Multivariate Methods in Statistical Data Analysis

- Web-Site: http://tmva.sourceforge.net/
- See also: "*TMVA - Toolkit for Multivariate Data Analysis , A. Hoecker, P. Speckmayer, J. Stelzer, J. Therhaag, E. von Toerne, H. Voss et al.,* *arXiv:physics/0703039v5 [physics.data-an]*

universitätbonn

Contents

- Introduction to Multivariate Analysis (MVA)
- Boosted Decision Trees
- ATLAS Machine Learning Challenge
- Non-HEP Applications and Data Science

universitätbonn

Introduction to Multi-variate Analysis

universitätbonn

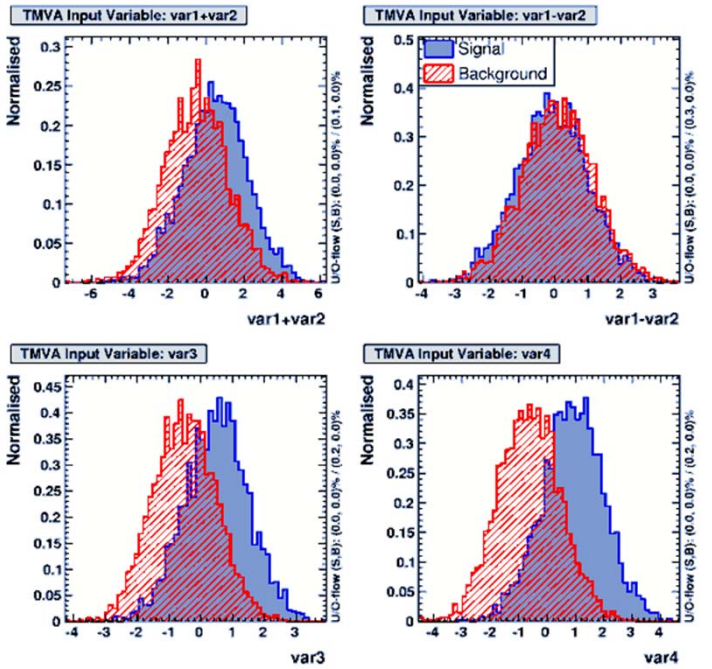
Event Classification in High-Energy Physics

- Most HEP analyses require discrimination of signal from background:
  - Event level (Higgs searches, …)
  - Cone level (Tau-vs-jet reconstruction, …)
  - Track level (particle identification, …)
  - Lifetime and flavour tagging (*b*-tagging, …)
  - Parameter estimation (*CP* violation in *B* system, …)
  - etc.
- The multivariate input information used for this has various sources
  - Kinematic variables (masses, momenta, decay angles, …)
  - Event properties (jet/lepton multiplicity, sum of charges, …)
  - Event shape (sphericity, Fox-Wolfram moments, …)
  - Detector response (silicon hits, *dE*/*dx*, Cherenkov angle, shower profiles, muon hits, …)
  - etc.
- Traditionally few powerful input variables were combined; new methods allow to use up to 100 and more variables w/o loss of classification power

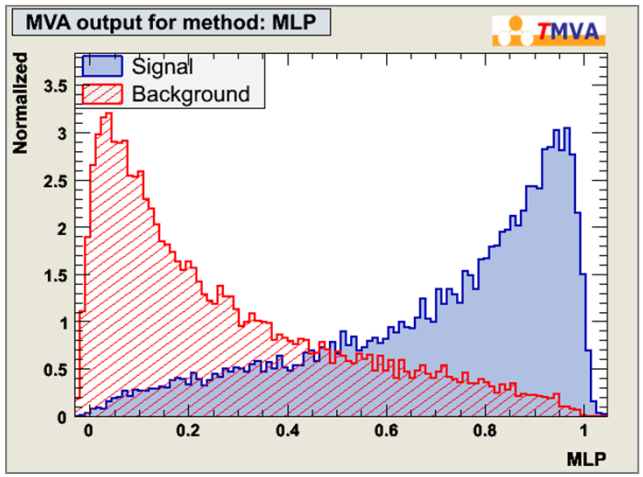
  e.g. MiniBooNE: NIMA 543 (2005), or D0 single top: Phys.Rev. D78, 012005 (2008)

What is a multi-variate analysis

- "Combine" all input variables into one output variable
- Supervised learning means learning by example: the program extracts patterns from training data
- Methods for un-supervised learning → not common in HEP, yet



