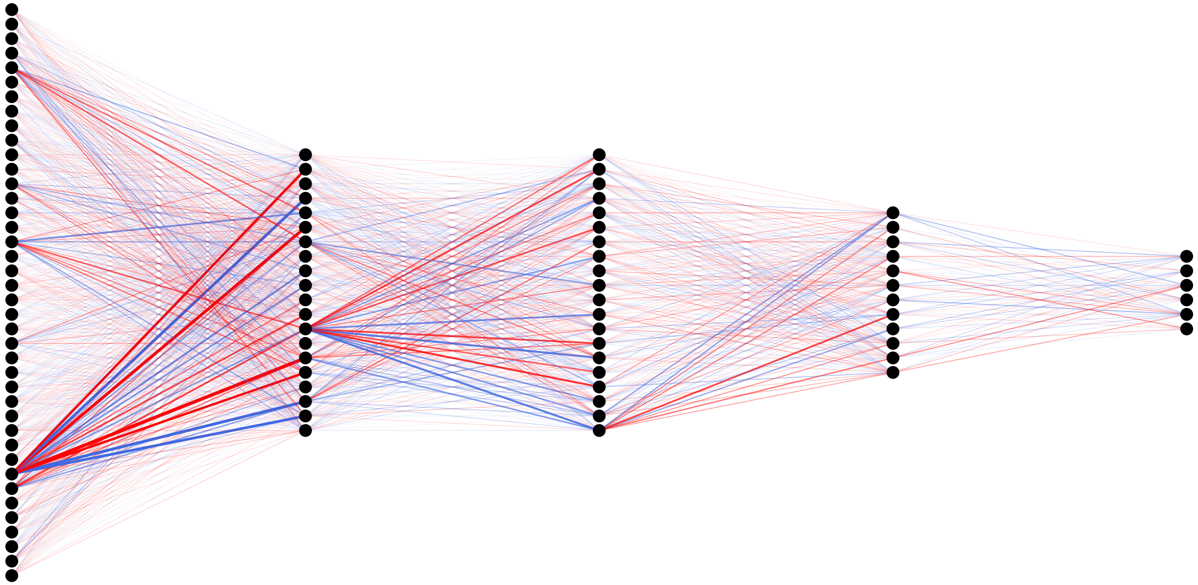
Analysis of Space Charge Dominated Beams using the Phase Advance Scan Technique and a Neural Network

PAHBB Workshop 2023



Frank Mayet
San Sebastian
22.06.2023

HELMHOLTZ

DESY.

Outline

Extra slides at the end of the presentation

- Introduction / Motivation
- Neural Networks – Short Primer
- Simulation Study
- Experimental Results
- Conclusion

More details in the manuscript


PHYSICAL REVIEW ACCELERATORS AND BEAMS **25,** 094601 (2022)

Predicting the transverse emittance of space charge dominated beams using the phase advance scan technique and a fully connected neural network

F. Mayet,* M. Hachmann, K. Floettmann, F. Burkart, H. Dinter,
W. Kuropka, T. Vinatier, and R. Assmann
Deutsches Elektronen-Synchrotron DESY, Notkestrase 85, 22607 Hamburg, Germany

(Received 24 May 2022; accepted 15 August 2022; published 6 September 2022)

The transverse emittance of a charged particle beam is an important figure of merit for many accelerator applications, such as ultrafast electron diffraction, free electron lasers, and the operation of new compact accelerator concepts in general. One of the easiest to implement methods to determine the transverse emittance is the phase advance scan method using a focusing element and a screen. This method has been shown to work well in the thermal regime. In the space charge dominated laminar flow regime, however, the scheme becomes difficult to apply because of the lack of a closed description of the beam envelope including space charge effects. Furthermore, certain mathematical, as well as beamline design criteria must be met in order to ensure accurate results. In this work, we show that it is possible to analyze phase advance scan data using a fully connected neural network (FCNN), even in setups, which do not meet these criteria. In a simulation study, we evaluate the performance of the FCNN by comparing it to a traditional fit routine based on the beam envelope equation. Subsequently, we use a pretrained FCNN to evaluate measured phase advance scan data, which ultimately yields much better agreement with numerical simulations. To tackle the confirmation bias problem, we employ additional mask-based emittance measurement techniques.

DOI: 10.1103/PhysRevAccelBeams.25.094601

I. INTRODUCTION

Many modern particle accelerators are tuned to achieve as small transverse beam emittance as possible. This is due to the fact that most users demand the highest beam brightness possible. Beam brightness is important for many accelerator applications, such as ultrafast electron diffraction [1], free-electron lasers [2], and the operation of new compact accelerator concepts in general (e.g., [3–6]). A common definition of brightness is [7]

$$B = \frac{\eta I}{\pi^2 \epsilon_x \epsilon_y}, \tag{1}$$

where η is a form factor close to unity, I is the beam peak current, and $\epsilon_{x,y}$ is the horizontal and vertical transverse emittance, respectively. Hence, in order to maximize B, transverse emittance has to be minimal.

There are multiple methods to characterize the transverse emittance. One of the most common techniques is the phase advance scan technique, where the transverse beam size is recorded on a screen vs the focusing strength of an upstream quadrupole or solenoid magnet [8–11]. The data can then be fitted based on the beam envelope equation. Alternatively, the beam images can be fed into tomography algorithms to reconstruct the transverse phase space, from which the emittance can be obtained [12,13]. Space charge effects can be included to some extent [12,14,15]. Instead of scanning the focusing strength of a magnet, also multiple screens can be used to record the beam size vs the phase advance. Other—potentially single-shot—methods involve the insertion of masks into the beamline, which then, subsequently, can be imaged on a downstream screen [16]. Coupled with advanced reconstruction algorithms, these methods are capable of delivering reconstruction of the core 4D phase space [17]. In this work, we concentrate on the phase advance scan technique, as this is the easiest one to implement, only requiring standard beamline components.

One of the limitations of the phase advance scan technique is that there is no closed description of the beam envelope for space charge dominated beams [14,15]. It is therefore difficult to apply the method in this regime [18]. Space charge dominated beams especially occur, for example, in the injector part of high-brightness electron sources, where the beam is still nonrelativistic. In order to quantify whether a beam is space charge dominated, the so-called *laminarity parameter* ρ can be calculated [19]. This parameter represents the ratio between the space charge term and the emittance term of the beam envelope equation. It is given by

*frank.mayet@desy.de

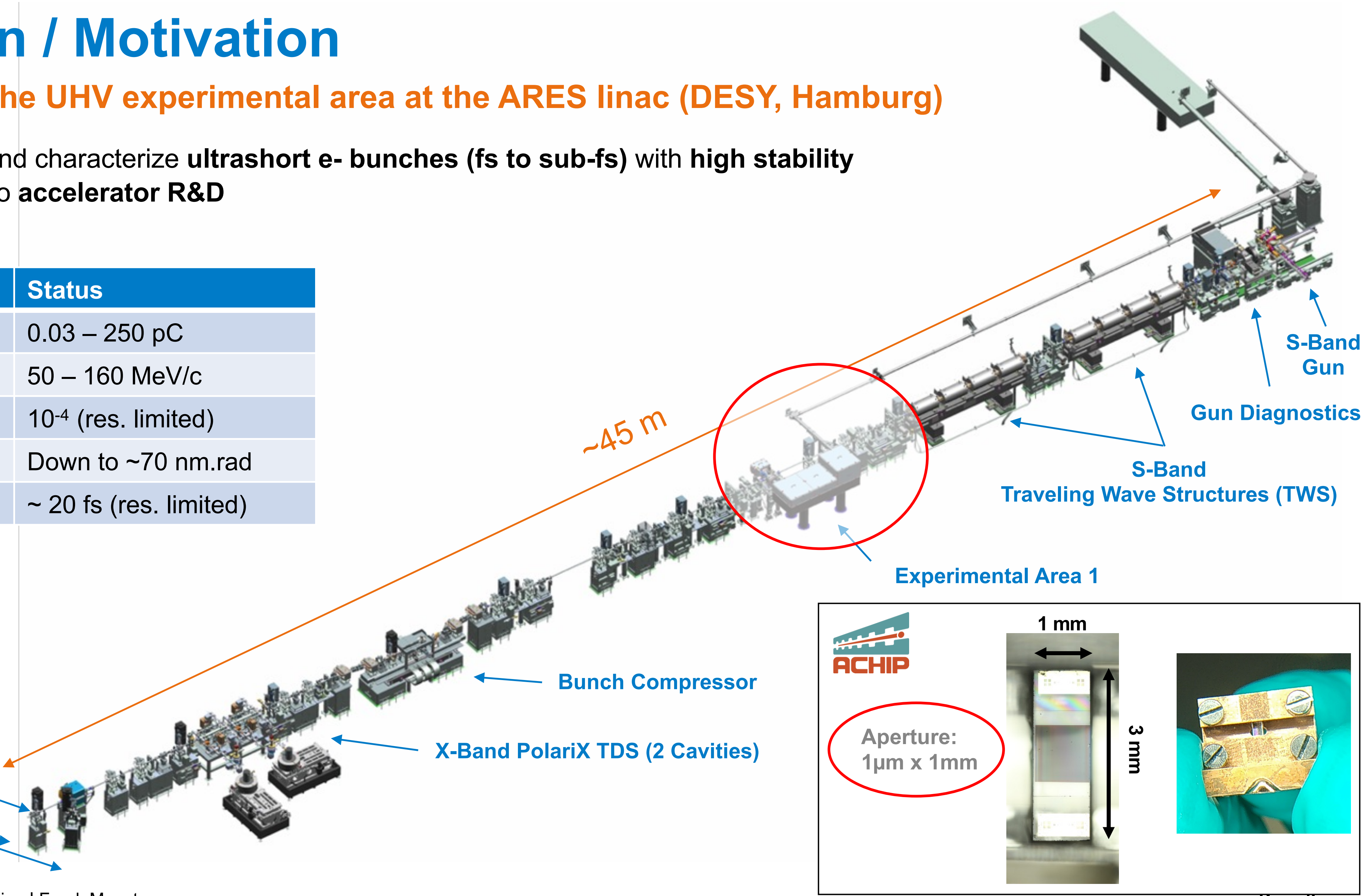
https://journals.aps.org/prab/pdf/10.1103/PhysRevAccelBeams.25.094601

Introduction / Motivation

My main project: The UHV experimental area at the ARES linac (DESY, Hamburg)

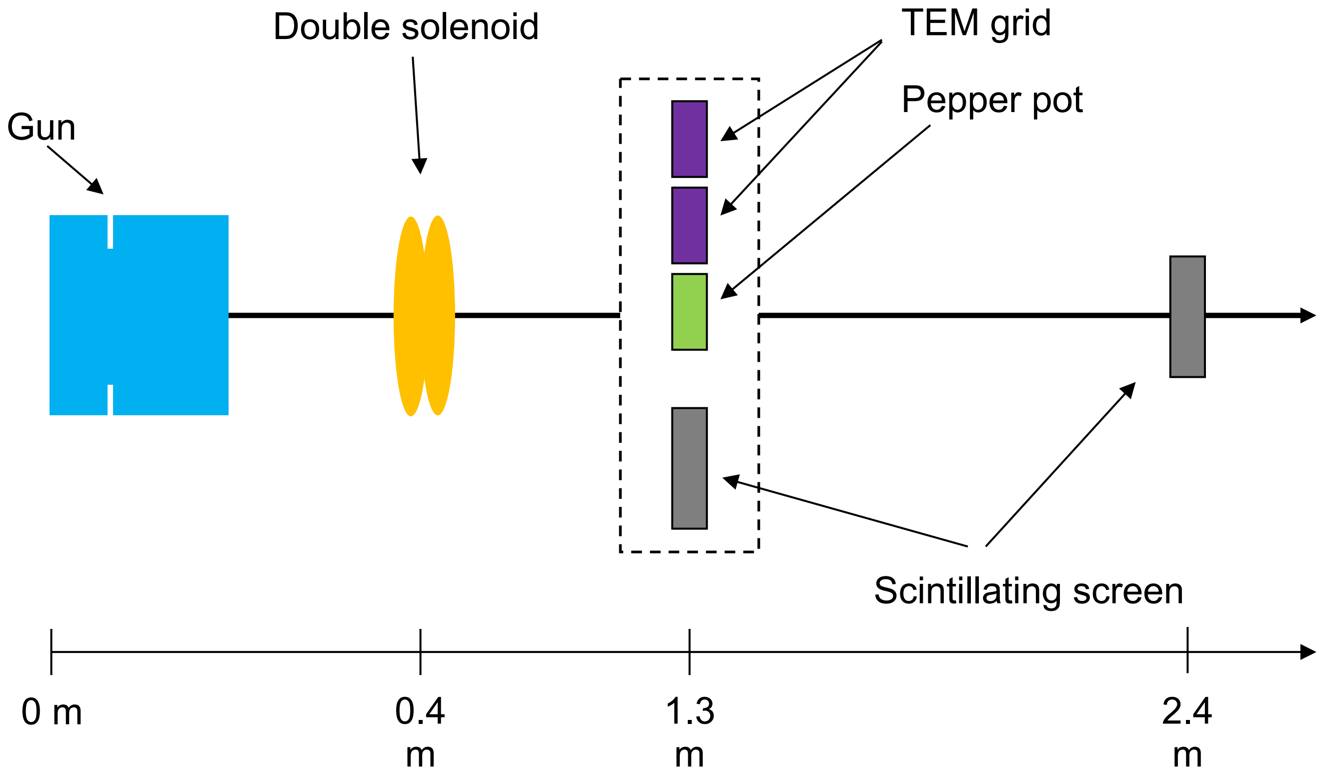
ARES goal: Generate and characterize **ultrashort e- bunches (fs to sub-fs)** with **high stability** for applications related to **accelerator R&D**

Parameter	Status
Charge	0.03 – 250 pC
Momentum	50 – 160 MeV/c
Momentum Spread	10^{-4} (res. limited)
Transv. Norm. Emittance	Down to ~70 nm.rad
Bunch Length	~ 20 fs (res. limited)

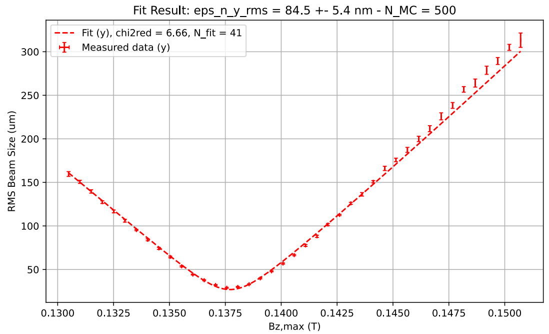


ARES Gun Region

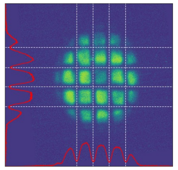
Options for an emittance measurement



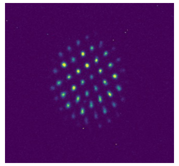
- Phase advance scan on either the first or second **screen** using the **double solenoid**



