

Toy generation



Marcin Chrzęszcz
Nicola Serra
Elena Graverini



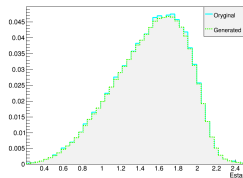
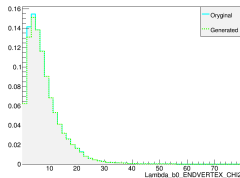
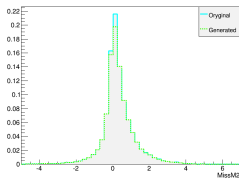
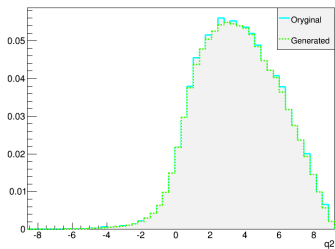
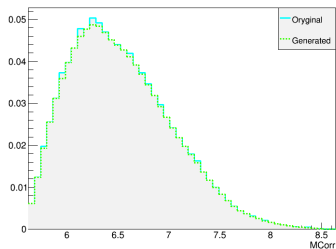
University of
Zurich ^{UZH}

$R(\Lambda_c^*)$ meeting, Zurich
February 17, 2016

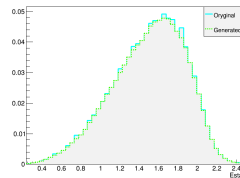
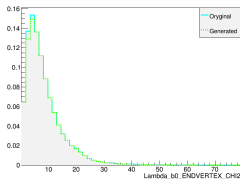
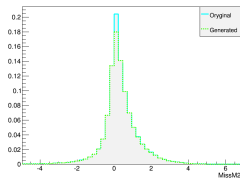
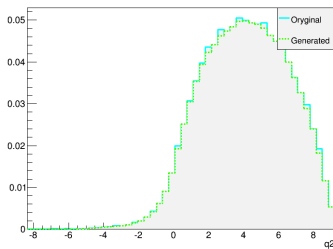
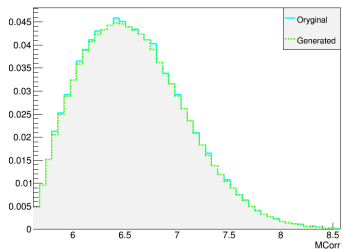
Introduction

- ⇒ Before we ask of $\mathcal{O}(500M)$ events we need to be sure we know how much we need.
- ⇒ Makes it also easier for convenor to argue for the MC production in the PPG.
- ⇒ To perform this studies I propose to produce a large number of TOY MC so that we can study the statistical properties of our approach.
- ⇒ The toys are generated from our data set using a kernel density estimators.
- ⇒ A word of practical advice: I didn't use as RooNDKesPDF as we want to integrate in at least 5D integral and it took over 24h for root to normalize everything.
- ⇒ Numpy libraries takes less then a minute to do that.
- ⇒ The variables I have generated:
 $q^2, m_{miss}^2, m_{corr}, E^*, \chi^2$ VTX
- ⇒ The sample had just the trigger applied!
- ⇒ Separate generation for $\tau, \mu,$ and Λ_c states
- ⇒ Over all it's quite efficient. In 5min I can generate 50M events on my laptop.

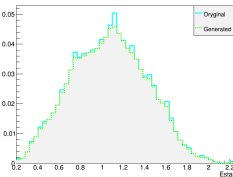
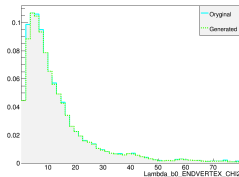
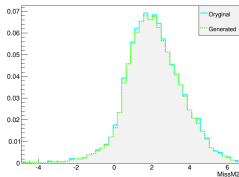
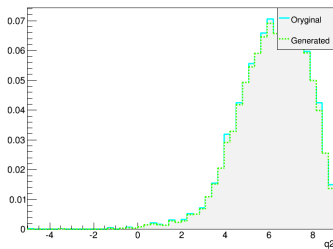
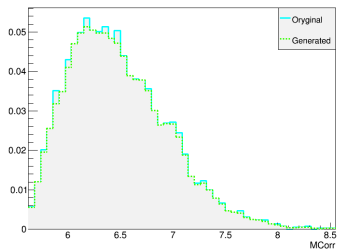
Sanity checks $\Lambda_b \rightarrow \Lambda_c(2625)\mu\nu$



