Uczenie maszynowe w fizyce wysokich energii

P. Grabiński, B. Meder, V. Mykhaylova

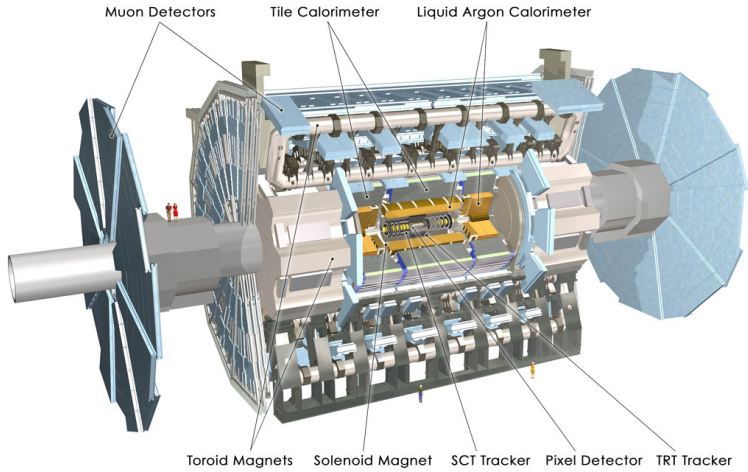
Instytut Fizyki Jądrowej
im. H. Niewodniczańskiego

24 lipca 2015

Porządek prezentacji

Uczenie maszynowe vs. fizyka cząstek

vs.



Definicja

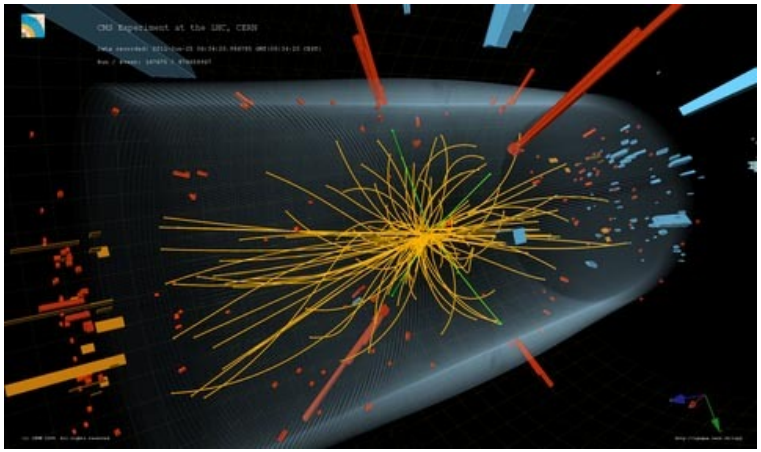
Uczenie maszynowe - algorytmy pozwolające zautomatyzować proces pozyskiwania i analizy danych.
Algorytmy uczą się zależności występujących w problemie na dostarczonych przykładach, poprawiając zdolność przewidywania z każdym kolejnym przejściem.
Wyróżniamy:

- Uczenie z nauczycielem - dla danych traningowych znana jest odpowiedź
- Uczenie bez nauczyciela - nieznana odpowiedź, algorytmy grupują podobne dane

Dlaczego potrzebujemy takich metod?

1. Ogromna ilość danych gromadzonych w trakcie zderzeń cząstek w detektorach
2. Nie znamy dokładnych zależności pomiędzy zmiennymi.