- Grid-based measurement on the second **screen**
    - Low charge: **TEM grid**



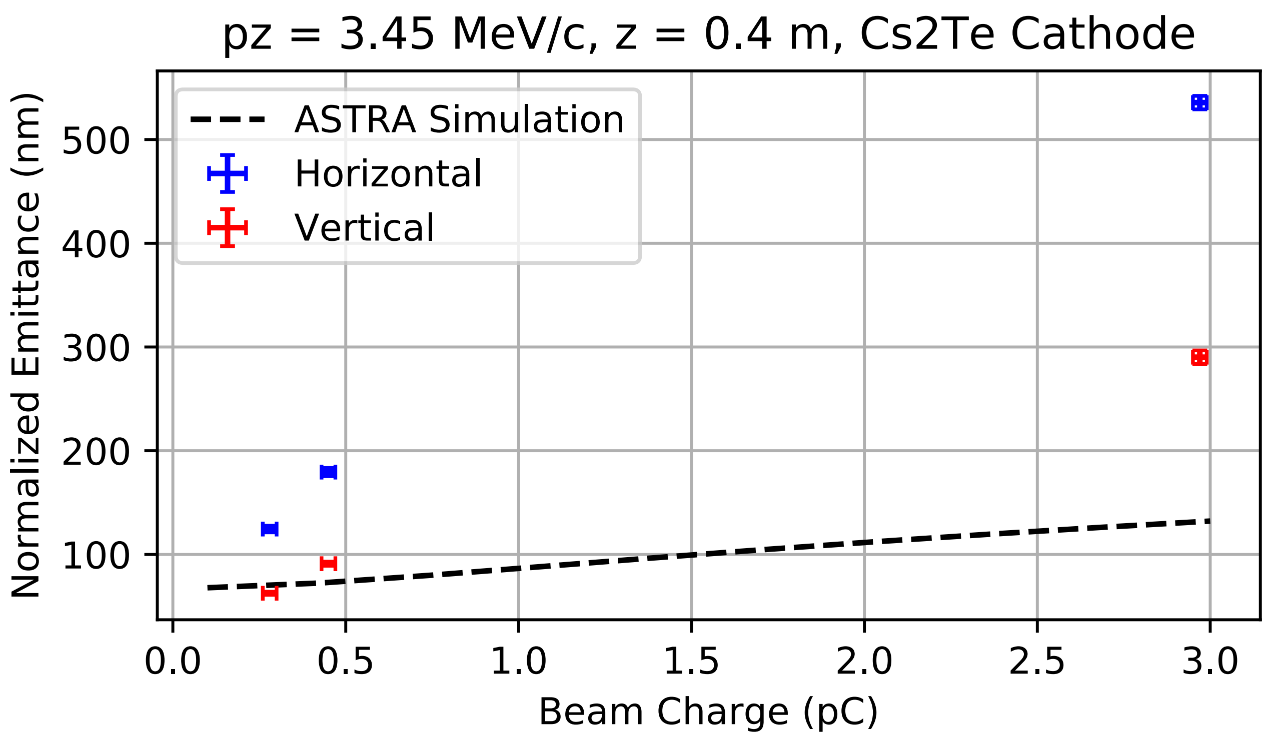
    - High charge: **Pepper pot**



Premise

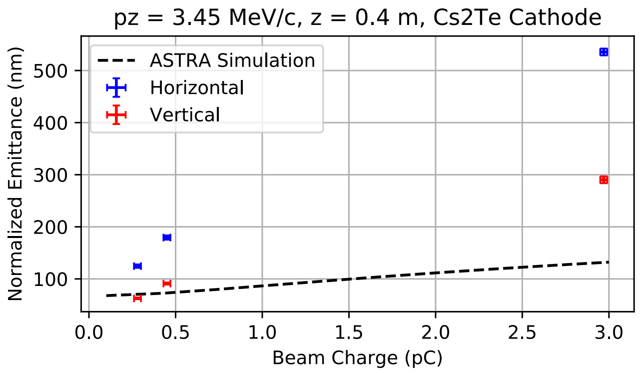
Why a new analysis method?

- Very first results, obtained using the phase advance scan technique and traditional data analysis
  → Turned out not to fit expectations; also any following measurement yielded generally too high values



Premise

What could be the reason?



- The fit method does not take space charge into account!
- Other possible reasons:
  - Numerical significance of the emittance could be too low
  - *For the horizontal plane: Beam diagnostics*
    → Screen image suffers from depth of field, because the angle between screen and camera is 45°

Laminarity Parameter

$$\rho = \frac{I\sigma^2}{2I_\mathrm{A}\gamma\varepsilon_n^2} \qquad I_\mathrm{A} \approx 17\,\mathrm{kA}$$

ρ >> 1 → Space charge dominated regime

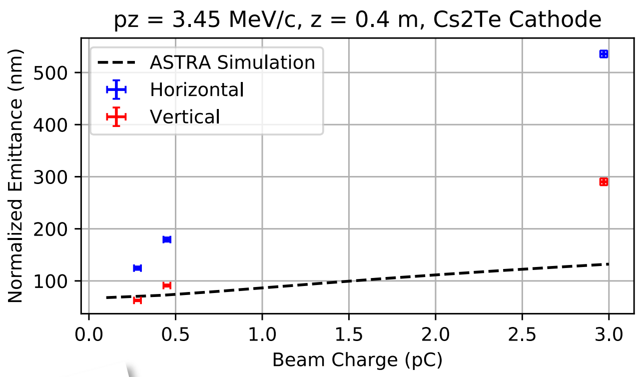
Fit Feasibility Criterion

$$\frac{\varepsilon_x^2}{\sigma_{x,0}^2 \cdot (\sigma_{x,0})'^2} \geq 0.01$$

$$\begin{aligned}\sigma_x^2 =& M_{11}^2\sigma_{x,0}^2 \\ &+ 2M_{11}M_{12}\sigma_{x,0}(\sigma_{x,0})' \\ &+ M_{12}^2\left(\frac{\varepsilon_x^2}{\sigma_{x,0}^2} + (\sigma_{x,0})'^2\right).\end{aligned}$$

Premise

What could be the reason?



- The fit method does not take space charge into account!
- Other possible reasons:
  - Numerical significance of the emittance could be too low
  - *For the horizontal plane: Beam diagnostics*
    → Screen image suffers from depth of field b
    between screen and camera is 45°

In order to not depend on these two criteria, can we maybe leverage machine learning to analyze the solenoid scan data instead?

$$\frac{}{2I_{\mathrm{A}}\gamma\varepsilon_n^2} \qquad I_{\mathrm{A}} \approx 17\,\mathrm{kA}$$

ρ >> 1 → Space charge dominated regime

Fit Feasibility Criterion

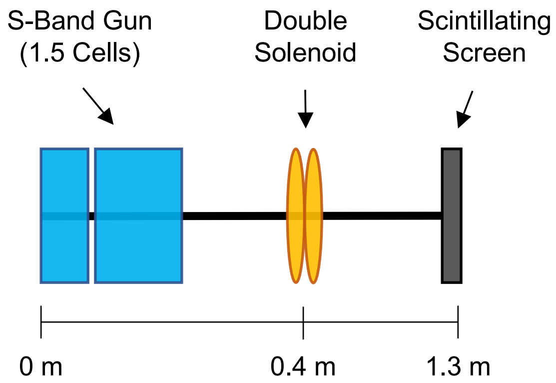
$$\frac{\varepsilon_x^2}{\sigma_{x,0}^2 \cdot (\sigma_{x,0})'^2} \geq 0.01$$

$$\begin{aligned}\sigma_x^2 =& M_{11}^2\sigma_{x,0}^2 \\ &+ 2M_{11}M_{12}\sigma_{x,0}(\sigma_{x,0})' \\ &+ M_{12}^2\left(\frac{\varepsilon_x^2}{\sigma_{x,0}^2} + (\sigma_{x,0})'^2\right).\end{aligned}$$

Simulation Study

Simulation Study

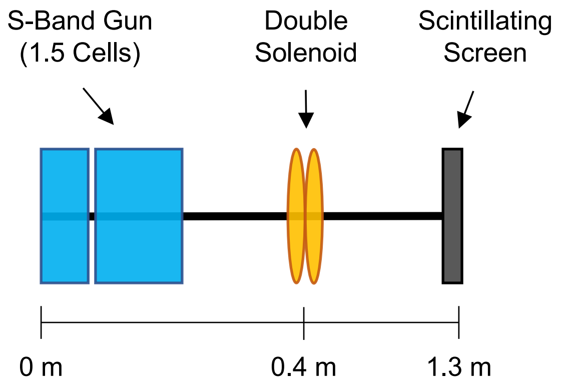
Phase Advance Scan Technique



- **What data do we record?**
  - RMS beam size vs. solenoid current
- **What do we want?**
  - Normalized emittance at the solenoid position

Simulation Study

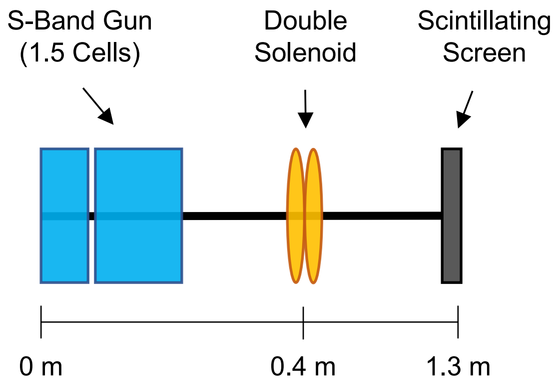
Phase Advance Scan Technique



- **What data do we record?**
  - RMS beam size vs. solenoid current
- **What do we want?**
  - Normalized emittance at the solenoid position
- **Potential training data must be constructed from two ASTRA simulations**
  - Tracking up to the solenoid position → Extract emittance
  - Beam size vs. solenoid field at the position of the first screen

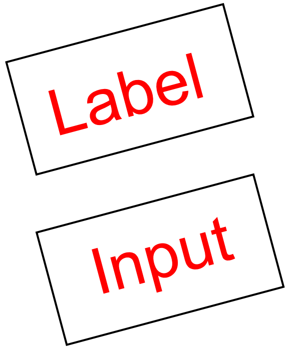
Simulation Study

Phase Advance Scan Technique



- **What data do we record?**
  - RMS beam size vs. solenoid current
- **What do we want?**
  - Normalized emittance at the solenoid position

- **Potential training data must be constructed from two ASTRA simulations**
  - Tracking up to the solenoid position → Extract emittance
  - Beam size vs. solenoid field at the position of the first screen



Simulation Study

Methodology

- Training data produced with parallelized ASTRA on the DESY HPC cluster (10k macro particles)

…determined via convergence study…

Data Set

Data

- *Beam size at the screen (M = 40)*
- *Solenoid field strength (M = 40)*

→ Interleaved combination (20 points centered around the minimum)
→ M_{in} = 20 + 20 = 40

Label

- ***Emittance** at the solenoid*
- ***Beam size** at the solenoid*
- ***Beam divergence** at the solenoid*
- ***Bunch length** at the solenoid*
- *ASTRA parameter **XYrms** (laser spot)*
- *ASTRA parameter **Trms** (emission time)*
- *ASTRA parameter **Qbunch** (charge)*

→ M_{out} = 7

→

Split Data Sets

- N_{tra} Training Sets
- N_{val} Validation Sets
- N_{tes} Test Sets

N_{tra}
N_{val}
N_{tes}

Parameter	Value
Bunch charge	[0.01, 2.1] pC
Laser spot size (flat top diameter)	[240, 400] µm
Cathode emission time (rms)	[60, 100] fs

Laser input parameters according to these ranges – uniformly distributed

N = 16,066
N_{tra} = 0.6 * N (*Training*)
N_{val} = 0.2 * N (*Performance validation during the training*)
N_{tes} = 0.2 * N (*Performance validation after the training*)

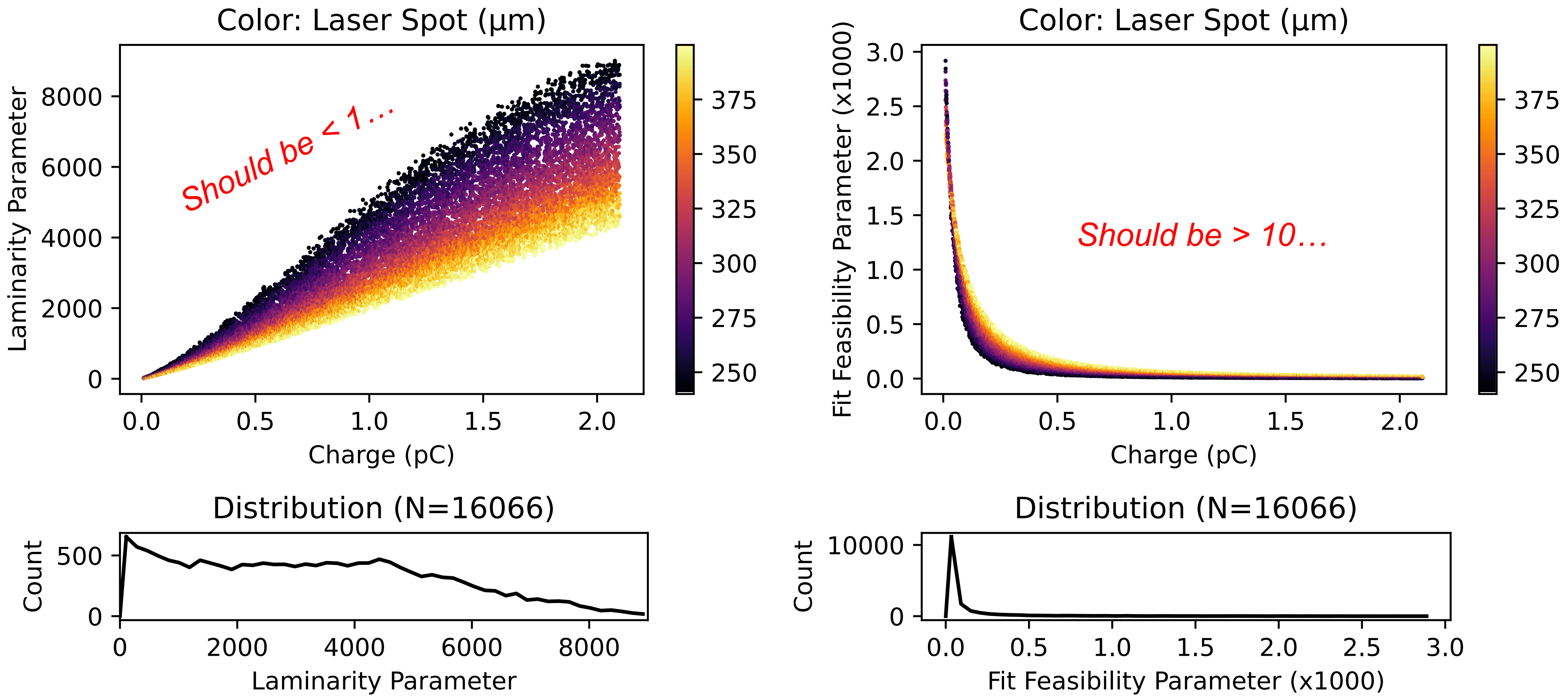
NN-Layout (3 hidden layers):

