- Goal: unsupervised detection of faulty BPMs
- Features
  - Betatron tune
  - Amplitude of obtained FFT (scaled with respect to oscillation frequency)
  - Noise to amplitude ratio







E. Fol et al., Detection of faulty beam position monitors using unsupervised learning, Phys. Rev. Accel. Beams 23 (2020)

Further Applications in Fault Diagnosis

For particle accelerator

- G. Azzopardi and G. Ricci, New Machine Learning Model Application for the Automatic LHC Collimator Beam-Based Alignment, Proc. 18th Int. Conf. on Accelerator and Large Experimental Physics Control Systems, Shanghai, China, pp. 953–958 (2022).
- E. Fol et al., Detection of faulty beam position monitors using unsupervised learning, Phys. Rev. Accel. Beams 23 (2020).
- Chris Tennant et al., Superconducting radio-frequency cavity fault classification using machine learning at Jefferson Laboratory, Phys. Rev. Accel. Beams 23 (2020).
- A. Nawaz et al., Anomaly Detection for the European XFEL using a Nonlinear Parity Space Method, IFAC PapersOnLine 51 (2018).
- Ayla Nawaz et al., Probabilistic model-based fault diagnosis for the cavities of the European XFEL, at -Automatisierungstechnik 69 (2021).
- Sichen Li et al., A Novel Approach for Classification and Forecasting of Time Series in Particle Accelerators, Information 12 (2021).
- E. Fol et al., Optics Corrections Using Machine Learning in the LHC, Proc. 10th International Particle Accelerator Conference, Melbourne, Australia, pp. 3990–3993 (2019).
- E. Fol et al., Machine Learning Methods for Optics Measurements and Corrections at LHC, Proc. 9th International Particle Accelerator Conference, Vancouver, Canada, pp. 1967–1970 (2018).
- A. Grünhagen et al. "Fault analysis of the beam acceleration control system at the European XFEL using data mining." 2021 IEEE 30th Asian Test Symposium (ATS). IEEE, 2021.
- G. Martino et al. "Comparative evaluation of semi-supervised anomaly detection algorithms on high-integrity digital systems." 2021 24th Euromicro Conference on Digital System Design (DSD). IEEE, 2021.
- A. Eichler et. al., "Anomaly detection at the European X-ray Free Electron Laser using a parity-space-based method", Physical Review Accelerators and Beams 26 (1), 012801 (2023)

ML for Accelerators

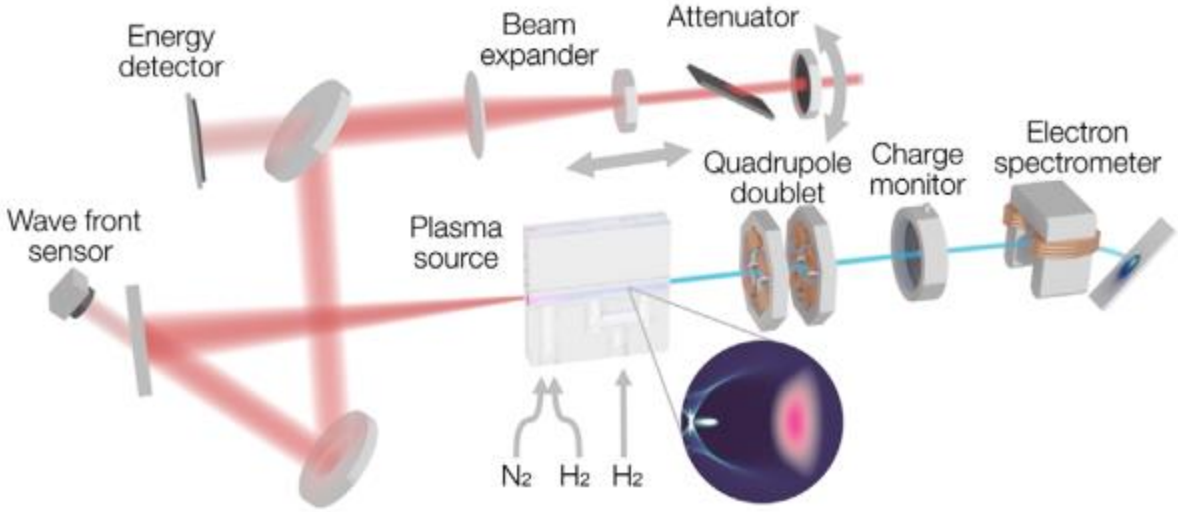
What are the most important fields?