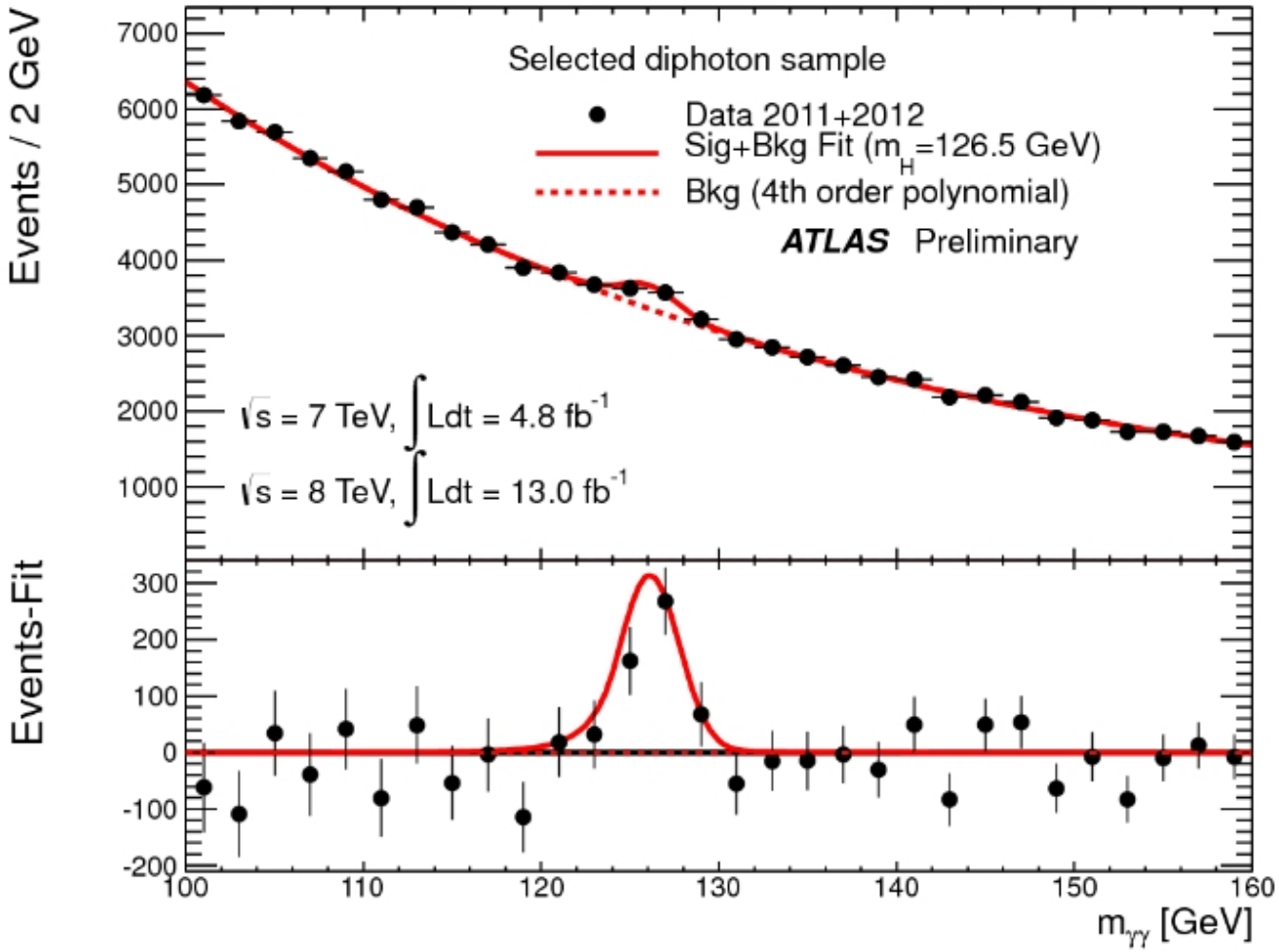
Analiza wielu zmiennych - dowolna metoda statystyczna, w której podczas obliczeń stosujemy więcej niż jedną zmienną.



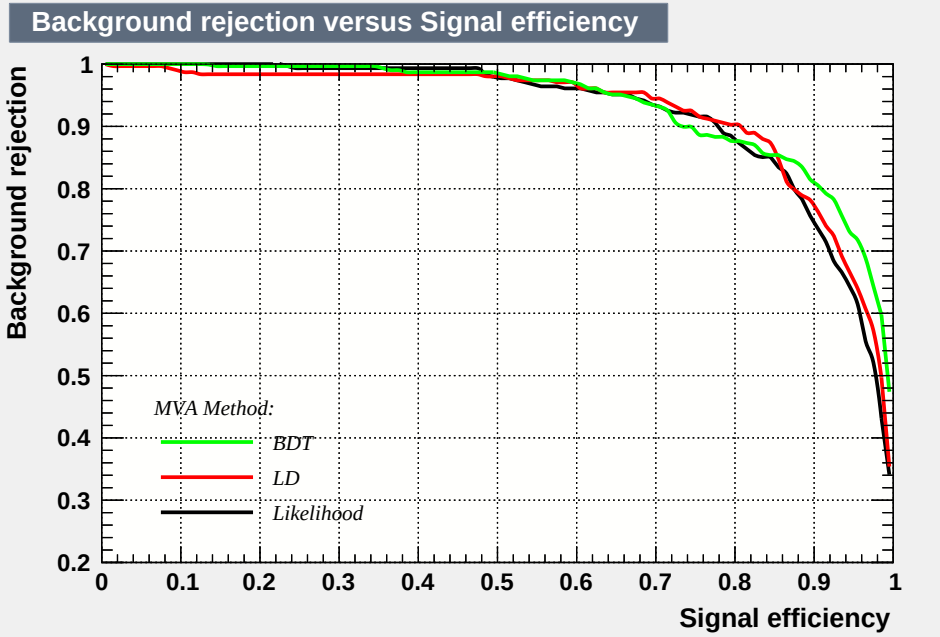
Analiza danych

Odkrycie bozonu Higgsa kanał rozpadu $H \to \gamma\gamma$.

