Outline

• Machine Learning in HEP
• Status and Overview
• Software and Tools
• New Directions
Machine Learning in HEP
Machine Learning

What is Machine Learning?

• Study of algorithms that improve their performance $P$ for a given task $T$ with more experience $E$

Sample tasks: identifying faces, Higgs bosons
How do LHC Experiments Use Machine Learning?
Identifying boosted objects

Mauro Donegà: Data Science @ LHC 2015

Photon Energy regression

- How to improve the corrections? Add more variables in the description:
  - difficult to model correctly the correlations
  - curse of dimensionality
- Move to a multivariate approach: BDT (Gradient Boosting)
- Use many more variables (first try O(80) then down to O(20))
- correct treatment of the correlations by the BDT.
- Basically add whatever variable makes sense to describe
  the photon
  "photon shape" variables
  photon coordinates (eta, phi)
  median energy density $\rho$ in the event

Target Variable: $E_{rec}/E_{true}$

10-30% improvement on resolution depending on the energies and region of the detector

Training sample: again single particle gun MC

(uniform energy spectrum [3-300] GeV and uniform in the detector volume ($\eta$, $\phi$))

H → $\gamma\gamma$ MC

Illustration only

Classification

- Particle identification
  - Is this particle a photon or a jet?
- Advanced Pattern Recognition
  - Clustering detector hits, jet sub-structure
- Searches for new Physics
  - Is this a Higgs/SUSY event or background

Function Estimation

- Energy/Momentum estimation
  - Better estimate using Machine Learning regression
From CMS

Mass fit

\( H \rightarrow \gamma\gamma \)

“Golden channel”
From LHCb

ML Trigger
PID
Event Selection
Dalitz Plots
From ALICE

Better
Particle ID

Higher Signal
Significance
From ATLAS

b and c jet identification

- Example: b-jet Identification
  - Compression of information: hits → tracks → jet-based quantities, e.g.
    - Displaced vertex
    - b probability of jet as product of b track probabilities

- Inputs: we currently use jet-based quantities
- Output: b-jet or not using NN & BDT: mainly TMVA, also test AGILEPack (C++ framework for deep learning designed for HEP purposes by Luke de Oliveira: https://github.com/lukedeo/AGILEPack)

- Origin
  - b and c jet identification

- Example: charm-jet Identification
  - Define 2 discriminants based on 3 NN outputs:

- Boosted objects
  - Gluon Radiation
  - Boosted Boson Type Tagging
  - Jet ETmiss
  - SLAC, Stanford University
  - March 26, 2014

- Physics meets Computer Vision: Jet Images
  - Powerful discrimination
  - Intuitive visualization to understand what physics has been learned

- The Future
  - Beyond optimizing discrimination, can we learn what ML algorithms learn about physics?

- Blurring jet clustering algorithms with Fuzzy Jets
  - IRC safe likelihood-based approach to jet clustering

- We have shown two examples:
  - Sub-jet
    - Radiation around 1st subjet in QCD jets
    - Radiation along direction between subjet in W-jets
    - No info in presence of 1st subjet
    - Hard 2nd subjet in QCD jets
    - 0.6 < Subjet $R < 0.8$
    - Radiation along direction between subjet in W-jets
    - Wide 2nd subjet in QCD jets
    - 0.5 < Subjet $R < 0.9$

- Boosted objects
ML in HEP Today

• **Large ensembles** of classifiers

• **Deep vs. shallow learning**
  – Neural networks with many more hidden layers

• **Combination of semi/un-supervised with supervised learning**

• **Bayesian approaches**
Deep Learning
Deep Learning

Computer Vision (CV) Benchmarks

First super-human result in 2015*

* Google/Microsoft 4.9% → 3.5%
Deep Learning

Deep Learning Neural Networks:

• Significant performance improvement
  – Training more complex models
    • Increased Depth
    • Enlarged Width
    • Feedback/Convolution
    • Novel activation functions
  – Effective strategies against over-fitting
    • Regularization
Deep Learning

Higgs Boson Example:

Tuning deep neural network architectures.

Hyper parameters | Choices
--- | ---
Depth | 2, 3, 4, 5, 6 layers
Hidden units per layer | 100, 200, 300, 500
Learning rate | 0.01, 0.05
Weight decay | 0, 0.00001
Pre-training | none, autoencoder, multi-task autoencoder
Input features | low-level, high-level, complete set

Best:
- 5 hidden layers
- 300 neurons per layer
- Tanh hidden units, sigmoid output
- No pre-training
- Stochastic gradient descent
- Mini batches of 100
- Exponentially-decreasing learning rate
- Momentum increasing from .5 to .99 over 200 epochs
- Weight decay = 0.00001

8% improvement

P. Baldi, et. al. 2014
Deep Learning

Jet flavors:

Jet Flavor Classification in High-Energy Physics with Deep Neural Networks

Daniel Guest*, 1 Julian Collado*, 2 Pierre Baldi, 2 Shih-Chieh Hsu, 3 Gregor Urban, 2 and Daniel Whiteson 1

1 Department of Physics and Astronomy, University of California, Irvine, CA 92697
2 Department of Computer Science, University of California, Irvine, CA 92697
3 Department of Physics, University of Washington, Seattle, WA 98195

(Dated: August 1, 2016)

Classification of jets as originating from light-flavor or for inferring the nature of particles produced in high-energy dimensionality of the data provided by the tracking detector state-of-the-art tools require expert data-reduction to conform that can be effectively managed by shallow classification networks to this task, attempting classification at several tracks. We find that the highest-level lowest-dimensionality needed for classification, that the performance of current or slightly exceeded by deep-network-based taggers using classification using only lowest-level highest-dimensionality task for deep networks, and that adding lower-level track provides a significant boost in performance compared to t

Best: High + Low level

Deep Learning

Convolutional Neural Networks:

- Began with image and sequence-based problems in computer vision
  - Images (2D)
  - CNN's learn features with simple structures
- Filters: repeatedly applied
- Unsupervised learning during first stage
- Jet images and evolution
Deep Learning

Convolutional Neural Networks:

Neural Network (NN)  Deep NN  Convolutional NN
Deep Learning

Neutrinos:

A Convolutional Neural Network Neutrino Event Classifier

A. Aurisano, et. al. 2016


aUniversity of Cincinnati,
Cincinnati, Ohio 45221, USA
bCollege of William & Mary,
Williamsburg, Virginia 23187, USA
cUniversity of Minnesota,
Minneapolis, Minnesota 55455, USA
dFermi National Accelerator Laboratory,
Batavia, Illinois 60510, USA
eIndiana University,
Bloomington, Indiana 47405, USA
HEP-ML Software

Toolkits

• TMVA: developed by and for HEP users
• External: scikit-learn, R

Deep learning: Theano, Tensorflow, Caffee, Keras, Torch…

Individual packages for specific tasks
ML Formally Meets HEP

9 - 13 November 2015, CERN

DS@LHC

Local Organising Committee
- Xabier Cid (CERN)
- Gilles Louppe (CERN)
- Michelangelo Manganaro (CERN)
- Maurizio Pierini (CERN)
- Jean-Roch Vlimant (Caltech)

Program Committee
- Kyle Cranmer (New York U)
- Cécile Germain (LRI)
- Vladimir Vava Gigorov (CERN)
- Gilles Louppe (CERN)
- Andrew Lowe (Wigner RCP)
- Maurizio Pierini (CERN)
- David Rousseau (LAL-Orsay)
- Maria Spiropulu (Caltech)
- Jean-Roch Vlimant (Caltech)
- Daniel Whitson (UC Irvine)

sponsored by
LHC Physics Center at CERN
Fermilab National Laboratory

Higgs Challenge
When High Energy Physics meets Machine Learning
May to September 2014

QCHS 2016
Inter-experimental LHC Machine Learning working group

- Exchange of HEP-ML expertise and experience among LHC experiments
- ML Forum for LHC-related development and discussions
- ML software development and maintenance
- Exchange between HEP and ML communities
- Education (Tutorials)
• Website: http://iml.web.cern.ch
• Last meeting Aug. 27 (monthly) https://indico.cern.ch/event/548789
• Discussion Forum: http://iml.web.cern.ch/content/forum
• LPCC group from 05/2016
Diana-HEP
AMVA4NewPhysics

Advanced Multi-Variate Analysis for New Physics Searches at the LHC

AMVA4NewPhysics is a Marie Skłodowska-Curie ITN Network funded by the Horizon2020 program of the European Commission.

Who we are and what we do:

Participating institutions
Supervisors
Network participants
Research goals

Documents:

CALLS FOR APPLICATIONS
Public network documents
Private area

Events:

Network meetings
Schools
Workshops
Conferences
Seminars

AMVA4NewPhysics is a consortium of 15 among Universities, research institutes, and industrial partners that have the common aim of developing advanced statistical learning tools for applications to particle physics problems as well as for industrial applications.

AMVA4NewPhysics will train a pool of early-stage researchers in particle physics and advanced statistical methods, to study data produced by the Large Hadron Collider experiments. The double goal of our studies is to search for new physics and to study the detailed properties of the Higgs boson.

On the left column of this site you may find links to documentation on our research program, our research output, and all our recruitment offers.

RECENT NEWS

August 16th -
AMVA4NewPhysics is a sponsor of the XIth Quark Confinement and the Hadron Spectrum (QCHS12) conference which takes place from 28 August - 3 September, 2016 in Thessaloniki (Greece).

August 16th -
Three new ESRs have started in the network: Anna Stokia at CERN, Cecilia Tosciari at the University of Oxford and Fabricio Jiménez at UBP.

