Results from Flavour Tagging using a neural network algorithm in LHCb Data

Matt Williams

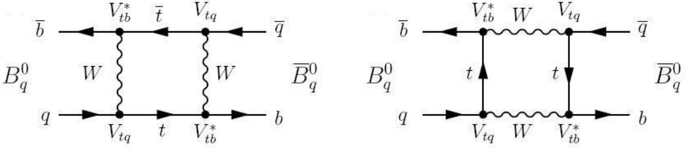
University of Warwick/RAL

April 3, 2012

Flavour Tagging

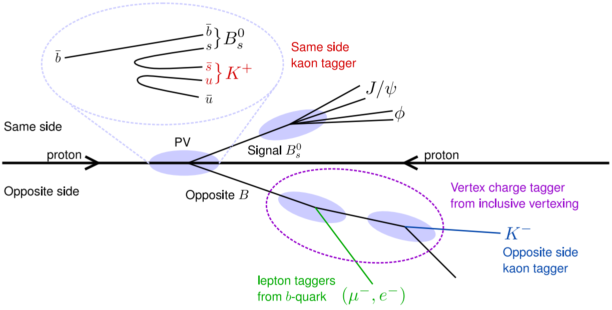
Flavour tagging is the process of establishing the quark content of a meson at the time it was created.

Since B mesons can oscillate while travelling, their flavour at decay time may not be the same as their initial flavour.



The decay-time flavour can sometimes be inferred from its decay products but to get the initial flavour, alternative methods must be used.

Tagging at LHCb



In the proton collision, a $b\bar{b}$ pair is created. Each quark will form a hadron; one of which will be the signal which is reconstructed for use in analysis.

The non-signal hadron will contain information about the flavour of the signal meson. It can decay in a number of channels, both hadronically and leptonically.

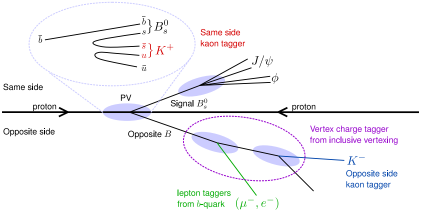
Existing muon tagger

The opposite-side muon tagger selects a muon coming from the decay of the opposite-side B meson. The muon is created in a $b \to c\mu^-\bar{\nu}_\mu$ process (branching fraction of $\sim$10%).

The charge of the muon is then directly correlated to the flavour of the signal-side B meson.

The muon is selected through a series of simple cuts on p, p_T, primary vertex impact parameter, muon PID etc.

There are a number of backgrounds such as non-muons misidentified as muons and real muons which have come from secondary charm decays.



A different approach – NeuroBayes

Instead of using 'square' cuts, use a multivariate method to select the muon.

Multivariate selections have in the past been shown to improve signal yields and background rejection, even when trained on the same variables.

NeuroBayes was chosen since it provides:

- Automated input preprocessing
- Decorrelation of inputs
- Low-significance input pruning
- Simple interface allows to quickly get good results

Training samples for Muon tagger

The neural network was trained on Monte Carlo simulation data since it allows cleaner separation of signal and background samples.

In order to test sensitivity to the training channel, two data sets were used: $B \to DX$ (where X is any hadron) and $B^+ \to J/\psi K$.

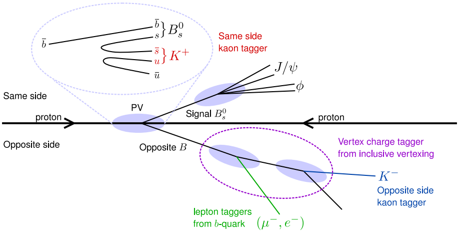
Monte Carlo from 2010 was used, with running conditions as close to the physics run as possible.

Training scheme

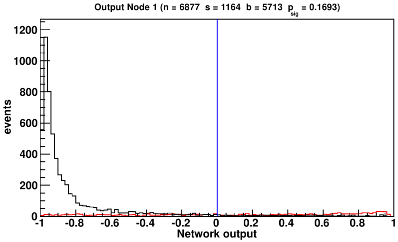
In order to train the network, signal and background samples must be defined.

First a set a loose cuts (muon PID, p, p_T etc.) are applied. Monte Carlo truth information is used to determine exactly which muons came from the decay of the opposite-side B hadron.

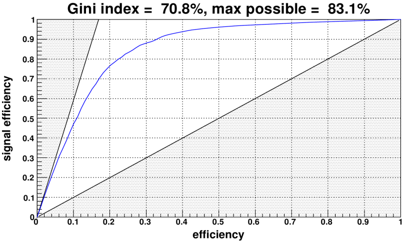
Those muons which came from the hadron are defined as signal and all others (muons and non-muons) are defined as background.



Results of training



The response of the neural network on a testing sample. Black is background and red is signal.



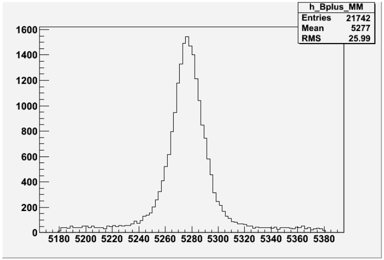
ROC curve of the neural network.

Testing on real data

The neural network was used to select tagging muons in real data to test its effectiveness.

$B^+ \to J/\psi K^+$ was chosen as the testing channel as it is self-tagging. The B^+ doesn't oscillate and so the reconstructed flavour can be compared with that from the tagging algorithm.

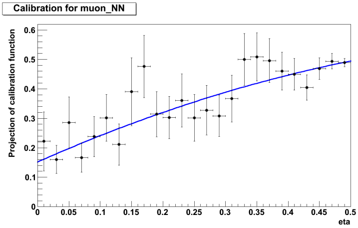
The flavour estimate and the neural network output (NN_{out}) is recorded for each event. NN_{out} will be used as an estimate of how sure we are of any particular event being tagged correctly.



Calibration

NN_{out} ranges from -1 to 1. It is more usual to measure in terms of mistag probability (η_N) which ranges from 0 to 0.5 so we apply a linear transformation.

η_N is still an estimate and so it must be calibrated. For each bin of η_N we calculate the true mistag fraction (ω) of the events in that bin. Plotting the two against each other and fitting gives the calibration.



$$NN_{out} \xrightarrow{transform} \eta_N \xrightarrow{calibrate} \omega$$

Tagging power

The effective efficiency of a tagging algorithm is given as

$$\varepsilon_{eff} = \varepsilon_{tag} D^2$$

where ε_{tag} is the efficiency with which the tagger returns a result and D is the dilution, defined as $D = 1 - 2\omega$.

An ε_{eff} of 10% gives data with equivalent statistics with having 10% of the data with perfect tagging.

Results

Tagger	$\varepsilon_{tag}(\%)$	$\omega(\%)$	$\varepsilon_{eff}\ (\%)$
Standard	5.15	29.3 ± 1.9	0.88 ± 0.11
$B \to DX$ Neural Network	12.44	36.0 ± 1.2	1.00 ± 0.12
$B^+ \to J/\psi K$ Neural Network	9.91	34.0 ± 1.2	0.99 ± 0.10

The results of the neural network-based tagger compared to the existing muon tagger. ε_{eff} is the figure of merit.

This muon tagger results feeds into the full LHCb tagger which has a total ε_{eff} of approximately 2.5%.

Summary

- Existing taggers use simple cut-based selections.
- We used a neural network to select the muons instead.
- Gained a 12.5% increase in efficiency.
- Currently being integrated into LHCb's tagging package for others to experiment with.

Questions?