Duże tło, słaby sygnał - bez analizy danych nie widać odpowiedzi.



Miara jakości algorytmu - Krzywa ROC



Krzywa ROC(angl. receiver operating characteristic)
Miarą jakości algorytmu jest ilość odrzuconego tła w funkcji rejestrowanego sygnału.
AUC - Area Under the Curve
Wartością libczową jaką operujemy jest powierzchnia pod krzywą.

TMVA

TMVA 4

Toolkit for Multivariate Data Analysis with ROOT

Users Guide

A. Hoecker, P. Speckmayer, J. Stelzer, J. Therhaag, E. von Toerne, H. Voss

Contributed to TMVA have:

M. Backes, T. Carli, O. Cohen, A. Christov, D. Dannheim, K. Danielowski, S. Henrot-Versillé, M. Jachowski, K. Kraszewski, A. Krasznahorkay Jr., M. Kruk, Y. Mahalalel, R. Ospanov, X. Prudent, A. Robert, D. Schouten, F. Tegenfeldt, A. Voigt, K. Voss, M. Wolter, A. Zemla

TMVA - opis

TMVA - The Toolkit for Multivariate Data Analysis with ROOT - niezależny projekt, zintegrowany z ROOT, który zapewnia środowisko uczenia maszynowego do przetwarzania i oceny zaawansowanych wielowymiarowych technik klasyfikacyjnych.

Zalety:

- integralny z **Root**
- napisany w **C++**
- obszerna dokumentacja i wiele przykładów
- automatyczna wizualizacja danych

Wady:

- integralny z **Root**
- **brak wielowątkowości**

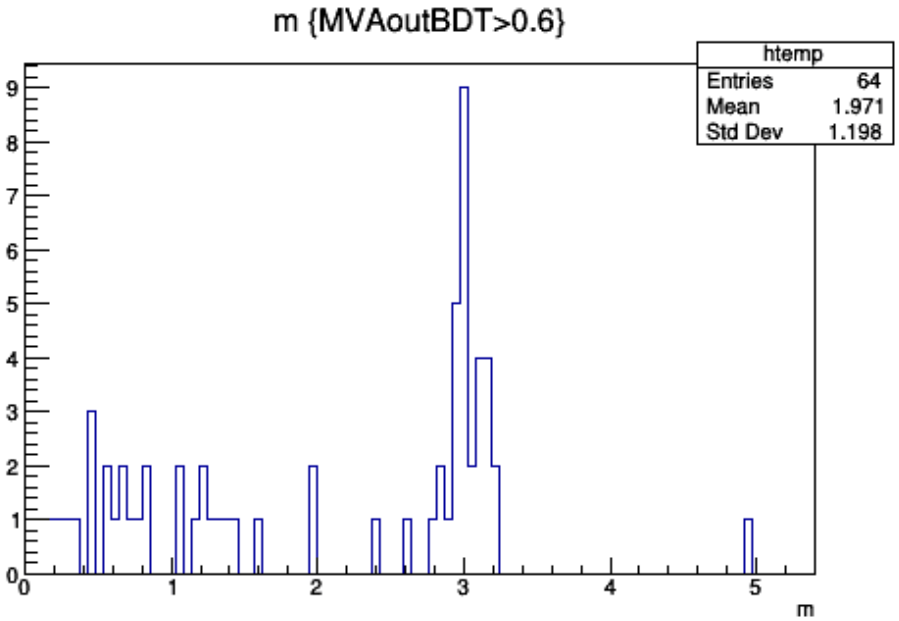
Możliwości TMVA

Zawiera algorytmy:

- Rectangular Cut Optimisation (cięcia)
- Likelihood Estimator
- Fischer's Discriminants
- Function Discriminant Analysis
- **Artificial Neuron Networks (ANN)**
- Multidimensional K-Nearest Neighbour Classifier (KNN)
- **Boosted Decision Trees (BDT)**
- Support Vector Machine (SVM)