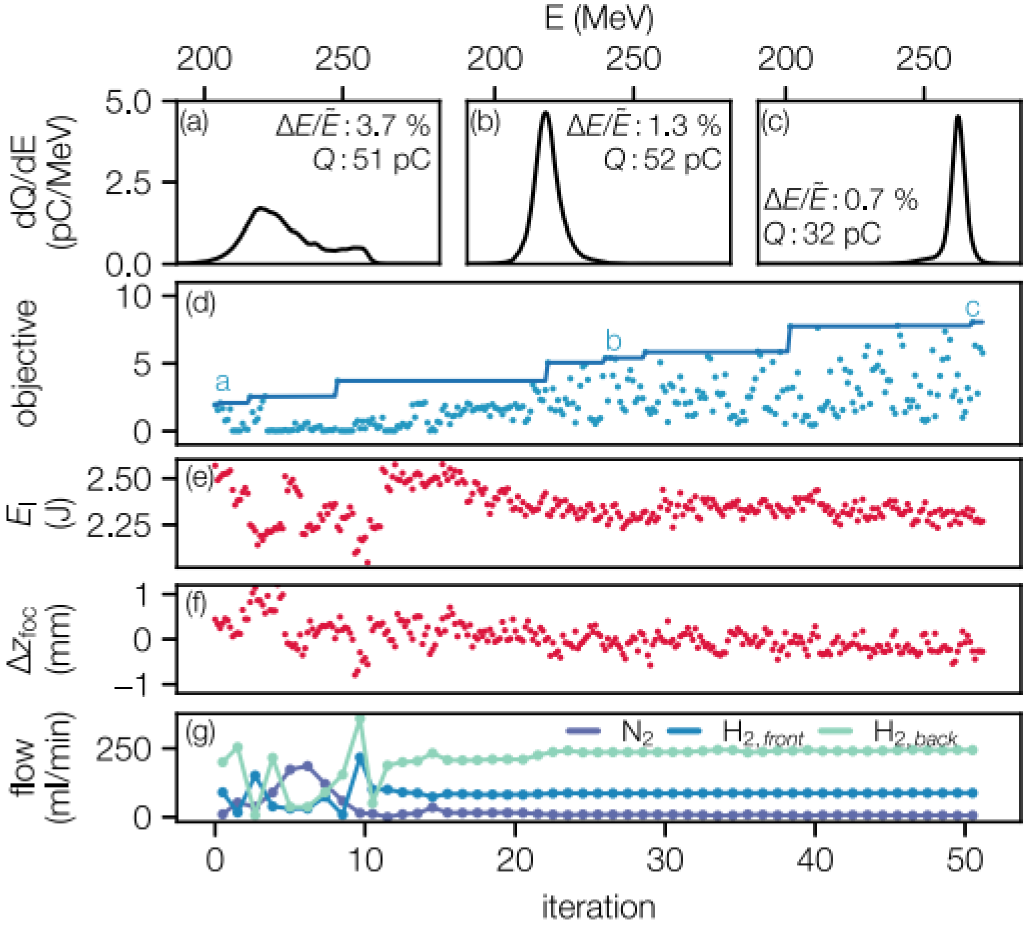
Data analysis	Estimating and predicting	Fault diagnosis	Tuning and control
• Understanding physics • Find new correlations of parameters • Identify relevant data channels → New physical insight	Surrogate models → Models for online control and optimization, and for accelerator design Virtual diagnostics → Additional, nondestructive, (online) information	• Predict & prevent failures • Protect the system • Identify poor conditions • Find the root cause of errors encountered → Improve the availability/ reliability of machine operation	• Exploit data to retrieve desired machine settings • Push the way of operation • Optimize performance → Better performance for users
• Unsupervised Learning	• Supervised Learning	• Supervised Learning • Unsupervised Learning • (Statistics/Control)	• Reinforcement Learning • Optimization • (Control)

