What did we learn from Machine Learning?

Elena Fol

R. Tomás, G. Franchetti

CERN

Goethe-University Frankfurt

Outline

I. Introduction to Machine Learning
   - *What is "Learning"?*
   - *Relevant concepts and definitions*

II. Applications in Accelerator Physics
   - *Motivation*
   - *Examples*

III. Experience with ML in optics measurements and corrections

IV. Conclusions

Part I. Introduction to Machine Learning

Teaching machines to learn from experience

- Tasks that are extremely easy and obvious for us are difficult to program in traditional ways
- Impossible to learn every possible rule to perform a task
  - ➢ learn from examples instead

Teaching machines to learn from experience

- Tasks that are extremely easy and obvious for us are difficult to program in traditional ways
- Impossible to learn every possible rule to perform a task
    - ➢ learn from examples instead

Teaching machines to learn from experience

- Tasks that are extremely easy and obvious for us are difficult to program in traditional ways
- Impossible to learn every possible rule to perform a task
  - ➢ learn from examples instead

→ **Cat?**

Relevant ML concepts and definitions

*"… computer programs and algorithms that automatically **improve with experience** by **learning from examples** with respect to some class of task and performance measure, **without being explicitly programmed**." **

Supervised Learning

- Input/output pairs available
- Make prediction for unknown input based on experience from given examples

Object detection in computer vision, speech recognition, predictive control

Unsupervised Learning

- Only input data is given
- Learn structures and patterns

Anomaly detection, pattern recognition, clustering, dimensionality reduction

Reinforcement Learning

- No training data
- Interact with an environment
- Trying to learn optimal sequences of decisions

Robotics, industrial automation, dialog systems

* Thomas M. Mitchell. Machine Learning. McGraw-Hill, Inc., New York, 1997.

Supervised Learning

How does the learning work in practice?

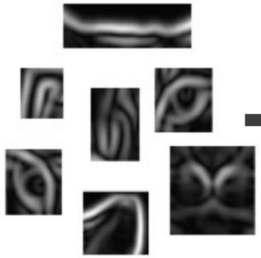
1. Collect examples

2. Preliminary processing

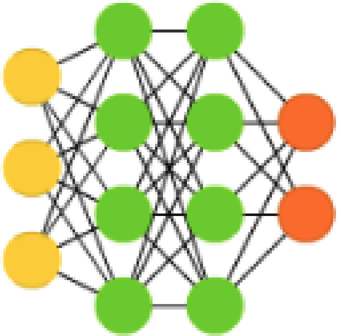
3. Training, tuning, validation

4. Prediction

Data sample



Input Features



Model

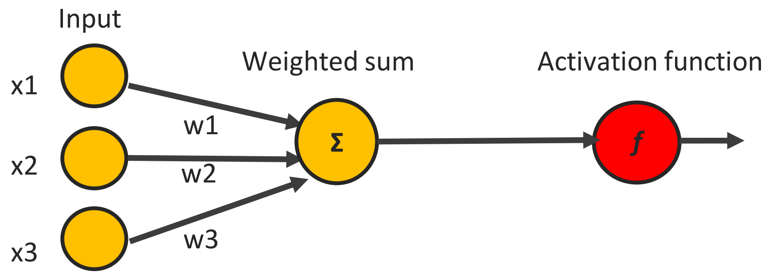
Cat

Output
(target variable)

Supervised Learning

Neural Network as an example:

- **Weights w** from the **inputs x**
- **Activation function f**
- **Output y** of a single neuron: $y = f(\sum x_i w + b)$

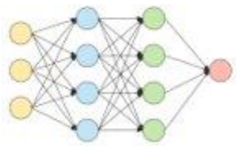


Universal Approximation Theorem: A **simple neural network** including only a single hidden layer can approximate any bounded continuous target function with arbitrary small error. *(Cybenko, 1989, for sigmoid activation functions)*

How does the learning work in practice?

example 1
example 2
example 3
.
.
.

Training input data



Function with **adjustable parameters** (weights, bias)

Model output

$$\boldsymbol{y = f\left(\sum x_i w_i + b\right)}$$

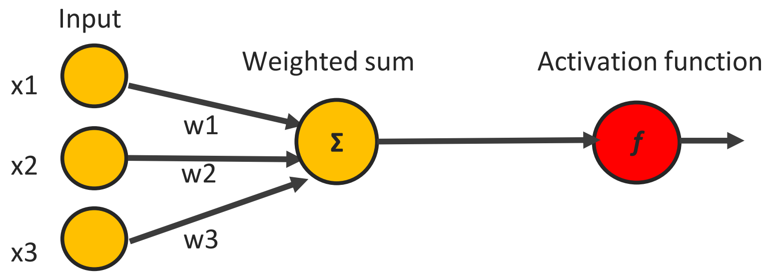
Compute the **loss** (approximation error):
e.g. MSE, MAE

Training output data

Supervised Learning

Neural Network as an example:

- **Weights w** from the **inputs x**
- **Activation function f**
- **Output y** of a single neuron: $y = f(\sum x_i w + b)$

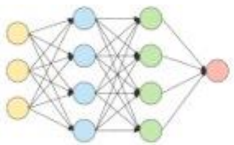


Universal Approximation Theorem: A **simple neural network** including only a single hidden layer can approximate any bounded continuous target function with arbitrary small error. *(Cybenko, 1989, for sigmoid activation functions)*

How does the learning work in practice?

example 1
example 2
example 3
.
.
.