M_{in} – | M_{in} – $M_{in}/2$ – $M_{in}/2$ | – M_{out}

Fully connected with ReLU activation

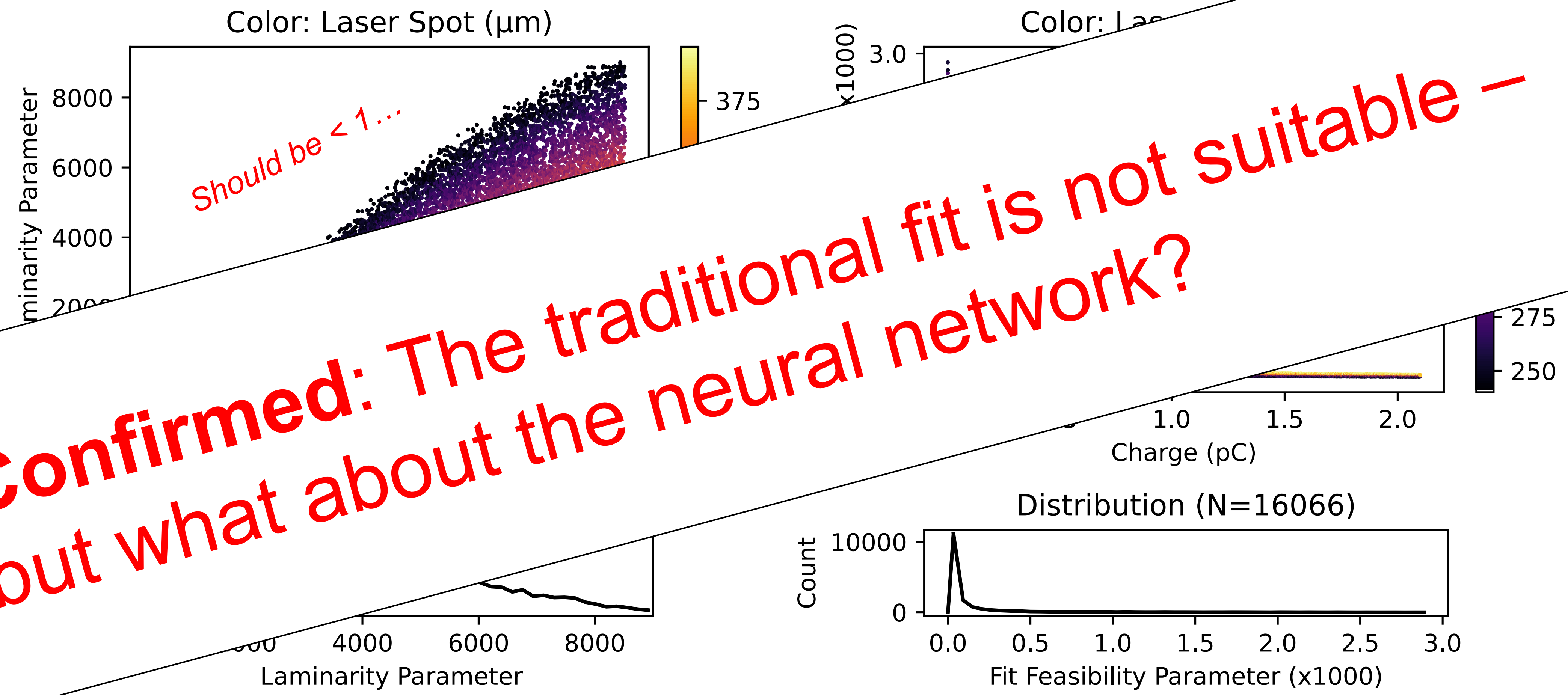
Simulation Study

Data – Let’s check the fit feasibility criteria…



Simulation Study

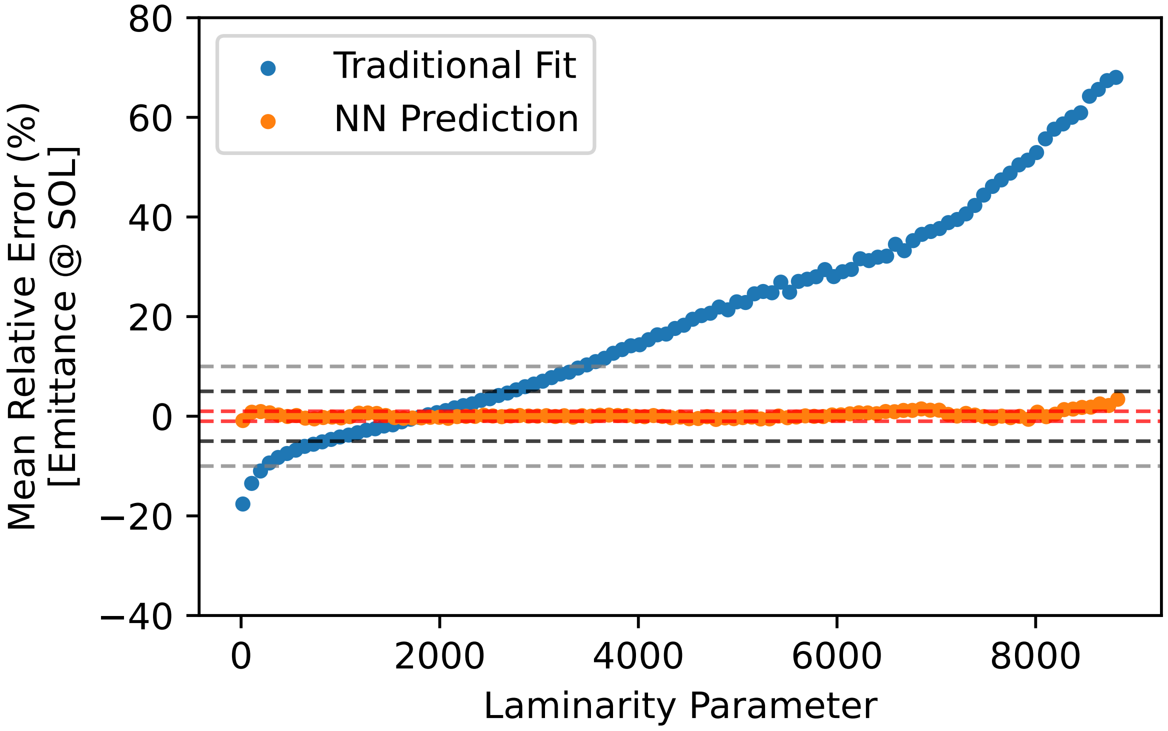
Data – Let's check the fit feasibility criteria…

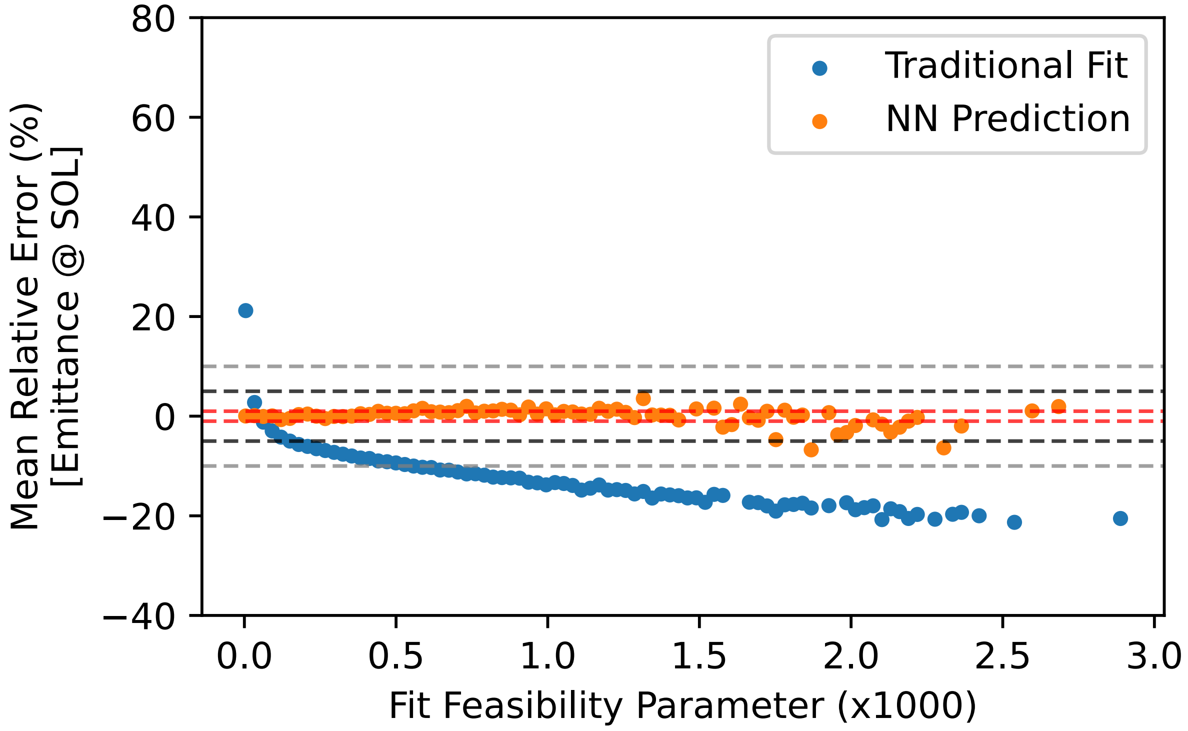


Confirmed: The traditional fit is not suitable – but what about the neural network?

Simulation Study

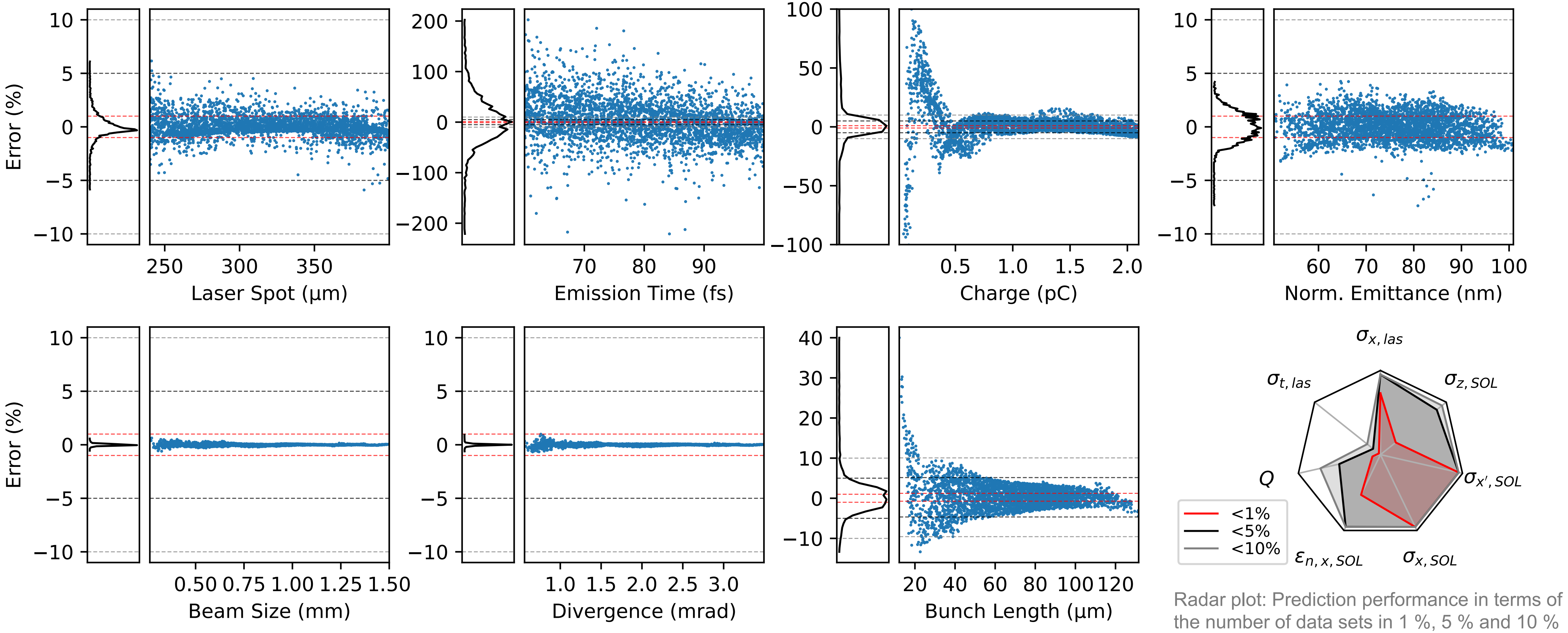
Results – Test data sets





Simulation Study

Results – Test data sets – Not only transverse emittance!



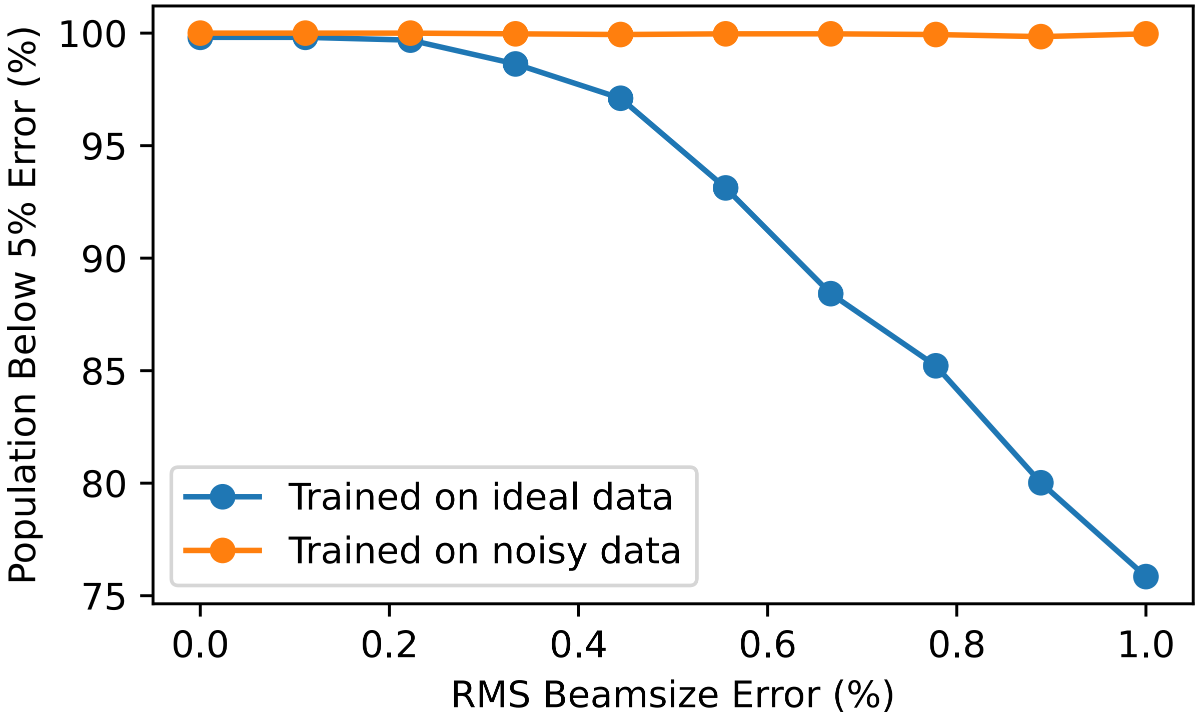
Radar plot: Prediction performance in terms of the number of data sets in 1 %, 5 % and 10 % relative error intervals