Input Variables

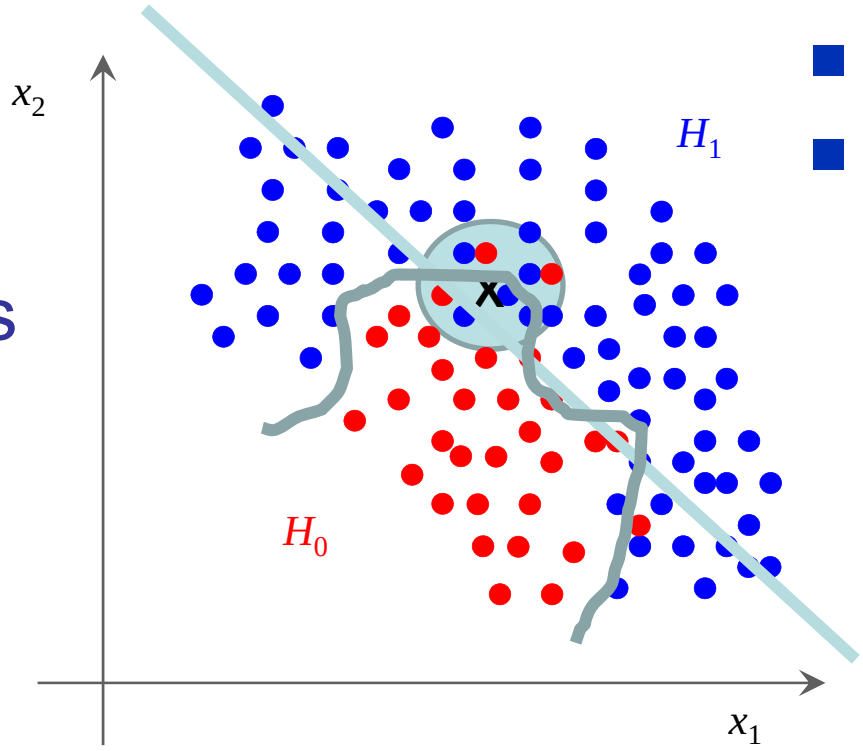


Classifier Output

universitätbonn

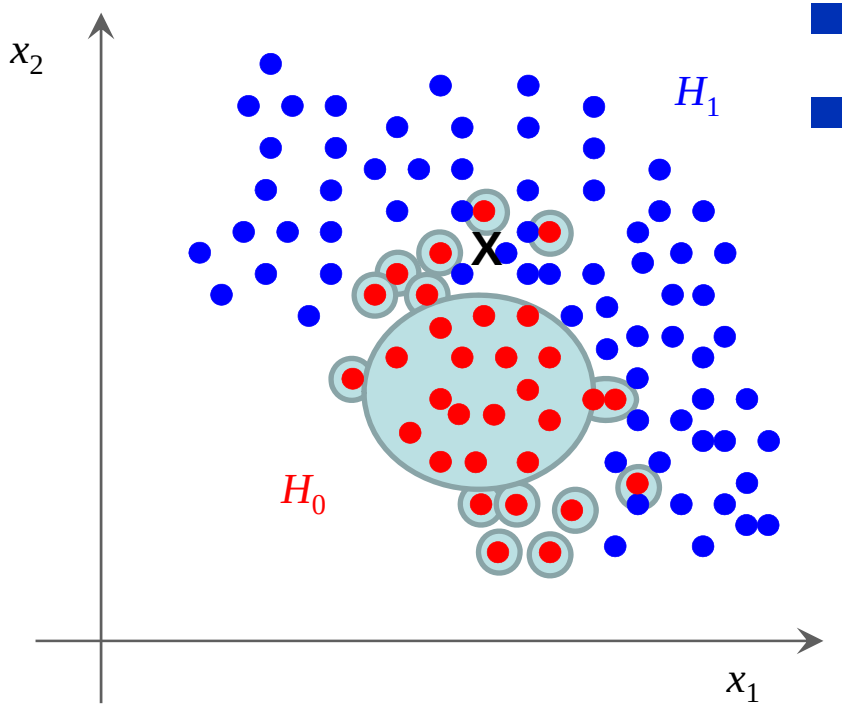
MVA methods

- Kernel PDE (probablility density estimators)
- functional approaches (Linear, Likelihoods, …)
- General methods:
  - Neural nets,
  - Boosted Decision trees
  - Support Vector machines



Overtraining

- If the MVA follows statistical fluctuations of input training data → performance will not be reproducable on independent training data.



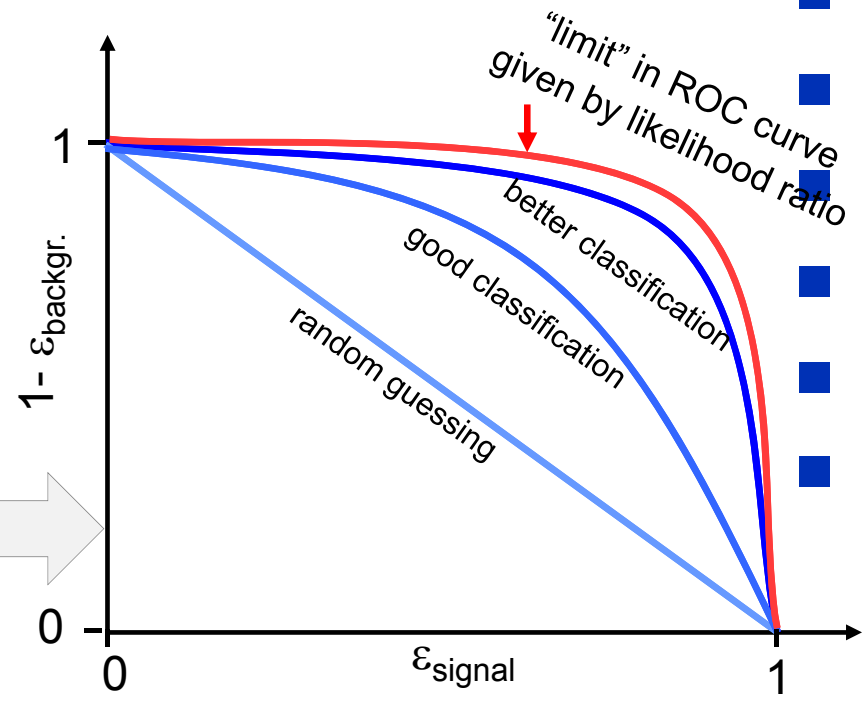
Neyman-Pearson Lemma

$$\text{Likelihood Ratio}: \quad y(x) = \frac{P(x|S)}{P(x|B)}$$

<u>Neyman-Peason</u>:

The Likelihood ratio used as "selection criterion" y(x) gives for each selection efficiency the best possible background rejection.

i.e. it maximises the area under the "*Receiver Operation Characteristics*" (ROC) curve



Varying y(x)>"cut" moves the working point (efficiency and purity) along the ROC curve

How to choose "cut"? → need to know prior probabilities (S, B abundances)

- Measurement of signal cross section: maximum of $S/\sqrt{(S+B)}$ or equiv. $\sqrt{(\varepsilon \cdot p)}$
- Discovery of a signal : maximum of $S/\sqrt{(B)}$
- Precision measurement: high purity (p)
- Trigger selection: high efficiency (ε)

universitätbonn

Toolkit for MultiVariate Analysis (TMVA)

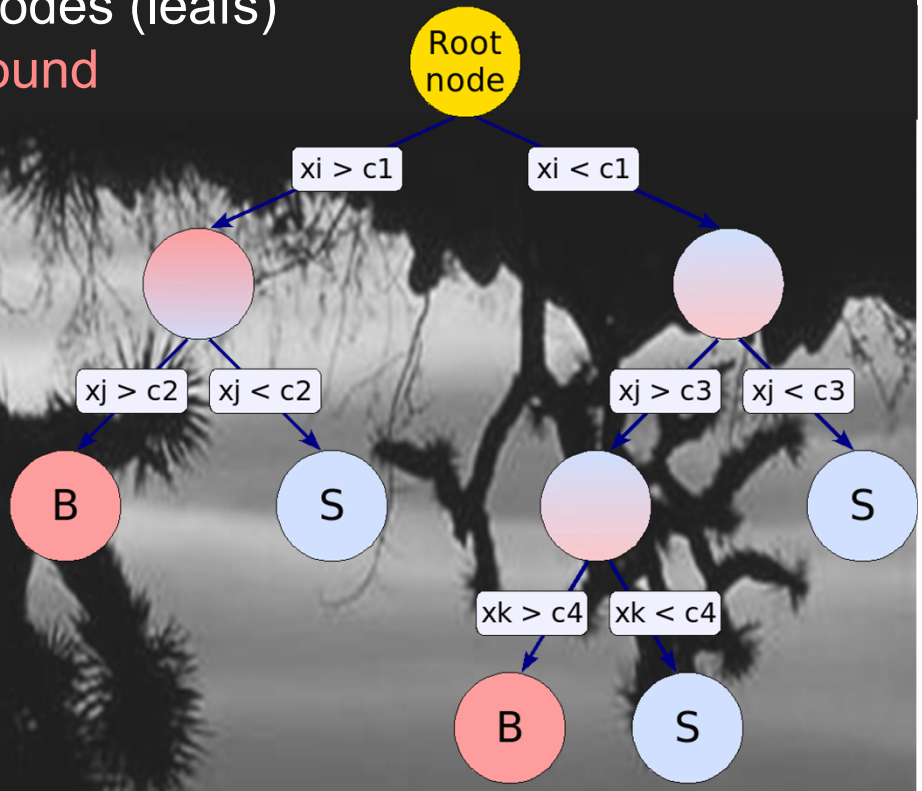
universitätbonn

What is TMVA?

