Machine Learning

//////////

Introduction

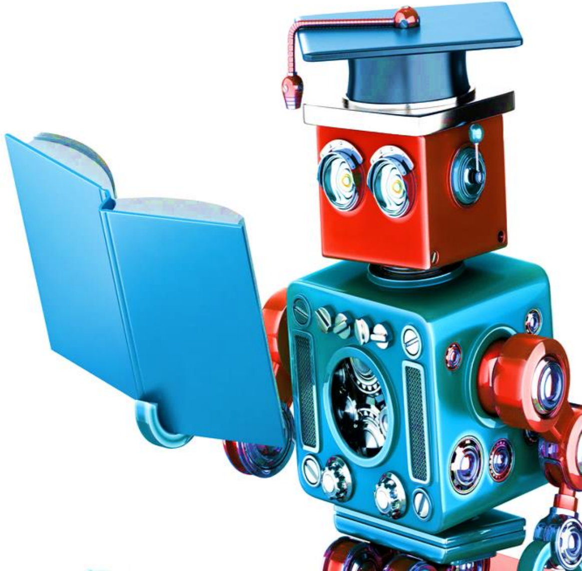
May 2018 / Katja Glaß

Agenda

- Overview
- Neural Networks
- CTCAE Grading as Example
- Use Cases
- Summary

Overview

Machine learning is a field of computer science that gives computers the ability to learn without being explicitly programmed.



Overview

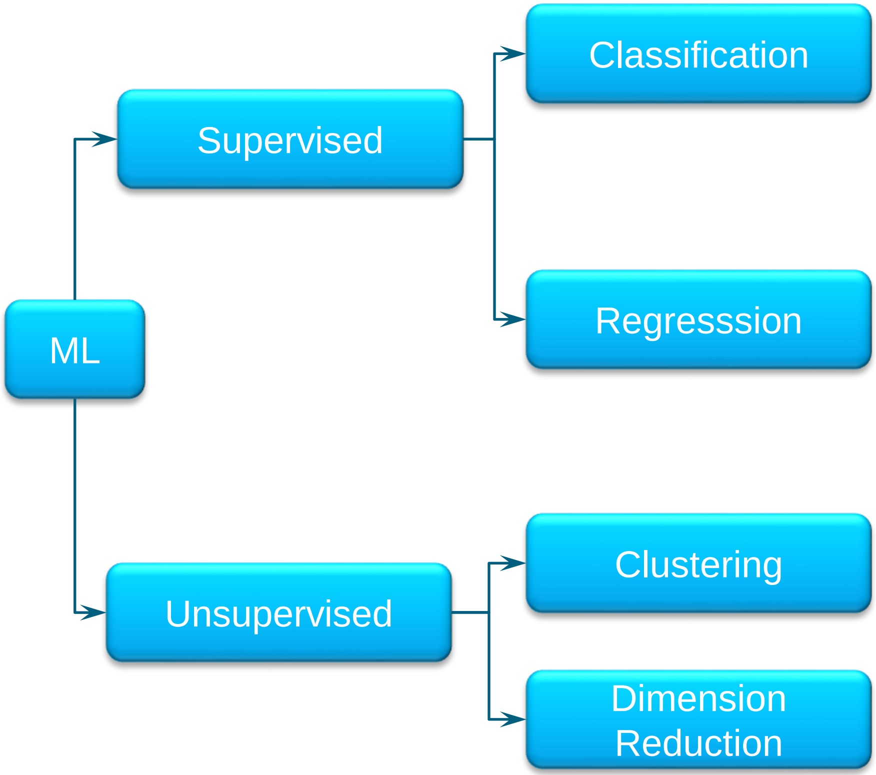
Different approaches for different purposes

Big data / high complexity

- Deep Neural Networks
- Recurrent Neural Networks
- Reinforcement Learning

Overview

Different approaches for different purposes



- Artificial Neural Networks
- Logic Regression
- Support Vector Machines
- Random Forest …

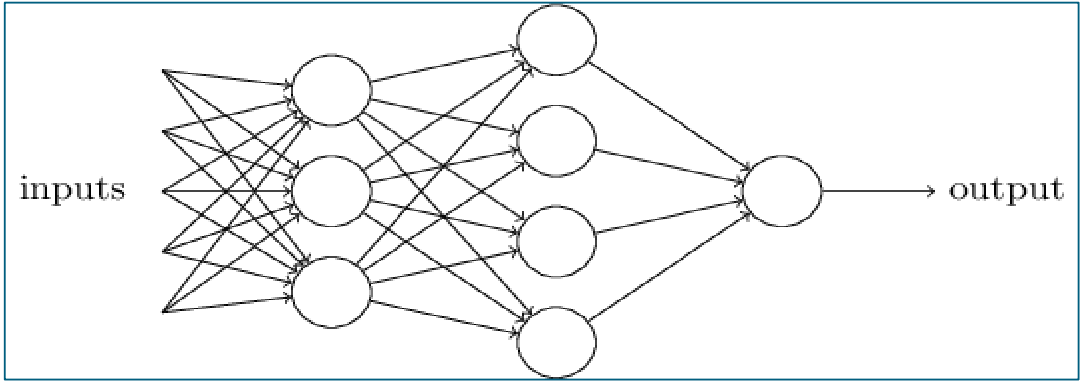
- Artificial Neural Networks
- Linear Regression
- Decision Trees
- Bayesian Networks …

