Michał Wójcik
Avast Software
Research Department

Flavours of Physics - Kaggle competition (4th place)

December, 11th 2015

NIPS conference, Montreal

Features

FlightDistance - *Distance between t and PV (primary vertex, the original protons collision point).*

FlightDistanceError - *Error on FlightDistance*.

mass - *reconstructed t candidate invariant mass,* ***which is absent in the test samples***.

LifeTime - *Life time of tau candidate*.

IP - *Impact Parameter of tau candidate*.

IPSig - *Significance of Impact Parameter*.

VertexChi2 - *Chi2 of t vertex.*

dira - *Cosine of the angle between the t momentum and line between PV and tau vertex*.

pt - *transverse momentum of t*.

DOCAone - *Distance of Closest Approach between p0 and p1.*

DOCAtwo - *Distance of Closest Approach between p1 and p2*.

DOCAthree - *Distance of Closest Approach between p0 and p2.*

IP_p0p2 - *Impact parameter of the p0 and p2 pair.*

IP_p1p2 - *Impact parameter of the p1 and p2 pair.*

isolationa - *Track isolation variable.*

isolationb - *Track isolation variable*.

isolationc - *Track isolation variable.*

isolationd - *Track isolation variable*.

isolatione - *Track isolation variable.*

isolationf - *Track isolation variable.*

iso - *Track isolation variable.*

CDF1 - *Cone isolation variable.*

CDF2 - *Cone isolation variable*.

CDF3 - *Cone isolation variable.*

production - *source of t. This variable is absent in the test samples.*

ISO_SumBDT - *Track isolation variable.*

p0_IsoBDT - *Track isolation variable.*

p1_IsoBDT - *Track isolation variable.*

p2_IsoBDT - *Track isolation variable*.

p0_track_Chi2Dof - *Quality of p0 muon track.*

p1_track_Chi2Dof - *Quality of p1 muon track.*

p2_track_Chi2Dof - *Quality of p2 muon track.*

p0_pt - *Transverse momentum of p0 muon.*

p0_p - *Momentum of p0 muon.*

p0_eta - *Pseudorapidity of p0 muon.*

p0_IP - *Impact parameter of p0 muon.*

p0_IPSig - *Impact Parameter Significance of p0 muon.*

p1_pt - *Transverse momentum of p1 muon.*

p1_p - *Momentum of p1 muon.*

p1_eta - *Pseudorapidity of p1 muon*.

p1_IP - *Impact parameter of p1 muon.*

p1_IPSig - *Impact Parameter Significance of p1 muon*.

p2_pt - *Transverse momentum of p2 muon.*

p2_p - *Momentum of p2 muon.*

p2_eta - *Pseudorapidity of p2 muon.*

p2_IP - *Impact parameter of p2 muon*.