…detailed discussion and table in the paper…

Experimental Results

Experimental Results

Extension to measured data

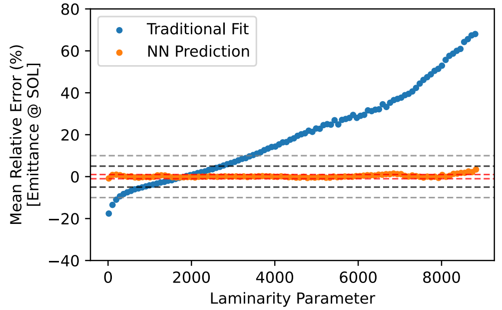


- Data is not perfect in reality
  - Uncertainty on beam sizes / B-fields
  - Scan range
  - Scan point spacing
  - Absolute value of the B-fields
- Trained on noisy data
  - From each data set:
    - 100 noisy sets with relative errors
    - 100 noisy sets with abs. errors
    - With and without beam size resolution limit
  - Based on typical ARES values
  - **→ 2(2*100 + 1)*N = 6,458,532 sets**

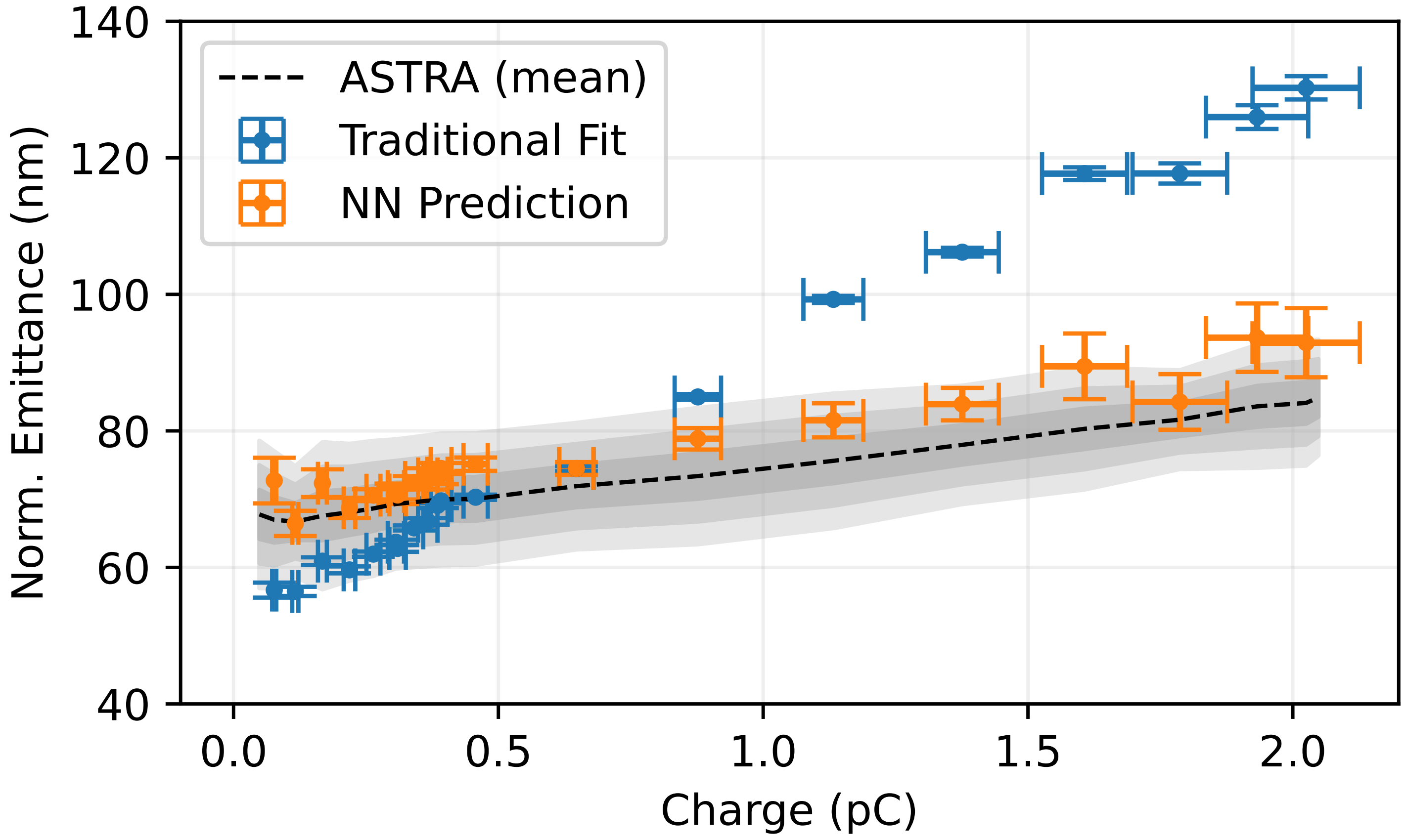
Experimental Results

Measurements performed at ARES

Recall…



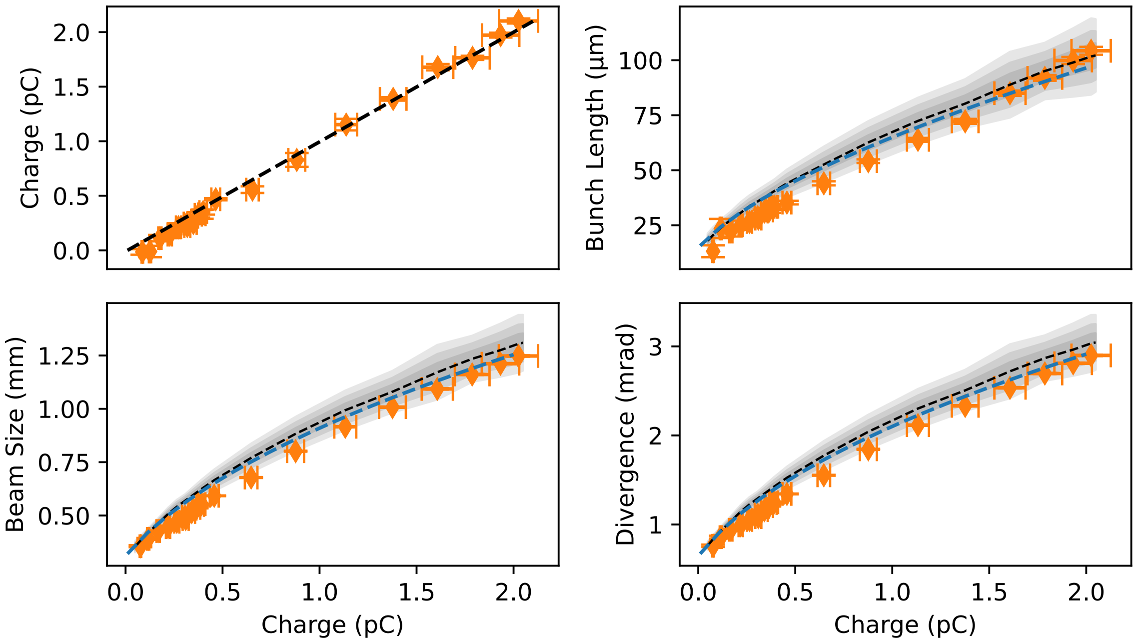
The FCNN results are much closer to the expected values than the results obtained from the traditional fit!



Experimental Results

Measurements performed at ARES – Other predicted parameters

Charge dependent parameters



Fixed machine parameters

Parameter	Prediction	Experiment
Laser Spot Size (µm)	338.5 ± 0.5	320 ± 30
Laser Pulse Length (fs, rms)	87.34 ± 0.04	76 ± 8
Solenoid Field – Start (mT)	130.516 ± 0.001	130.5 ± 0.1
Solenoid Field – End (mT)	150.783 ± 0.001	150.7 ± 0.1

We actually suspect a larger laser spot, because we can extract more charge at the moment than should be possible

Blue dashed line: Simulation with the predicted laser parameters

Conclusion

- **A pre-trained FCNN can be used to predict the transverse emittance from phase advance scan data, even in setups where the traditional fit method does not work anymore**
- The FCNN was adapted to real-world data and used to analyze measurements conducted at ARES
    - During the same measurement run, mask-based emittance measurements showed that the machine can indeed be described with ASTRA (→ confirmation bias)
- **Measurement results obtained using the fit method show wrong results, with the expected charge dependence. This dependence was discovered from the large number of produced data sets (~16k)**
- **The pre-trained FCNN also predicts other fixed and charge dependent beam parameters to varying accuracy.**

Conclusion

- **A pre-trained FCNN can be used to predict the transverse emittance from phase advance scan data, even in setups where the traditional fit method does not work anymore**
- The FCNN was adapted to real-world data and used to analyze measurements conducted at ARES
  - During the same measurement run, mask-based emittance measurements showed that the machine can indeed be described with ASTRA (→ confirmation bias)
- **Measurement results obtained using the fit method show wrong results, with the expected charge dependence. This dependence was discovered from the large number of produced data sets (~16k)**
- **The pre-trained FCNN also predicts other fixed and charge dependent beam parameters to varying accuracy.**

Thank You!

Acknowledgements

The ARES team @ DESY: F. Burkart, H. Dinter, W. Kuropka, T. Vinatier, and R. Aßmann

The REGAE team @ DESY: M. Hachmann, and K. Flöttmann

DESY beam diagnostics (MDI), DESY laser operations (FS-LA), DESY HPC (Maxwell)

Contact

Deutsches Elektronen-
Synchrotron DESY

www.desy.de

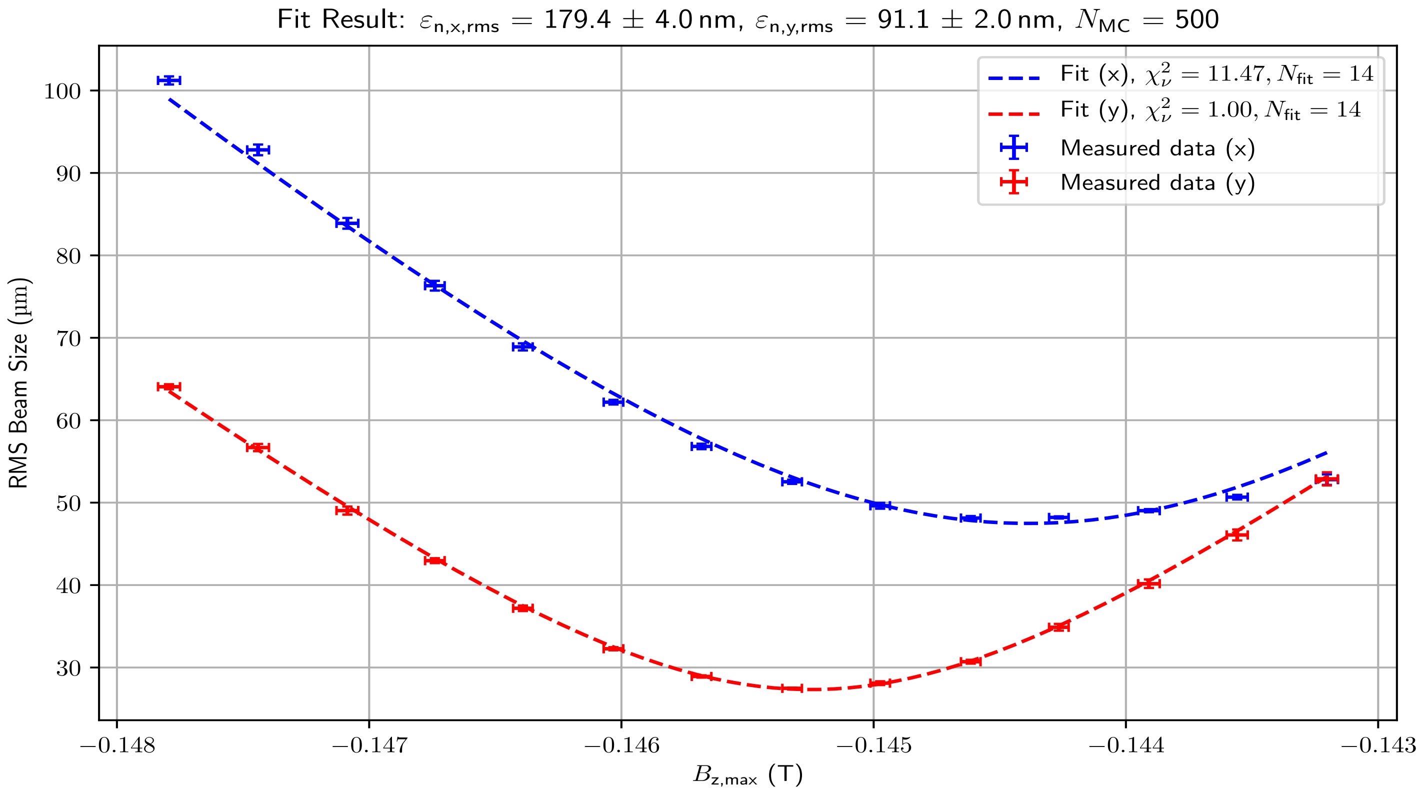
Frank Mayet
Department: MPY1
E-mail: frank.mayet@desy.de
Phone: +49 40 8998 3851

Supplementary Information

Methods

Phase advance scan

- Retrieve the emittance, based on the beam envelope equation
  - Find scannable transport matrix elements M_{12} and M_{11}
    - Drift → Multiple screens
    - Focusing element → Adjustable focusing strength



How to get the emittance?
Fit the envelope equation!
Ingredients below

$$\sigma_x^2 = M_{11}^2 \sigma_{x,0}^2 + 2M_{11}M_{12}\sigma_{x,0}(\sigma_{x,0})' + M_{12}^2\left(\frac{\varepsilon_x^2}{\sigma_{x,0}^2} + (\sigma_{x,0})'^2\right).$$

Double solenoid in thin lens approximation

$$\mathbf{M}_{\mathrm{DS}} = \mathbf{M}_{\mathrm{TL}} \cdot \mathbf{M}_{\mathrm{D}} \cdot \mathbf{M}_{\mathrm{TL}} = \begin{pmatrix} 1 & 0 \\ -\frac{1}{f} & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & l_{\mathrm{D}} \\ 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 \\ -\frac{1}{f} & 1 \end{pmatrix} = \begin{pmatrix} 1 - l_{\mathrm{D}}/f & l_{\mathrm{D}} \\ (l_{\mathrm{D}} - 2f)/f^2 & 1 - l_{\mathrm{D}}/f \end{pmatrix},$$

$$f(B_{z,\max}) = \left[\left(\frac{q}{2\langle p_z\rangle}\right)^2 F_2\right]^{-1}$$

Methods

Phase advance scan *(Emittance at the focusing element)*

Pros

- Measurement is easy to perform
- Tools available on many beamlines *per default*
- Data analysis is in principle easy

Cons

- It's a scan-based measurement and thus
  - ...takes time
  - ...is inherently multi-shot

Mathematical criterion originating from the envelope equation → Relative significance of the emittance! Is it numerically buried?

$$\frac{\varepsilon_x^2}{\sigma_{x,0}^2 \cdot (\sigma_{x,0})'^2} \geq 0.01$$

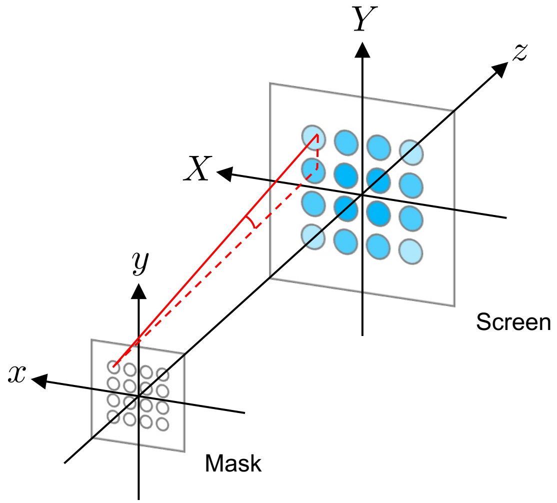
- Limitations
  - Feasibility considerations (see e.g. Max Hachmann's diploma thesis)
    - Constraints on beam size and divergence at the focusing element
    - Constraints on the setup (distance between focusing element and screen; available focusing strength)
  - Numerical problems (e.g. sine oscillation of the fit result, based on the position of the minimum w.r.t. the sampling)
  - Space charge not included! → May lead to overestimation of the emittance *(can be tackled somewhat → Add perveance term)*
  - Space charge and non-linearities during the scan! (Emittance changes during the scan, etc.)