Overview

Neural Networks

Current Hype are Neural Networks & Deep Neural Networks

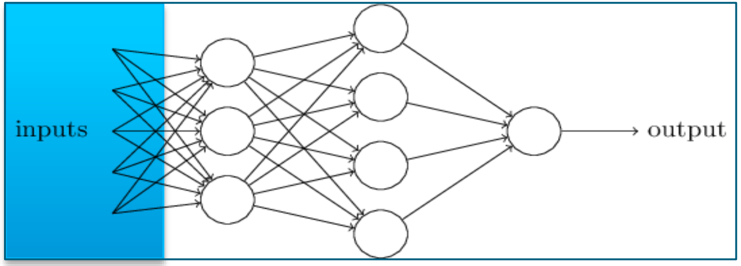
- Extremly powerful
  - Image recognition
  - Natural language processing



Neural Networks

Rough understanding

1) Transform issues to numeric representation
   - Number of inputs (neurons)
   - Value of each input



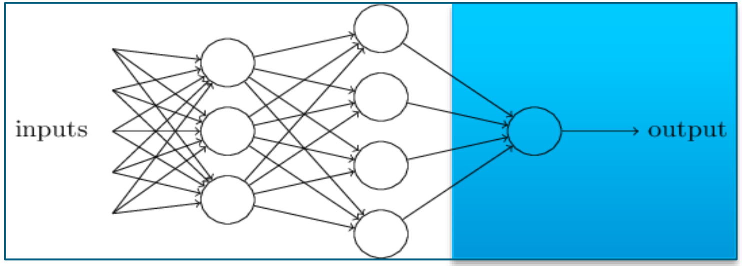
Example:

Example	Neurons	Neuron Values
Grey Image	Each pixel one neuron	Grey value
Text	Each word a neuron	Frequency
Lab analysis	Each numeric variable Each characteristic of character variable	Numeric value 0 / 1 appearance

Neural Networks

Rough understanding

1) Transform issues to numeric representation
   - Number of inputs (neurons)
   - Value of each input



Example:

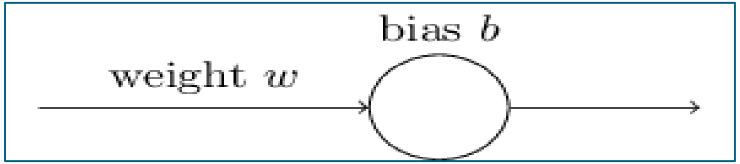
Example	Neurons	Neuron Values
Elephant? (image)	One	Elephant 0 .. 1 (probability)
Text classification (SDTM, ADAM, TLF?)	Each characteristic one neuron	ADAM 0 .. 1 (probability) SDTM 0 .. 1 (probability) TLF 0 .. 1 (probability)
Grading value? 0..4	Classification -> one neuron per classification	Grade 0 0 .. 1 (prob) …

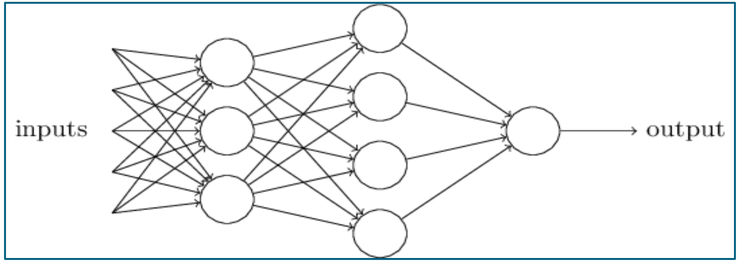
Neural Networks

Rough understanding

2) Structure

- Weight & Bias (optimized by algorithm)
- Number of neurons (5,3,4,1)
- Number of layers (3 with output layer)



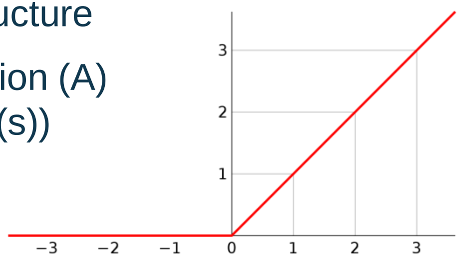


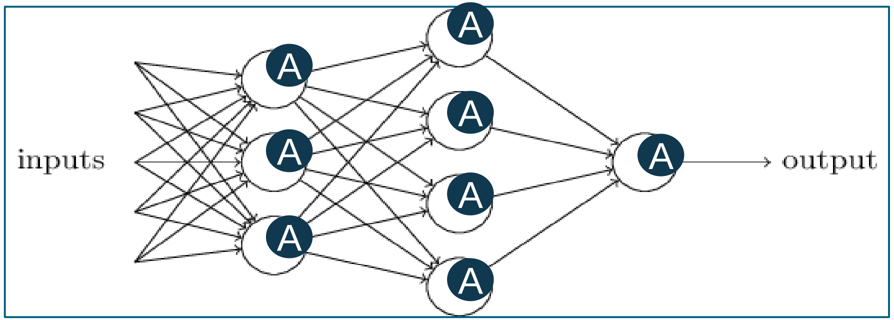
Neural Networks

Rough understanding

3) Mathematics Structure

- Activation Function (A)
  (output = f(input(s))