Bayesian Optimization

Of a Laser Plasma Accelerator

LUX accelerator





Input parameters

- Laser energy (attenuator)
- Focus position (motorized lens of beam expander)
- Gas flows (N2, H2, H2)

Objective function

median energy

$$f = \sqrt{Q}\,\frac{\tilde{E}}{\Delta E}$$

bunch charge

mean absolute deviation of energy

Sören Jalas et. al. **"Bayesian Optimization of a Laser-Plasma Accelerator."** PHYSICAL REVIEW LETTERS 126, 104801 (2021)

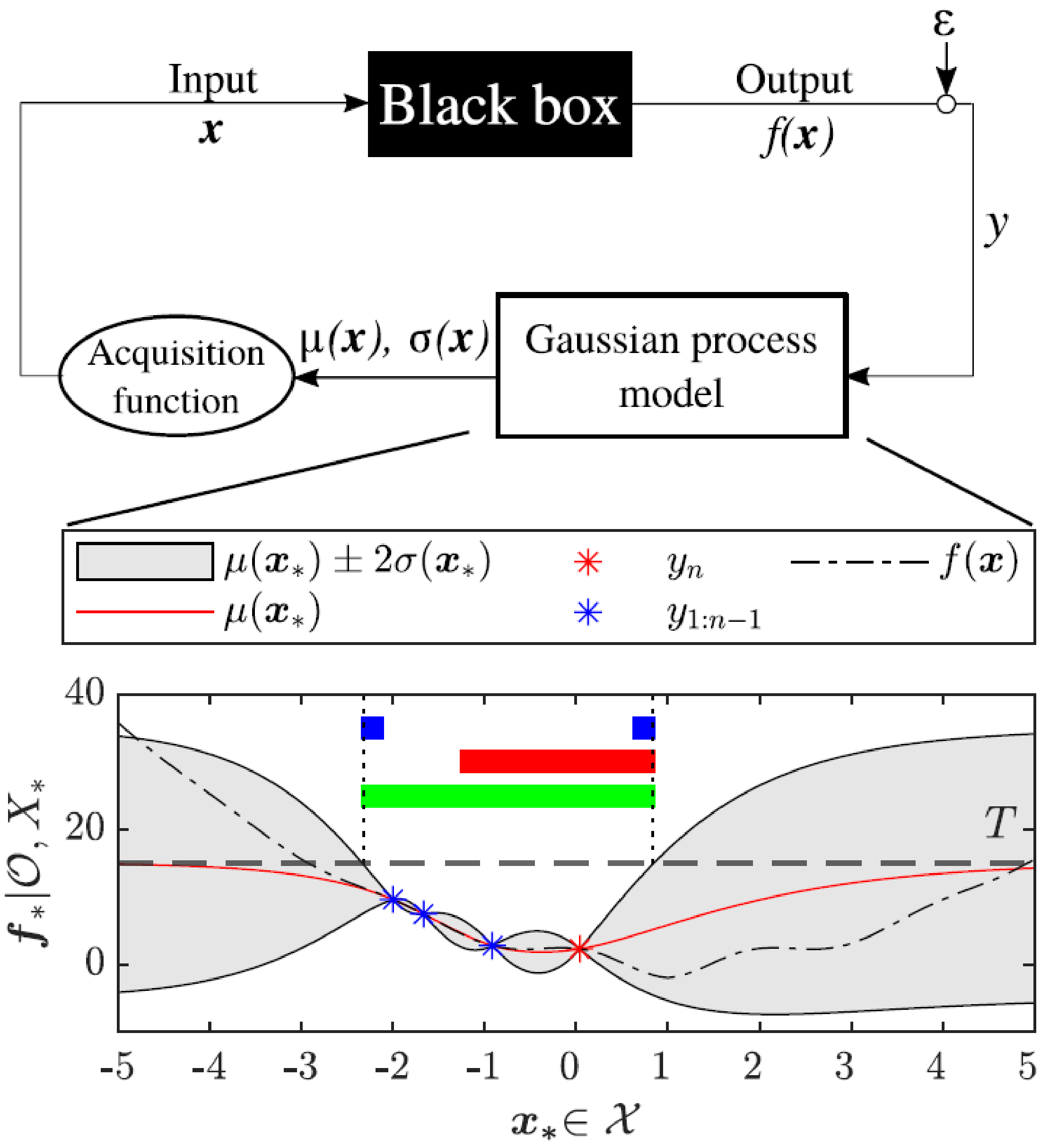
BO extensions

e.g. safe BO

- Controller optimization for the optical synchronization

min (timing jitter)^2
Controller
such that controller is decentralized, fixed order and
stabilizing
laser is never out of lock

- Safe Bayesian optimization
  - Black Box approach
  - Safe during optimization
  - Learns a probabilistic surrogate model



Further BO Applications

For particle accelerators

- J. Kirschner et al., Tuning particle accelerators with safety constraints using Bayesian optimization, Phys. Rev. Accel. Beams 25 (2022).
- Y. Gao et al., Bayesian optimization experiment for trajectory alignment at the low energy RHIC electron cooling system, Phys. Rev. Accel. Beams 25 (2022).
- C. Xu et al, Optimization Studies of Simulated THz Radiation at FLUTE, Proc. 13th International Particle Accelerator Conference, Bangkok, Thailand (2022).