Training input data



Function with **adjustable parameters** (weights, bias)

Model output

$$\boldsymbol{y = f\left(\sum x_i w_i + b\right)}$$

Compute the **loss** (approximation error): e.g. MSE, MAE

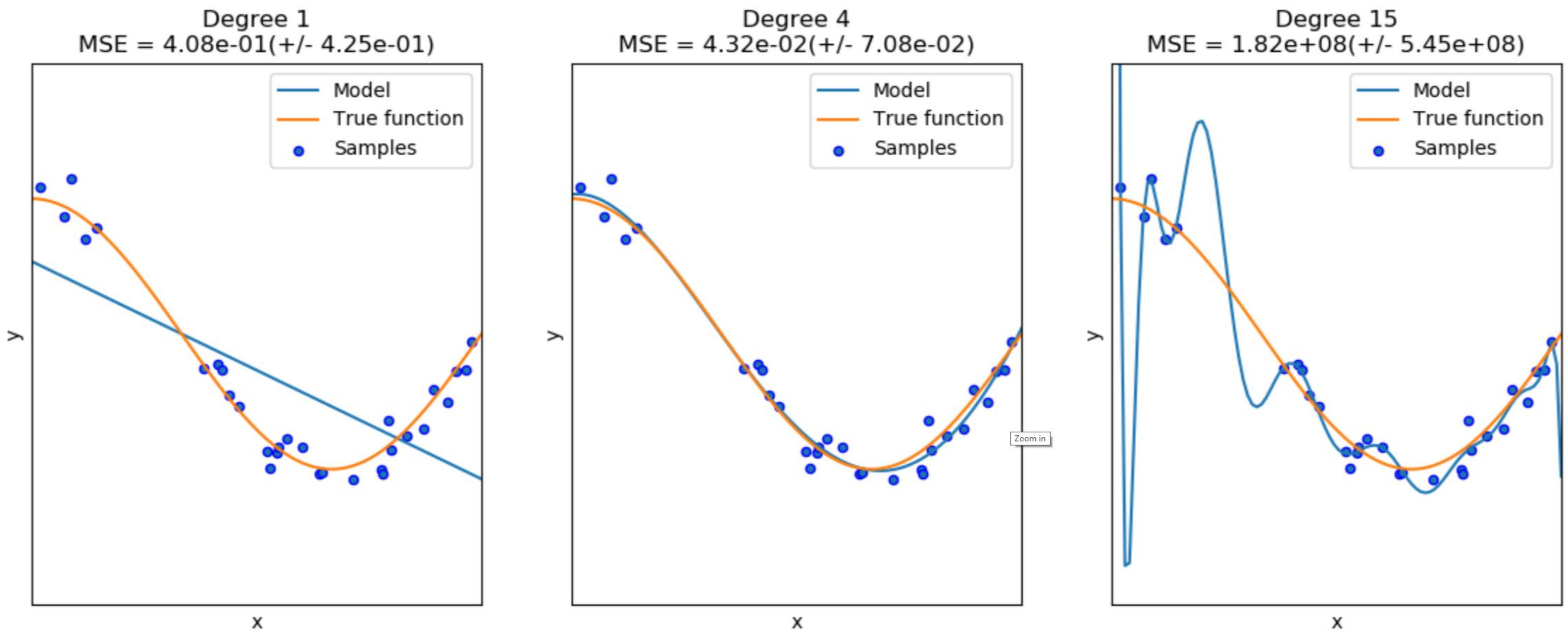
Training output data

Adjust parameters

Minimizing the loss

e.g. Gradient Descent

Training and generalization: no perfect model needed!



Simple models underfit

- Derivate from data (high bias)
- Do not correspond to data structure (low variance)

Complex models overfit

- Very low systematical deviation (low bias)
- Very sensitive to data (high variance)

We don‘t want „look up tables“

We don‘t want unreliable prediction

→ Bias-Variance tradeoff

ML is more than Neural Networks…

- **Regression and Classification Models:** *resolve correlation between input variables and dependent target variables*
  - Simple Linear Regression, Multivariate Regression, Logistic regression, Support Vector Machine
- **Dimensionality reduction techniques:** *reduce the number of independent variables (features) without significant decrease on prediction accuracy*
  - Independent Component Analysis, Principle Component Analysis, Features Importance Analysis
- **Decision Trees**: *split the input data based on a sequence of variables (thresholds) to estimate the target output value or to separate data points into regions*
  - Ensemble methods: Train several slightly different models and take majority vote/ average of the prediction
- **Clustering:** *grouping or separating data objects into clusters*
  - Identify hidden patterns in the data, similarities and differences

ML is more than Neural Networks…

- **Regression and Classification Models:** *resolve correlation between input variables and dependent target variables*
  - Simple Linear Regression, Multivariate Regression, Logistic regression, Support Vector Machine

- **Dimensionality reduction techniques:** *reduce the number of independent variables (features) without significant decrease on prediction accuracy*
  - Independent Component Analysis, Principle Component Analysis, Features Importance Analysis

- **Decision Trees**: *split the input data based on a sequence of variables (thresholds) to estimate the target output value or to separate data points into regions*
  - Ensemble methods: Train several slightly different models and take majority vote/ average of the prediction

- **Clustering:** *grouping or separating data objects into clusters*
  - Identify hidden patterns in the data, similarities and differences

Machine Learning is about learning from the data, not about application of a particular "intelligent" technique.

CERN

Part II. Application in Accelerator Physics

Motivation

Accelerators

Limitations of traditional optimization and modeling tools?