May 6th -
The Workshop on Machine Learning for Particle Physics, 6 - 10 June, 2016, CERN is open for registration.
ML in SFT

CERN EP-SFT ML efforts led by

S. Gleyzer, L. Moneta

– Technical/Doctoral
– Masters + undergraduates
– GSoC Students

ROOT
Data Analysis Framework

TMVA
TMVA

Toolkit for Multivariate Analysis:

• HEP ML workhorse
• Easy to get started with
• ROOT integrated
• In production by LHC experiments

Next release at the end of September will include many new features
New TMVA Features
New Features

Modular Design and Interfaces
Deep Learning (CPU + GPU)
Parallelism
• Multi-threading
• Multi-processing
• Spark
• GPUs
Cross-Validation
HyperParameter Tuning
Interactive Training
Unsupervised Learning
Greater integration with Jupyter
Deep Learning

Deep Learning is expensive but possible as more and better hardware available

• CPU/APACHE SPARK clusters
• GPUs
• Clusters of GPUs
• Dedicated low-level libraries and high-level interfaces
TMVA Deep Learning

Back-propagation

Design

cpu/gpu

DNN

S. Pfroentschuh
CPU Performance

**Implementation:** Multithreaded OpenBLAS and TBB

**Hardware:** Intel Xeon E5-2650, 8 × 4 cores, 2 GHz, estimated peak performance per core: 16 GFLOP/s

---

S. Pfreundschuh
TMVA Deep Learning

GPU Performance

Optimization:

- Use compute streams to expose more parallelism to the device.
- Compute gradients for multiple batches in parallel.
- Using 2 streams:

![Graph showing numerical throughput and percentage of peak performance](image)
TMVA Deep Learning

DNN Throughput Comparison

S. Pfreundschuh
TMVA Deep Learning

Shallow vs. Deep Networks

Deep Networks vs. BDT

Performance Improvement

S. Pfreundschuh
TMVA Parallelization

Parallel Architecture

- Base
- Algorithm

Basic requirements.

Results

- #Jobs
- Time
- Etc..

Programming model.

- Mpi
- Threads
- Spark
- Etc..

O. Zapata Mesa
TMVA Parallelization

SPARK Parallelization

Node 1
- Driver Program
- SparkContext
- Master
- Worker 1
  - 4 cores

Node 2
- Worker 2
  - 4 cores

K-fold cross validation

16-fold cross validation

G. Douzas
Classifier output: Neural networks, decision trees

Simple neural network
- Python function reads the network, converts to JSON; JS with d3js make the visualization from JSON
- Interactive: focusing connections, zooming, moving

Deep neural network
- HTML5 Canvas visualization (speed)
- Less interactive: zooming, moving

Decision trees
- Ipywidgets: input field for selecting the tree
- Visualization from JSON with D3js
- Interactive: closing subtree, showing the path, focusing, moving, zooming, reset

Tutorial notebook:
http://nbviewer.jupyter.org/github/qati/GSOC16/blob/master/notebooks/ROQTbooks-TMVA-NeuralNetwork-DecisionTree-Visualization.ipynb

A. Bagoly
Trends and New Directions
Trends and Directions

- **Deep(er) learning**
- **Better regression**
  - Single- and Multi-objective
- **Unsupervised learning**
  - Standalone applications
  - as preprocessing step (input to DNN)
- **Integrated with hardware**
  - Next-Gen FPGAs, neuromorphic computing
Feedforward NNs

Convolutional NNs

Deep Belief Nets

Recurrent NNs

Recursive NNs

Deep Q Learning

Neural Turing Machines

Memory NNs
Unsupervised Features

- Transformation of raw data input to a representation that can be effectively exploited in machine learning tasks
- New framework for variable transformations in TMVA with VarTransformHandler class
Deep Autoencoders

- Deep neural network is trained to output the input i.e. learn the identity functions.
- Constrain number of units in hidden layer, thus learning compressed representation.

Abhinav Moudgil

Variance Threshold

Input Features → Calculate Variance → Compare Threshold → Selected Features
Loss Function choice important:

New Loss Function Options

A. Carnes, UF
More Information

Websites:  http://root.cern.ch
          http://iml.web.cern.ch
          http://oiproject.org
Backup
JUPYTER Notebooks

• Open source **web-based** application blends code with elements such as text, figures, links
  – Excellent integration of instructions and executable code
  – Perfect for interactive analysis and teaching demonstrations of anything that involves code
  – Can be run locally, on a server, laptop or smartphone
  • All you need is a browser

http://www.mybinder.org/repo/qati/GSOC16/notebooks/notebooks/ForSergei.ipynb
Service for Web-based Analysis:

- Browser based service (CERN)
- Jupyter Notebook interface
- Integrated with

- In the “cloud”
- Storage: eos, CERNBox

– http://swan.web.cern.ch