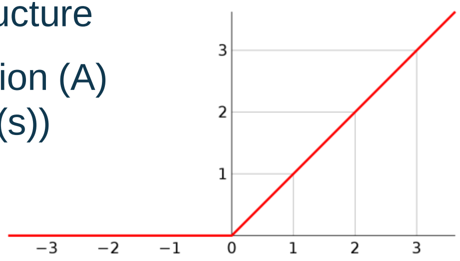


Neural Networks

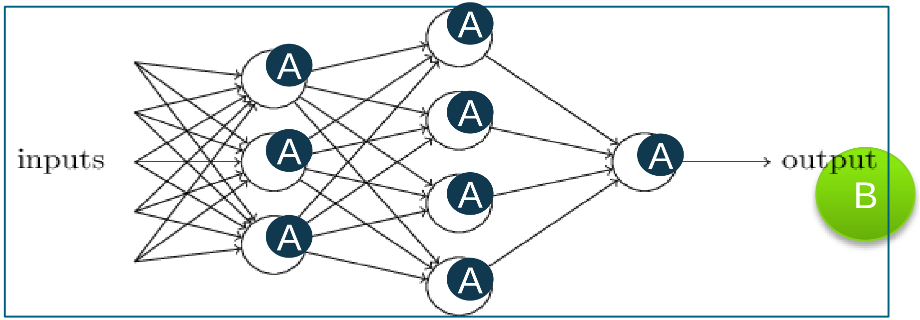
Rough understanding

3) Mathematics Structure

- Activation Function (A) (output = f(input(s)))



- Success-Function (B)
  - Cost / Utility
  - Accuracy
  - Square Error …

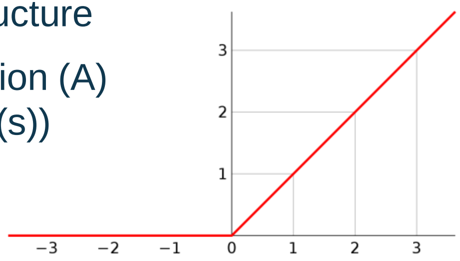


Neural Networks

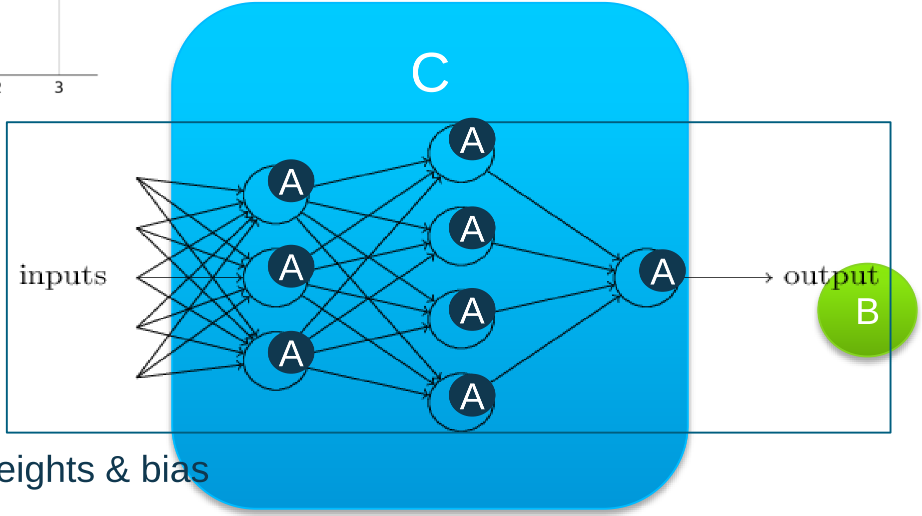
Rough understanding

3) Mathematics Structure

- Activation Function (A) (output = f(input(s)))



- Success-Function (B)
  - Cost / Utility
  - Accuracy
  - Square Error …



- Optimization function (C) – updates weights & bias
  - Gradient Decent …

Neural Network

Recommended Literature

Free EBook “Neural Networks and Deep Learning” (http://neuralnetworksanddeeplearning.com/)

- Introduction & Statistical insights
- MNIST Example, large database with labeled handwritten digits

Video Introduction “Tensorflow and deep learning - without a PhD”
(https://www.youtube.com/watch?v=vq2nnJ4g6N0&list=LL3uReggFn2MOSMZlG69vzEw)

- 2,5h video introduction on Tensorflow (free Google machine learning toolset)
- Top for first hands-on experiences, implementation complexity low

Cool real world applications in a nutshell (~4 minutes each)

- https://www.youtube.com/watch?v=Bui3DWs02h4
- https://www.youtube.com/watch?v=aKSILzbAqJs

Neural Network

Machine Learning Crash Course

Machine Learning Crash Course with TensorFlow APIs

- 40+ exercises
- 25 lessons
- 15 hours
- Lectures from Google researchers
- Real-world case studies
- Interactive visualizations of algorithms in action