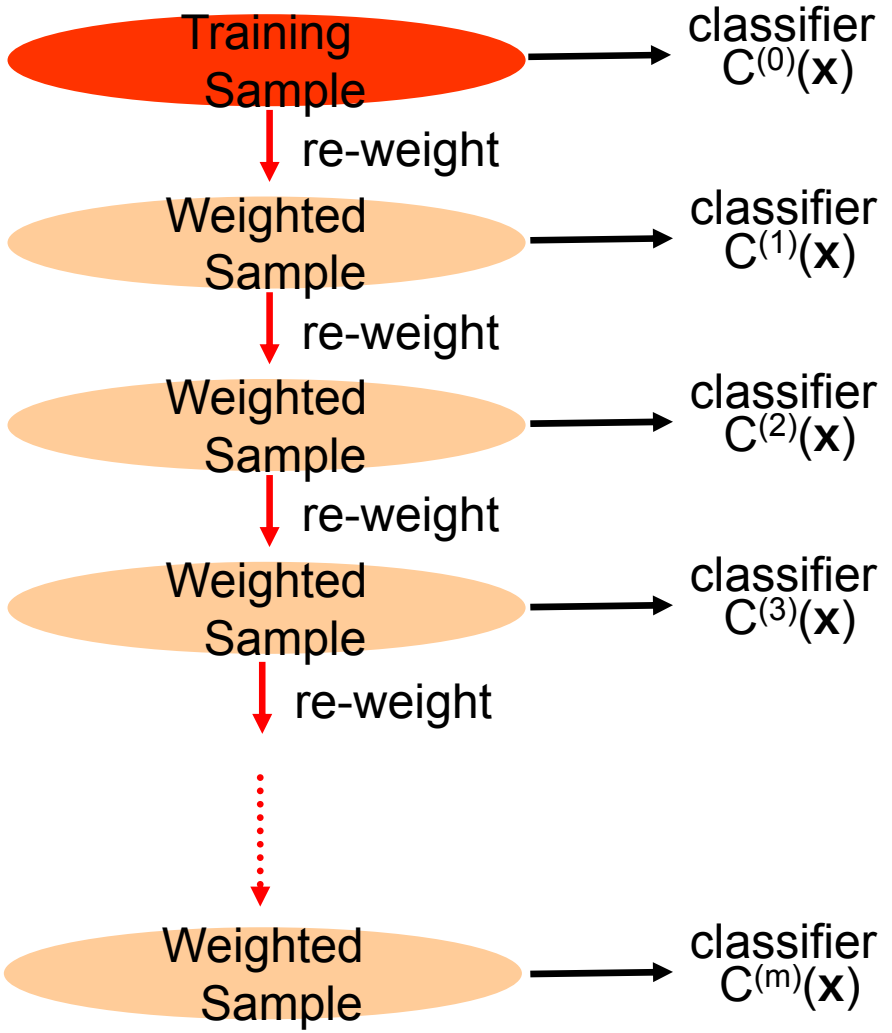
- TMVA: not just a collection of MVA methods for supervised learning, but also provides…
  - a common interface for all MVA techniques
  - a common interface for classification and regression
  - easy training and testing of all methods on the same datasets
  - a complete user analysis framework and examples
  - embedded in ROOT
  - an understandable Users Guide
  - "*TMVA - Toolkit for Multivariate Data Analysis , A. Hoecker, ..., E.v.Toerne et al., arXiv:physics/0703039v5 [physics.data-an]*

Boosted Decision Trees

Decision Tree: Sequential application of cuts splits the data into nodes, where the final nodes (leafs) classify an event as signal or background



Adaptive Boosting (AdaBoost)



AdaBoost re-weights events misclassified by previous classifier by:

$$\frac{1-f_{err}}{f_{err}} \quad \text{with :}$$

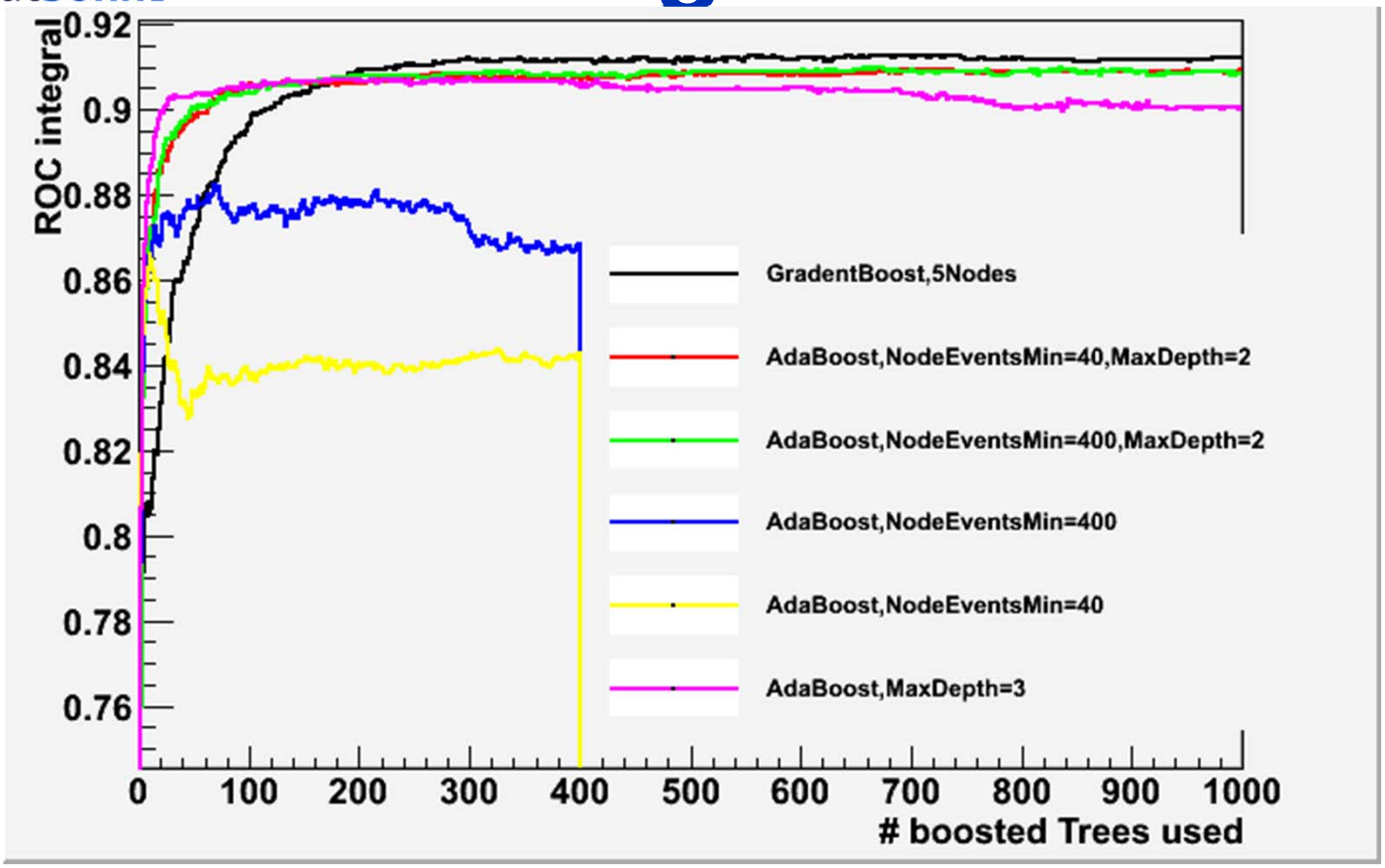
$$f_{err} = \frac{\text{misclassified events}}{\text{all events}}$$