All positions are in relation to the centroid of the distribution

Methods

Grid-based measurement

- Retrieve the emittance, by mapping the x-x' phase space using a mask
  - Scan either a slit (multi-shot), or use dense slit mask (single shot)
  - Both planes can be measured at the same time, by using a hole pattern
- Try to sample the phase space with as many beamlets as possible to determine the overall emittance as good as possible – *Otherwise*: Emittance in a subspace!
- It is also possible to analyze shadow images of grid bars instead of beamlets
  - Beamlets: Characterize beamlets directly, by measuring their size
  - Grid bars: Determine beamlet properties, by fitting two error functions to the shadow
- The resolution depends on both the feature size of the mask and the detection system
- If the features of the mask are small enough, space charge during transport to the screen is not an issue



Each analyzed beamlet adds a spot in the x-x' phase space – the weighted area in phase space is then proportional to the emittance at the mask
→ Not analyzing all beamlets is like collimating the beam and the emittance will be underestimated
→ Ideally the mask is scanned to map even more details (but then: scan…)

How to get the emittance?
Calculate moments!
Ingredients below

Small angle approximation!

Beamlet pos. — Rms divergence of the beamlet — Beamlet size

$$\overline{x}'_{ij} = \frac{\overline{X}_{ij} - x_{ij}}{L} \qquad \sigma_{x'_{ij}} = \frac{\sigma_{X_{ij}}}{L}$$

Mean local angular deviation — Distance between mask and screen

Position of the hole in mask plane

$$\langle x^2 \rangle = \frac{\sum_{ij} I_{ij} x_{ij}^2}{\sum_{ij} I_{ij}} \qquad \langle x x' \rangle = \frac{\sum_{ij} I_{ij} x_{ij} \overline{x}'_{ij}}{\sum_{ij} I_{ij}}$$

Intensity of the beamlet (→ weight / fraction of the overall charge)

$$\langle x'^2 \rangle = \frac{\sum_{ij} I_{ij} \left(\overline{x}'^2_{ij} + \sigma^2_{x'_{ij}} \right)}{\sum_{ij} I_{ij}}$$

From these moments, the emittance can be readily obtained
(correlation terms can also be calculated, but are omitted for brevity)

Methods

Grid-based measurement *(Emittance at the mask position)*

Pros

- Single shot!
- Space charge dominated beams can be measured, as the beamlets are not themselves SC dominated

Cons

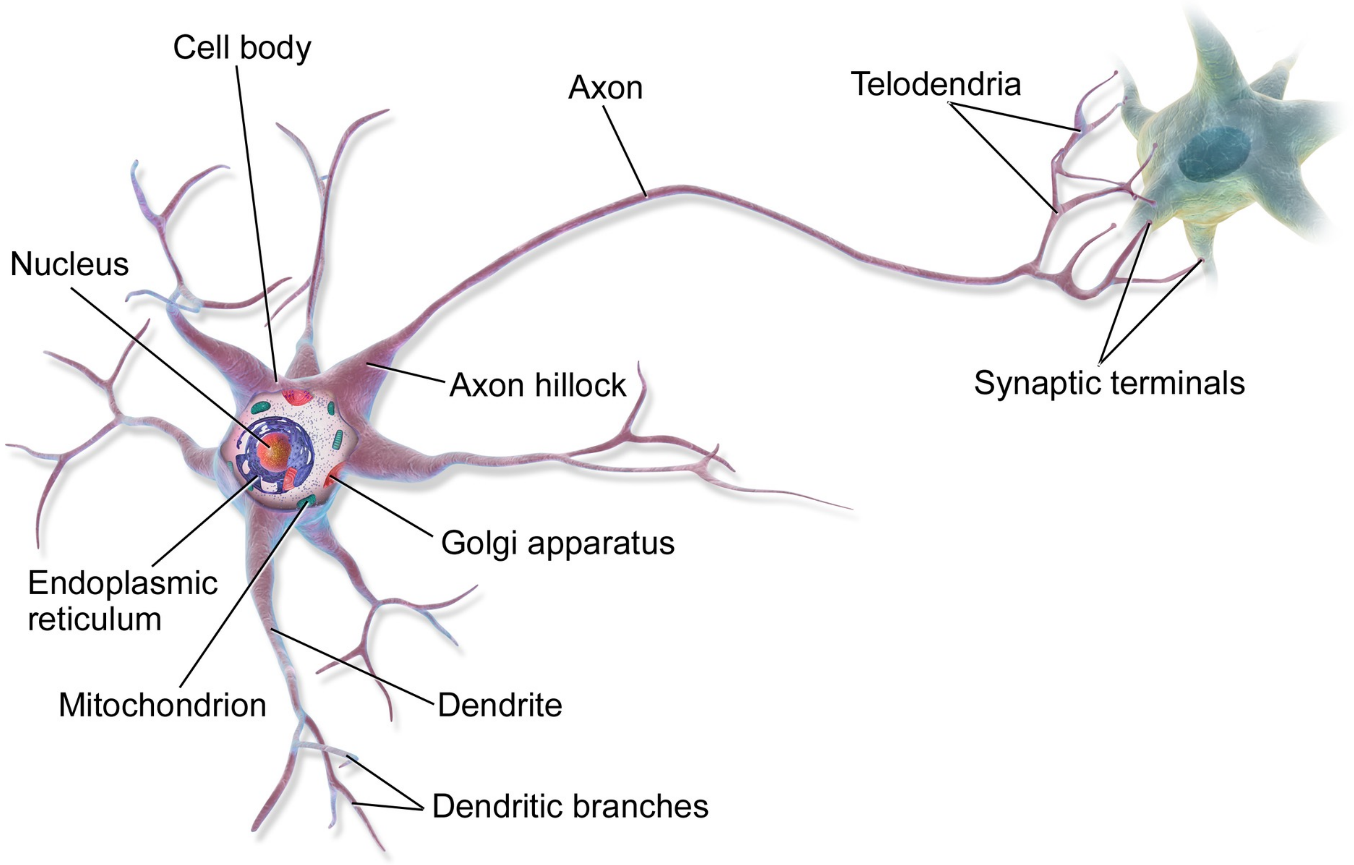
- Needs small mask features and high-res diagnostics
- Is a dedicated setup
- Might not measure *the beam you actually want to use (see below)*
- Data analysis is surprisingly tedious

- Limitations
  - If the ratio between feature size and thickness of the mask don't match, an angle cut can underestimate the emittance
  - The mask might be difficult to manufacture (TEM grids are readily available commercially though)
  - Distance to the screen determines the max. emittance that can be measured → Beamlets overlap at some point
  - Screen resolution determines the minimum emittance that can be measured via beam size measurement
  - Space charge can still be an issue if the features are too large
  - Space charge and non-linearities are still an issue in the sense that the beam needs to be focused before the grid (need diverging beam!). → This can lead to emittance growth between the focusing element and the grid
    → Measurement still *correct*, but the beam might not be the same compared to the nominal working point!