CTCAE Grading as Example

Anemia – simple test case

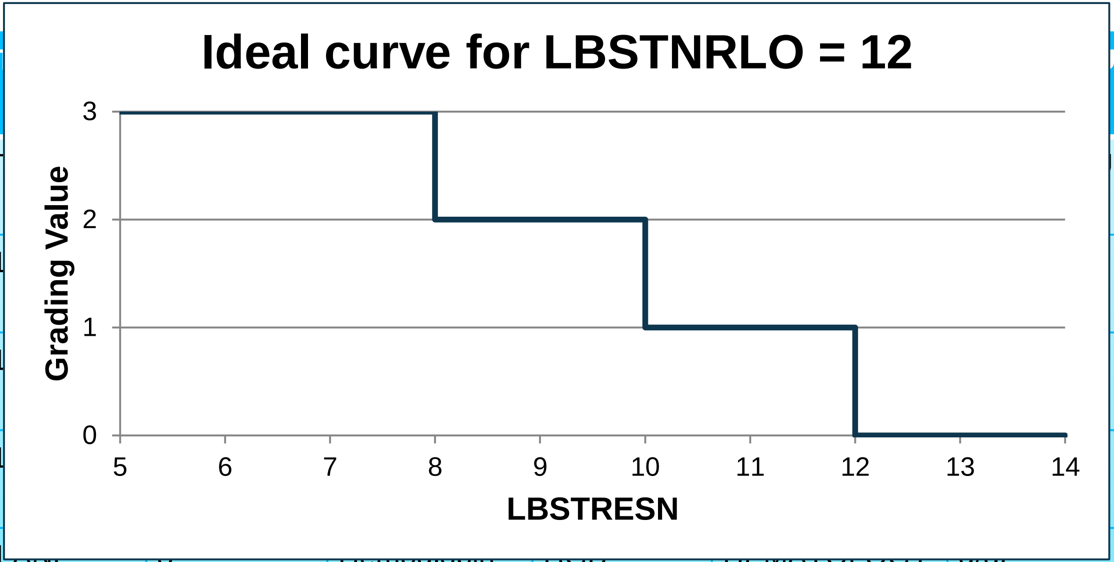
- First hands-on-experiences
- See how / why it is working
- Student (4 weeks) with Phython, Keras & TensorFlow experiences
- Two input parameters (Limit, Lab value)
- One output parameter (Grading 0 .. 3)

CTCAE Grading as Example

Anemia – simple test case

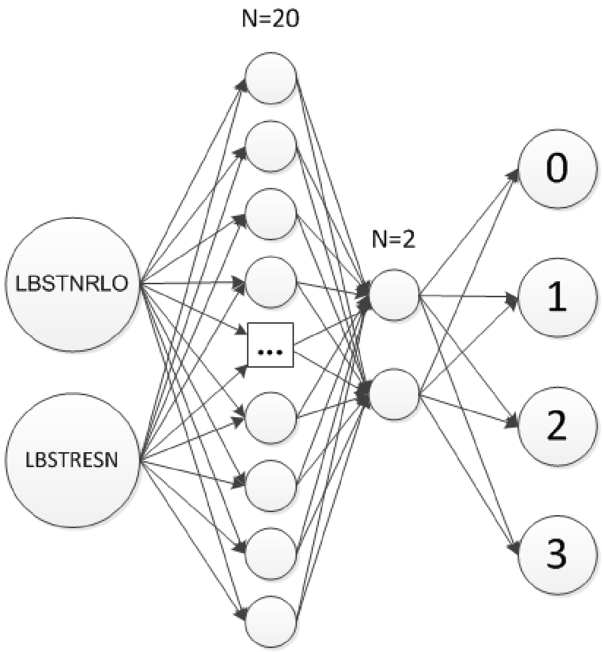
Input: LBSTNRLO (low), LBSTRESN (value)

XTTEST	XT						RANGE_LOW	RANGE_HIGH
XTTEST	XT						lower end	upper end
Anemia	BL						GE LBSTNRLO	
Anemia	BL						GE 10.0	LT LBSTNRLO
Anemia	BL						GE 8.0	LT 10.0
Anemia	BLANE	3	Hemoglobin	HGB	HEMATOLOGY	g/dL		LT 8.0

