Advanced Analysis Methods

Tulika Bose

April 6\textsuperscript{th}, 2016
Searching for rare signals

Examples of background to $H \rightarrow ZZ$ searches

5 orders of magnitude!
Over huge backgrounds

To achieve a discovery, huge background reduction rate needed

- Example of $H \to \gamma\gamma$ : typically 9 orders of magnitude under the QCD jets background
- **Reducible background** : jet-jet, photon-jet
  - Jets can be mis-identified as photons
  => can be suppressed by tight photon identification criteria
- **Irreducible background** : photon-photon
  - Non-resonant diphoton continuum
  => Can be discriminated using kinematic properties
With experimental challenges

**Experimental challenges:**
- Detector calibration
- Identification of the tracks / energy deposits in the sub-detectors
- Particle reconstruction
- Particle identification
- Finding the vertex of hard interaction among all pile-up vertices
- Discriminate the signal process against all other background processes
- ...

- Multivariate methods can help for that

Collision with 20 pile-up events recorded with the ATLAS detector
Multi-variate methods: definitions

**MultiVariate Analysis:**
- Set of statistical analysis methods that simultaneously analyze multiple measurements (variables) on the object studied
- Variables can be dependent or correlated in various ways

**Classification / regression:**
- **Classification**: discriminant analysis to separate classes of events, given already known results on a training sample
- **Regression**: analysis which provides an output variable taking into account the correlations of the input variables

**Statistical learning:**
- **Supervised learning**: the multivariate method is trained over a sample where the result is known (e.g. Monte Carlo simulation of signal and background)
- **Unsupervised learning**: no prior knowledge is required. The algorithm will cluster events in an optimal way
Multivariate analyses in HEP

- **Signal/background discrimination**:
  - **Object reconstruction**: discriminate against instrumental background (electronic noise...)
  - **Object identification**: e.g. electron, bottom quark identification, to improve the rejection of other objects resembling (e.g. jets)
  - **Discriminating physics process against physics backgrounds**: Many examples, e.g. single top against W+jets, H->WW against WW background...

- **Improving the energy measurement**, via regression. Allows to narrow the reconstructed mass peak, improve the resolution.

- **Estimate the sensitivity of the analysis**:
  - **Sensitivity to signal exclusion or discoveries**: Likelihood of the data to be consistent with background only or signal+background hypothesis
  - **Combination** of many channels

=> exclusion limits or discoveries
MVA example from Tevatron

Single top discovery

- When published, very controversial
- 36 boosted decision trees used to discriminate signal from background
- First measurement of the single top cross-section, today well established

PhysRevLett.98.181802
No turning back!

• Over the past ten years, Multivariate analysis (MVA) methods gained gradual acceptance in HEP.
• In fact, they are now “state of the art”
• Some of the most important physics results in HEP have come from the use MVA methods.
MVA use in Higgs Discovery

- MVA used in every possible analysis aspect
  - Electrons/photons ID
  - MVA for EM cluster energy corrections
  - Vertex identification (diphotons)
  - b-tagging
  - S/B discrimination in all channels
    - $\gamma\gamma$, $ZZ \rightarrow 4\ell$, ($WW$, $bb$, $\tau\tau$)
Event Analysis Techniques

• Examples:

• Characteristics:
  – Level of complexity and transparency
  – Performance in term of background rejection
  – Way of dealing with non-linear correlations
  – Robustness while increasing the number of input variables
  …
“Rectangular” cuts

- Simplest multivariate method, very intuitive
  - All HEP analyses are using rectangular cuts, not always completely optimized

- Rectangular cuts optimization:
  - e.g. Grid search,

- Characteristics:
  - Difficult to discriminate signal from background if non-linear correlations
  - Optimization difficult to handle with high number of variables

Define the signal region:

\[
\begin{align*}
& a_1 < x_1 < a_2, \\
& b_1 < x_2 < b_2
\end{align*}
\]
Cut-based analysis

- Lepton $p_T > 41$ GeV
- $M(\text{top}) < 352$ GeV
- Event Energy $< 65$ GeV

In the final event set:
- Estimate background yield
- Compare to data
  \[ N_{\text{obs}} = N_{\text{data}} - N_B \]
- Calculate signal acceptance
  \[ \sigma = N_{\text{obs}} / (A \times L) \]
Decision Trees

- Machine-learning technique, widely used in the social sciences
- Idea: recover events that fail criteria in cut-based analysis
Including events that fail a cut

- Create a tree of cuts
- Divide sample into “pass” and “fail” sets
- Each node corresponds to a cut (branch)

Start at first “node” with “training sample” of 1/3 of all signal and background events
- For each variable, find splitting value with best separation between two children (mostly signal in one, mostly background in the other)
- Select variable and splitting value with best separation to produce two “branches” with corresponding events, (F)ailed and (P)assed cut
Repeat recursively on each node
Stop (terminate at leaf) when improvement stops or when too few events left

Trees and leafs

- Create a tree of cuts
- Divide sample into “pass” and “fail” sets
- Each node corresponds to a cut (branch)
- A leaf corresponds to an end-point
- For each leaf, calculate purity (from MC):
  purity = \( \frac{N_S}{N_S+N_B} \)
Decision tree output

- Train on signal and background models (MC)
  - Stop and create leaf when $N_{MC} < 100$
- Compute purity value for each leaf
- Send data events through tree
  - Assign purity value corresponding to the leaf to the event
- Result approximates a probability density distribution

Decision tree output for each event = leaf purity
Closer to 1 for signal and closer to 0 for background
Measure and Apply

- Take trained tree and run on independent simulated sample, determine purities.
- Apply to Data
- Should see enhanced separation (signal right, background left)
- Could cut on output and measure, or use whole distribution to measure.

