Machine learning classification for $D^0$ meson signal extraction in d+Au collisions

Mini-workshop on Machine Learning for Particle Physics, ICNFP 2021
August 25, 2021

Jaroslav Bielčík, Kateřina Hladká, Lukáš Kramárik, Václav Kůs
Czech Technical University in Prague
Katerina.Hladka@fjfi.cvut.cz

The work was supported from the project LTT18002 of the Ministry of Education, Youth, and Sport of the Czech Republic.
Outline

• The Solenoid Tracker At RHIC (STAR)
• Heavy-flavour quarks as a probe of QGP
• Dataset and topological properties of D⁰ mesons
• Random forests
• Boosted decision trees
• Deep neural network
• Comparison of trained algorithms
The **Solenoid Tracker At RHIC (STAR)**

- Designed to study the strongly interacting matter
- Excels in tracking and identification of charged particles at mid-rapidity with full azimuthal coverage
- Most of the subsystems are immersed in 0.5 T solenoidal magnetic field

**Time Projection Chamber (TPC)**
- Main tracking device; momentum determination
- Particle identification via specific energy loss $dE/dx$

**Time of Flight (TOF)**
- Particle identification at low transverse momentum $p_T$ via velocity $\beta$
The **Solenoid Tracker At RHIC (STAR)**

- Designed to study the strongly interacting matter
- Excels in tracking and identification of charged particles at mid-rapidity with full azimuthal coverage
- Most of the subsystems are immersed in 0.5 T solenoidal magnetic field

**Heavy Flavor Tracker (HFT)**
- Inner tracking system
- Excellent DCA$_{xy}$ and DCA$_z$ resolution: $\approx 50$ $\mu$m for kaons at $p_T = 750$ MeV/c
Heavy-flavour quarks as a probe of QGP

- Heavy-flavour quarks possess large masses
  - they are produced primarily at the initial stages of heavy-ion collisions
  - they experience the evolution of the collision

- Charm quarks are confined in hadrons, that due their lifetimes could not be observed directly in the detector

- Open charm hadrons could be studied via hadronic decays: \( D^0(\bar{D}^0) \rightarrow K^-\pi^+(K^+\pi^-) \)

- Our goal is to reconstruct \( D^0 \) meson in order to study cold nuclear matter effects in asymmetric d+Au collisions at \( \sqrt{s_{NN}} = 200 \) GeV
Motivation

Goal is to compare performance of the various machine learning methods (and packages) for finding $D^0$ signal

- **BDT - Boosted Decision Trees (TMVA, ROOT)**\(^1\)
  Widely used, since implemented within ROOT package

- **RF - Random forest (scikit-learn, Python)**\(^2\)
  Robust and interpretable, one of the most popular supervised machine learning methods, fast to train and simple to optimize

- **DNN - Deep neural network (Keras, Python)**\(^3\)
  The most complex machine learning method tested in this study.

---


\(^3\)F. Chollet et al., “Keras.” https://keras.io, 2015
Dataset

- **HIJING full-event simulation of 2016 d+Au collisions at $\sqrt{s_{NN}} = 200$ GeV**
  - official STAR's embedding to zero-bias collisions
  - at least one $D^0$ meson decayed to $K$ and $\pi$ in the event
  - events with primary vertex $|V_z| < 6$ cm

- **$D^0 \rightarrow K \pi$ decay channel is studied**
  - pairs of kaons and pions are reconstructed in the same way as in the data
  - correct charge pairs are used for studies

- **Track selection**
  - HFT tracks
  - At least 15 space points in the TPC for track reconstruction
  - $p_T > 0.15$ GeV/$c$
  - Track pseudorapidity $|\eta| < 1$
  - TPC particle identification via $dE/dx$: pions: $|n\sigma_{\text{pion}}| < 3$, kaons: $|n\sigma_{\text{kaon}}| < 2$
Topological reconstruction of $D^0$ mesons

- Topological properties of $D^0$ decays used for their reconstruction:

  - **DCA** = distance of closest approach
  - **PV** = primary vertex, place of the d+Au collision
  - **SV** = secondary vertex, place of the $D^0$ meson decay

  - **decay length**:  
    - distance between primary vertex (PV) and secondary vertex (SV) of $D^0$ meson candidate  
    - ideally $\approx 200 \, \mu m$

  - **DCA$_{daughters}$** between kaon and pion tracks  
    - ideally $\rightarrow 0 \, \mu m$

  - **cosine of angle $\theta$** between $D^0$ momentum and decay length vector

  - **cosine of angle $\zeta$** between $D^0$ momentum and kaon momentum

  - **DCA$_{D0}$** of $D^0$ meson to PV  
    - ideally $\rightarrow 0 \, \mu m$

  - **DCA$_{K, \pi}$** of kaon and pion to PV  
    - ideally $\gg 0 \, \mu m$
Distribution of topological variables

- Correct and wrong charge combination pairs have nearly similar distributions which makes $D^0$ separation challenging.
- $D^0$ signal shape is as we expect from topological properties of the decay.
Data pre-processing

- Each reconstructed correct-charge Kπ pair is associated to the tracks in the HIJING simulation.