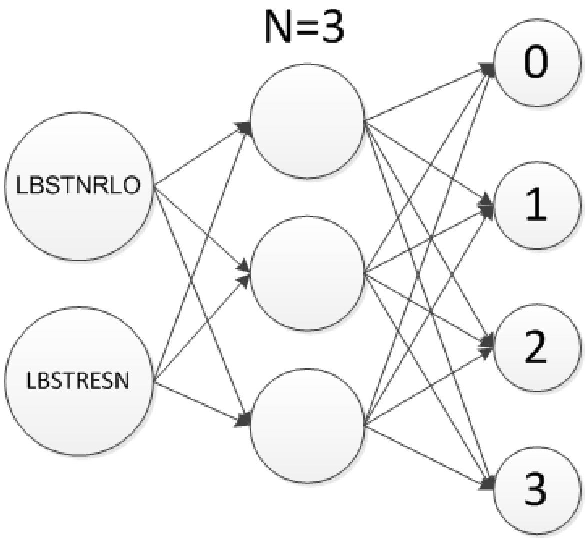


CTCAE Grading as Example

Network Layout



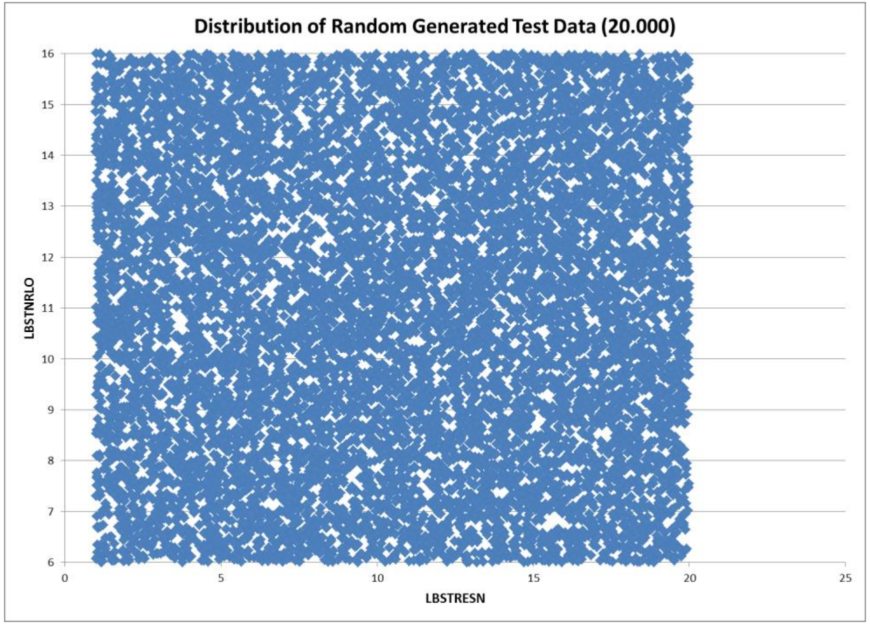
Sequential 20/2/4



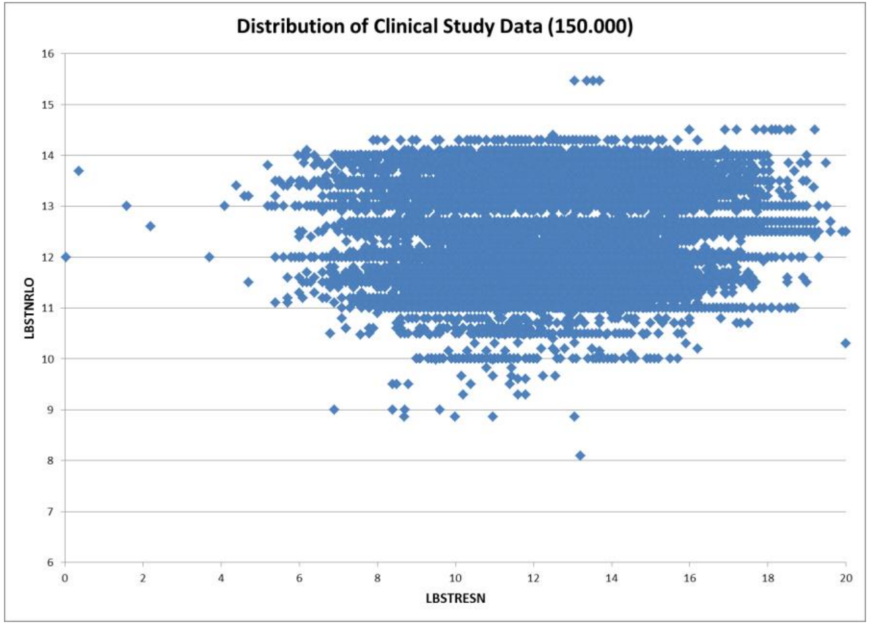
Sequential 3/4

CTCAE Grading as Example

Input Data



Random



Study

CTCAE Grading as Example

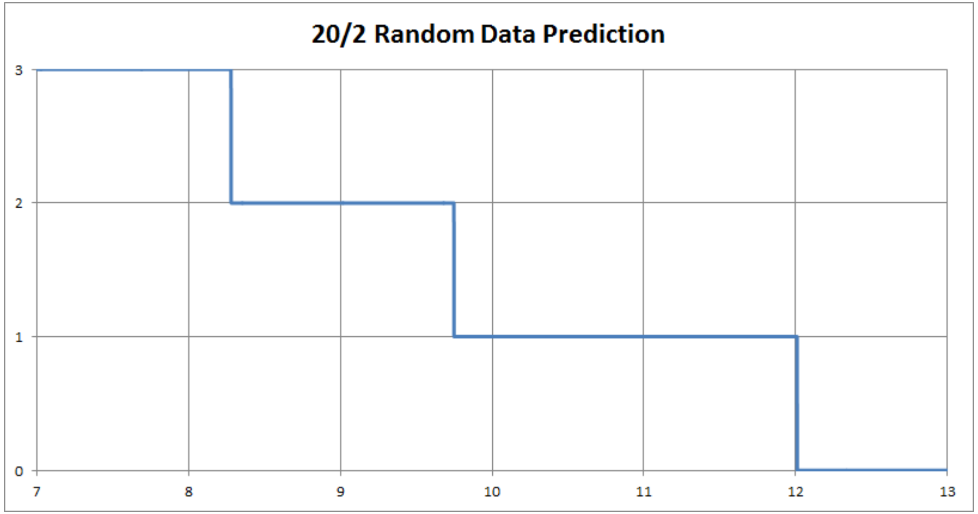
Implementing a Model with Keras (TensorFlow)