ML is a powerful tool for prediction and data analysis

Which limitations can be solved by ML with reasonable effort?

- How to deal with previously unobservable behavior?
- Required computational resources for large amount of optimization targets
- Objective functions, specific rules and thresholds have to be known

Machine Learning methods can learn an arbitrary model from given examples without requiring explicit rules

Current areas of ML application

Tuning
Optimization
Control

Virtual
Diagnostics

Anomaly and Fault
Detection

Surrogate and
Predictive
Modeling

Tuning and Control

➢ **Simultaneous optimization of multiple properties in a large parameter space, automation of manual tasks.**

- **Automatic alignment of LHC collimators based on beam loss spike recognition using an ensemble of several ML models.** *[G. Azzopardi et al., "Operational results on the fully automatic LHC collimator alignment", Phys. Rev. Accel. Beams 22, 093001 (2019)]*
- **Maximization of the average pulse energy in FELs by tune up to 105 components simultaneously based on average bunch energy.** *[A. Scheinker et al., "Model-independent tuning for maximizing free electron laser pulse energy", Phys. Rev. Accel. Beams (22), 082802 (2019)]*
- **Multi-objective optimization of minimizing transverse electron beam size at the end of the electron beam line of CERN's Advanced Proton Driven Plasma Wakefield Acceleration Experiment (AWAKE) while maintaining a design orbit.** *[A. Scheinker et al., "Online Multi-Objective Particle Accelerator Optimization of the AWAKE Electron Beam Line for Simultaneous Emittance and Orbit Control", arXiv:2003.11155]*
- **Reinforcement Learning based feedback system to stabilize the beam dynamics and control instabilities.** *[T. Boltz et al. "Feedback Design for Control of the Micro-Bunching Instability based on Reinforcement Learning", IPAC'19 (MOPGW017)]*
- **Bayesian approach for maximizing x-ray laser pulse energy by controlling groups of quadrupole magnets.** *[J. Duris et al. Bayesian Optimization of a Free-Electron Laser, Phys. Rev. Lett. 124, 124801 (2020)]*

Tuning and Control

- **Simultaneous optimization of multiple properties in a large parameter space, automation of manual tasks.**

- **Automatic alignment of LHC collimators based on beam loss spike recognition using an ensemble of several ML models.** *[G. Azzopardi et al., "Operational results on the fully automatic LHC collimator alignment", Phys. Rev. Accel. Beams 22, 093001 (2019)]*
- **Maximization of the average pulse energy in FELs by tune up to 105 components simultaneously based on average bunch energy.** *[A. Scheinker et al., "Model-independent tuning for maximizing free electron laser pulse energy", Phys. Rev. Accel. Beams (22), 082802 (2019)]*
- **Multi-objective optimization of minimizing transverse electron beam size at the end of the electron beam line of CERN's Advance** … **g a design orbit.** *[A. Schein* … *on Beam Line for Simultaneous E* …
- **Reinforcement** … **s.** *[T. Boltz et al. "Feedback Desi* … *9 (MOPGW017)]*
- **Bayesian appr** … **ets.** *[J. Duris et al. B* …

- **The widest application area since ML tools are naturally developed for automation/ control tasks,**
- **Becoming standard tools for operation.**

- **Various ML techniques can be applied to one problem – choice based on experience / already available frameworks,**
- **But also: one developed technique can be applied to several problems on different machines.**

Anomaly and Faults Detection

- **Detection and prevention of unusual, undesired events.**

- **Detection of outliers in the numerical computations of dynamic aperture** *[M. Giovannozzi et al., "Machine learning and beam dynamics activities at the CERN Large Hadron Collider.", to be published]*
- **Automatic detection of heating effects based on pressure reading, labeling datasets for supervised classification using clustering** *[F. Giordano et al. "Automatic classification of vacuum gauge", https://indico.cern.ch/event/927925]*
- **Instability detection for the LHC transverse feedback system using decision trees** *[L. Coyle et al., https://wiki.epfl.ch/fcc-epfl-lpap/machinelearning]*
- **Detection and classification of RF Cavity Faults using supervised models** *[A. Solopova et al. , "SRF Cavity Faults Classification Using Machine Learning at CEBAF", IPAC'19 (TUXXPLM2)]*
- **Hierarchical Temporal Memory and Recurrent NNs applied to anomaly detection for sensor failures at HIPA, PSI** *[J. Coello de Portugal, https://indico.psi.ch/event/8624]*

Anomaly and Faults Detection

- Detection and prevention of unusual, undesired events.

- **Detection of outliers in the numerical computations of dynamic aperture** *[M. Giovannozzi et al., "Machine learning and beam dynamics activities at the CERN Large Hadron Collider.", to be published]*
- **Automatic detection of heating effects based on pressure reading, labeling datasets for supervised classification using clustering** *[F. Giordano et al. "Automatic classification of vacuum gauge", https://indico.cern.ch/event/927925]*
- **Instability detection for the LHC transverse feedback system using decision trees** *[L. Coyle et al., https://wiki.epfl.ch/fcc-epfl-lpap/machinelearning]*
- **Detection an** … *Faults Classification*
- **Hierarchical T** … **PA, PSI** *[J. Coello de P*

- **Some operational examples**
- **Many studies under investigation**
- **Unsupervised learning can be used, without requiring large amount of data.**

- **Careful choice of data to perform the classification/ anomaly detection is crucial (feature engineering),**
- **Structured data logging can be very helpful.**