- Ryan Roussel, Adi Hanuka, and Auralee Edelen, Multiobjective Bayesian optimization for online accelerator tuning, Phys. Rev. Accel. Beams 24 (2021).
- S. Jalas et al, Bayesian Optimization of a Laser-Plasma Accelerator, Phys. Rev. Let. 126 (2021).
- R. Roussel et al, Turn-key constrained parameter space exploration for particle accelerators using Bayesian active learning, nature communications 12 (2021).
- A. Edelen et al., Machine learning for orders of magnitude speedup in multiobjective optimization of particle accelerator systems, Phys. Rev. Accel. Beams 23 (2020).
- R. J. Shalloo et al., Automation and control of laser wakefield accelerators using Bayesian optimization, nature communications 11 (2020).
- J. Duris et al., Bayesian Optimization of a Free-Electron Laser, Phys. Rev. Lett. 124 (2020).
- J. Kirschner et al., Adaptive and Safe Bayesian Optimization in High Dimensions via One-Dimensional Subspaces, Proc. 36th International Conference on Machine Learning, Long Beach, USA (2019).
- J. Kirschner et al., Bayesian Optimization for Fast and Safe Parameter Tuning of SwissFEL, Proc. 39th Free Electron Laser Conf. Hamburg, German, pp. 707–710 (2019).
- M. McIntire et al., Bayesian optimization of FEL performance at LCLS, Proc. 7th International Particle Accelerator Conference, Busan, Korea, pp. 2972–2975 (2016).