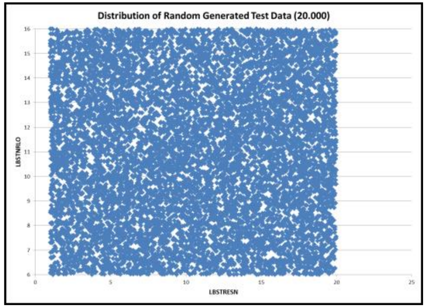
```
# create model
model = Sequential()
model.add(Dense(3, input_dim=2, activation='sigmoid'))
model.add(Dense(4, activation='sigmoid'))

# Compile model
model.compile(loss='categorical_crossentropy', optimizer='adam', metrics=['accuracy'])
# Fit the model
model.fit(dataTrain, labelTrain, epochs=100, batch_size=500)
# evaluate the model
scores = model.evaluate(dataFit, labelFit)
model.save("Trained_Models/my_model_sigmoid_2_3_Cri_V1.h5")
print("\n%s: %.2f%%" % (model.metrics_names[1], scores[1]*100))
```

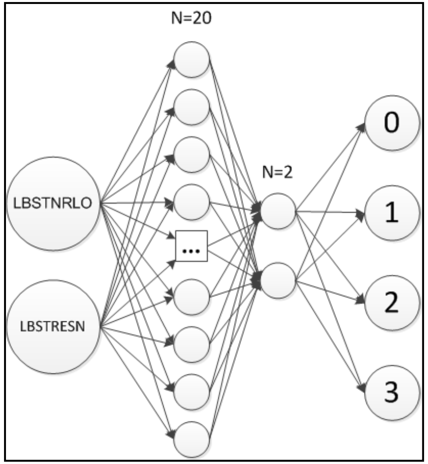
CTCAE Grading as Example

Results - Random generated Data

Model	Data	Loop1 Accuracy	Loop2 Accuracy	Loop3 Accuracy	Range = 12 Accuracy
20 / 2	Random	89.67%	93.33%	96.33%	92,28%



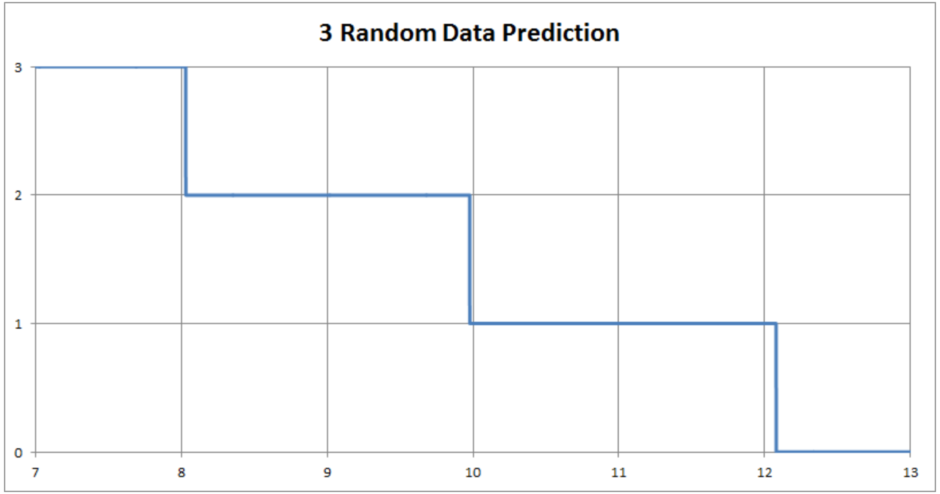


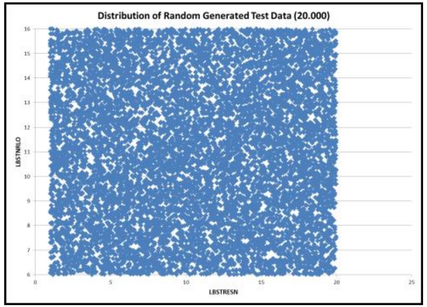


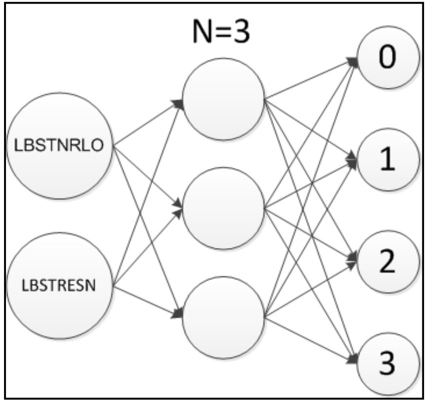
CTCAE Grading as Example

Results - Random generated Data

Model	Data	Loop1 Accuracy	Loop2 Accuracy	Loop3 Accuracy	Range = 12 Accuracy
20 / 2	Random	89.67%	93.33%	96.33%	92,28%
3	Random generated Data	88.18%	96.37%	98.63	98,13%