Neural Networks

A short primer

- Artificial neuron

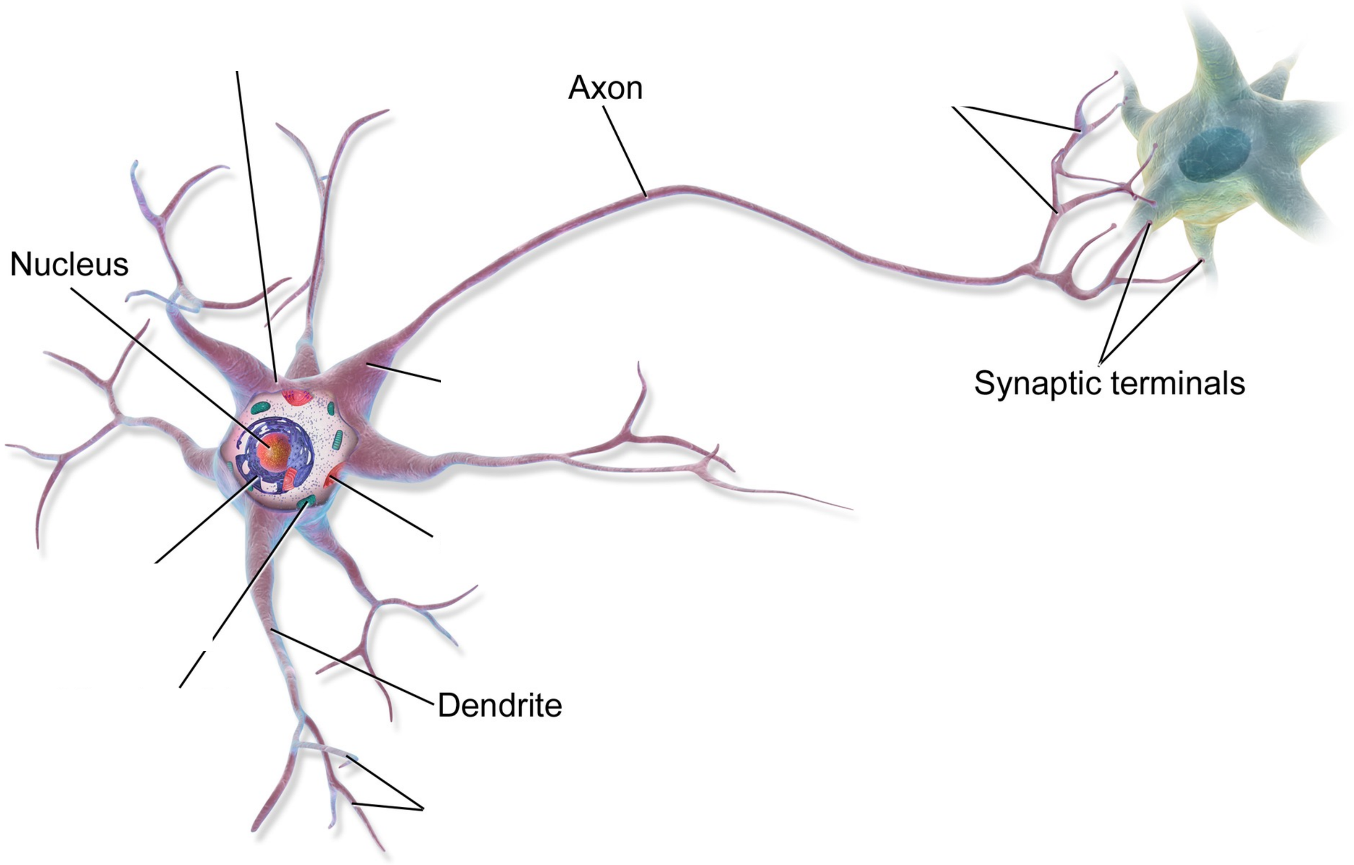


https://en.wikipedia.org/wiki/Neuron

Neural Networks

A short primer

- Artificial neuron

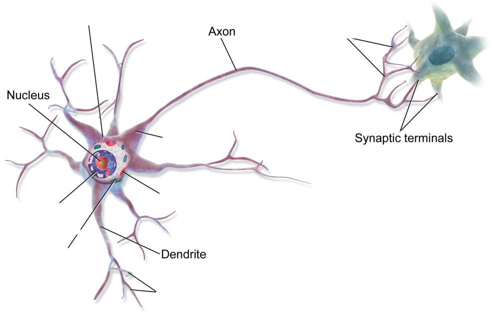


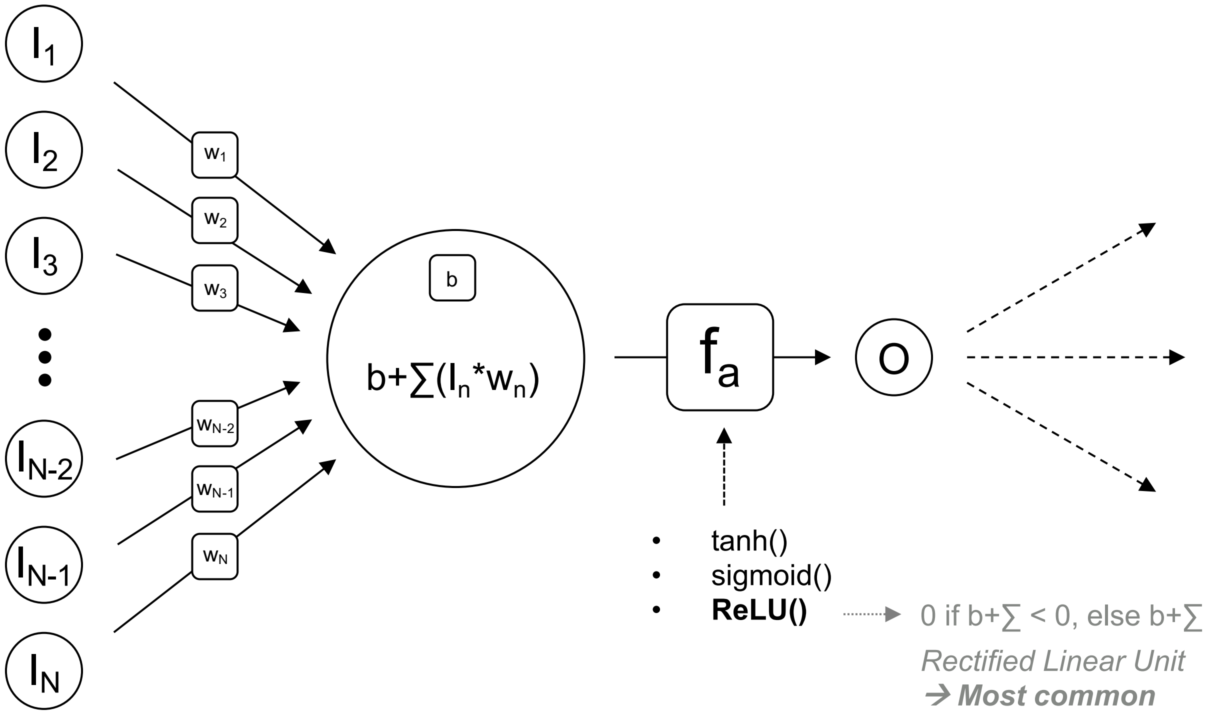
https://en.wikipedia.org/wiki/Neuron

Neural Networks

A short primer

- Artificial neuron / Perceptron

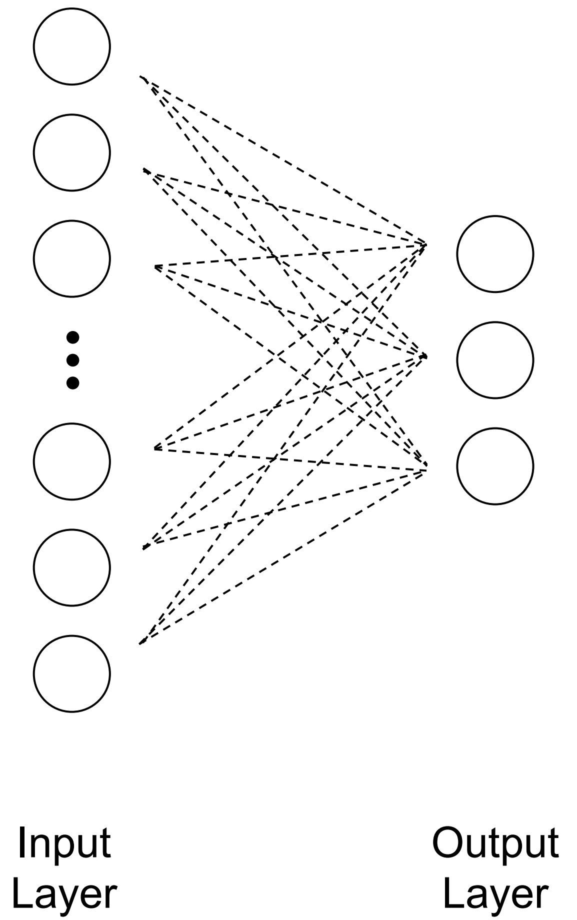




Artifical	Input	Weights	Biased Node	Activation	Output
Biological		Dendrites	Nucleus	Axon	Synaptic Terminals

Neural Networks
A short primer

- Most basic neural network
  - Can only classify *linearly separable* data sets
    - *The weights can only appear to 1st order*

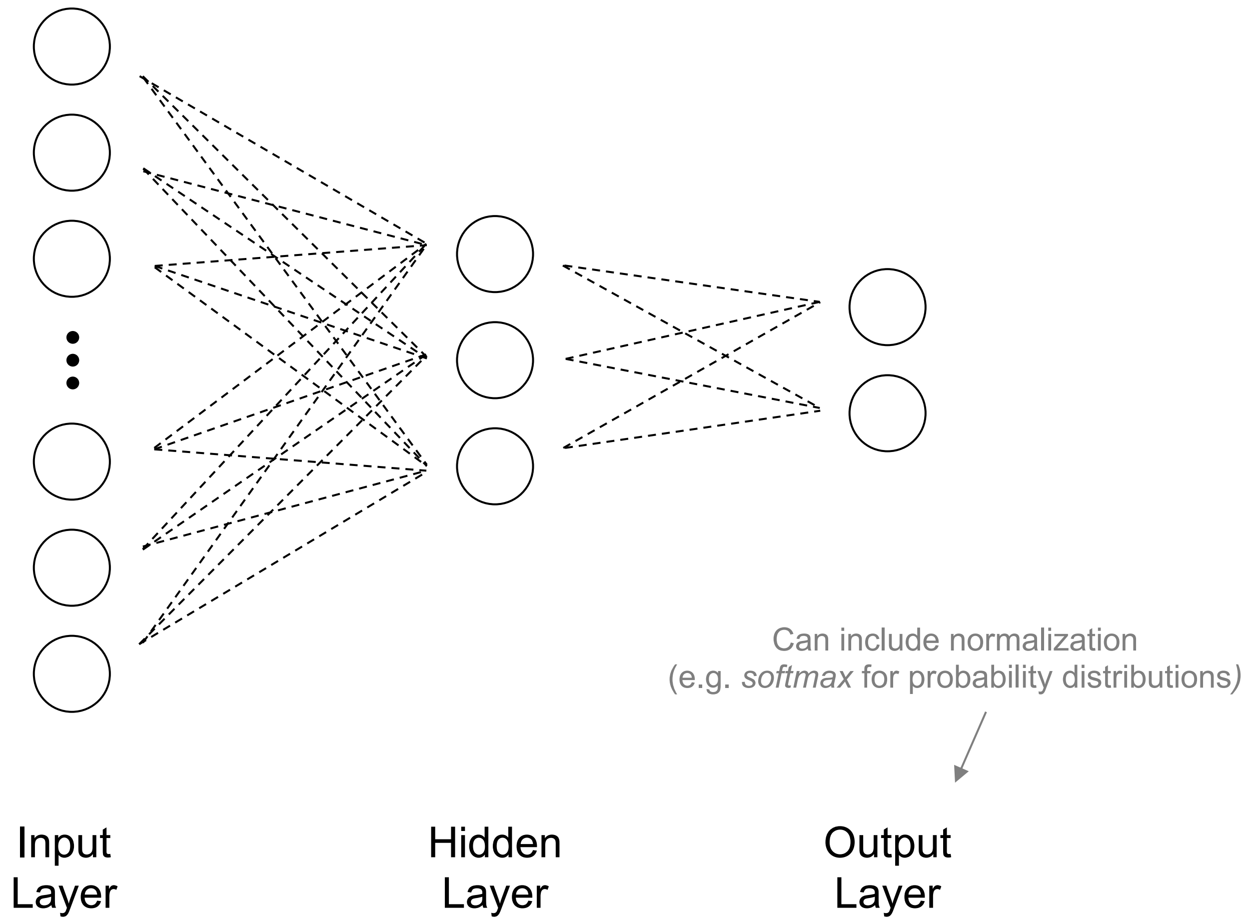


Fully connected, or densely coupled

Neural Networks

A short primer

- Simple general purpose neural network
  - Can have an arbitrary number of hidden layers of arbitrary width (i.e. number of neurons)
  - Networks with more then one hidden layer
    → Deep neural networks
  - In general: 1 IL + N HL + 1 OL
- *A single sufficiently large hidden layer is adequate for approximation of most functions*
  - *Why more than one hidden layer then?*
    → The problem is: What is 'sufficiently large'?
- …There are many variations of this, for example Convolutional Neural Networks (CNN), which reduce the number of connections, which can speed up the process for large input widths

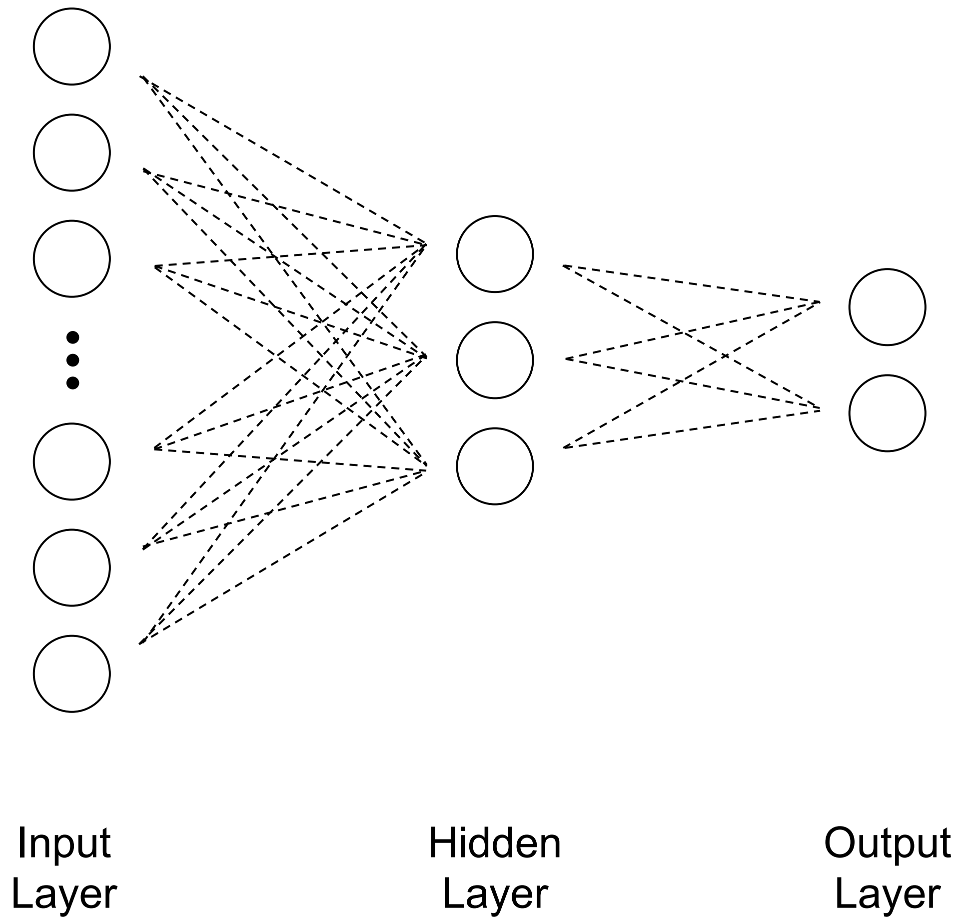


D. Stathakis (2009) How many hidden layers and nodes?, International Journal of Remote Sensing, 30:8, 2133-2147

Neural Networks

A short primer

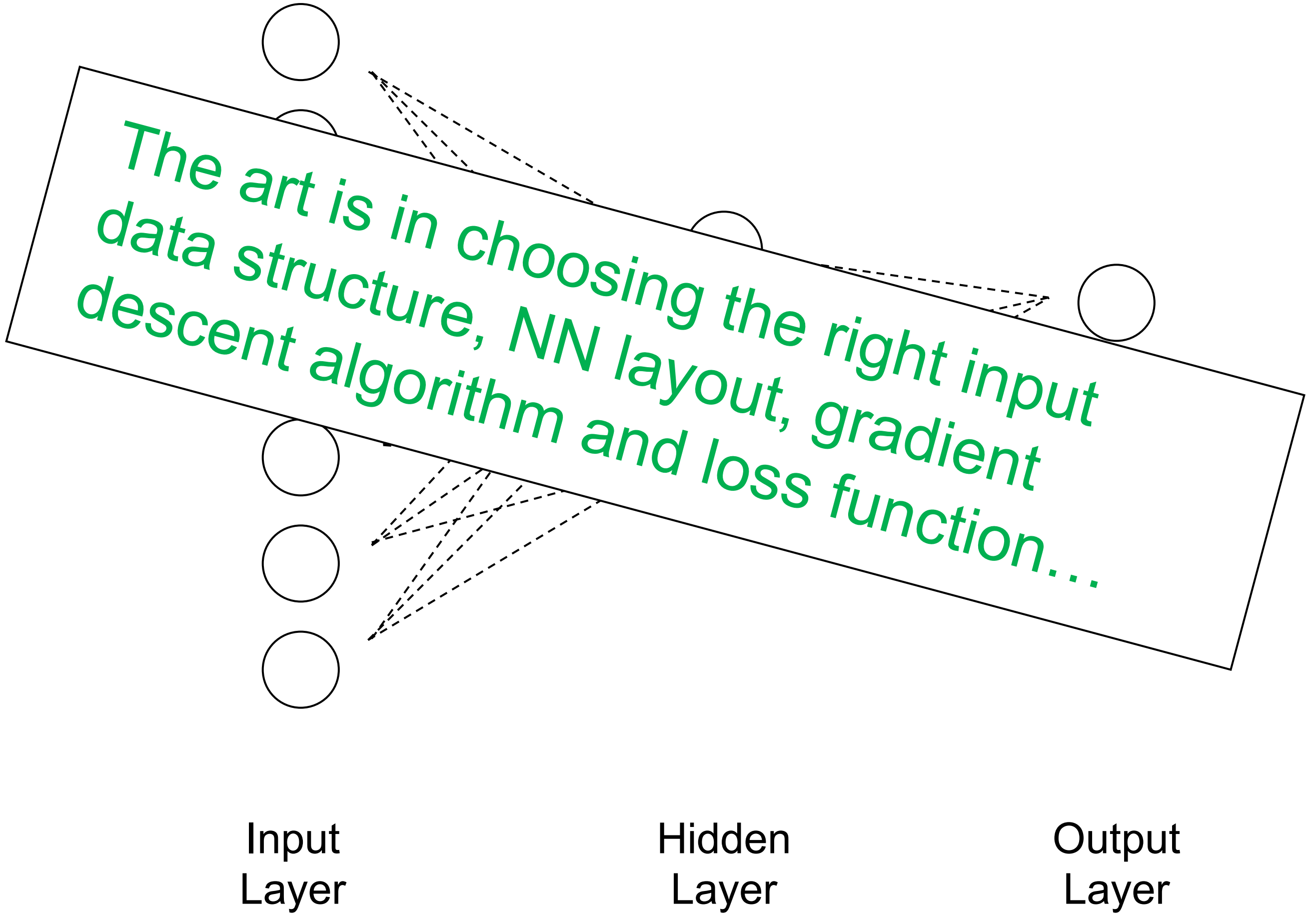
- How to train the network?
  - *Most basic: Supervised learning*
    - *Labeled data sets*
    - *Feedforward + Backpropagation*
      - *Gradient descent*



Neural Networks

A short primer

- How to train the network?
  - *Most basic: Supervised learning*
    - *Labeled data sets*
    - *Feedforward + Backpropagation*
      - *Gradient descent*