- Those coming from the real D⁰ meson in the simulation are used as signal sample in the classification training, other correct-charge pairs are considered to be the background.

- For the testing phase of the methods, combined signal and background correct-charge pairs are used.

- All of the methods are independently trained and tested in the five D⁰ meson transverse momentum $p_{T,D^0}$ intervals.

  - $p_{T,D^0}$ intervals for training: 0–1, 1–2, 2–3, 3–5 and 5–8 GeV/c.

- For ML training, datasets with and without precuts were tested.

---

All presented models were trained and evaluated over balanced dataset: number of signal samples $\approx$ number of background samples.

---

<table>
<thead>
<tr>
<th>Variable</th>
<th>Min.</th>
<th>Max.</th>
</tr>
</thead>
<tbody>
<tr>
<td>$DCA_{K,\pi}$ [mm]</td>
<td>0.002</td>
<td>2.0</td>
</tr>
<tr>
<td>$DCA_{daughters}$ [mm]</td>
<td>0.0</td>
<td>0.2</td>
</tr>
<tr>
<td>decay length [mm]</td>
<td>0.005</td>
<td>2.0</td>
</tr>
<tr>
<td>$DCA_{D^0}$ [mm]</td>
<td>0.0</td>
<td>0.5</td>
</tr>
<tr>
<td>$\cos \theta$</td>
<td>0.7</td>
<td>1.0</td>
</tr>
</tbody>
</table>
Random forest

- **Random forest (RF):**
  - scikit-learn package used
  - Machine learning algorithm built with ensemble of independent trees trained using bagging (bootstrap aggregating)
  - Parameters may be optimized during the training:
    - maximum depth of trees
    - number of trees within ensemble
    - function used to measure quality of split of samples within tree’s node (impurity measure)

- **Hyper-parameter space for RF optimization:**
  - maximum depth of trees $d_{\text{max}}$: \{5, 7, 10, 12, 15, 17, 20\}
  - number of trees $N_{\text{trees}}$: \{200, 400, 600, 800, 1000, 1200\}
  - impurity measure: \{Gini, Entropy\}

- **Dataset split:**
  - 60\% of the samples used for training and hyper-parameter optimization (grid search) using 3-fold cross-validation
  - 40\% of the samples used for classifier evaluation

<table>
<thead>
<tr>
<th>Optimized parameters of random forest classifier</th>
<th>With precuts</th>
<th>Without precuts</th>
</tr>
</thead>
<tbody>
<tr>
<td>$p_T^{1,0}$ [GeV/c]</td>
<td>0-1</td>
<td>0-1</td>
</tr>
<tr>
<td>$d_{\text{max}}$</td>
<td>10</td>
<td>15</td>
</tr>
<tr>
<td>$N_{\text{trees}}$</td>
<td>1200</td>
<td>1200</td>
</tr>
<tr>
<td>Impurity measure</td>
<td>Entropy</td>
<td>Entropy</td>
</tr>
</tbody>
</table>
Boosted decision trees

- **Boosted decision trees (BDT):**
  - TMVA package used
  - Ensemble of shallow trees linked together using boosting algorithm (AdaBoost)
  - Training of individual trees is not independent, data misclassified by a tree have higher weight in the training of the next tree (class weights are balanced using weights coming from classification error fraction)

- **Error fraction:**
  - Calculated as how much signal events have positive BDT response (defined in [-1,1] range) and vice-versa for the background
  - Error fraction is getting higher with increasing $N_{\text{trees}}$ and converges
  - The trees with error fraction close to 0.5 are very weak classifiers

- **BDT parameters:**
  - $N_{\text{trees}}$ : selected as 500 (based on error fraction convergence)
  - Impurity measure: Gini
  - maximum tree depth $d_{\text{max}}$: 3
  - bagging is applied to minimize overtraining
Deep neural network

- Keras package used
- Supervised machine learning algorithm able to perform non-linear transformation of input space
- May address complex classification problems more effectively when optimized properly
- Requires large amount of CPU time, computational power and training data
- Feature-wise standardization was applied to topological variables

Hyper-parameter space for deep neural network setup optimization

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Activation ( \sigma )</th>
<th>Init. Learning rate ( \cdot 10^{-3} )</th>
<th>Dropout rate ( \rho )</th>
<th>Optimizer</th>
<th>Kernel init.</th>
<th>Batch size</th>
<th>Neurons in hidden layer ( w )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Options</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ReLU</td>
<td></td>
<td>0.1</td>
<td>0.2</td>
<td>Adam</td>
<td>Lecun uniform</td>
<td>64</td>
<td>64</td>
</tr>
<tr>
<td>Tanh</td>
<td></td>
<td>0.5</td>
<td>0.3</td>
<td>Stoch. grad. Descent</td>
<td>He Normal</td>
<td>128</td>
<td>128</td>
</tr>
<tr>
<td>1.0</td>
<td></td>
<td>1.0</td>
<td>0.4</td>
<td>AdaMax</td>
<td>He uniform</td>
<td></td>
<td>256</td>
</tr>
</tbody>
</table>
Deep neural network