Courtesy: Jan Kaiser

Reinforcement Learning for Beam Steering

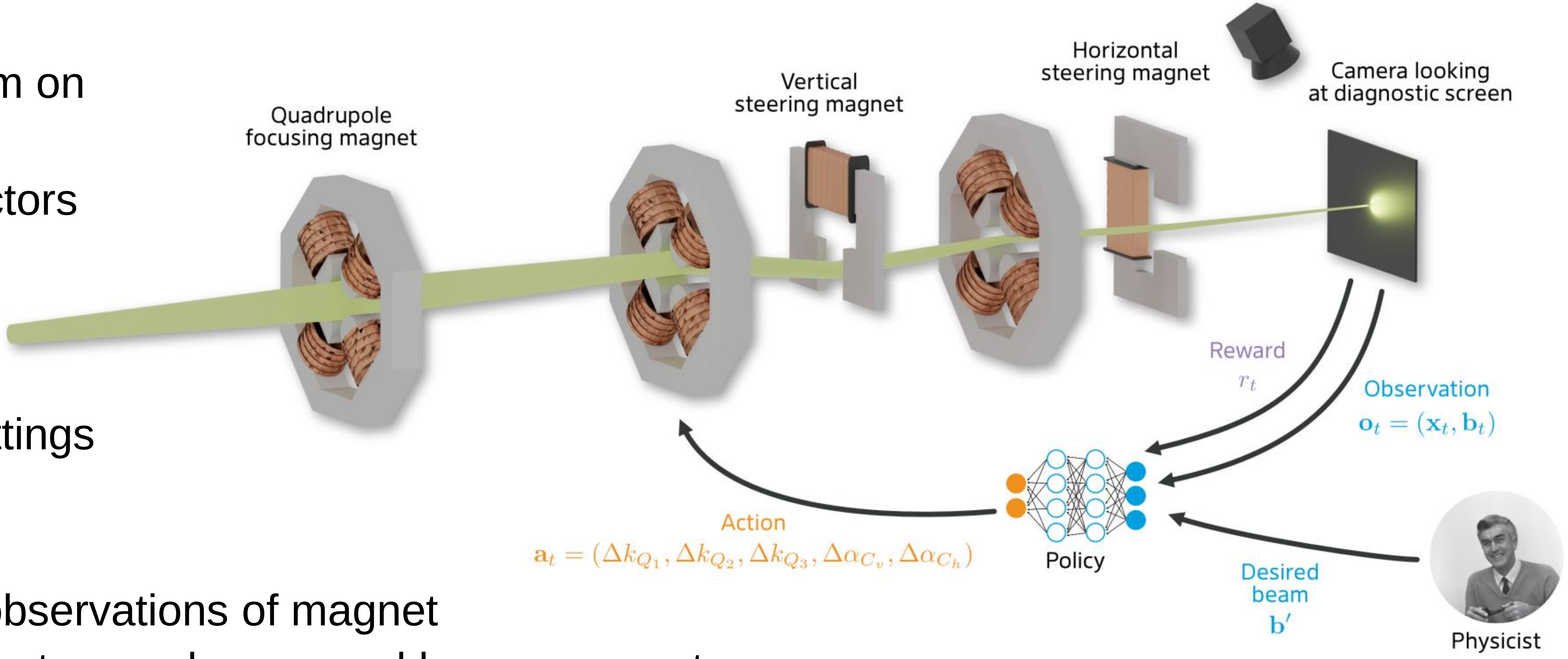
At ARES

The Task

Focus and position electron beam on a diagnostic screen using a quadrupole triplet and two correctors

Actions

Continuous 5-dimensional action defined as change to magnet settings



Observations

- Continuous 13-dimensional observations of magnet settings, desired beam parameters and measured beam parameters
- Partially observability of a more than 29-dimensional state space

Reward

Observed beam Desired beam

Improvement of the objective $O(o_t) = \ln \sum_{p \in b_t, p' \in b_t'} w_p \, |p - p'|$

Jan Kaiser, Oliver Stein, Annika Eichler. "Learning-based Optimization of Particle Accelerators Under Partial Observability Without Real-World Training." *Proceedings of the 39th International Conference on Machine Learning*, PMLR 162:10575-10585, 2022.