Surrogate and Predictive Modeling

➢ **Learning underlying physical processes in the correlations of input and output data.**

- **Orbit corrections studies, early attempts:**

  Y. Kijima, "A beam diagnostic system for accelerator using Neural Networks", 1992
  E. Bozoki, "Neural Network technique for orbit correction in accelerators", 1994
  E. Meier, "Orbit correction studies using Neural Networks", 2012

- **Bayesian approach for linear optics correction** *[Y. Li, R. Rainer, W. Cheng, "Bayesian approach for linear optics correction", Phys. Rev. Accel. Beams (22), 012804 (2019)]*
- **Nonlinear, fast-executing surrogate models that are trained on sparse sampling of the physics simulation** *[A. Edelen et al., "Machine Learning for Orders of Magnitude Speedup in Multi-Objective Optimization of Particle Accelerator Systems ", Phys. Rev. Accel. Beams (23), 044601 (2020)]*
- **Training of NN to predict vertical beam size at one position from multiple Insertion Device settings, allows to correct the beam size** *[S.C. Leemann, "Demonstration of Machine Learning-Based Model-Independent Stabilization of Source Properties in Synchrotron Light Sources", Phys. Rev. Lett. 123, 194801 (2019)]*

Surrogate and Predictive Modeling

➢ **Learning underlying physical processes in the correlations of input and output data.**

- **Orbit corrections studies, early attempts:**

  Y. Kijima, "A beam diagnostic system for accelerator using Neural Networks", 1992
  E. Bozoki, "Neural Network technique for orbit correction in accelerators", 1994
  E. Meier, "Orbit correction studies using Neural Networks", 2012

- **Bayesian approach for linear optics correction** *[Y. Li, R. Rainer, W. Cheng, "Bayesian approach for linear optics correction", Phys. Rev. Accel. Beams (22), 012804 (2019)]*
- **Nonlinear, fast-executing surrogate models that are trained on sparse sampling of the physics simulation** *[A. Edelen et al., "Machine Learning for Orders of Magnitude Speedup in Multi-Objective Optimization of Particle Accelerator Systems ", Phys. Rev. Accel. Beams (23), 044601 (2020)]*
- **Training of** … **s to correct the** … *Stabilization* …

➢ **Neural Networks can be very useful to model non-linear tasks,**
➢ **Also simpler liner models can be applied if the problem is known to be linear.**
➢ **Different ML methods are investigated demonstrating good suitability for modeling tasks.**
➢ **Wide application range from simulations to real time diagnostics.**

CERN

Virtual Diagnostics

- **Provide real-time information using non-invasive diagnostics by predicting beam parameters from other available properties.**

- **Single-shot x-ray diagnostics in an XFEL : learning physical process behind correlations between input and output of ML model, while analytical modelling of every experimental aspect is not possible.** *[A. Sanchez-Gonzalez, et al., “Machine learning applied to single-shot x-ray diagnostics in an XFEL”,* https://arxiv.org/abs/1610.03378*]*

- **Longitudinal phase space (LPS) prediction using ANN, based on the correlation between LPS distribution and various accelerator parameters.** *[C. Emma et.al "Machine learning-based longitudinal phase space prediction of particle accelerators", Phys. Rev. Accel. Beams (21), 112802 (2018)]*
- **Computer vision techniques to predict x-ray temporal power profile .** *[X. Ren et.al. “Temporal power reconstruction for an x-ray free-electron laser using convolutional neural networks ”, Phys. Rev. Accel. Beams 23, 040701 (2020)]*
- **Prediction of longitudinal phase information from noisy beam position monitors data based on ML-image processing.** *[X. Xu, Y. Zhou, and Y. Leng, “Machine learning based image processing technology application in bunch longitudinal phase information extraction”, Phys. Rev. Accel. Beams 23, 032805 (2020)]*
- **Classification of beam losses to understand the impact on luminosity and lifetime of accelerator components**

[G. Valentino, B. Salvachua , “Machine Learning applied at the LHC for beam loss pattern classification”, IPAC18 (WEPAF078)]

Virtual Diagnostics

- **Provide real-time information using non-invasive diagnostics by predicting beam parameters from other available properties.**

- **Single-shot x-ray diagnostics in an XFEL : learning physical process behind correlations between input and output of ML model, while analytical modelling of every experimental aspect is not possible.** *[A. Sanchez-Gonzalez, et al., "Machine learning applied to single-shot x-ray diagnostics in an XFEL", https://arxiv.org/abs/1610.03378]*

- **Longitudinal phase space (LPS) prediction using ANN, based on the correlation between LPS distribution and various accelerator parameters.** *[C. Emma et.al "Machine learning-based longitudinal phase space prediction of particle accelerators", Phys. Rev. Accel. Beams (21), 112802 (2018)]*
- **Computer vision techniques to predict x-ray temporal power profile .** *[X. Ren et.al. "Temporal power reconstruction for an x-ray free-electron laser using convolutional neural networks ", Phys. Rev. Accel. Beams 23, 040701 (2020)]*
- **Prediction of longitudinal phase information from noisy beam position monitors data based on ML-image processing.** *[X. Xu, Y. Zhou, and Y. Leng, "Machine learning based image processing technology application in bunch longitudinal phase information extraction", Phys. Rev. Accel. Beams 23, 032805 (2020)]*
- **Classifica … onents** *[G. Valentin … C18 (WEPAF078)*

> - **Virtual diagnostics tools mostly implemented using neural networks,**
> - **For image-based diagnostics: Convolutional Neural Networks (known to be very efficient in computer vision)**
> - **Simulation studies, but some methods also demonstrated on the machines.**