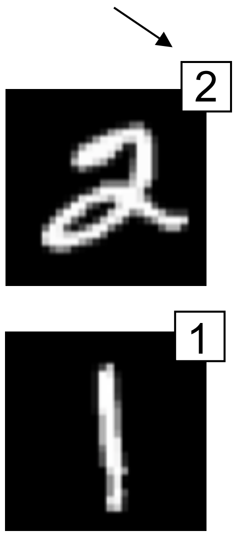


Neural Networks

A short primer

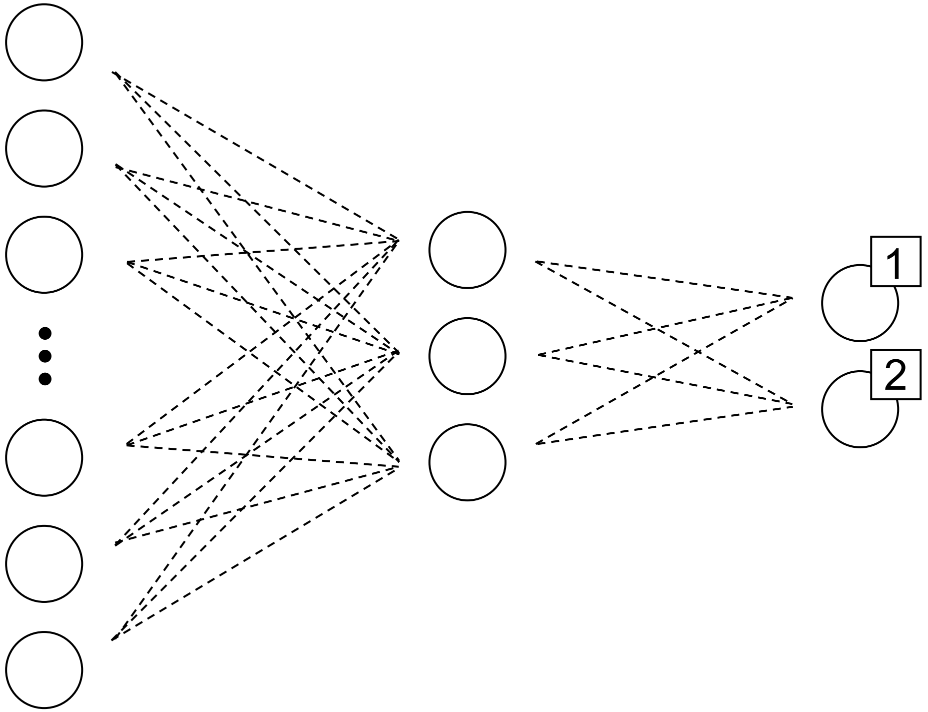
Classifier → Handwritten Digits

Label (we know the correct result)



Examples from the MNIST Dataset

Input
→ 1D representation of the handwritten digit
(2D matrix to 1D vector)



Output → Probability whether it is a 1 or a 2

Input Layer

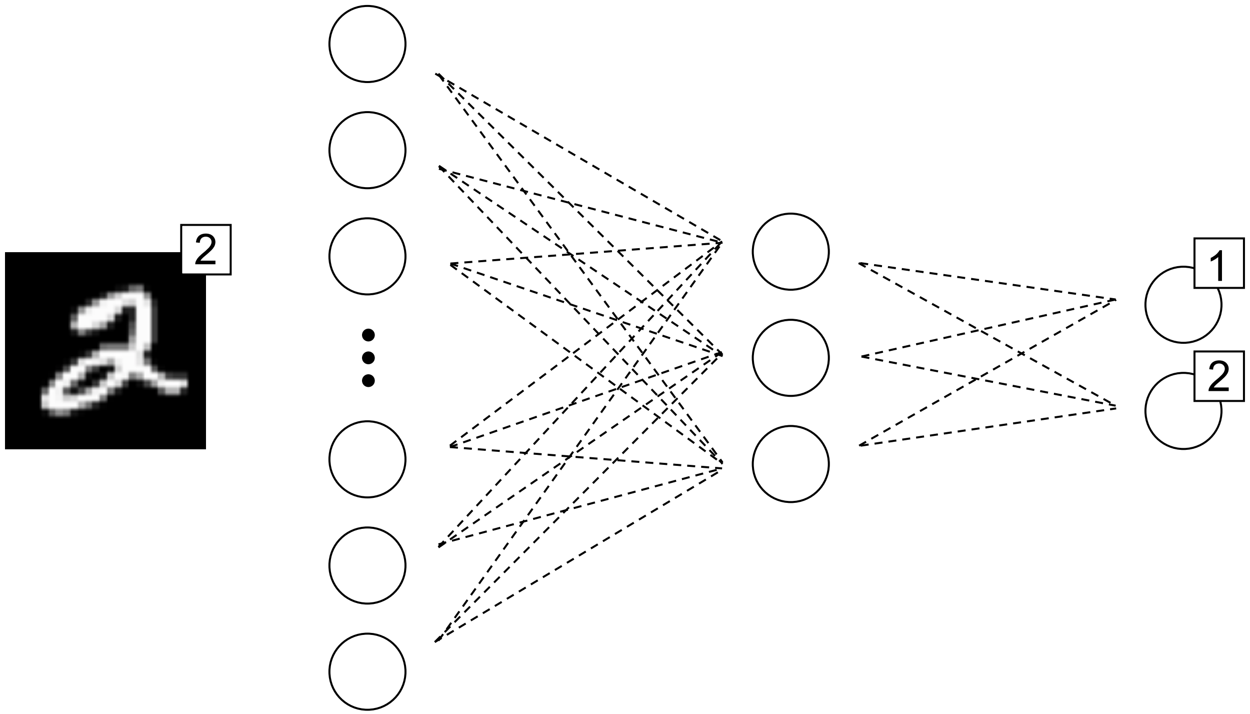
Hidden Layer

Output Layer

Neural Networks

A short primer

Classifier → Handwritten Digits



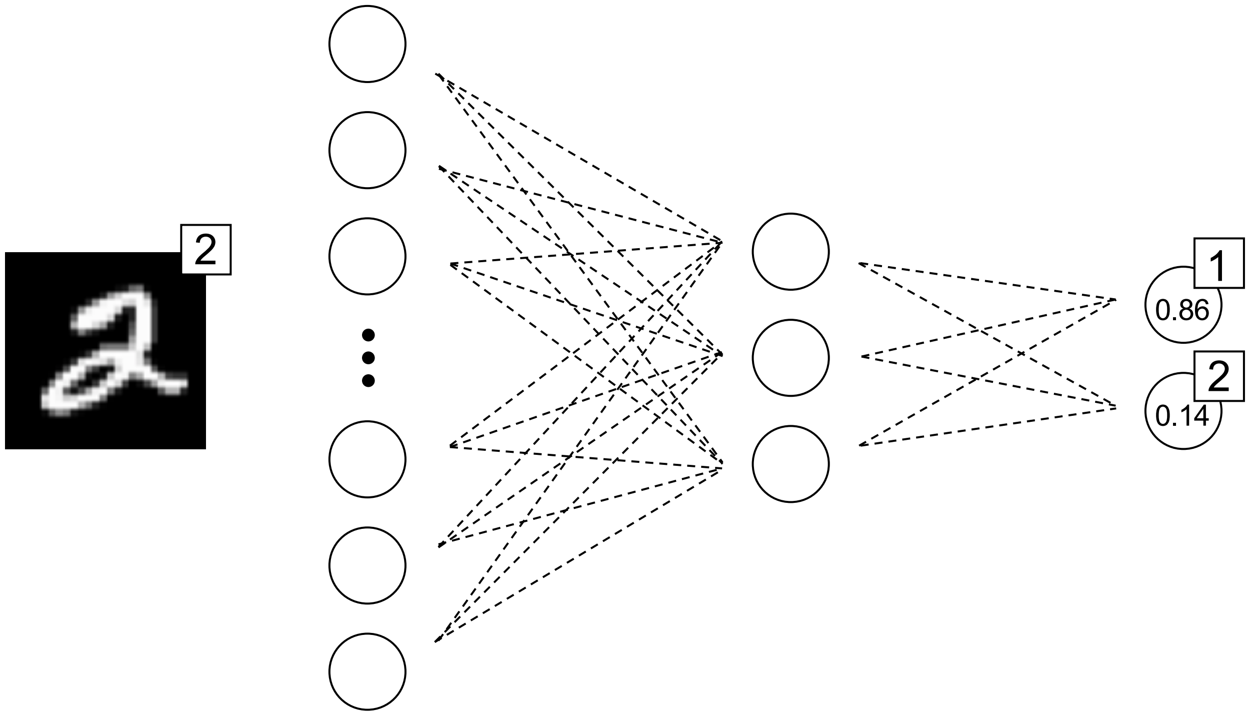
Learning procedure

1. Step: Initialize random weights and biases

Neural Networks

A short primer

Classifier → Handwritten Digits



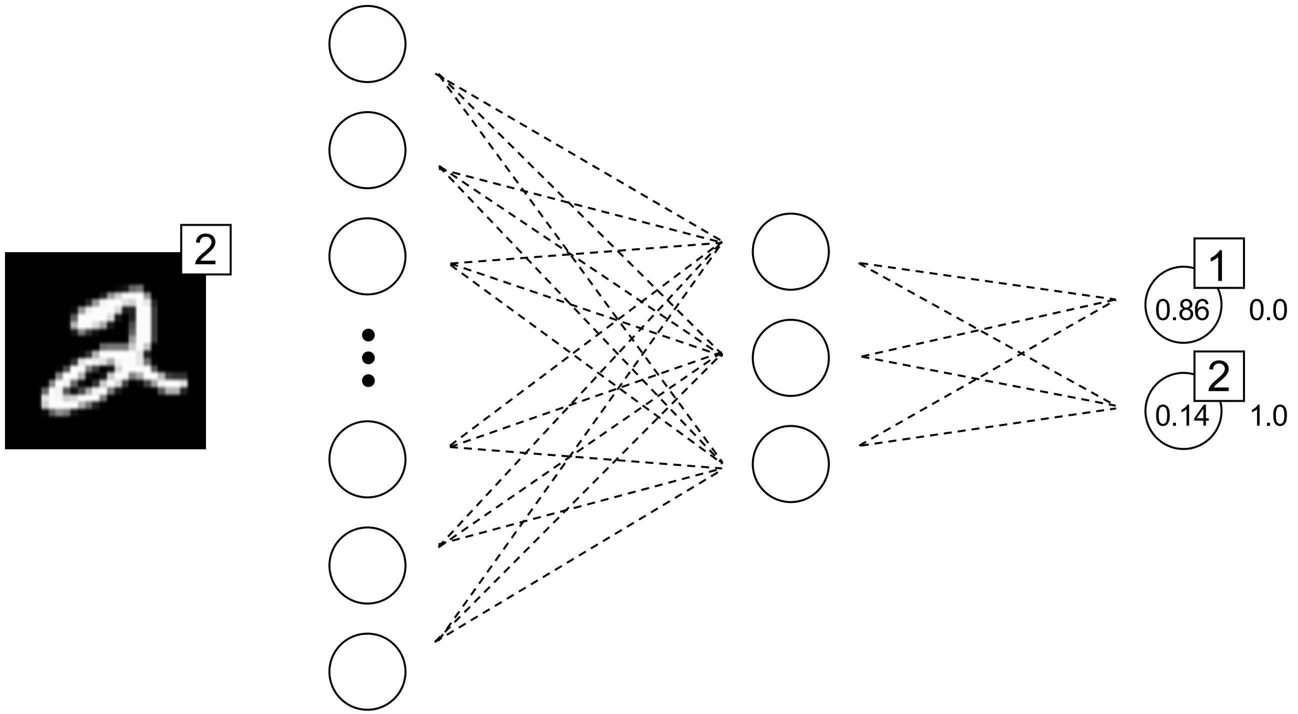
Learning procedure

1. Step: Initialize random weights and biases
2. Step: Operate the network in feedforward mode → *Forward pass*

Neural Networks

A short primer

Classifier → Handwritten Digits



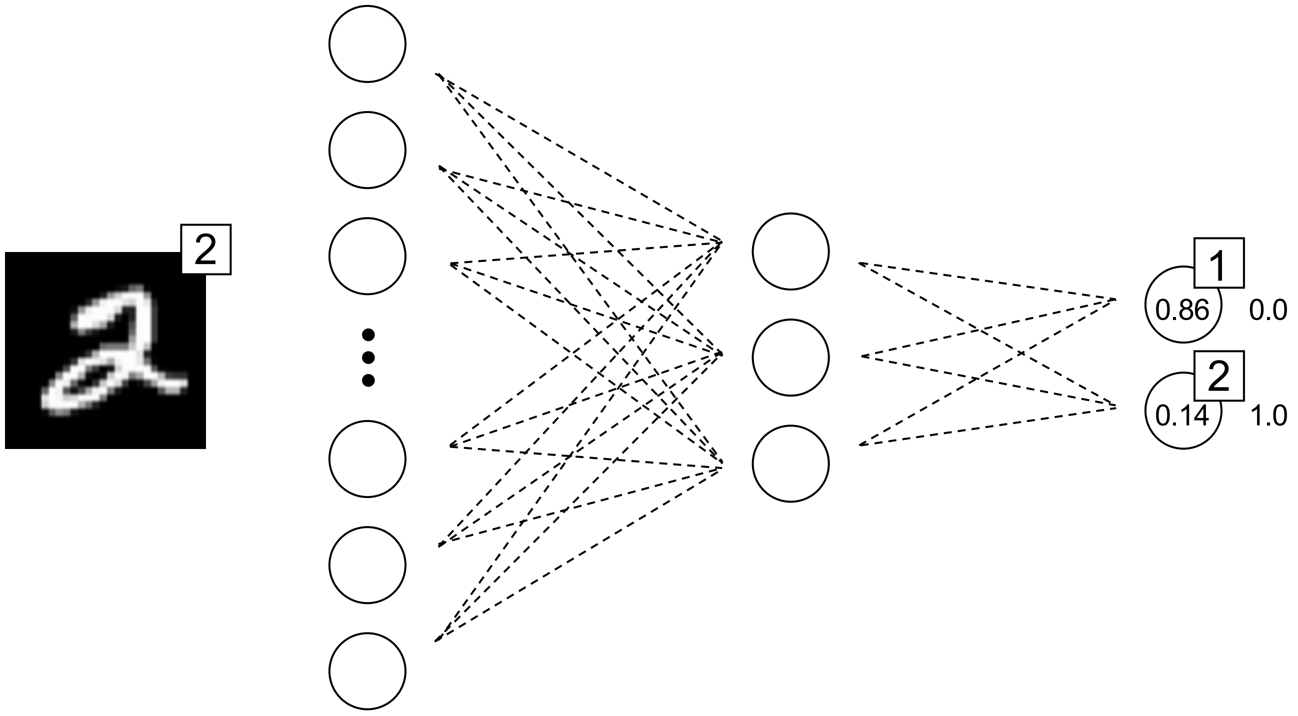
Learning procedure

1. Step: Initialize random weights and biases
2. Step: Operate the network in feedforward mode → *Forward pass*
3. Step: Calculate the *loss function* based on the *data label* (for example MSE)