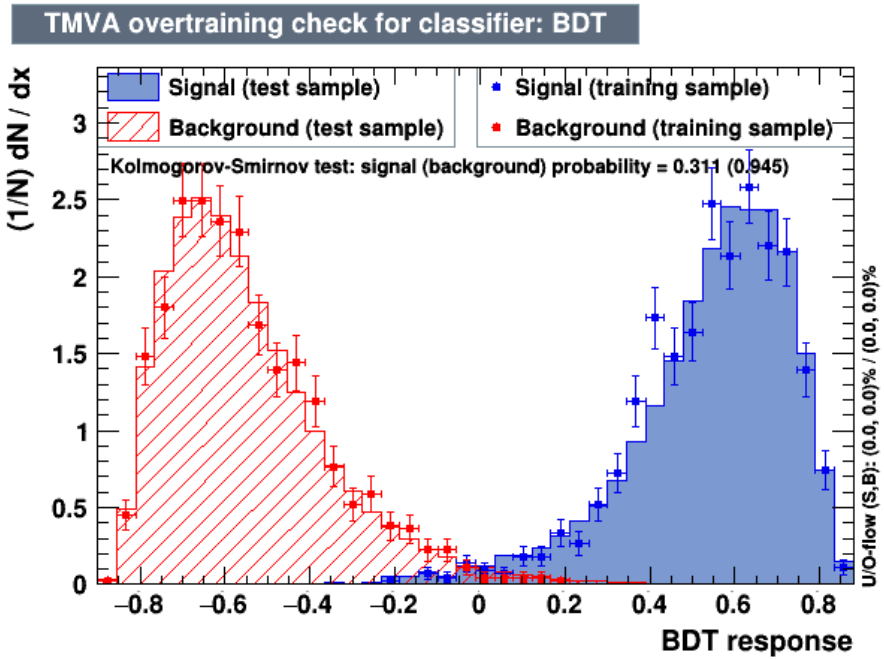
TMVA vs. Ćwiczenie 6

Masa inwariantna:



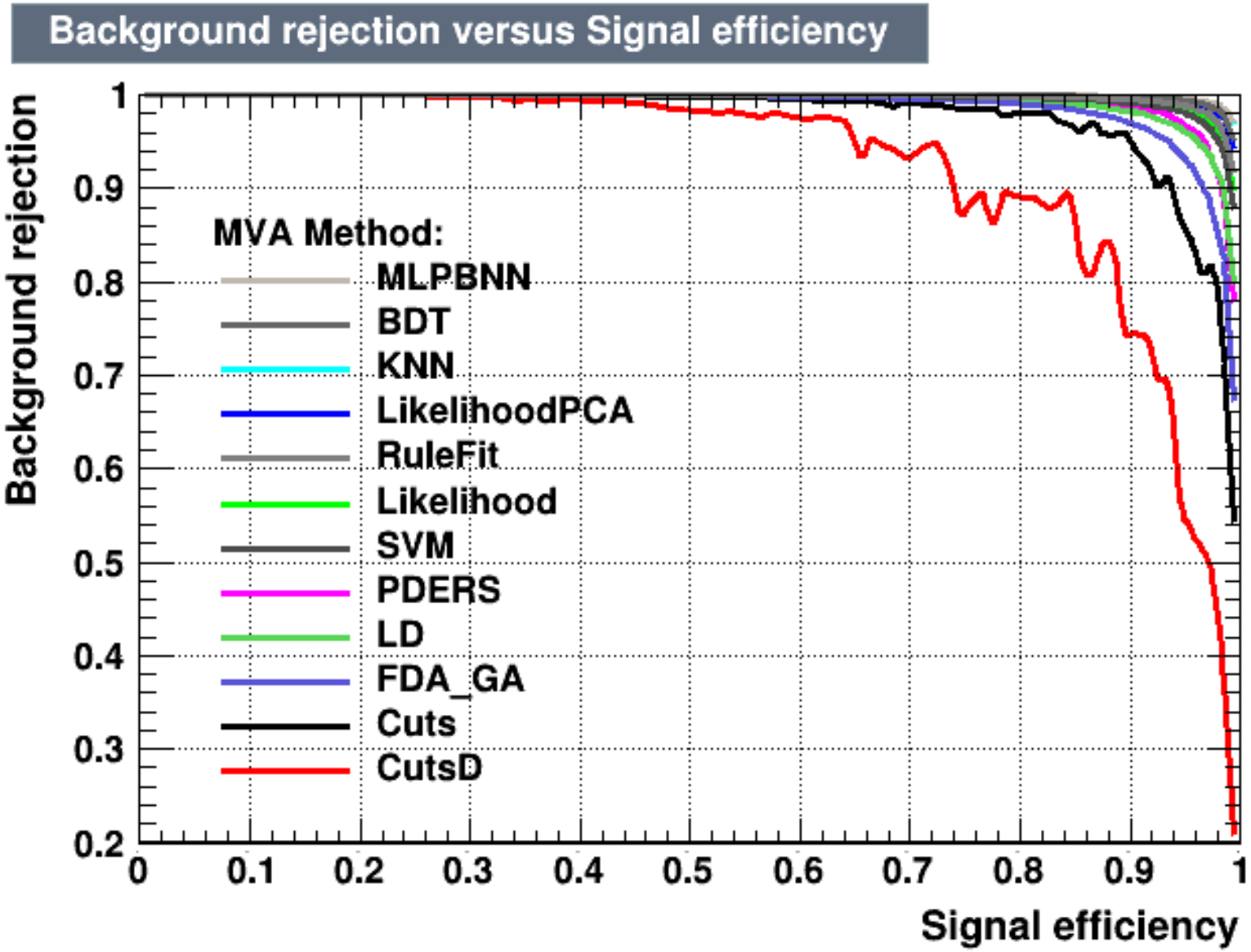
Widoczny sygnał dla $m = 3$

Współczynik przynależności:



Bardzo dobra separacja

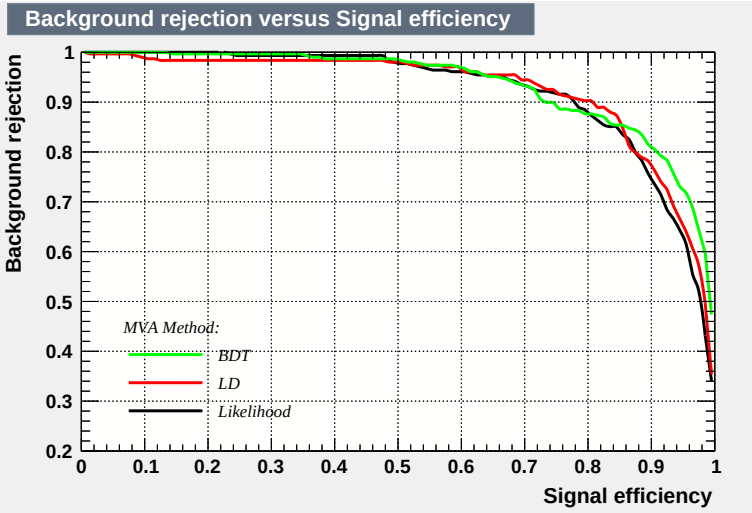
TMVA vs. Ćwiczenie 6 - krzywa ROC



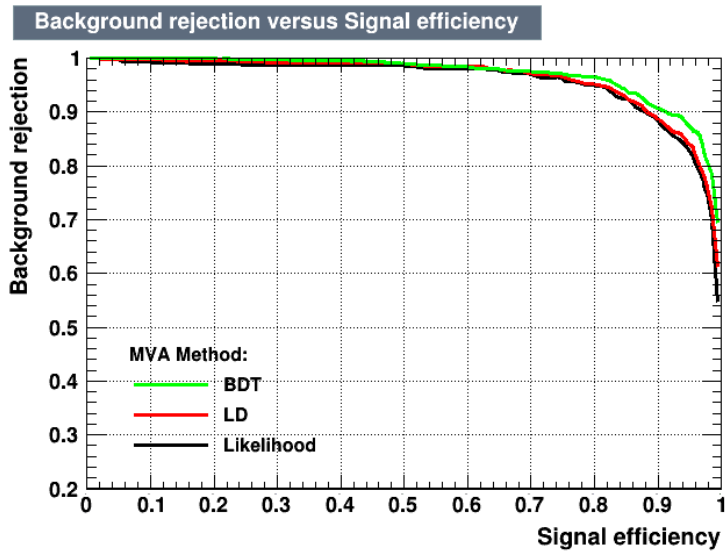
TMVA vs. Dane ATLAS

Sygnał: $Z \to \tau\tau$ Tło: $W \to \mu\upsilon_\mu$

Dwie skorelowane zmienne



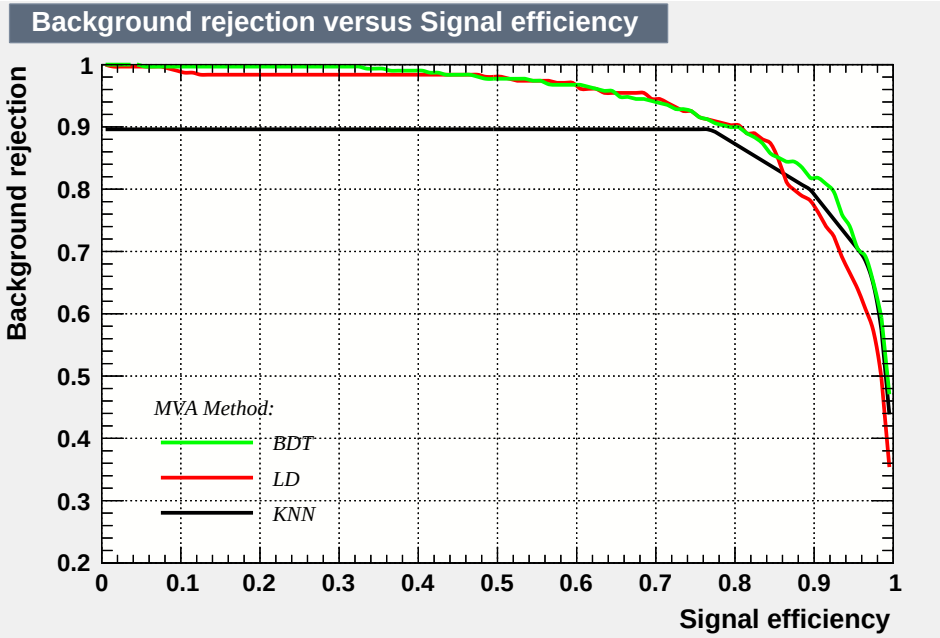
Usunięta jedna ze skorelowanych zmiennych