AdaBoost weights the classifiers also using the error rate of the individual classifier according to:

$$y(x) = \sum_{i}^{N_{Classifier}} \log\left(\frac{1-f_{err}^{(i)}}{f_{err}^{(i)}}\right) C^{(i)}(x)$$

universitätbonn

Boosting at Work



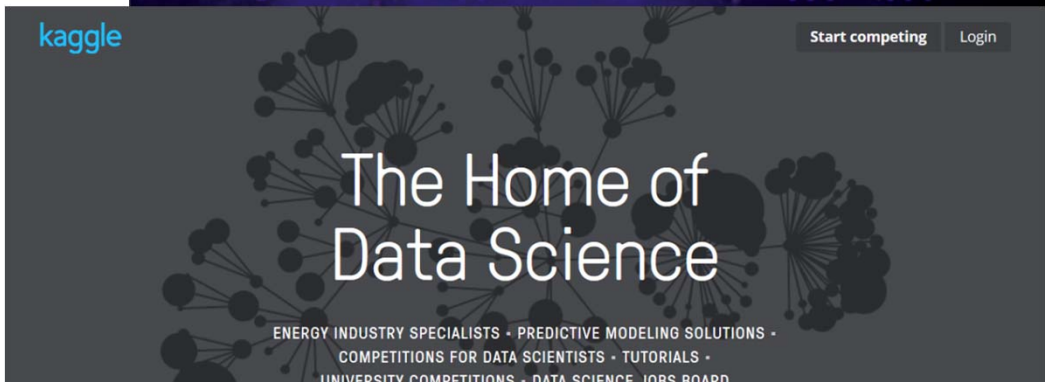
Boosting seems to work best on “weak” classifiers (i.e. small, dumb trees)
Tuning (tree building) parameter settings are important
For good out of the box performance: Large numbers of very small trees

universitätbonn

The ATLAS Higgs Machine Learning Challenge

universitätbonn

- Hosted on **kaggle**
- training data in VBF $H \rightarrow \tau\tau$ (~ 30 var)
- Publicly available: 250k training events
- 550k evaluation data with hidden truth-tag.
- Participants submit tag list for evaluation data
- Evaluation feed-back in two-step process



universitätbonn

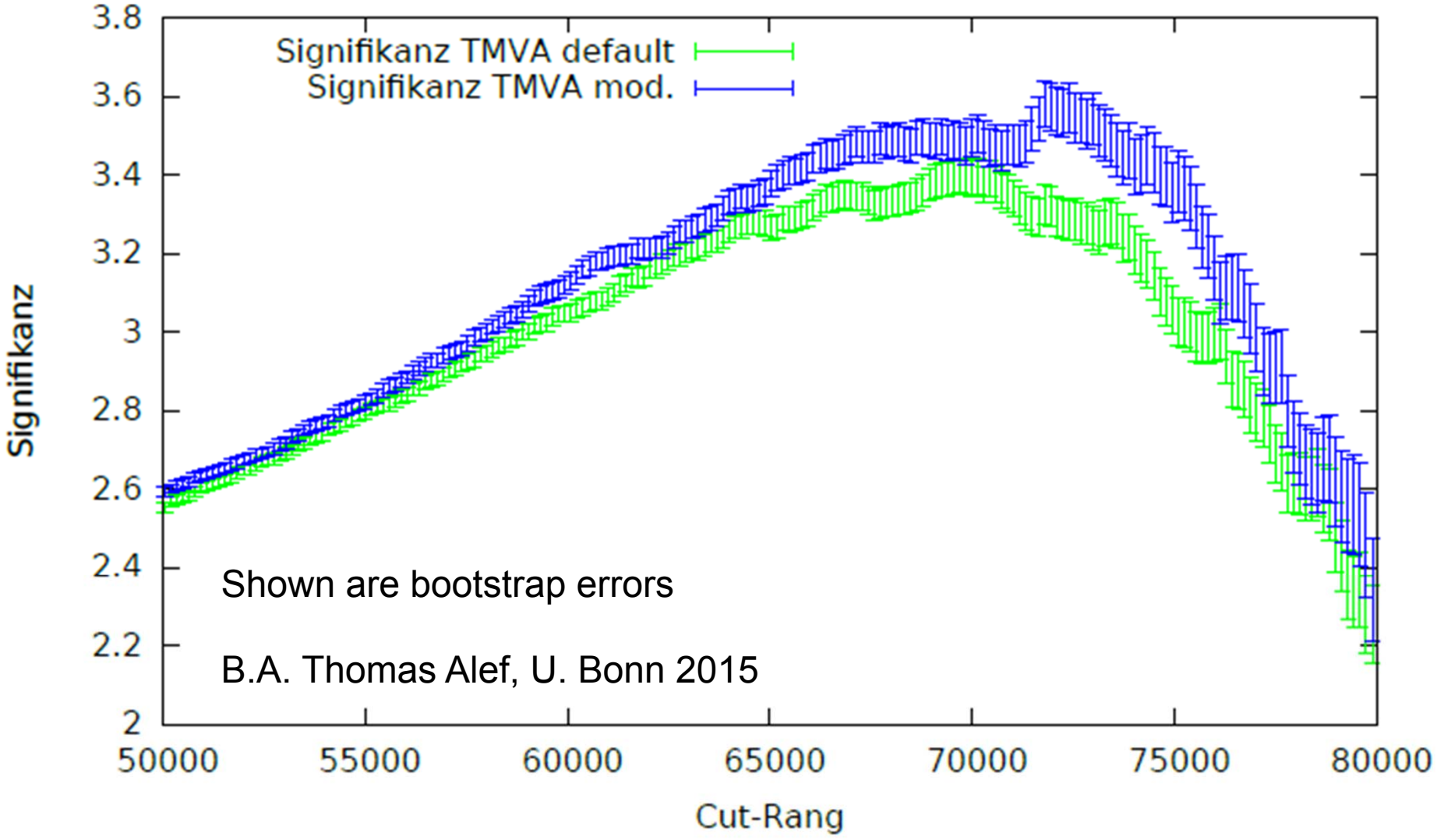
Approx. Median Significance

$$AMS = \sqrt{2\left((s+b+b_r)\cdot log\left(1+\frac{s}{b+b_r}\right)-s\right)}$$