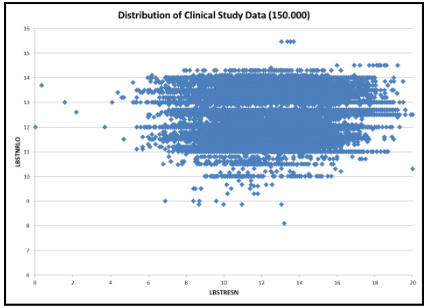


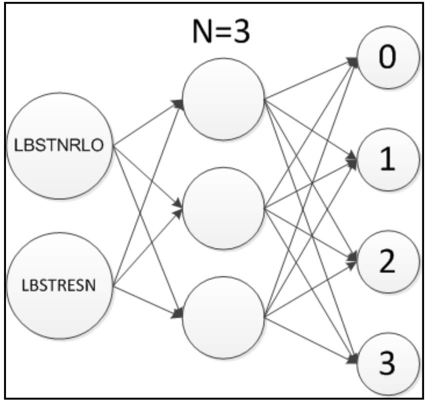


CTCAE Grading as Example

Results - Random generated Data

Model	Data	Loop1 Accuracy	Loop2 Accuracy	Loop3 Accuracy	Range = 12 Accuracy
20 / 2	Random	89.67%	93.33%	96.33%	92,28%
3	Random	88.18%	96.37%	98.63%	98,13%
3	Clinical	96.03%	98.91%	**99.03%**	

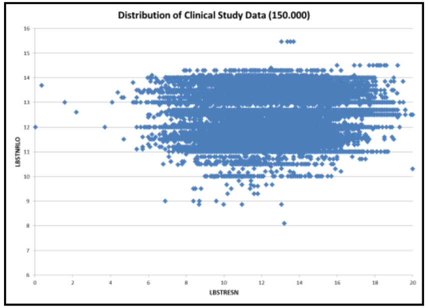


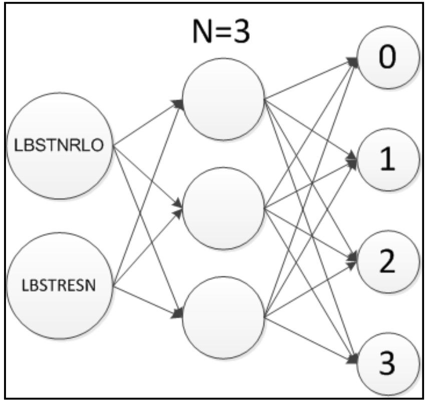


CTCAE Grading as Example

Results - Random generated Data

Model	Data	Loop1 Accuracy	Loop2 Accuracy	Loop3 Accuracy	Range = 12 Accuracy
20 / 2	Random	89.67%	93.33%	96.33%	92,28%
3	Random	88.18%	96.37%	98.63%	98,13%
3	Clinical	96.03%	98.91%	**99.03%**	**68.75%**

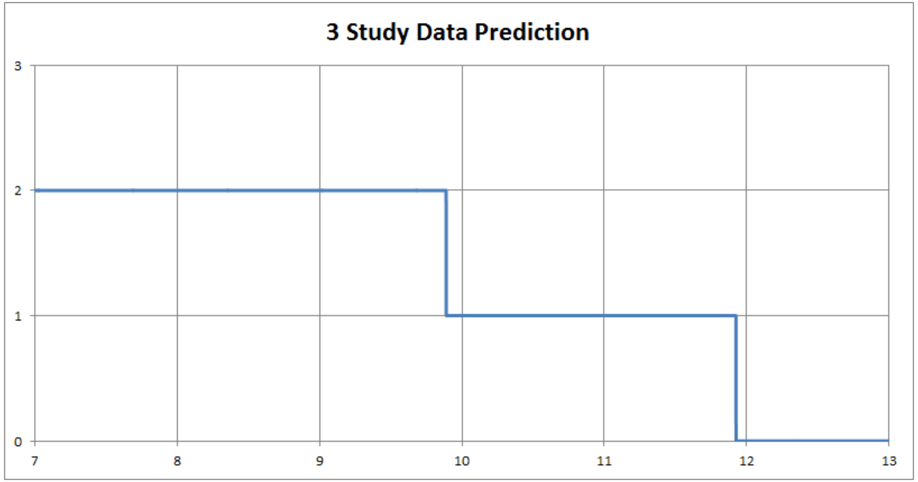


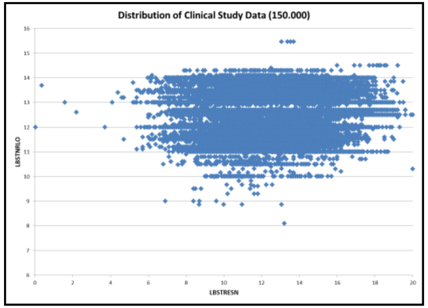


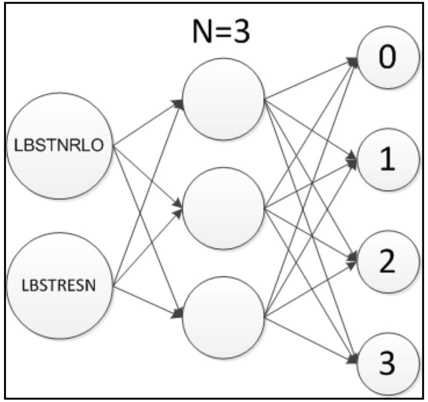
CTCAE Grading as Example

Results - Random generated Data

Model	**Data**	**Loop1 Accuracy**	**Loop2 Accuracy**	**Loop3 Accuracy**	**Range = 12 Accuracy**
20 / 2	Random	89.67%	93.33%	96.33%	92,28%
3	Random	88.18%	96.37%	98.63%	98,13%
3	Clinical	96.03%	98.91%	**99.03%**	**68.75%**







Use Cases

Where can we apply Machine Learning in our area?