Reinforcement Learning for Beam Steering

Setup and results

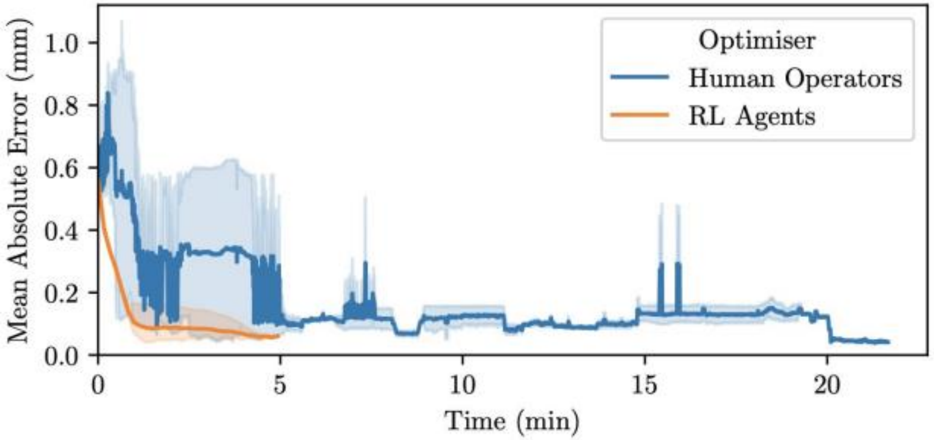
Reinforcement Learning setup

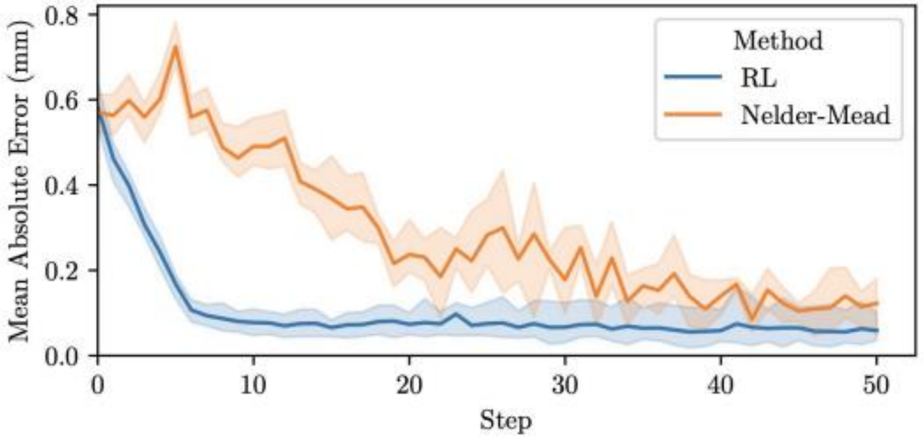
- TD3 algorithm used for training
- Agents are trained in simulation running 6 million steps
- Simulation environment: Cheetah (Simple high-speed linear beam dynamics simulation written in Python

Sim2Real transfer

- Domain randomization

Algorithm	MAE Median (mm)	Convergence Median (Steps)
Do Nothing	1.122	0
Zero	0.588	1
FDF	0.699	1
Random	0.267	101
Powell	0.259	119
COBYLA	0.105	34
Nelder-Mead	0.007	112
Bayesian	0.081	101
Ours	0.008	7
Ours (Machine)	0.036	12





O Stein, J Kaiser, A Eichler, I Agapov. "Accelerating linear beam dynamics simulations for machine learning applications." *Proceedings of the 13th International Particle Accelerator Conference, 2022*

Further RL Applications

for particle accelerator

- Learning-based optimization of particle accelerators under partial observability without real-world training - Tuning of electron beam properties on a diagnostic screen using RL.
- Sample-efficient reinforcement learning for CERN accelerator control - Beam trajectory steering using RL with a focus on sample-efficient training.
- Autonomous control of a particle accelerator using deep reinforcement learning - Beam transport through a drift tube linac using RL.
- Basic reinforcement learning techniques to control the intensity of a seeded free-electron laser - RL-based laser alignment and drift recovery.
- Real-time artificial intelligence for accelerator control: A study at the Fermilab Booster - Regulation of a gradient magnet power supply using RL and real-time implementation of the trained agent using field-programmable gate arrays (FPGAs).
- Magnetic control of tokamak plasmas through deep reinforcement learning - Landmark paper on RL for controlling a real-world physical system (plasma in a tokamak fusion reactor).
- …