- S: signal, b: background (in signal region)
- Figure of merit, superior to S/Sqrt(S+B), S/Sqrt(B)
- Approximate significance of profile likelihood fit
- Function of b_r: regularizes expression as b→0

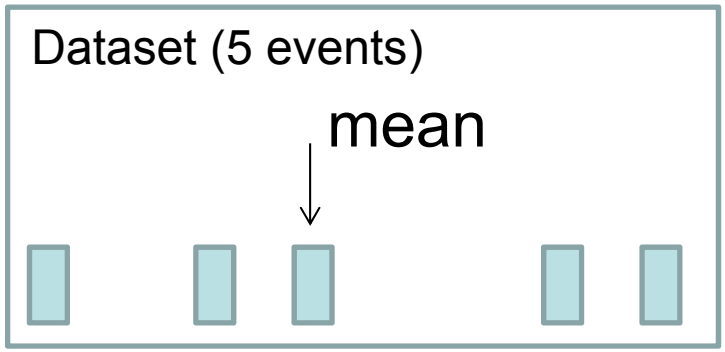
universitätbonn

TMVA performance



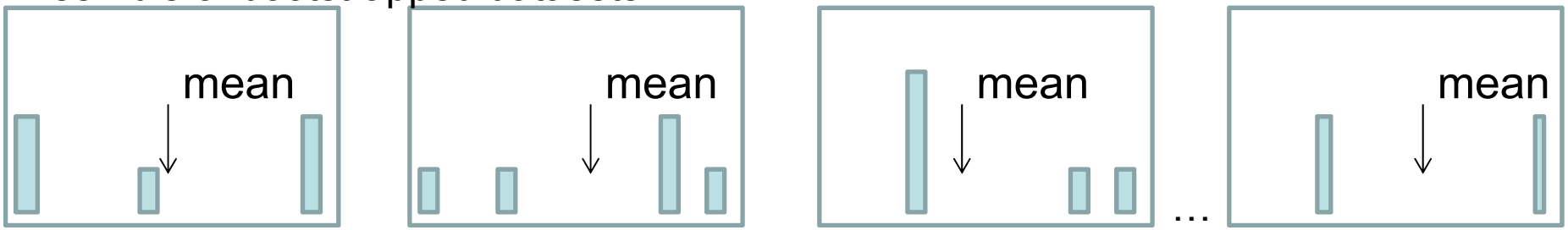
Bootstrapping example

Estimating a mean from 5 events

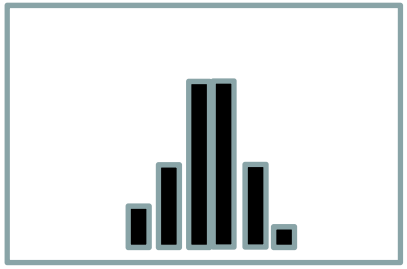


- **Draw events from your set.**
- **Permitted to draw same event several times.**
- **Draw same number as events in dataset.**

Ensemble of bootstrapped datasets



Distribution of mean values obtained from bootstrapped datasets



Standard deviation is the bootstrap error

General method bootstrapping works for any estimator

Final leaderboard

#	Δrank	Team Name ‡ model uploaded * in the money		Score	Entries	Last Submission UTC (Best – Last Submission)
1	↑1	Gábor Melis ‡ *	7000$	3.80581	110	Sun, 14 Sep 2014 09:10:04 (-0h)
2	↑1	Tim Salimans ‡ *	4000$	3.78913	57	Mon, 15 Sep 2014 23:49:02 (-40.6d)
3	↑1	nhlx5haze ‡ *	2000$	3.78682	254	Mon, 15 Sep 2014 16:50:01 (-76.3d)
4	↑38	ChoKo Team		3.77526	216	Mon, 15 Sep 2014 15:21:36 (-42.1h)
5	↑35	cheng chen		3.77384	21	Mon, 15 Sep 2014 23:29:29 (-0h)
6	↑16	quantify		3.77086	8	Mon, 15 Sep 2014 16:12:48 (-7.3h)
7	↑1	Stanislav Semenov & Co (HSE Yandex)		3.76211	68	Mon, 15 Sep 2014 20:19:03
8	↓7	Luboš Motl's team	Best physicist	3.76050	589	Mon, 15 Sep 2014 08:38:49 (-1.6h)
9	↑8	Roberto-UCIIIM		3.75864	292	Mon, 15 Sep 2014 23:44:42 (-44d)
10	↑2	Davut & Josef		3.75838	161	Mon, 15 Sep 2014 23:24:32 (-4.5d)
45	↑5	crowwork ‡	HEP meets ML award XGBoost authors Free trip to CERN	3.71885	94	Mon, 15 Sep 2014 23:45:00 (-5.1d)
782	↓149	Eckhard		3.49945	29	Mon, 15 Sep 2014 07:26:13 (-46.1h)
991	↑4	Rem.		3.20423	2	Mon, 16 Jun 2014 21:53:43 (-30.4h)
		simple TMVA boosted trees		3.19956		

Slide from D. Rousseau, “ML meets CERN“ meeting, 19 May 2015

Machine Learning Wins the Higgs Challenge

By 20th November 2014

https://www.kaggle.com/c/higgs-boson

The winner of the four-month long Higgs Machine Learning Challenge, launched on 12 May, is Gábor Melis from Hungary, followed closely by Tim Salimans from The Netherlands and Pierre Courtiol from France. They will receive cash prizes, sponsored by Paris-Saclay Centre for Data Science and Google, of $7000, $4000, and $2000 respectively. The three winners have been invited to participate at the Neural Information Processing Systems conference on 13 December in Canada.

The Special High Energy Physics meets Machine Learning Award was given to team Crowwork's Tianqi Chen and Tong He. Though their score was 3.72 to Melis' 3.81, a thorough scrutiny showed that Crowwork's algorithm was an excellent compromise between performance and simplicity, which could improve tools currently used in high-energy physics. The team has been invited to CERN next year for a workshop where they will discuss the application of machine learning techniques in high-energy physics. The Challenge, hosted by Kaggle, had the all-time record of 1,785 teams participating.

Winners of the Higgs Machine Learning Challenge: Gábor Melis and Tim Salimans (top row), Tianqi Chen and Tong He (bottom row).

"The Challenge is done but we are only really half-way through the project. We now have to digest the many ideas submitted by the participants, and establish long-term collaboration between high energy physics and machine learning communities," says David Rousseau, ATLAS physicist and organizer of the Challenge.