- Cut on BDT yields signal enriched sample
- Allows to study top quark properties
Decision Tree Verification

- Use “mystery” ensembles with many different signal assumptions
- Measure signal cross section using decision tree outputs
- Compare measured cross sections to input ones
- Observe linear relation close to unit slope

\[ \chi^2/\text{ndof} = 4.89/4 \]
\[ \text{Slope} = 1.07 \pm 0.03 \]
\[ \text{Intercept} = -0.12 \pm 0.10 \]
Object Kinematics

\( p_T(jet1) \)
\( p_T(jet2) \)
\( p_T(jet3) \)
\( p_T(jet4) \)
\( p_T(best1) \)
\( p_T(notbest1) \)
\( p_T(notbest2) \)
\( p_T(tag1) \)
\( p_T(untag1) \)
\( p_T(untag2) \)

Angular Correlations

\( \Delta R(jet1,jet2) \)
\( \cos(best1,lepton)_{besttop} \)
\( \cos(best1,notbest1)_{besttop} \)
\( \cos(tag1,alljets)_{alljets} \)
\( \cos(tag1,lepton)_{btaggedtop} \)
\( \cos(jet1,alljets)_{alljets} \)
\( \cos(jet1,lepton)_{btaggedtop} \)
\( \cos(jet2,alljets)_{alljets} \)
\( \cos(jet2,lepton)_{btaggedtop} \)
\( \cos(lepton,Q(lepton) \times z)_{besttop} \)
\( \cos(lepton,besttopframe)_{besttopCMframe} \)
\( \cos(lepton,btaggedtopframe)_{btaggedtopCMframe} \)
\( \cos(notbest,alljets)_{alljets} \)
\( \cos(notbest,lepton)_{besttop} \)
\( \cos(untag1,alljets)_{alljets} \)
\( \cos(untag1,lepton)_{btaggedtop} \)

Event Kinematics

Aplanarity(alljets, \( W \))
\( M(W,best1) \) (“best” top mass)
\( M(W,tag1) \) (“b-tagged” top mass)
\( H_T(alljets) \)
\( H_T(alljets - best1) \)
\( H_T(alljets - tag1) \)
\( H_T(alljets,W) \)
\( H_T(jet1,jet2) \)
\( H_T(jet1,jet2,W) \)
\( M(alljets) \)
\( M(alljets - best1) \)
\( M(alljets - tag1) \)
\( M(jet1,jet2) \)
\( M(jet1,jet2,W) \)
\( M_T(jet1,jet2) \)
\( M_T(W) \)
Missing \( E_T \)
\( p_T(alljets - best1) \)
\( p_T(alljets - tag1) \)
\( p_T(jet1,jet2) \)
\( Q(lepton) \times \eta(untag1) \)
\( \sqrt{S} \)
Sphericity(alljets, \( W \))

- Adding variables does not degrade performance
- Tested shorter lists, lose some sensitivity
- Same list used for all channels
Random Forest

- Random Forests is an ensemble method that combines different trees
- Final output is determined by the majority vote of all trees

A Random Forest combines the votes of all trees
Random forest

• Average over many decision trees
  – Typically O(100)
• Each tree is grown using m variables
  – For N total variables, m<<N
• Very fast algorithm
  – Even with large number of variables
• Very few parameters to adjust
  – Typically only m
Random Forest

- Random Forests is an ensemble method that combines different trees
- Final output is determined by the majority vote of all trees
- The idea is, that a sum of weak learners results in a stronger learner

Simple example:
- 3 different trees which are uncorrelated and are correct in 60% of cases
- In order to correctly classify an event, only $\frac{2}{3}$ trees have to be correct. That means, the misclassification probability is either 3 wrong or $\frac{2}{3}$ wrong:
  - $P = \binom{3}{2} \times 0.4^2 \times 0.6 + \binom{3}{3} \times 0.4^3 \times 0.6^0 = 0.352$
- Therefore the ensemble of trees is better than only one tree even though their separation power is the same (if uncorrelated)
Boosted Decision Trees

**Boosting**
- Recent technique to improve performance of a weak classifier
- Recently used on DTs by GLAST and MiniBooNE
- Basic principal on DT:
  - train a tree $T_k$
  - $T_{k+1} = \text{modify}(T_k)$

**AdaBoost algorithm**
- Adaptive boosting
- Check which events are misclassified by $T_k$
- Derive tree weight $\alpha_k$
- Increase weight of misclassified events
- Train again to build $T_{k+1}$
- Boosted result of event $i$:
  $$T(i) = \sum_{n=1}^{N_{\text{tree}}} \alpha_k T_k(i)$$
Neural Networks