Tools

What is out there

Machine Learning

Weights & Biases

PyTorch

TensorFlow

BoTorch

Keras

OpenCV

scikit learn

Xopt

21 Essential Python Tools | DataCamp

Reinforcement Learning

Gymnasium

Dopamine

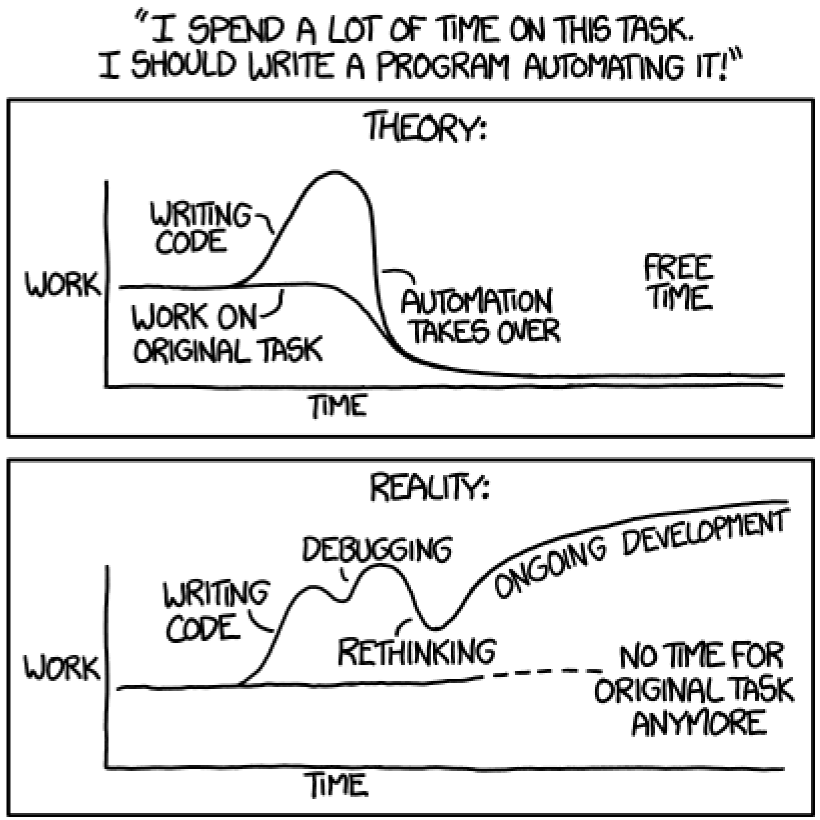
Stable baselines

rllib

Chainer RL

DeepMind TRFL

The Best Tools for Reinforcement Learning in Python You Actually Want to Try - neptune.ai



Data Visualization

matplotlib

plotly

seaborn

PANDAS PROFILING

Thank you

And many thanks to the group IPC in MSK

Contact

DESY. Deutsches Elektronen-Synchrotron

www.desy.de

Annika Eichler
MSK
annika.eichler@desy.de
+49 (0)40 8998 4041

TUHH
Hamburg University of Technology
www.tuhh.de

Annika Eichler
ICS
annika.eichler@tuhh.de

Machine Learning Round Table

Questions

- Voting:
  - Are you using machine learning? YES NO
  - Are you planning to use machine learning? YES NO

- Where are limitations of current methods? Where do you see applications? Where we are not meeting the requirements?
- How do we need to plan the infrastructure to enable the usage of ML? What is the current status of the infrastructure?
- Where do you see limitations of ML methods?
- Do you see advantages given by LLRF community?