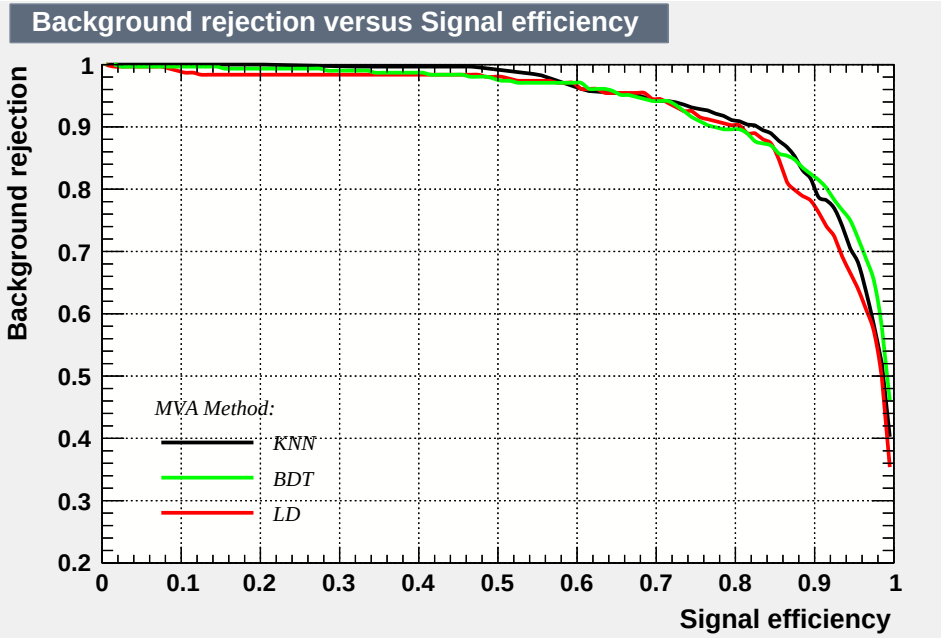
Zmienne analizowane: evtsel_tau_et, evtsel_dPhiSum, evtsel_tau_pi0_n, evtsel_transverseMass, sum_cos_dphi, evt_sel_lep_pt,tau_leadTrkPt_at0. W drugim podejściu usunięto zmienną sum_cos_dphi.

TMVA vs. Dane ATLAS - parametry KNN

Parametr $n_{KNN} = 30$



Parametr $n_{KNN} = 500$



Widać, że przy odpowiednich parametrach algorytm K-Nearest Neighbours osiągnął wyniki lepsze od innych algorytmów.

XGBoost - opis

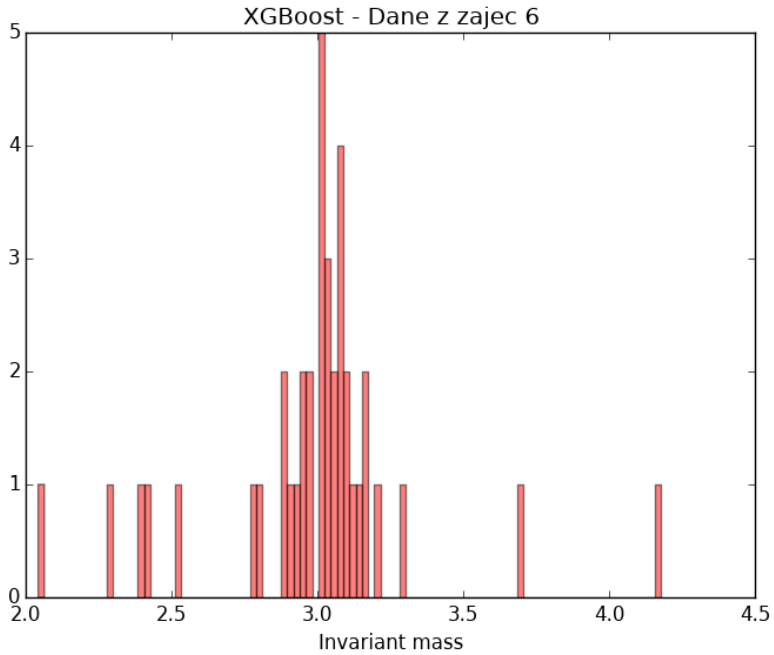
- Napisany:
  - w **C++(wydajność)**
  - z użyciem technologii **OpenMP(multithreading - więcej wydajności)**
  - z interfejsem w **Pythonie**
- Wykorzystuje algorytmy:
  - **Gradient Boosted Decision Trees (GBDT)**
  - **Generalized Linear Model (GLM)**
- Zajął 9. miejsce w **KHBMLC**
- Umożliwił wygraną w konkursach **Kaggle**:
  - **Predict the relevance of search results from eCommerce sites** sponsorowanym przez **CrowdFlower**
  - **Microsoft Malware Classification Challenge (BIG 2015)**
- nagrodzony nagrodą **HEP meets ML award**