- The activity of the input units represents the raw info that is fed into the network.
- The activity of each hidden unit is determined by the activities of the input units and the weights on the connections between the input and the hidden units.
- The behavior of the output units depends on the activity of the hidden units and the weights between the hidden and output units.
Neural networks

**Input Nodes:** One for each variable $x_i$

- $M_T$ (jet1,jet2)
- $M$ (alljets)
- $p_T$ (jet1,jet2)
- $p_T$ (notbest2)
- $p_T$ (notbest1)
- $\cos(l,Q(l)x_z)$ _besttop_
- $M$ (W,best)
- $M$ (W,tag1)
- $\Delta R$ (jet1,jet2)
- $\sqrt{s}$
- $p_T$ (tag1)

**Output Node:** linear combination of hidden nodes

$$f(\vec{x}) = \sum w'_k n_k(\vec{x}, \vec{w}_k)$$

**Hidden Nodes:** Each is a sigmoid dependent on the input variables

$$n_k(\vec{x}, \vec{w}_k) = \frac{1}{1 + e^{-\sum w_{ik} x_i}}$$
What method is best?

• The “no free lunch” theorem tells you that there is no one method that is superior to all others for all problems.

• In general, one can expect neural networks (NN), Boosted decision trees (BDT) and random forests (RF) to provide excellent performance over a wide range of problems.
The Buzz about Deep Learning

• A lot of excitement about “Deep Learning” Neural Networks (DNN) in the Machine Learning community
  – Spreading to other areas!
  – Some studies already in HEP!

• Multiple non-linear hidden layers to learn very complicated input-output relationships

• Huge benefits in applications in computer vision (image processing/ID), speech recognition and language processing
Deep Learning

Use raw data inputs instead of derived “intelligent” variables (or use both)

Final learning better than shallow networks, particularly when inputs are unprocessed raw variables!

However, need a lot of processing power (implement in GPUs) time….
Deep Neural Networks for HEP

- Baldi, Padowski, Whiteson [arXiv:1402.4735v2]
- Studied two benchmark processes
  - Charged Higgs vs ttbar events
  - SUSY: Chargino pairs vs WW events into dilepton+MET final state

Significant improvement in Higgs case, not so dramatic in case of SUSY

<table>
<thead>
<tr>
<th>Technique</th>
<th>Low-level</th>
<th>High-level</th>
<th>Complete</th>
</tr>
</thead>
<tbody>
<tr>
<td>NN</td>
<td>2.5σ</td>
<td>3.1σ</td>
<td>3.7σ</td>
</tr>
<tr>
<td>DN</td>
<td>4.9σ</td>
<td>3.6σ</td>
<td>5.0σ</td>
</tr>
</tbody>
</table>

SUSY Study

<table>
<thead>
<tr>
<th>Technique</th>
<th>Low-level</th>
<th>High-level</th>
<th>Complete</th>
</tr>
</thead>
<tbody>
<tr>
<td>NN</td>
<td>6.5σ</td>
<td>6.2σ</td>
<td>6.9σ</td>
</tr>
<tr>
<td>DN</td>
<td>7.5σ</td>
<td>7.3σ</td>
<td>7.6σ</td>
</tr>
</tbody>
</table>
Unsupervised Learning

• The most common approach is to find clusters or hidden patterns or groupings in data


http://chem-eng.utoronto.ca/~datamining/Presentations/SOM.pdf

• We have not tapped these methods for identifying unknown components in data, unsupervised classification, for exploratory data analysis

• Could be useful in applications for topological pattern recognition
  – Use in Jet-substructure, boosted jet ID
Summary

• Multivariate methods brought a paradigm shift in HEP analysis ~10 years ago. Now they are state of the art.
• Applications of new ideas/algorithms such as deep learning should be explored, but the resources involved may not justify the use in every case.
• Well established techniques of the past – neural networks, Boosted Decision Trees will continue to be the ubiquitous general purpose MVA methods.
Extra
Resources

- PhyStat code repository
  https://plone4.fnal.gov:4430/P0/phystat/

- PhyStat 2007 conference
  http://phystat-lhc.web.cern.ch/phystat-lhc/

- Jim Linnemann's collection of statistics links:
  http://www.pa.msu.edu/people/linnemann/stat_resources.html

- Statistical analysis tool R
  http://www.r-project.org/

- TMVA (multivariate analysis tools in root)
  http://tmva.sourceforge.net/

- Neural Networks in Hardware
  http://neuralnets.web.cern.ch/NeuralNets/nnwInHep.html

- Boosted Decision Trees in MiniBoone
  http://arxiv.org/abs/physics/0508045

- Decision Tree Introduction
  http://www.statsoft.com/textbook/stcart.html

- GLAST Decision Trees
  http://scipp.ucsc.edu/~atwood/Talks20Given/CPAforGLAST.ppt
Summary

- Neural networks, simple decision trees, etc.
- Cut-based or likelihood
- Random guess
- Boosted decision trees, bayesian neural networks, random forests