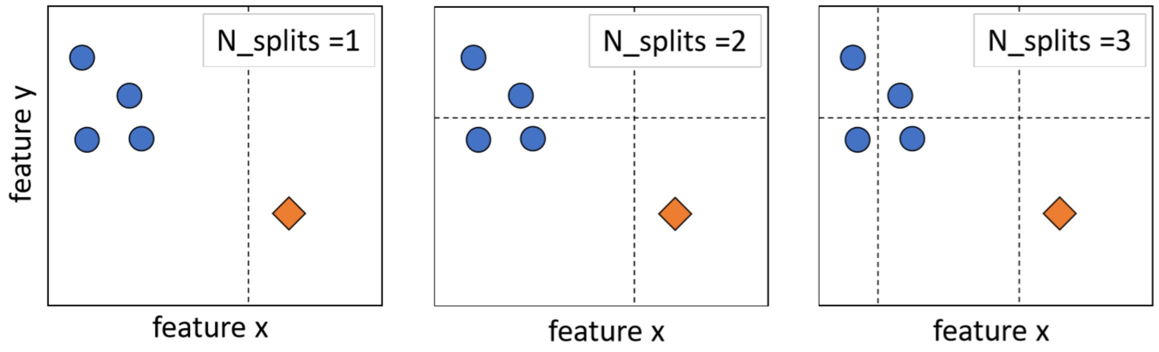
Part II. Experience with ML in optics measurements and corrections

I. Detection of faulty BPMs

- Optics is reconstructed from Beam Position Monitors (BPM) signal which is denoised and cleaned using SVD and signal cuts.
- Presence of remaining faulty signal can be observed only in the last analysis step – optics reconstructed from BPM signal → manual cleaning and repeating optics computation are required.

→ How to detect as many faulty BPMs as possible before they appear as outliers in optics functions?

→ **Unsupervised Learning using Isolation Forest (Ensemble of Decision Trees)**



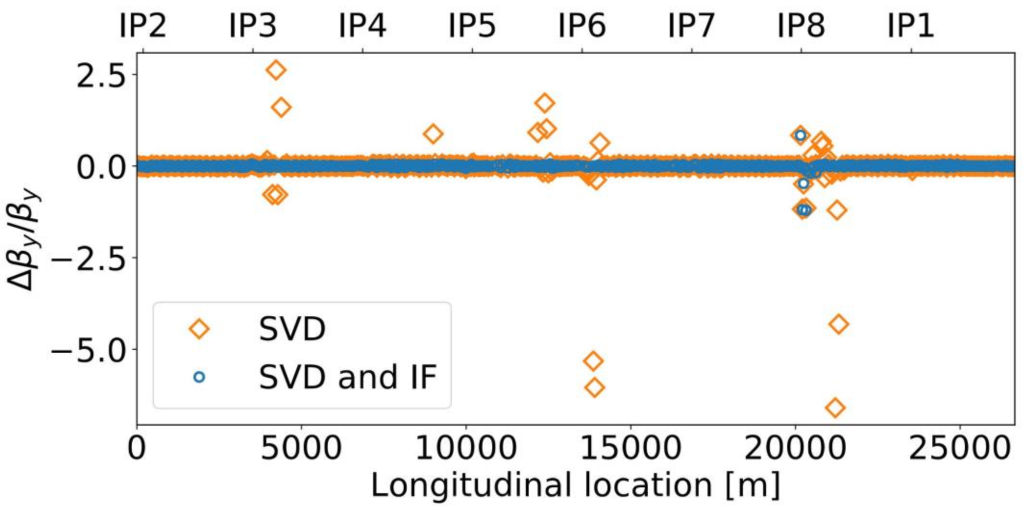
Conceptual illustration of Isolation Forest algorithm

- **Random splits aiming to "isolate" each point.**
- The less splits are needed, the more "anomalous".
- Expected proportion of outliers is a parameter of the algorithm.

I. Detection of faulty BPMs

Using harmonic properties of BPM turn-by-turn signal as input data and setting the expected proportion of anomalies in the data to 1%.

➢ **Isolation Forest detects most of the faulty BPMs remaining after the cleaning with traditional tools.**



IF-cleaning is based on the structures in given data
→ ***Ability to identify anomalies without predefined thresholds or rules.***

False identification of good BPMs as anomalies is possible
→ ***Choice of contamination parameter and interplay with other cleaning tools are carefully studied on simulations.***

✓ IF is fully integrated into optics measurements at LHC
✓ Successfully used in operation under different optics settings.

II. Estimation of quadrupole errors

- Corrections aim to minimize the difference between the measured and design optics by changing the strength of corrector magnets – single quadrupoles and quadrupoles powered in circuits.