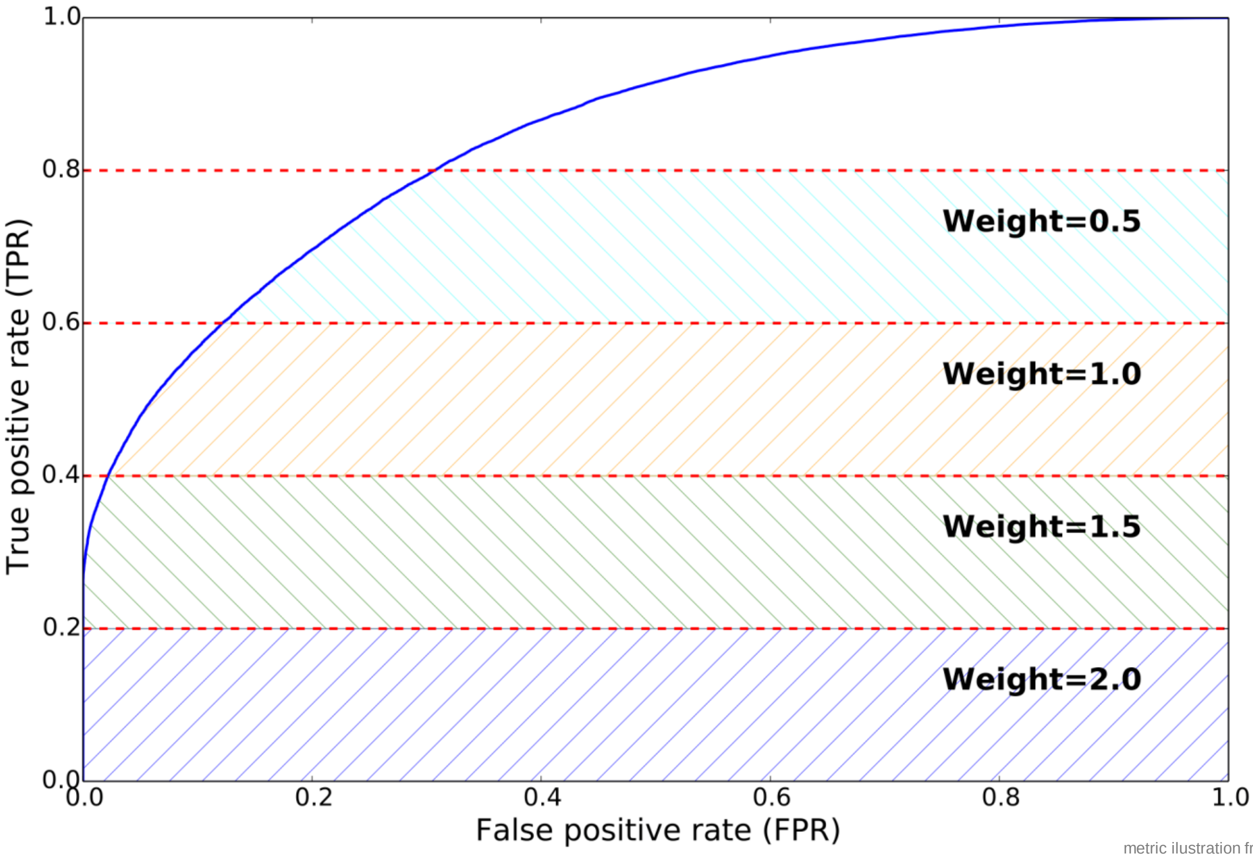
p2_IPSig - *Impact Parameter Significance of p2 muon.*

SPDhits - *Number of hits in the SPD detector.*

min_ANNmuon - *Muon identification. LHCb collaboration trains Artificial Neural Networks (ANN) from informations from RICH, ECAL, HCAL, Muon system to distinguish muons from other particles. This variables denotes the minimum of the three muons ANN. min ANNmuon should not be used for training. This variable is absent in the test samples.*

signal - *This is the target variable for you to predict in the test samples.*

Receiver operating characteristic (ROC) curve



metric ilustration from kaggle.com

(KS) Kolmogorov–Smirnov test ≤ 0.09

(CVM) Cramer-von Mises test ≤ 0.002

Features a, nnp01, nnp02, nnp12

Let p be a vector of momentum of mother particle and p_0, p_1, p_2 be momentum vectors of outcoming particles. We don't know these vectors from data, but we can calculate their magnitudes.

We assume that $p = p_0 + p_1 + p_2$ thus

$|p|^2 = |p_0|^2 + |p_1|^2 + |p_2|^2 + 2cos(\alpha_{01})|p_0||p_1| + 2cos(\alpha_{02})|p_0||p_2| + 2cos(\alpha_{12})|p_1||p_2|$, where α_{ij} is an angle between particles i and j.