Neural Networks

A short primer

Classifier → Handwritten Digits



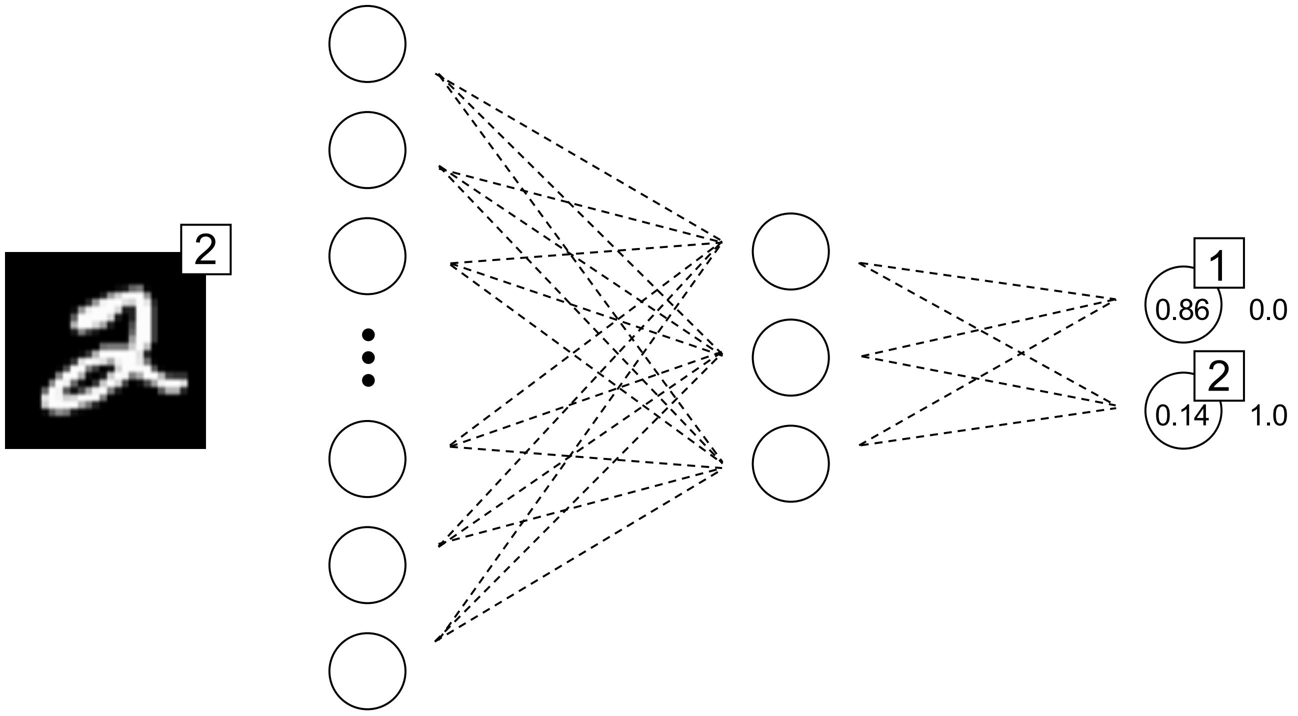
Learning procedure

1. Step: Initialize random weights and biases
2. Step: Operate the network in feedforward mode → *Forward pass*
3. Step: Calculate the *loss function* based on the *data label* (for example MSE)
4. Step: Propagate the loss back through the network towards the input and adjust the weights according to a *gradient descent* algorithm → *Backpropagation*

Neural Networks

A short primer

Classifier → Handwritten Digits



Learning procedure

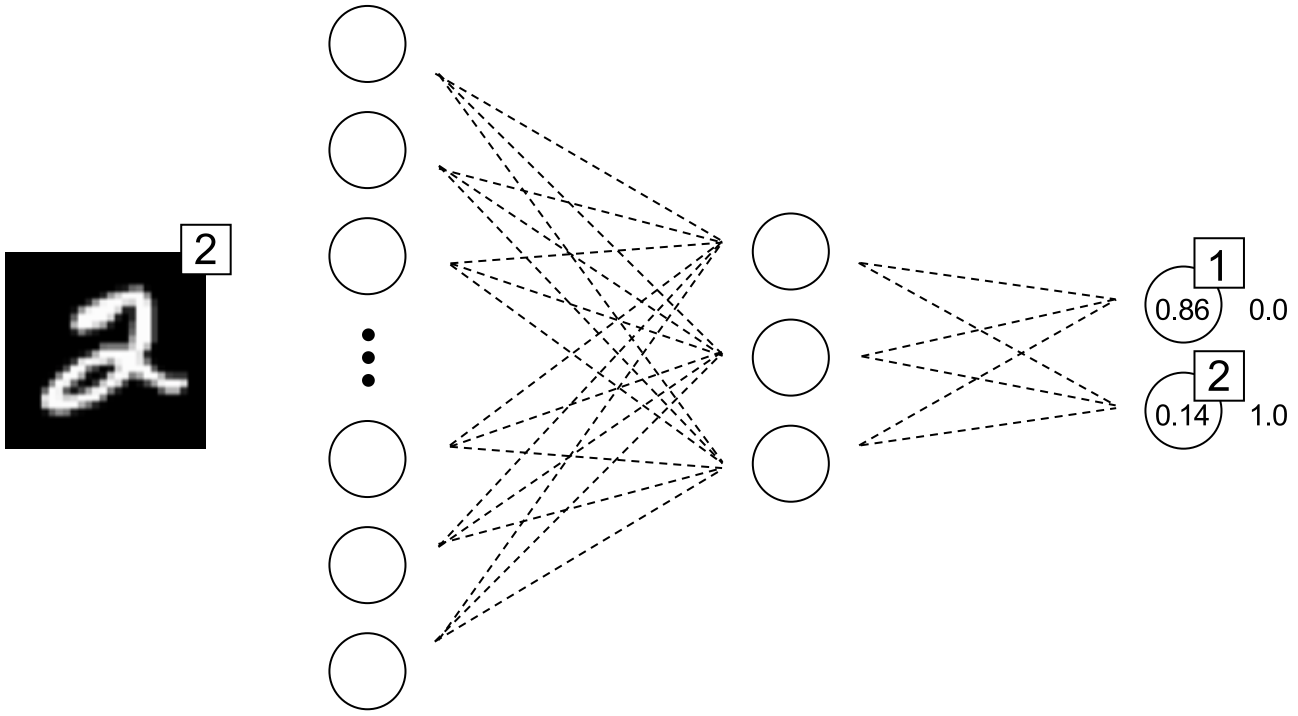
1. Step: Initialize random weights and biases
2. Step: Operate the network in feedforward mode → *Forward pass*
3. Step: Calculate the *loss function* based on the *data label* (for example MSE)
4. Step: Propagate the loss back through the network towards the input and adjust the weights according to a *gradient descent* algorithm → *Backpropagation*

→ Do this for all training data sets
→ This is then called a training *epoch*
→ *Train the network for many epochs until the loss function converges*

Neural Networks

A short primer

Classifier → Handwritten Digits



Learning procedure

1. Step: Initialize random weights and biases
2. Step: Operate the network in feedforward mode → *Forward pass*
3. Step: Calculate the *loss function* based on the *data label* (for example MSE)
4. Step: Propagate the loss back through the network towards the input and adjust the weights according to a *gradient descent* algorithm → *Backpropagation*

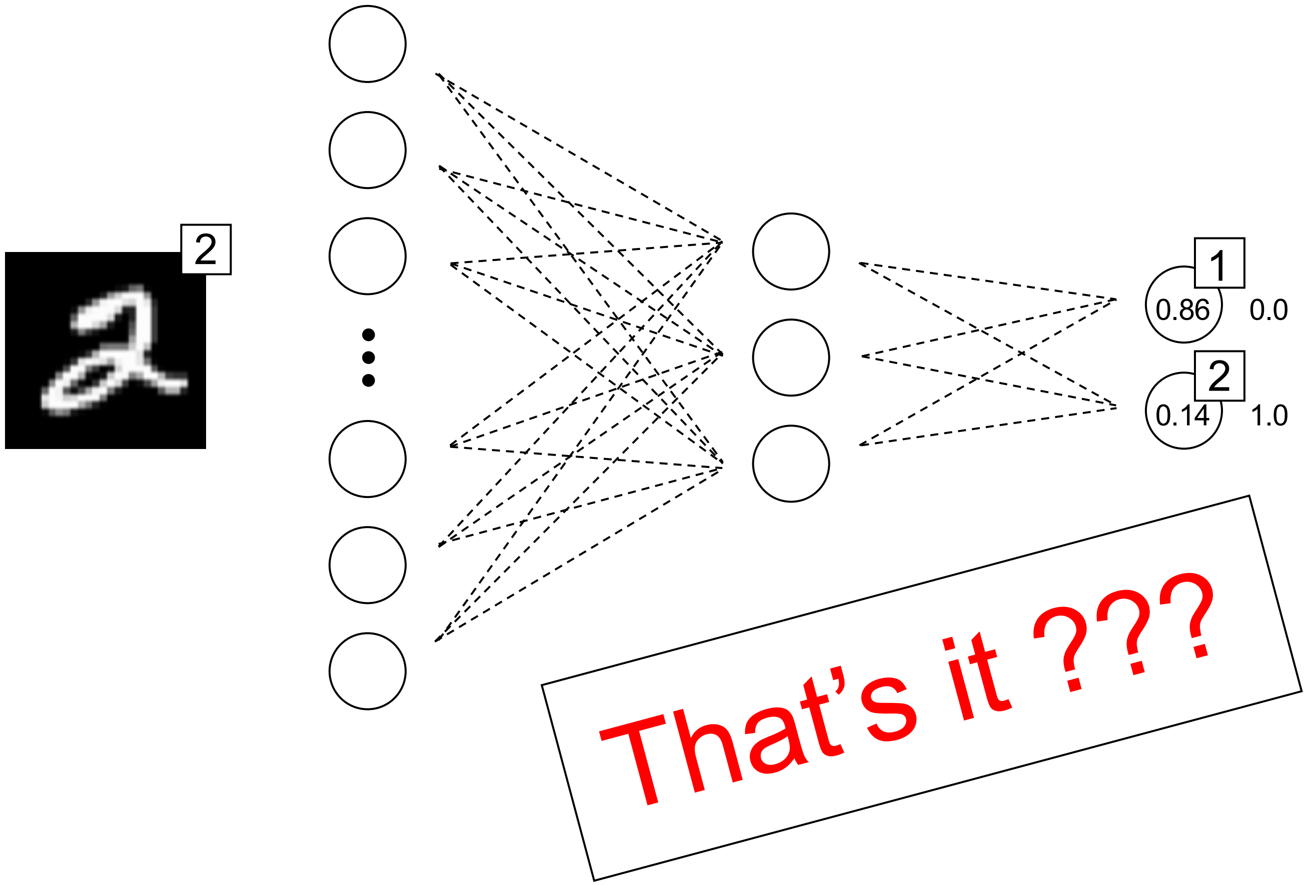
→ Do this for all training data sets
→ This is then called a training *epoch*
→ *Train the network for many epochs until the loss function converges*

Prediction is just a normal *forward pass*

Neural Networks

A short primer

Classifier → Handwritten Digits



Learning procedure

1. Step: Initialize random weights and biases
2. Step: Operate the network in feedforward mode → *Forward pass*
3. Step: Calculate the *loss function* based on the *data label* (for example MSE)
4. Step: Propagate the loss back through the network towards the input and adjust the weights according to a *gradient descent* algorithm → *Backpropagation*

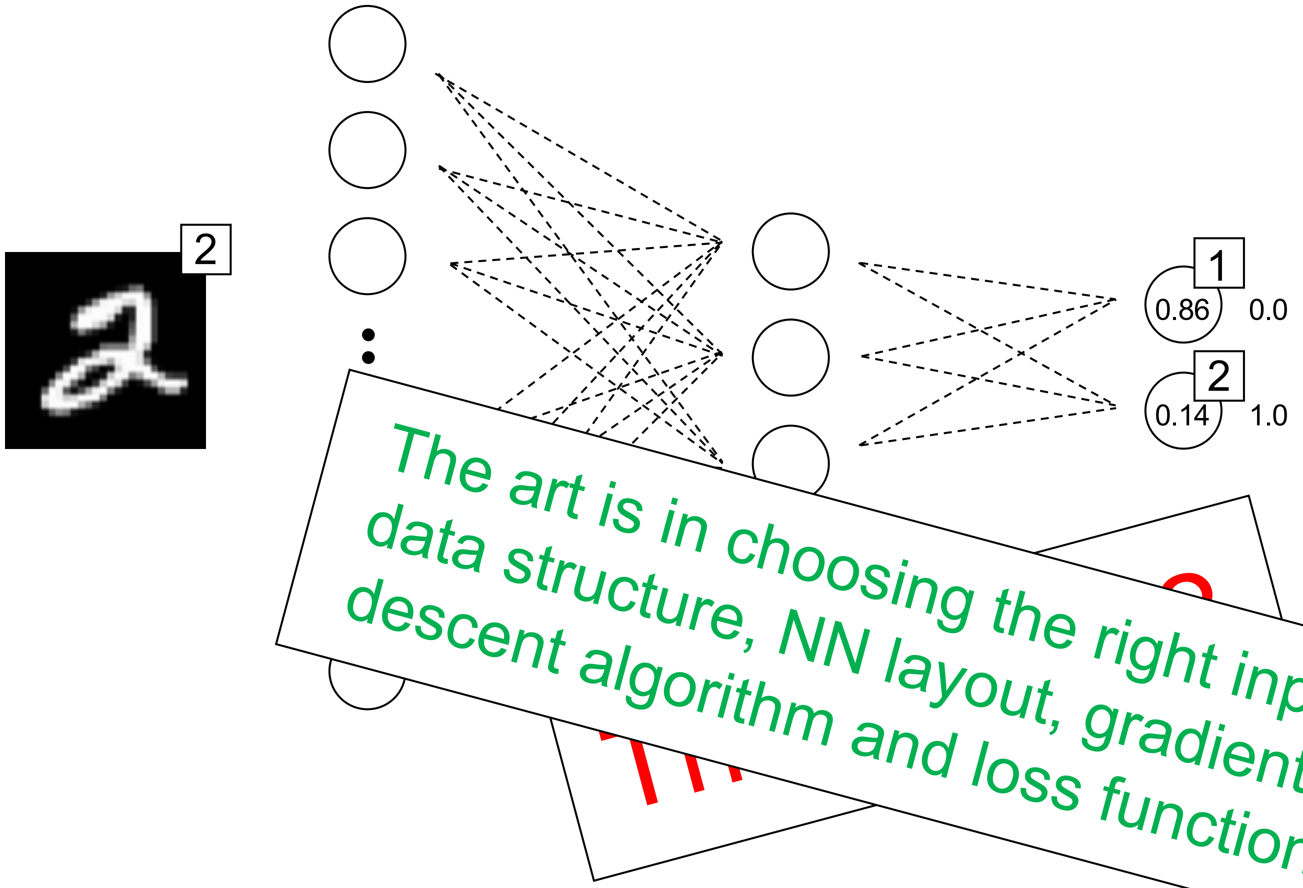
→ Do this for all training data sets
→ This is then called a training *epoch*
→ *Train the network for many epochs until the loss function converges*

Prediction is just a normal *forward pass*

Neural Networks

A short primer

Classifier → Handwritten Digits



The art is in choosing the right input data structure, NN layout, gradient descent algorithm and loss function…

Learning procedure

1. Step: Initialize random weights and biases
2. Step: Operate the network in feedforward mode → *Forward pass*
3. Step: Calculate the *loss function* based on the *data label* (for example MSE)
4. Step: Propagate the loss back through the network towards the input and adjust the weights according to a *gradient descent* algorithm → *Backpropagation*

→ Do this for all training data sets

→ This is then called a training *epoch*

the network for many epochs until

oss function converges

on is just a normal *forward pass*