Lessons learned

- There are many tools out there apart from TMVA.
- BDTs did well compared to neural nets (winner Gabor Melis expressed only slight preference for NN)
- Cross validation very important to fight overtraining and statistical fluctuations
- Why challenge results are not directly transferable to ATLAS analyses
  - optimal working point with systematics differs strongly from bare w.p.
  - Things get easier with sufficient M.C. stat.

universitätbonn

Cross-validation

- Training data introduce a bias when used to evaluate performance.
- Split data into subsamples in many different ways for training and evaluation and obtain several classifiers that can be combined.
- Simple Example: 2-folded training

- Difference between classifier1 and 2 used to eval. Fluctuationbs in performance

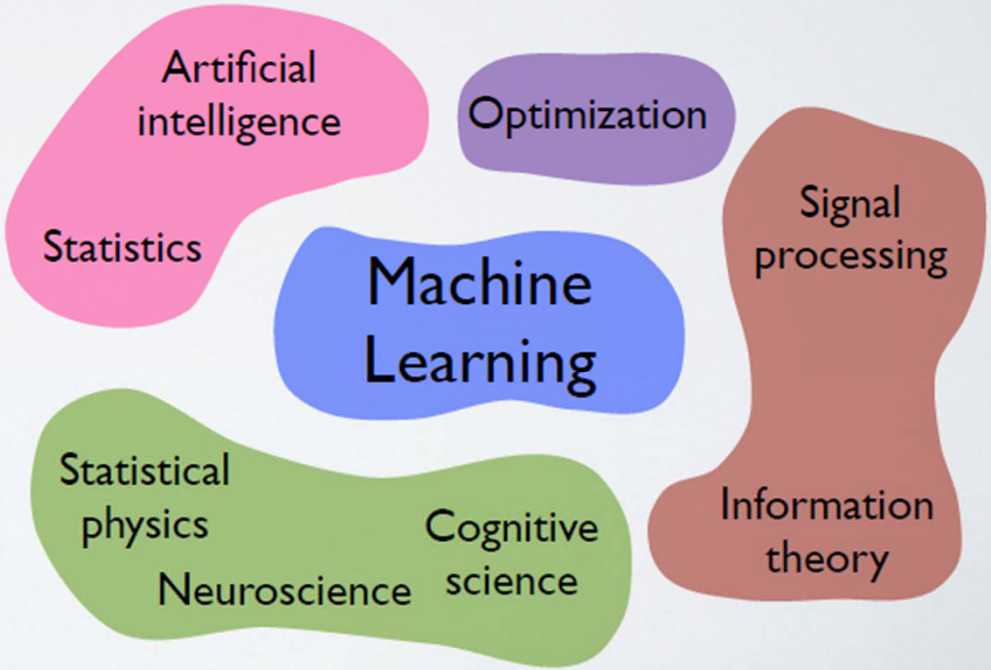
universitätbonn

Non-HEP applications of Machine Learning

test

WHAT IS MACHINE LEARNING?

- "The science of getting computers to act without being explicitly programmed" - Andrew Ng (Stanford/Coursera)
  - part of standard computer science curriculum since the 90s
  - inferring knowledge from data
  - generalizing to unseen data
  - usually no parametric model assumptions
  - emphasizing the computational challenges



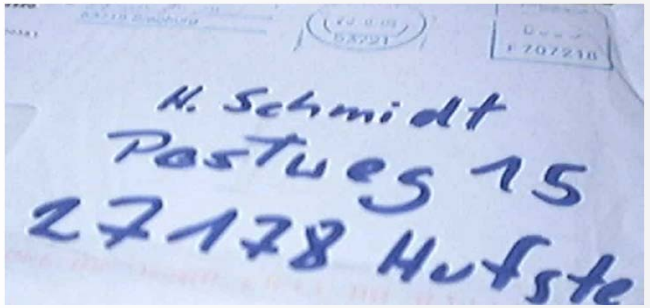
Slide from Balazs Kegl, ATLAS ML challenge

universitätbonn

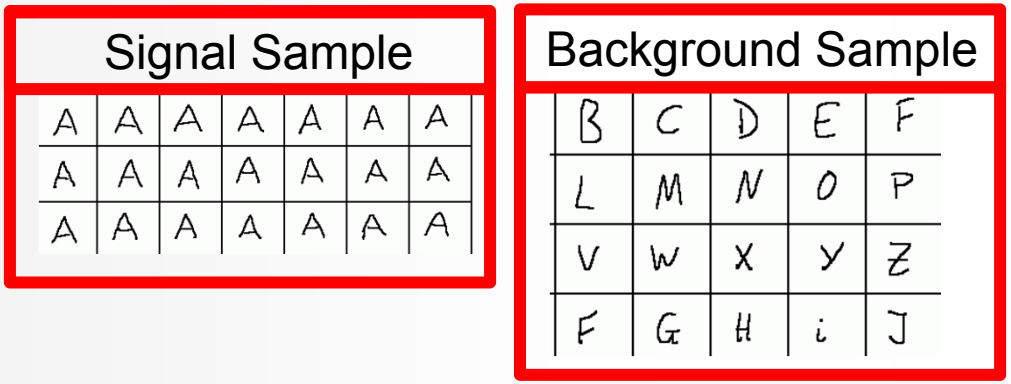
A simple non-HEP example for a multi-variate classification task

Pattern Recognition of handwritten digits

- Automatic reading of handwritten digits for Zip-code processing in mail services (Postleitzahlerkennung)



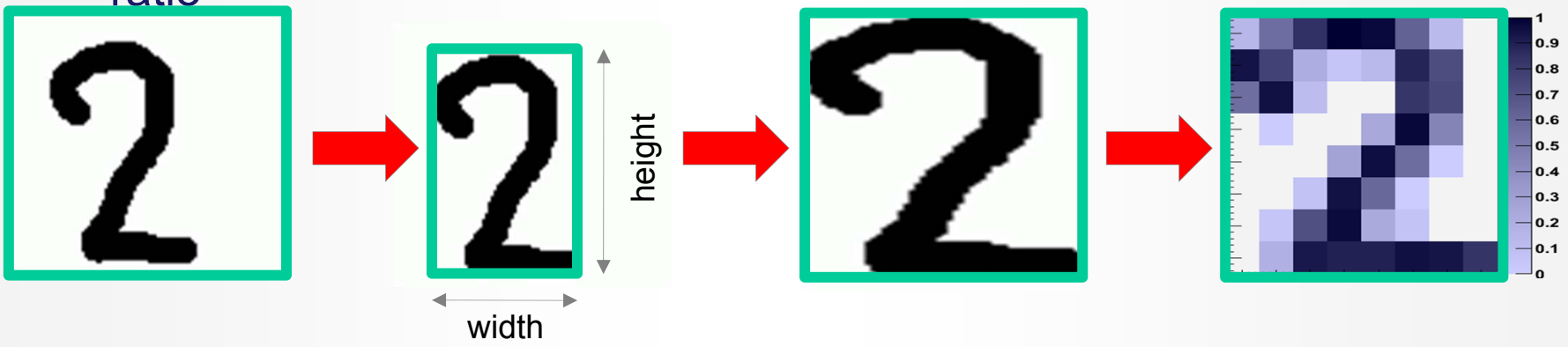
- One MVA methods for each digit: one digit is Signal, everything else is Background