XGBoost vs. Ćwiczenie 6

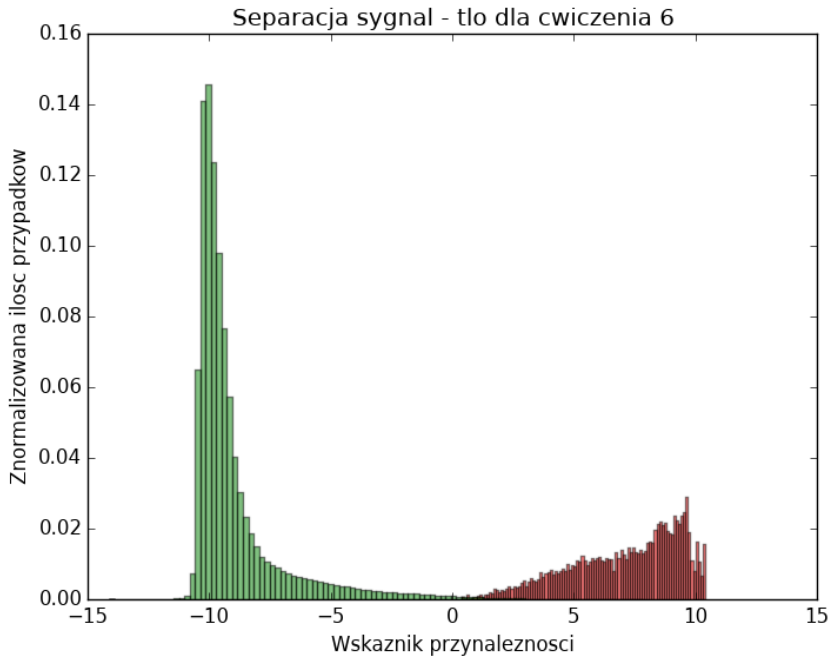
Parametry przez nas użyte:

- Głębokość maksymalna 9
- Współczynnik uczenia 0.01
- Liczba drzew 1000
- Część przykładów 1

Masa niezmiennicza z widocznym sygnałem:



Bardzo dobra separacja:



Kaggle Higgs Boson Challenge

Completed • $13,000 • 1,785 teams

Higgs Boson Machine Learning Challenge

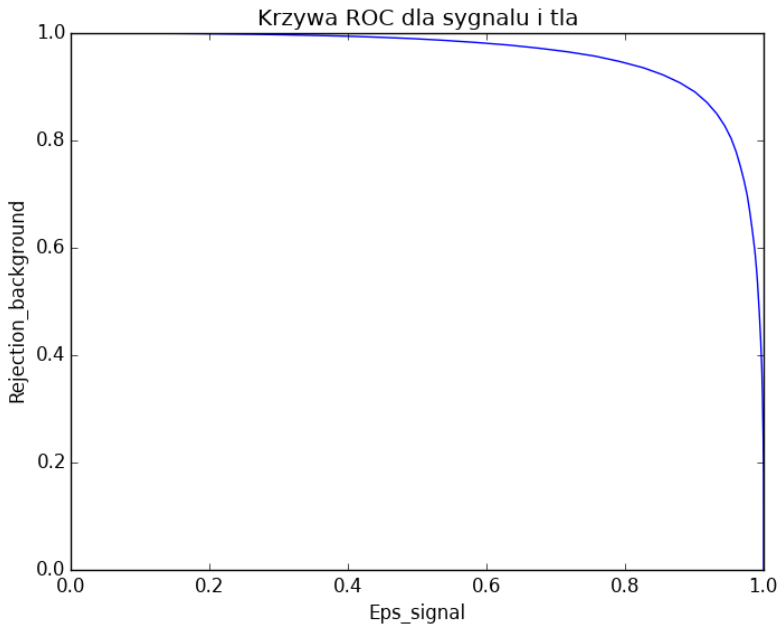
Mon 12 May 2014 – Mon 15 Sep 2014 (10 months ago)