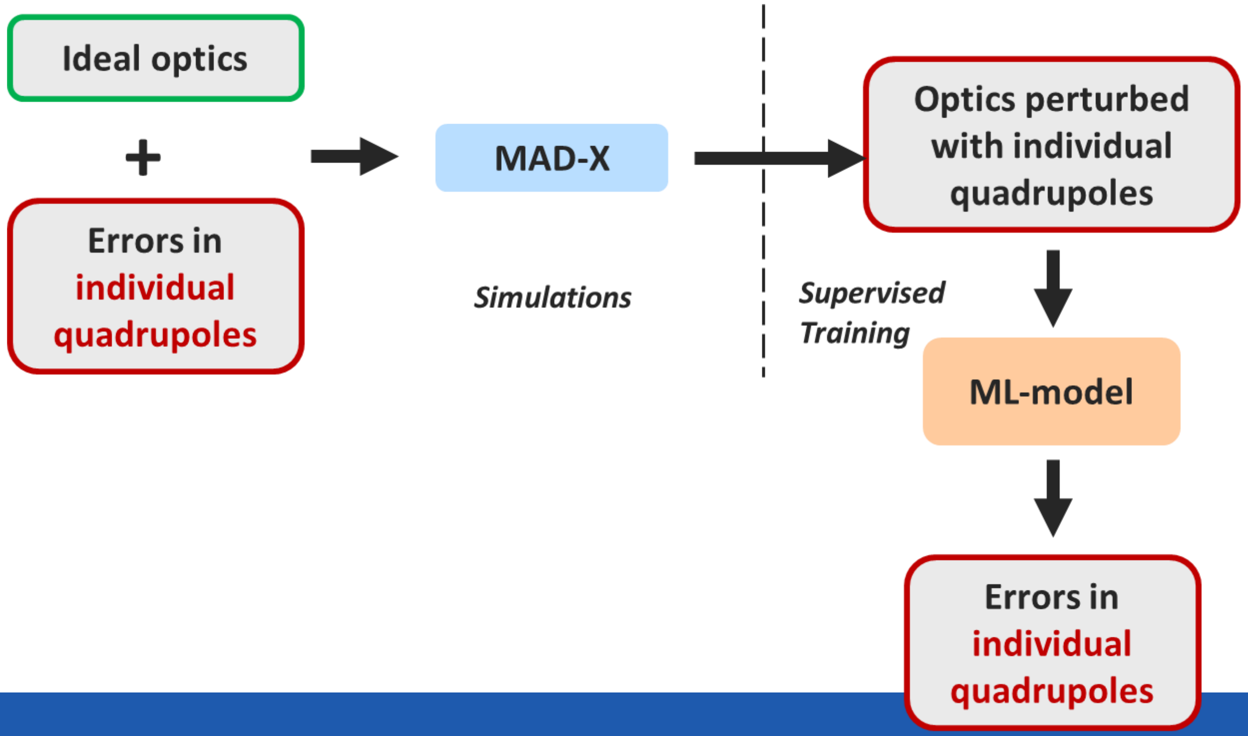
How to get the entire set of currently present magnet errors in one step?
→ Train **supervised regression model** to predict magnet errors from optics perturbations caused by these errors.

Training samples:
- **1256 target** variables

- randomly assigned gradient errors in the **all** quadrupoles, **both** beams

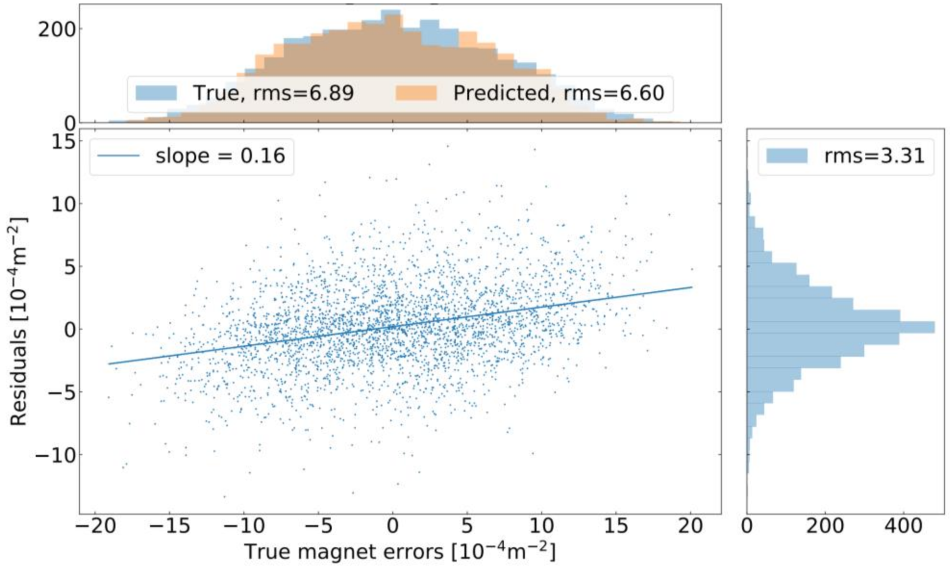
- **3304 input** variables: simulated betatron phase advance, normalized dispersion at all BPMs, β at BPMs next to IPs.

- Adding realistic noise estimated from the measurements.