- Input values: brightness of each individual pixel
  → need to reduce number of pixels
  → preprocessing necessary

Handwritten digits/letters

- Preprocessing:
  - [step 1] Find frame around digit, determine aspect ratio
  - [step 2] Transform to aspect ratio=1
  - [step 3] Merge pixels into 8x8 array
  - Input to multivariate analysis: 64 pixels plus the original aspect ratio



Original digit
~100 dpi

Step (1)

Step (2)
Aspect ratio=1

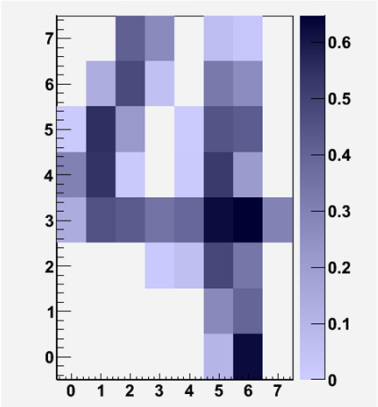
Step (3)
8x8 pixel array

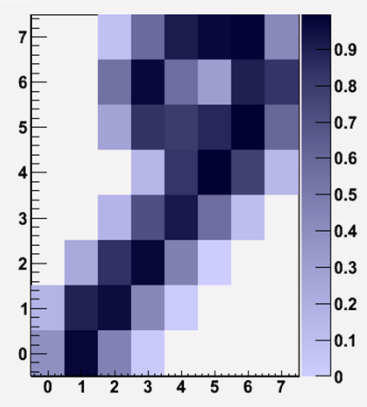
$$\text{Aspect ratio} = \frac{\text{height}}{\text{width}}$$

Pattern Recognition of handwritten digits

- Boosted Decision Trees with gradient boost (3000 trees)

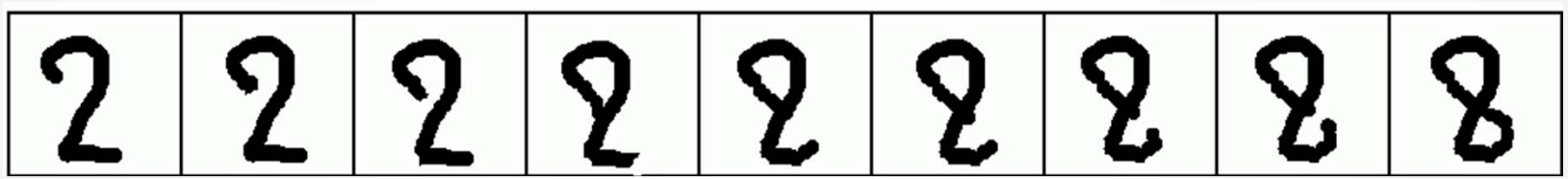
- Training: One digit is signal, all others are background
- Data sample:
  - MNIST database: 60k training digits, 10k test
  - (http://yann.lecun.com/exdb/mnist/)
  - Strict separation of test and training sample
  - persons contributing to training sample do NOT contribute to test sample (and vice versa).