Let $B = |p|^2 - |p_0|^2 - |p_1|^2 - |p_2|^2$

and let $x_{01} = \frac{|p_0||p_1|}{B}$, $x_{02} = \frac{|p_0||p_2|}{B}$, $x_{12} = \frac{|p_1||p_2|}{B}$. (they are features 'nnp01', 'nnp02', 'nnp12')

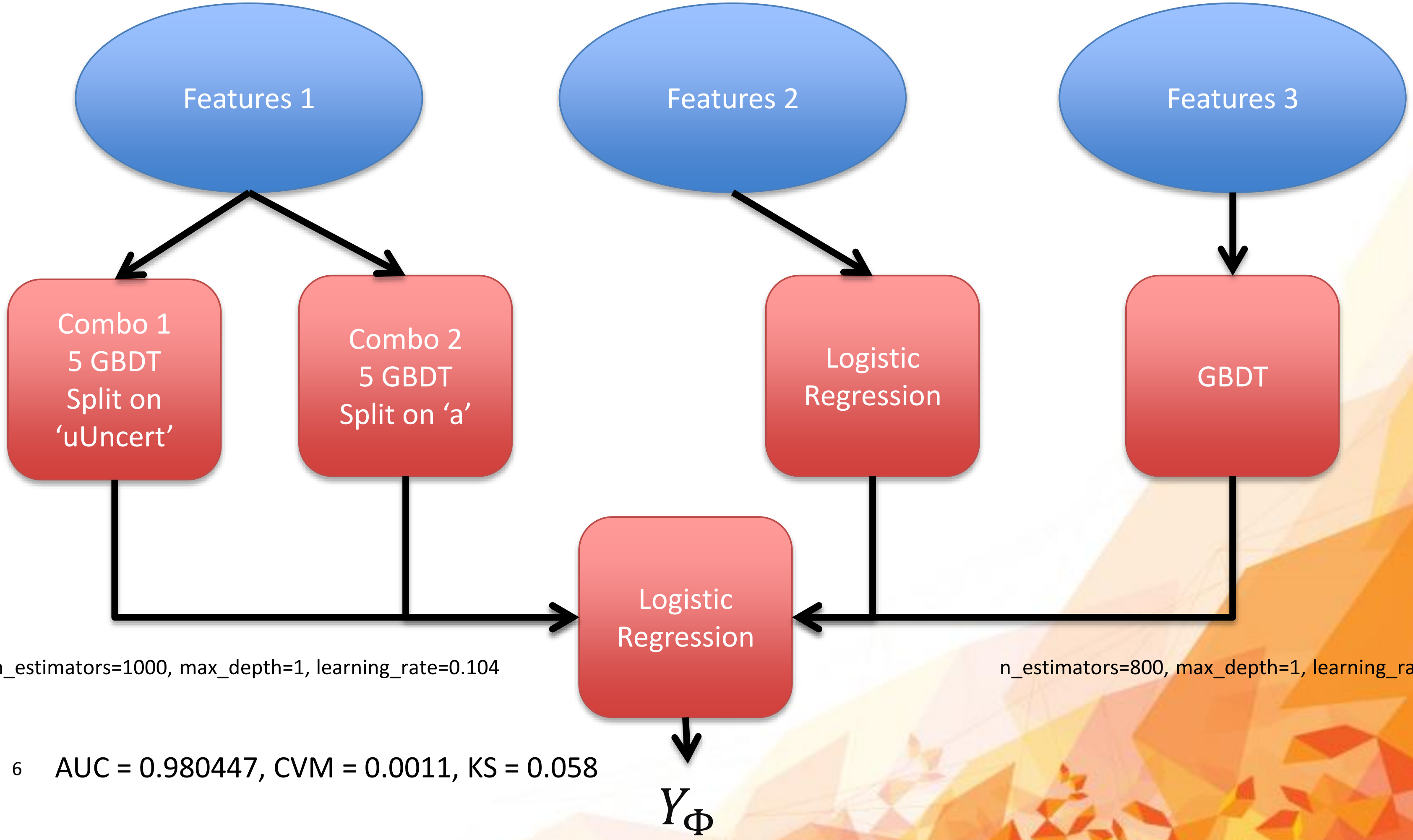
Now we get an equation $0.5 = cos(\alpha_{01})x_{01} + cos(\alpha_{12})x_{12} + cos(\alpha_{12})x_{12}$

Because angles α_{ij} are quite small all points (x_{01} , x_{02} , x_{12}) are almost on the plane $x + y + z = 0.5$, which is clearly visible on data plot. This is the plane of 0-angles. We can now measure how far from this plane our real (x_{01} , x_{02} , x_{12}) are. And this is exactly feature 'a'.