II. Estimation of quadrupole errors

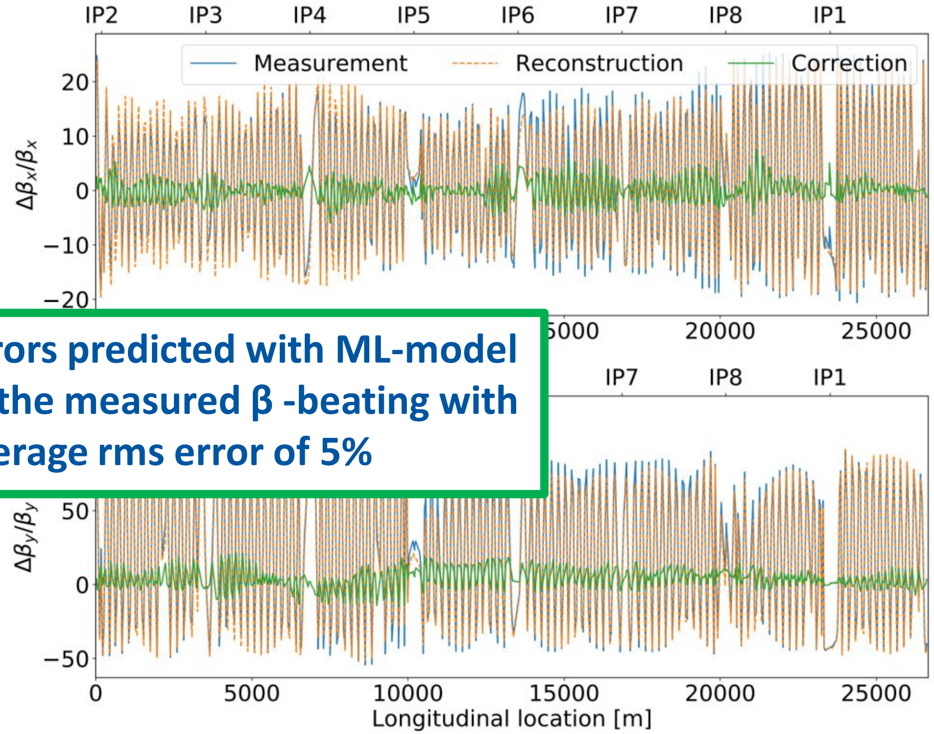
- Linear regression with weights regularization (Ridge)
- 75 000 samples from **simulations** (80% training, 20% test)
- Systematic error of prediction **16%,** random error ~30%



Prediction of simulated individual magnet errors.

Supervised Learning allows to determine realistic quadrupole errors directly from optics deviations.

- Verification on measurements data: provide LHC optics **measurement** as input to the trained model.
- Real individual magnet errors are unknown
  → Simulate optics perturbation with predicted errors
  → Compare to the measurement used as input.



Magnet errors predicted with ML-model reproduce the measured β -beating with average rms error of 5%

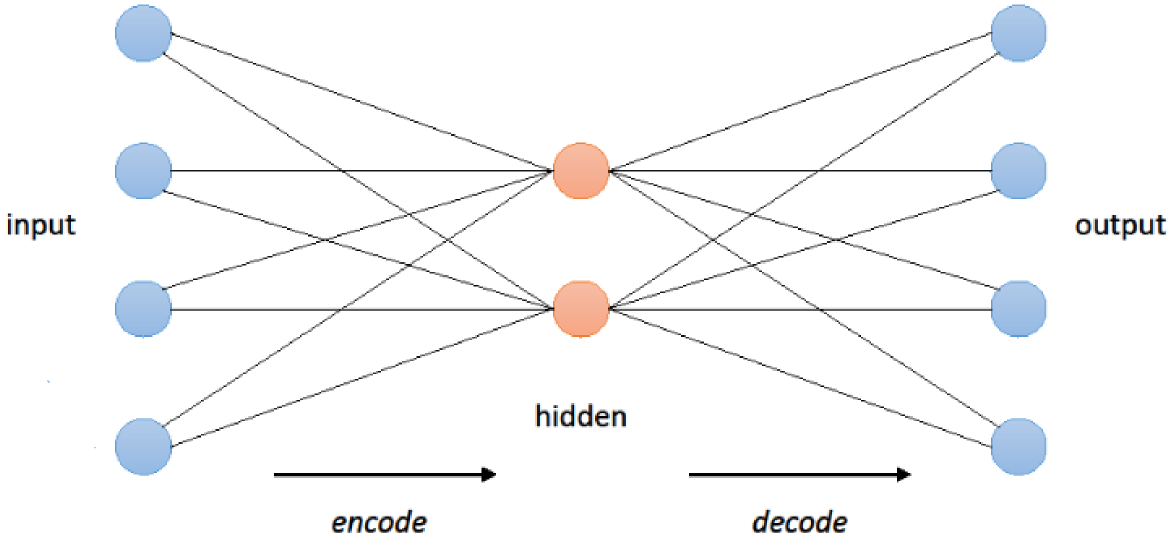
II. Estimation of quadrupole errors

- Results on real data are less optimistic than simulations
  → **More realistic simulations of training data** by introducing more error sources.
- In some machine sections individual quadrupoles cannot be directly used for corrections – only circuits.
  → **Translate the predicted single magnets errors into correction settings.**

- Quality of prediction depends on **noise in the input data** and available BPMs:

→ **Autoencoder Neural Network to denoise and reconstruct phase measurements.**

First results on simulations:
→ Noise is reduced by a factor of 2
→ Phase advance measurements at simulated missing BPMs reconstructed with 1% accuracy.



Part V. Conclusions

ML in accelerators: summary

Accelerator Problem	ML methods	Benefits	To be considered
• Automation of particular components	Supervised techniques for classification: Decision Trees, SVR, Logistic Regression, NN	Saving operation time, reducing human intervention, preventing subjective decisions	Dedicated machine time usually required to collect training data and to fine tune developed methods.
• Online optimization of several targets which are coupled • Unexpected drifts, continuous settings readjustment needed to maintain beam quality	Reinforcement Learning, Bayesian optimization, Gaussian Process, Adaptive Feedback	Simultaneous optimization targeting several beam properties, automatically finding trade-off between optimization targets, allows faster tuning offering more user time.	Ensuring that all important properties are included as optimization targets.
• Detection of anomalies	Unsupervised methods: clustering, ensembles of decision trees (e.g. Isolation Forest), supervised classification, Recurrent NN for time-series data.	Preventing faults before they appear, no need to define rules/ thresholds, no training is needed and can be directly applied on received data	In unsupervised methods, usually no “ground truth” is available → methods can be verified on simulations.