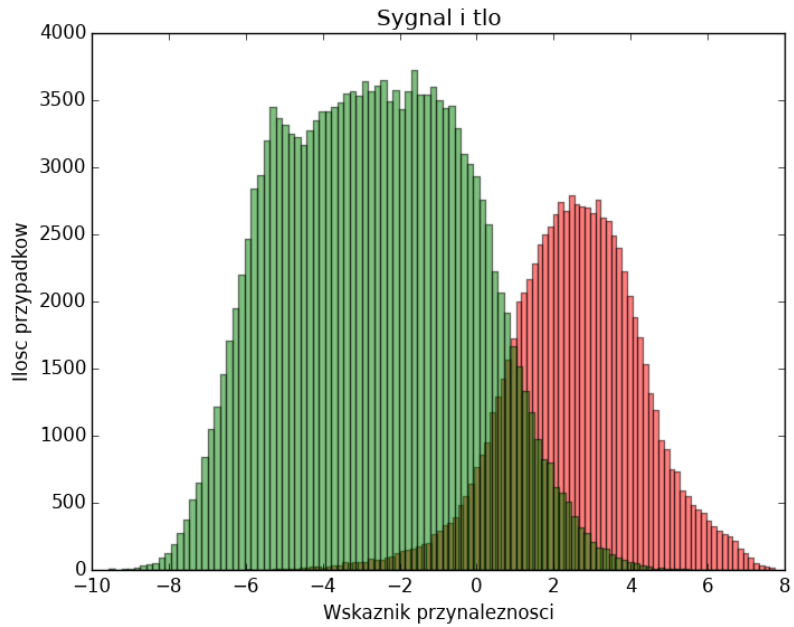
- Nazwa: Higgs Boson Machine Learining Challenge
- Cel: Nauczyć się nowych metod od specjalistów zajmujących się ML profesjonalnie
- Suma nagród: 13.000$
- Wnioski:
  - **ANN** $>$ **BDT**
  - warto myśleć o technologiach wielowątkowych

Dane Kaggle Higgs

Parametry najlepszego rozwiązania wykorzystującego XGBoost z konkursu Kaggle Higgs:

- Głębokość maksymalna 9
- Współczynnik uczenia 0.01
- Liczba drzew 3000
- Część przykładów 0.9





Hyperopt - opis

Pakiet napisany w **Pythonie** służący do optymalizacji funkcji skalarnych na skomplikowanych przestrzeniach, które mogą być rzeczywiste, dyskretne lub warunkowe.
Posiada dwa algorytmy wyszukiwania:

- Random Search
- Tree of Parzen Estimators (TPE)

Możliwe równoległe obliczenia przy użyciu pakietu **MongoDB**.
Użyty w takich projektach jak:

- **hyperopt-sklearn** - optymalizacja algortymów samouczących zawartych w **scikit-learn**
- **hyperopt-convnet** - optymalizacja **Convolutional neural network (CNN)**

Rozwiązania Kaggle-Higgs vs Hyperopt

Porównianie wyników uzyskanych przez nas automatycznie z wynikami z najlepszymi znalezionymi parametrami dla XGBoost.

Kto	9. K-H	M. Wolter	Nasze obliczenia
Maks. głębokość	9	10	9
Wsp. uczenia	0.01	0.089	0.059
Liczba drzew	3000	150/250/500	300
Liczba testów	-	300	100
Sub_sample	0.9	1	0.9
Maks. ROC	0.987	0.933/0.934/0.933	0.934

Sub_sample - jaka cześć danych brana jest do procesu uczenia - wprowadza pewną losowość i zapobiega przeuczaniu

Jak widać wyniki przez nas osiągnięte są znacznie słabsze. Prowadziliśmy poszukiwania w innym regionie parametrów.

Podsumowanie

Rozwiązaliśmy następujące problemy:

- Powtórzyliśmy ćwiczenie nr 6 przy użyciu TMVA oraz XGBoost
- Przeanalizowaliśmy dane rozpadów $Z \rightarrow \tau\tau$ z detektora ATLAS
- Użyliśmy XGBoost na danych z konkursu Kaggle Higgs Boson Challenge
- Zastosowaliśmy Hyperopt do automatycznej optymalizacji rozwiązania problemu z konkursu Kaggle Higgs Boson Challenge

Referencje

A. Hoecker, P. Speckmayer, J. Stelzer, J. Therhaag, E. von Toerne, H. Voss (2009)
TMVA 4 Package Documentation
https://tmva.sf.net

Tianqi Chen, Tong He, Bing Xu and Michael Benesty (2014)
XGBoost Package Documentation
https://github.com/dmlc/xgboost

James Bergstra, Dan Yamins, and David D. Cox (2013)
Hyperopt Package Documentation
https://github.com/hyperopt

Dziękujemy za uwagę!