Multivariate Machine Learning Methods: New Developments and Applications in HEP

Pushpa Bhat
Fermilab

August 6, 2015
No turning back!

- Over the past 25 years, Multivariate analysis (MVA) methods have gained gradual acceptance in HEP.
- In fact, they are now “state of the art”
- Some of the most important physics results in HEP, in the past two decades, have come from the use MVA methods.
- In 1990’s, I’d have on my title slide  
  “We are riding the wave of the future”
- That future is here, and MVA methods are here to stay!
Important Physics Results

- From top quark to the Higgs
  ... and many smart applications in object ID, energy corrections, as well
- Top-antitop event selection optimization – 1990-95 (D0)
- Top quark mass measurement -- 1996-97
- Top cross section measurements in all channels (1995 - )
- Top observation in the all-jets channel (D0) (1999)
- New particle/physics searches (1997 - )
- Observation of single top quark production (2009)
- Evidence for Higgs→ bb at the Tevatron (2012)
- Higgs Discovery at the LHC in 2012
MVA for Top quark in the mid-90’s

**DØ Lepton+jets**

The Discriminants

- **LB**: Low-bias maximum likelihood
- **NN**: Neural Networks

**Top Quark Mass Measurement**

- $m_t = 173.3 \pm 5.6\text{(stat.)} \pm 6.2\text{ (syst.) GeV/c}^2$

**Fit performed in 2-D:**

$(D_{LB/NN}, m_{fit})$

**Observation in all-jets channel**

- 31 Events (5 tagged)
MVA use in Higgs Discovery

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

- Classification
  - Object ID with high efficiency and low fake rates
    - Identification of electrons, photons, taus, b-quark jets, ..
  - signal/background discrimination
- Parameter Estimation
  - Measurement of quantities; observables ↔ parameters
- Function fitting
  - Energy correction functions, tag-rate functions, ...

Mathematically, all of these are Functional Approximation problems.
Classification

- In classification, the function to be approximated is

\[ f(x) = p(S | x) = \frac{p(x | S)p(S)}{p(x | S)p(S) + p(x | B)p(B)} \]

where \( S \) and \( B \) denote signal and background, respectively.

- In practice, it is sufficient to approximate the discriminant

\[ D(x) = \frac{p(x | S)}{p(x | S) + p(x | B)} \]

because \( D(x) \) and \( p(S | x) \) are related one-to-one:

\[ p(S | x) = \frac{D(x)}{D(x) + [1 - D(x)]/A} \]

where \( A = p(S) / p(B) \) is the prior signal to background ratio.
Multivariate Methods

A list of popular methods

- Random Grid Search
- Linear Discriminants
- Quadratic Discriminants
- Support Vector Machines
- Naïve Bayes (Likelihood Discriminant)
- Kernel Density Estimation
- Neural Networks
- Bayesian Neural Networks
- Decision Trees
- Random Forests
- Genetic Algorithms
Machine Learning

- Paradigm for automated learning from data, using computer algorithms
  - Has origins in the pursuit of artificial intelligence starting ~1960

- Requiring little *a priori* information about the function to be learned

- A method that can approximate a continuous non-linear function to arbitrary accuracy is called a *universal approximator*
  - e.g. Neural Networks
Machine Learning Approaches

- **Supervised Learning**
  - Supervised learning with a training data set containing feature variables (inputs) and target to be learned: \( \{y, x\} \)

- **Unsupervised Learning**
  - No targets provided during training.
  - Algorithm finds associations among inputs.

- **Reinforcement Learning**
  - Correct outputs are rewarded, incorrect ones penalized.
“Rectangular” Cuts

Regular Grid search

Signal eff. Vs bkgd. eff

Random Grid search (RGS)

RGS can serve as a benchmark for comparisons of efficacy of variables, variable combinations, and classifiers.

H.B.Proper, P.Bhat, et al. CHEP’ 95

Find “best” cuts

$N_{\text{tot}} = \# \text{ events before cuts}$

$N_{\text{cut}} = \# \text{ events after cuts}$

Fraction $= \frac{N_{\text{cut}}}{N_{\text{tot}}}$

ROC
The output of a neural network can approximate the Bayesian posterior probability $p(s \mid x)$:

$$f(x,w) = g\left(\sum_j w_j h_j + \theta\right) = p(s \mid x)$$

where

$$h_j = g\left(\sum_i w_{ij} x_i + \theta_i\right);$$

$$g(a) = \frac{1}{1 + e^{-a}}$$

Flexible, non-linear model
Random Grid Search for “cut” optimization

- The best “cut-based” analysis you can do!
- Notice that NN can provide significant gains even in this simple 2D analysis, at lower backgrounds which is the region of interest

A simple illustration of MVA

NN (or any other fully multivariate technique) can provide discrimination close to the Bayes limit.

Bayesian Neural Networks

- Instead of attempting to find a single “best” network, i.e., a single “best” set of network parameters (weights), with Bayesian training we get a posterior density for the network weights, \( p(w | T) \), \( T \equiv \) Training data.

- The idea here is to assign a probability density to each point \( w \) in the parameter space of the neural network. Then one takes a weighted average over all points, i.e., over all possible networks.

\[
\tilde{y}(x) = \int f(x, w) p(w | T) dw
\]

- Advantages:
  - Less likely to be affected by “over training”
  - No need to limit the number of hidden nodes
  - Good results with small training sample


PUSHPA BHAT

DPF2015
Boosting Decision Tree (BDT)

- A Decision Tree (DT) recursively partitions feature space into regions or bins with edges aligned with the axes of the feature space.