CERN

ML in accelerators: summary

Accelerator Problem	ML methods	Benefits	To be considered
• Computationally heavy, slow simulations • Reconstruct unknown properties from measurements	Supervised Regression models, NN for non-linear problems	Learning underlying physics directly from the data, faster execution	100% realistic simulations are not possible → the model performance will be as good as your data is.
• Reduction of parameter space e.g. for optimization	Clustering, Feature Importance Analysis using Decision trees	Speed up of available methods, simpler defined problems, easier to interpret	Parameter selection and combination (feature engineering) can have significant impact on ML methods performance
• Missing or too noisy data	Autoencoder NN	Robust models, data quality	Significant information should not be removed from the signal.

Potentially useful, but not (widely) used yet

Some ideas…

- **Transfer Learning**

  - Train model on one problem domain, apply on another task after re-training,
  - Data set required for re-training can be much smaller than data set used in initial training,
  - Real-time application (e.g. re-training using data recorded in operation),
  possibility to take advantage from previous efforts.

- **Inverted Models**

  - Train to predict a set of output targets from a set of input parameters.
  - Invert the model and use the learned correlation to predict the "input" parameters from targets,
  e.g. to predict settings from desired beam properties.

- **Text processing**

  - Logbooks contain a lot of unstructured information, which can be relevant to build automation/control ML tools.
  - Extract relevant information automatically by analyzing text entries using e.g. Ontology Learning.
  - Use extracted information to build models for machine components failures prediction, to automatize operation, etc.

Conclusions

Important to identify where ML can surpass traditional methods

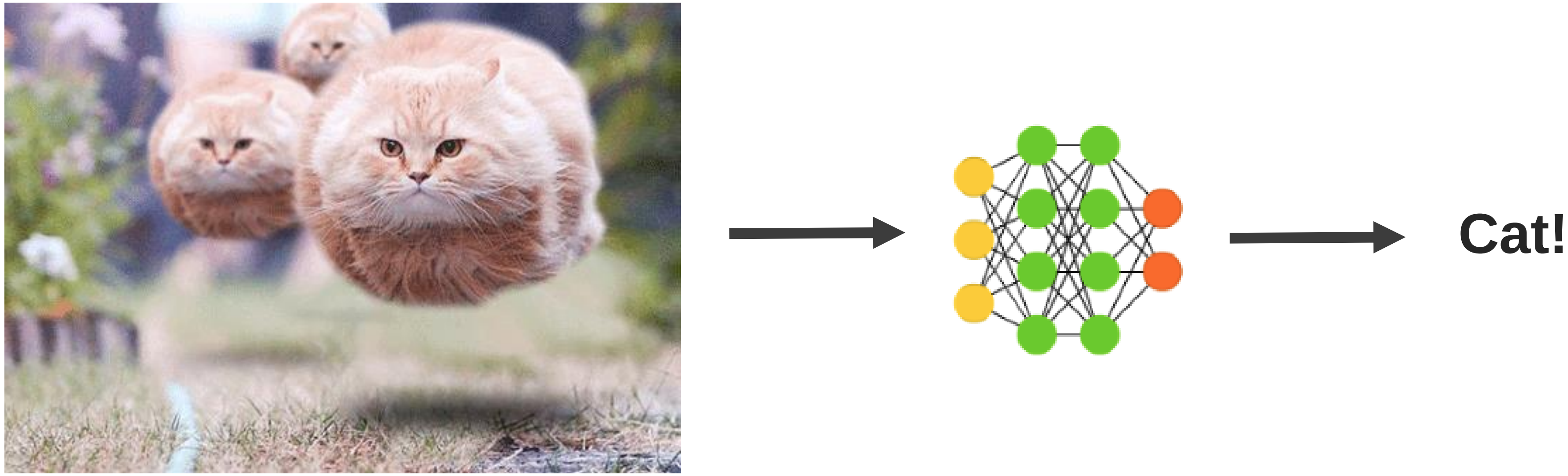
- How much effort is needed to implement a ML solution? Is appropriate infrastructure for data acquisition available? Enough resources to perform the training?

Universal Approximation Theorem
A simple neural network including only a single hidden layer can approximate any bounded continuous target function with arbitrary small error.

Does not say how big the effort could be…

- Define a **narrow task** (optimization of specific parameters rather than the entire machine)
- **Performance measure** of selected model (beam size, pulse energy, …)
- **Feature engineering** is highly important!

- Well structured data, extendable architecture of existing frameworks
  → possibility for the integration of ML tools.
- Extremely helpful e.g. when no analytical solution is available, for rapidly changing systems, when no direct measurements are possible.



Thank you for your attention!

Practical advice

- Often data preprocessing is needed before any model can be applied: rescaling, feature engineering, denoising, outlier elimination
  - data vizualisation can help
- Start with simple models - increase complexity only if needed
- Estimate model generalization (split into training, test and validation sets)

Frameworks to use:

- Prototyping, fast and easy implementation (very good documentation): http://scikit-learn.org/
- High-level package for Neural Networks: – https://keras.io/
- Deep Learning, specific complex model architectures:
  https://www.tensorflow.org/
  http://deeplearning.net/software/theano/
- Reinforcement Learning: OpenAI Gym https://gym.openai.com/