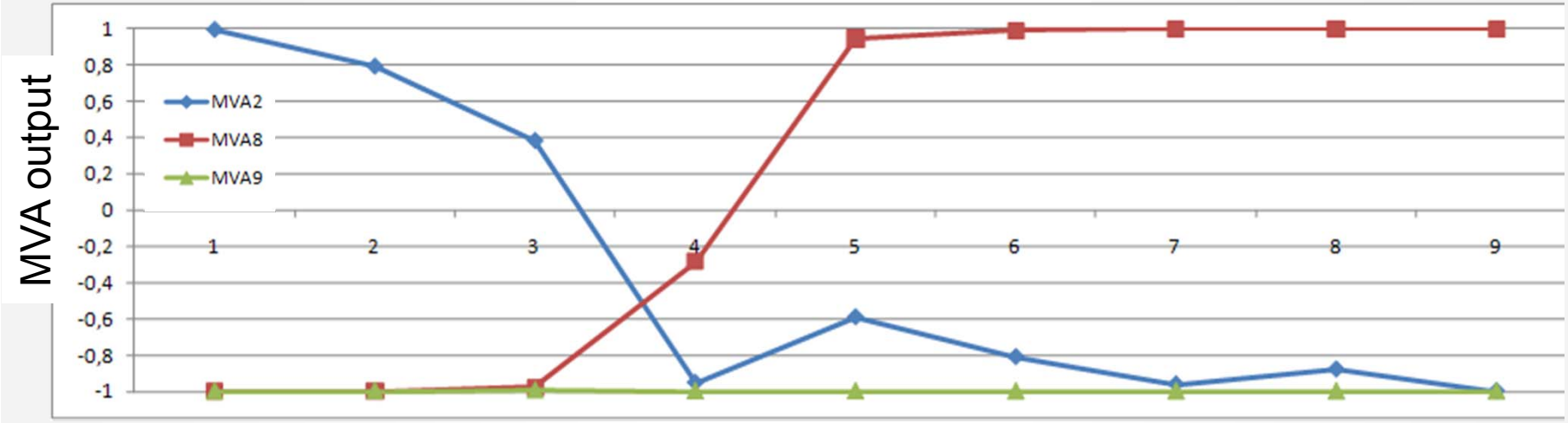




Pattern Recognition of handwritten digits

Example: stepwise morphing of “2“ into “8“





Selected as

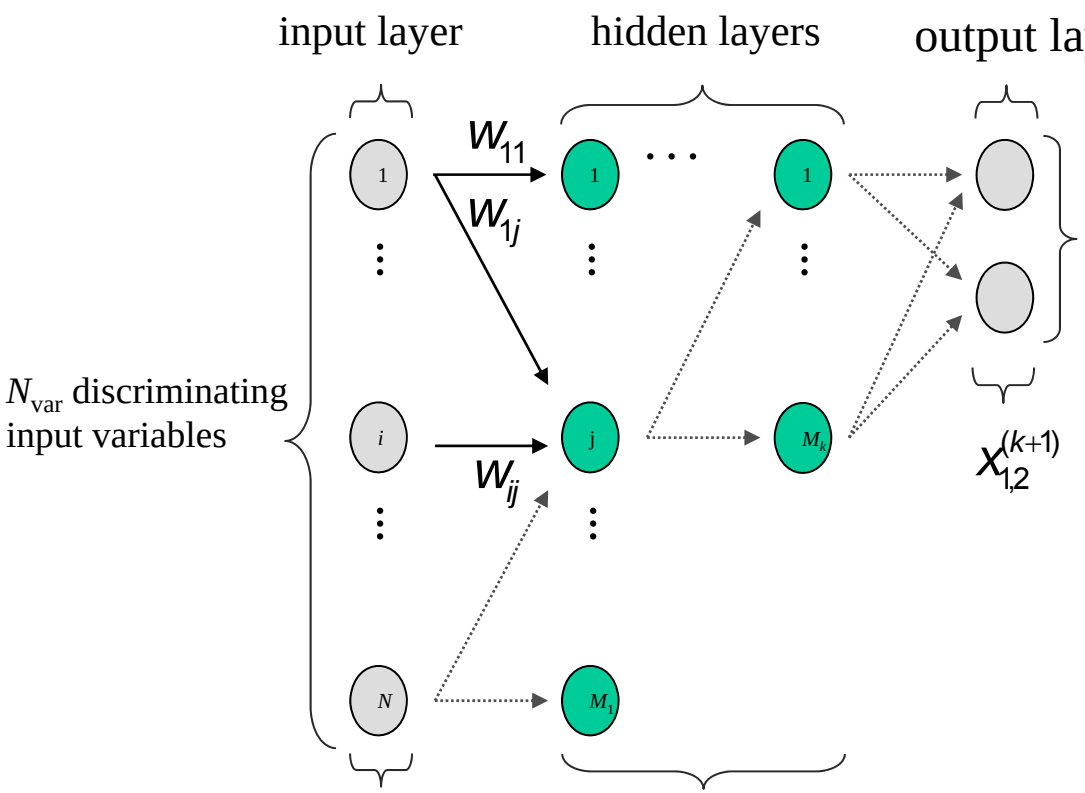
“2“ “2“ “2“ “8“ “8“ “8“ “8“ “8“ “8“

Output digit determined by MVA with largest output value

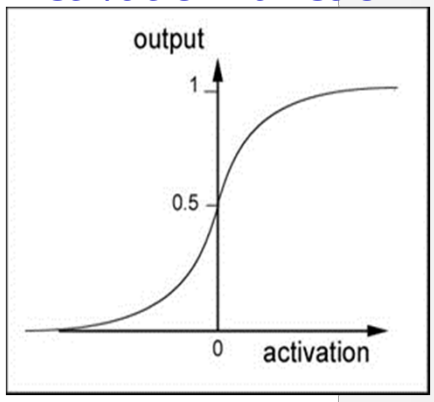
Artificial Neural Networks

- Modelling of arbitrary nonlinear functions as a nonlinear combination of simple „neuron activation functions“

Feed-forward Multilayer Perceptron



Activation function

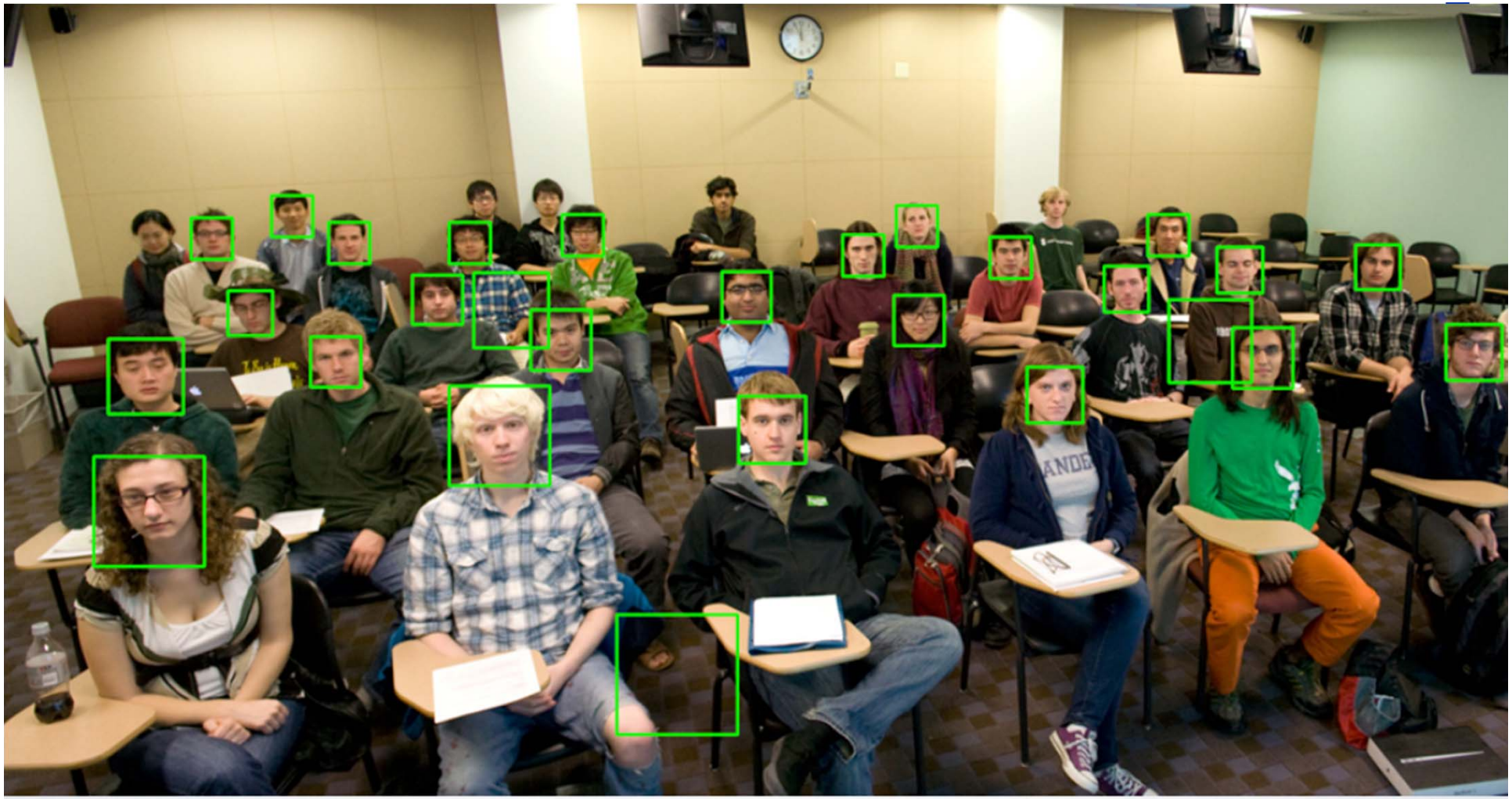


universitätbonn

Deep Neural Nets

- Neural nets in HEP work with ~2 internal layers.
- For decades deeper ($N_{internal}$ >> 2) nets would perform worse
- New ideas for training deep architectures emerged in last 10 years.
  - New training techniques to fight saturation effects
  - Feature learning (unsupervised pre-training)
  - New regularization effects to suppress noise
  - Clever combination of several nets

universitätbonn



Real time face detection