Use Cases

CSS Working Group „Machine Learning“

Educate for Future! Find use cases

- Drug discovery
- Drug candidate selection
- Clinical system optimization
- Medical image recognition
- Medical diagnoses
- **Optimum site selection / recruitment**
- **Data anomaly detection**
- **Personalized medicine**

Use Cases

More Use Cases

- Data mapping
- Coding
- Virtual monitoring / studies (medical devices)
- RWE evaluations (tons of data)
- Chat bots for questions / tickets
- Document search

Use Cases

CTCAE Grading

- Apply ML on available formula for learning (1)
- Investigate depending parameters for doctor’s required grading (2)
- Check doctor’s grading (3)

Use Cases

CTCAE Grading

- Apply ML on available formula for learning (1)
- Investigate depending parameters for doctor’s required grading (2)
- Check doctor’s grading (3)

Positive	Negative

// Learning opportunity with hands-on experiences (1)
// Lower machine-power / Laptop sufficient (1) (no clould required)
// Enough data available
	// Challenge doctor’s decision?

Use Cases

Optimum site selection / recruitment

- Build up site / recruitment data pool
- ML to learn quality / expected recruitment on parameters

Use Cases

Optimum site selection / recruitment

- Build up site / recruitment data pool
- ML to learn quality / expected recruitment on parameters

Challenges

- Get information of possible parameters/features
  - General (no. of staff, fluctuation rate, urban catchment, …)
  - Study specific (TAS, experts, experiences, past recruitment-time rate, …)
  - Subject specific (dropout rate, screening failure rate, …)

- Evaluation size, Cloud need
  - Likely need “machine learning” optimized machine

Use Cases

Optimum site selection / recruitment

- Build up site / recruitment data pool
- ML to learn quality / expected recruitment on parameters

Challenges

- Get information of possible parameters/features
  - General (no. of staff, fluctuation rate, urban catchment, …)
  - Study specific (TAS, experts, experiences, past recruitment-time rate, …)
  - Subject specific (dropout rate, screening failure rate, …)

Data

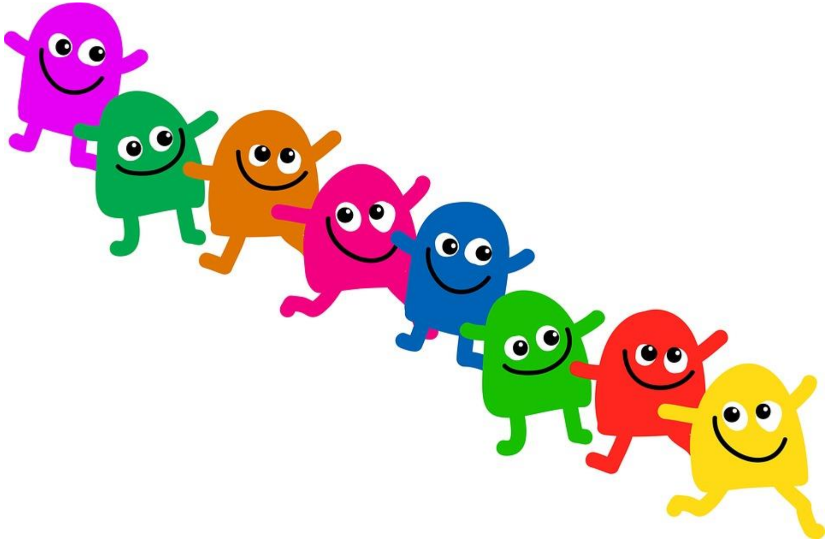
- Evaluation size, Cloud need
  - Likely need “machine learning” optimized machine

Cloud

Use Cases

Document search for relevance

- Data pool available (https://www.lexjansen.com/ >30.000 papers, >1700 from PhUSE)
- Excellent exercise for text processing
- Text / Paper processing is a generic issue
  - No data-security issues
  - Likely find publications / implementations
  - Ideal for community project



Summary

- Technical
  - Transfer issues into mathematic issue representation (frequency of words, …)
  - Have enough data!
    - 100 studies is not much for a machine learning, even 1000 not
    - 1.000.000 lab observations are likely sufficient

- Implementation
  - Many tutorials, easy implementation tools
  - Finding the right layout & parameters is the issue
  - Long run-times, probably cloud computing required

- Use Cases
  - Difficult to find practical ones

Summary

Think About

- What is the benefit using ML vs traditional data evaluations?
  - Getting with traditional methods 80% quickly
    - Data anomaly checks
    - Data mapping
  - Lack quite often data science, independent of ML or other methods
    - Recruitment / site selection
  - ML can easily scale up
    - Increasing amount of data
    - Process increasing amount of work

Summary

Other considerations

- Don’t start in silos
- Knowledge build up required

- Benefit from the community!
  - Join the “Machine Learning” CSS Work group
  - Visit the EU Connect Conference with
    - Machine Learning Stream
    - Machine Learning Hands-on-Workshop

Future Forward

Thank You!