- **Dataset split:**
  - **training data:** 40% of the samples used for training and hyper-parameter optimization (pre-training 70 epochs)
  - **validation data:** 20% of the samples used to evaluate the pre-trained models (selection of best hyper-parameters combination)
  - **training data** are then used again to train model with optimized hyper-parameters, training is stopped with EarlyStopping rule only (validation ACC was not improved during last 60 epochs)
  - **test data:** 20% of the samples used to evaluate the final model

<table>
<thead>
<tr>
<th>$p_{\text{max}}$ [GeV/c]</th>
<th>0-1</th>
<th>1-2</th>
<th>2-3</th>
<th>3-5</th>
<th>5-8</th>
<th>0-1</th>
<th>1-2</th>
<th>2-3</th>
<th>3-5</th>
<th>5-8</th>
</tr>
</thead>
<tbody>
<tr>
<td>Activation $\sigma$</td>
<td>ReLU</td>
<td>ReLU</td>
<td>ReLU</td>
<td>ReLU</td>
<td>ReLU</td>
<td>ReLU</td>
<td>ReLU</td>
<td>ReLU</td>
<td>ReLU</td>
<td>tanh</td>
</tr>
<tr>
<td>Init. Learning rate [10^{-3}]</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Dropout rate $p$</td>
<td>0.2</td>
<td>0.2</td>
<td>0.2</td>
<td>0.2</td>
<td>0.2</td>
<td>0.2</td>
<td>0.2</td>
<td>0.2</td>
<td>0.2</td>
<td>0.2</td>
</tr>
<tr>
<td>Optimizer</td>
<td>Adam</td>
<td>Adam</td>
<td>Adam</td>
<td>Adam</td>
<td>Adam</td>
<td>Adam</td>
<td>Adam</td>
<td>Adam</td>
<td>Adam</td>
<td>Adam</td>
</tr>
<tr>
<td>Batch size</td>
<td>64</td>
<td>64</td>
<td>64</td>
<td>64</td>
<td>64</td>
<td>64</td>
<td>64</td>
<td>64</td>
<td>64</td>
<td>64</td>
</tr>
<tr>
<td>Neurons in hidden layer $w$</td>
<td>256</td>
<td>256</td>
<td>256</td>
<td>256</td>
<td>256</td>
<td>256</td>
<td>256</td>
<td>256</td>
<td>256</td>
<td>256</td>
</tr>
</tbody>
</table>
Evaluation of receiver operating characteristics over test set

- Classification ability increases with higher $p_{T,D0}$
- For $p_{T,D0} < 5 \text{ GeV}/c$, classifiers trained over the data after preselection criteria (cuts) application achieve higher AUC values as those trained without their application.
- RF classification has significantly better performance than other methods.

### Area under receiver operating characteristic curve (AUC)

<table>
<thead>
<tr>
<th>$p_{T,D0}$ [GeV/c]</th>
<th>With precuts</th>
<th>Without precuts</th>
</tr>
</thead>
<tbody>
<tr>
<td>0-1</td>
<td>0.81</td>
<td>0.73</td>
</tr>
<tr>
<td>1-2</td>
<td>0.87</td>
<td>0.79</td>
</tr>
<tr>
<td>2-3</td>
<td>0.94</td>
<td>0.82</td>
</tr>
<tr>
<td>3-4</td>
<td>0.94</td>
<td>0.90</td>
</tr>
<tr>
<td>4-5</td>
<td>0.96</td>
<td>0.90</td>
</tr>
</tbody>
</table>

**Table legend:**
- **RF**: Random Forest
- **BDT**: Boosted Decision Tree
- **DNN**: Deep Neural Network
Conclusion

- Various machine learning methods (RF, BDT, DNN) were compared in order to obtain efficient classifier for finding signal in two-body decay of $D^0$ meson.

- HIJING full-event simulation of 2016 d+Au collisions at $\sqrt{s_{NN}} = 200$ GeV was used to study $D^0 \rightarrow K \pi$ decay channel.

- Tight cut preselection and the option with no preselection criteria was applied to topological variables.

- Despite the fact, that RF method is not common in high-energy physics, it shows notable enhancement of signal efficiency compared to the other presented methods.

- DNN in presented setup does not show significant improvement compared to the widely used BDT. However, their performance is close and the DNN might help to improve signal significance in similar analyses.
Backup
Signal and background efficiencies vs. reconstructed transverse momentum of Kπ pairs