$a = 0.5 - x_{01} - x_{02} - x_{12}$.

Features 1	Features 2	Features 3
nUncert, **a, nnp01, nnp02, nnp12**, np0_IsoBDT, np1_IsoBDT, np2_IsoBDT, np0_track_Chi2Dof, np1_track_Chi2Dof, np2_track_Chi2Dof, nVertexChi2', nIPSig, nIP, nFlightDistanceError, nFlightDistance, nLifeTime, nISO_SumBDT, dira, nIP, CDF1, CDF2, CDF3, **_pz0,_pz1,_pz2, _nE0,_nE1,_nE2**	niso, nisolationa, nisolationb, nisolationc, nisolationd, nisolatione,nisolationf	**a**, nVertexChi2, dira, nDOCAone, nDOCAtwo, nDOCAthree, nFlightDistanceError, nIP, nIPSig, nFlightDistance

nUncert = nFlightDistanceError * nVertexChi2



AUC = 0.980447, CVM = 0.0011, KS = 0.058

Features 1 + Features 2 + Features 3

+ **E, p0_p, p1_p, p2_p, p0_pt, p1_pt, p2_pt, E0, E1, E2** + Invariant mass from momentum features + Invariant mass from LifeTime, FlightDistance

MAE = 7.0

MAE = 4.0

Alexander Gramolin
Idea presented
on Kaggle forum

Random Forest Regression

n_estimators=150

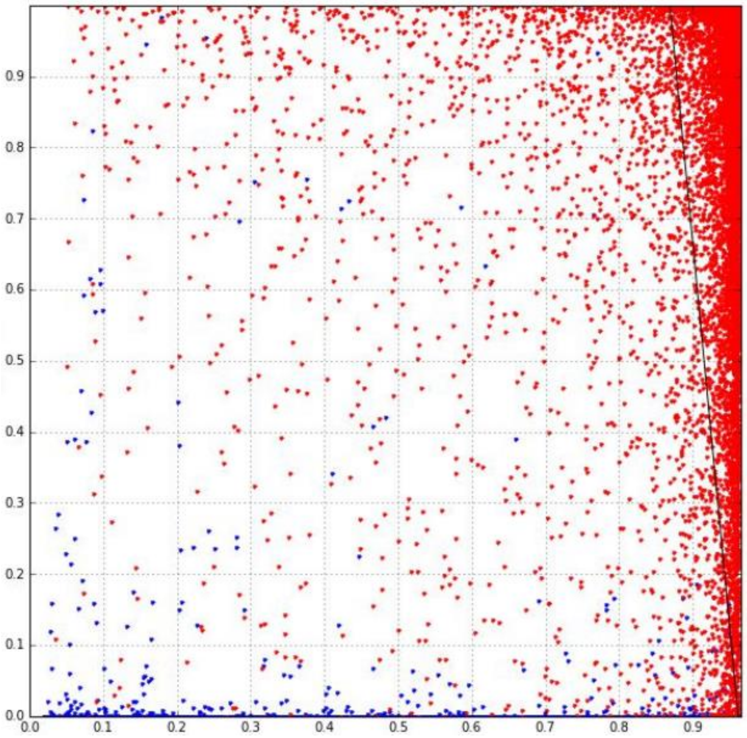
→ m

MAE = 2.6

$$Y_M(b) = \exp\left(\frac{\ln(0.5)(m-1776.82)^2}{b^2}\right)$$

$$Y = aY_\Phi + (1 - a)Y_M(b) = 0.915\, Y_\Phi + 0.085 \exp\left(\frac{\ln(0.5)(m - 1776.82)^2}{(11.0)^2}\right)$$

$$Y_M = \exp\left(\frac{\ln(0.5)(m - 1776.82)^2}{(11.0)^2}\right)$$



Y_Φ

$$0.88 \le 0.915\, Y_\Phi + 0.085\, Y_M$$

AUC = 0.999831, CVM = 0.00194, KS = 0.058

Thank you all for your attention!

Special thanks to Avast Software and Intel Corporation for helping me to be here!