- A response value is attached to each bin, $D(x) = s/(s+b)$

$$y(x) = \sum_{m=1}^{M} \alpha_m y_m(x, w_m)$$

$$\alpha_m = \ln \left[ \frac{1 - \varepsilon_m}{\varepsilon_m} \right]$$

Boosting:
Make a sequence of $M$ classifiers (DTs) that successively handle “harder” events and take a weighted average $\Rightarrow$ BDT
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 Bayesian neural networks (BNN), Boosted decision trees (BDT) and random forests (RF) to provide excellent performance over a wide range of problems.

- BDT is popular because of robustness, noise resistance (and psychological comfort!)
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 NN

- Use raw data inputs instead of derived “intelligent” variables (or use both)
  - Pre-processing or feature extraction in the DNN
- Pre-train initial hidden layers with unsupervised learning
- Multi-scale Feature Learning
  - Each high-level layer learns increasingly higher-level features in the data
- Final learning better than shallow networks, particularly when inputs are unprocessed raw variables!
- However, need a lot of processing power (implement in GPUs, time (and training examples)
Deep Learning

“Dropout” algorithm to avoid overfitting (pruning)

Pushpa Bhat
DPF2015
- Baldi, Padowski, Whiteson \(\text{arXiv:1402.4735v2}\)
- Studied two benchmark processes
  - Charged Higgs vs ttbar events
  - SUSY: Chargino pairs vs WW events into dilepton+MET final state

Exotic Higgs

<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>2.5(\sigma)</td>
<td>3.1(\sigma)</td>
<td>3.7(\sigma)</td>
</tr>
<tr>
<td>DN</td>
<td>4.9(\sigma)</td>
<td>3.6(\sigma)</td>
<td>5.0(\sigma)</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>6.5(\sigma)</td>
<td>6.2(\sigma)</td>
<td>6.9(\sigma)</td>
</tr>
<tr>
<td>DN</td>
<td>7.5(\sigma)</td>
<td>7.3(\sigma)</td>
<td>7.6(\sigma)</td>
</tr>
</tbody>
</table>

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

- The most common approach is to find clusters or hidden patterns or groupings in data
- Common and useful methods
  - K-Means clustering
  - Gaussian mixture models
  - Self-organizing maps (SOM)
- 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

http://chem-eng.utoronto.ca/~datamining/Presentations/SOM.pdf
Challenges in LHC Run 2 and beyond

- Challenges:
  - Pile-Up mitigation!
    - $<PU> \sim 40$ in Run2
      - Associating tracks to correct vertices
      - Correcting jet energies, MET, suppressing fake “pileup” jets,
      - Lepton and photon isolation
  - Boosted Objects
    - Complicates Object ID
    - W, Z, Higgs, top taggers!
      - Provides new opportunities
      - Use jet substructure
  - High energy Lepton ID
  - Signals of BSM could be very small
    - Small MET in SUSY signatures (compressed, stealth,...)
- Need new algorithms, approaches for reco and analysis
- New ideas in triggering and data acquisition
Summary

- Multivariate methods brought a paradigm shift in HEP analysis ~20 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.
- Revived emphasis on unsupervised learning is good and should be exploited in HEP.
- Well established techniques of the past – single hidden layer neural networks, Bayesian neural networks, Boosted Decision Trees should continue to be the ubiquitous general purpose MVA methods.
Extra slides
Optimal Discrimination

- More dimensions can help!
- One dimensional distributions are marginalized distributions of multivariate density.
- \( f(x_1) = \int g(x_1, x_2, x_3, \ldots) dx_2 dx_3 \ldots \)

Optimality criterion: minimize the error rate:

\[ \text{min } \alpha + \beta \]

Minimize the total misclassification error

\[ \alpha = \int_{-\infty}^{\chi_{\text{cut}}} p(x, B) \, dx \]
\[ \beta = \int_{-\infty}^{\chi_{\text{cut}}} p(x, S) \, dx \]

Significance level \( 1 - \beta \): Power
Minimizing Loss of Information .. And Risk

- General Approach to functional approximation
- Minimize Loss function:
  \[ L\{y, f(x, w)\} \]
- It is more robust to minimize average loss over all predictions
  \[ R(w) = \frac{1}{N} \sum_{i=1}^{N} L\{y_i, f(x_i, w)\}. \]

A common Risk function
  \[ R(w) = E(w) = \frac{1}{N} \sum_{i=1}^{N} (y_i - f(x_i, w))^2 \]

or a cost (or error) function: \( C(w) = R(w) + \lambda Q(w) \)
- There are many approaches/methods
Calculating the Discriminant

\[ p(S \mid x) = \frac{p(x \mid S)p(S)}{p(x \mid S)p(S) + p(x \mid B)p(B)} \]

\[ D(x) = \frac{p(x \mid S)}{p(x \mid S) + p(x \mid B)} \]

- Density estimation, in principle, is simple and straightforward.
- Histogramming:
  - **Histogram** data in \( M \) bins in each of the \( d \) feature variables
  \[ \Rightarrow M^d \text{ bins} \leftarrow \text{Curse Of Dimensionality} \]
  - In high dimensions, we would need a huge number of data points or most of the bins would be empty leading to an estimated density of zero.
  - But, the variables are generally correlated and hence tend to be restricted to a sub-space Therefore, **Intrinsic Dimensionality** \( < < d \)
- There are more effective methods for density estimation