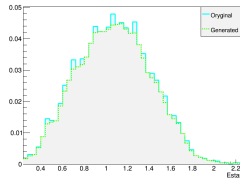
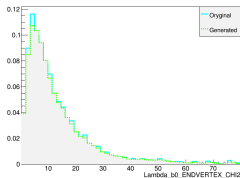
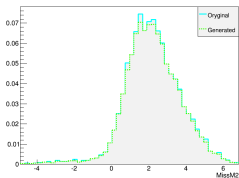
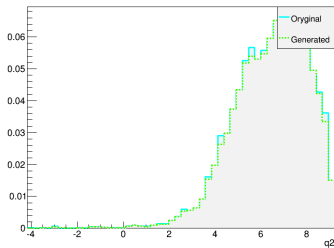
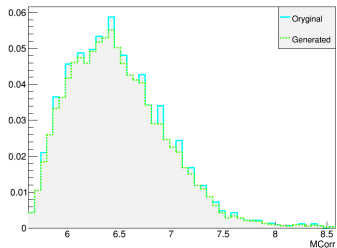
Sanity checks $\Lambda_b \rightarrow \Lambda_c(2595)\mu\nu$



Sanity checks $\Lambda_b \rightarrow \Lambda_c(2625)\tau\nu$



Sanity checks $\Lambda_b \rightarrow \Lambda_c(2595)\tau\nu$



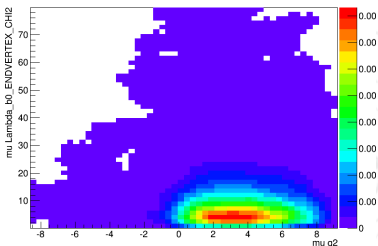
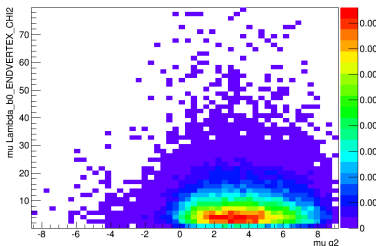
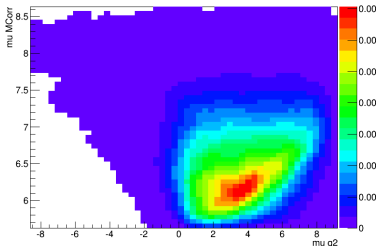
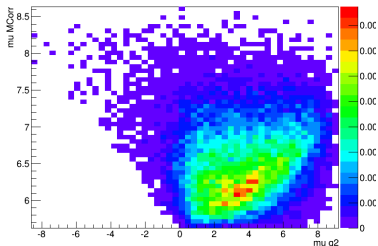
Sanity checks 2

- ⇒ Now it's not the full story. We need to be sure we generate the full correlation correctly!
- ⇒ I produced all the scatter plots so we can check!

Sanity checks 2, $\Lambda_b \rightarrow \Lambda_c(2625)\mu\nu$

Original

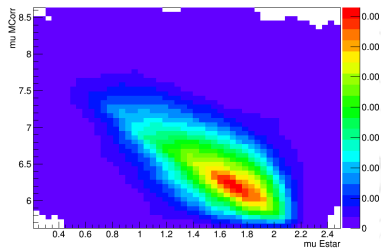
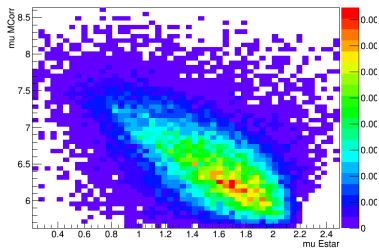
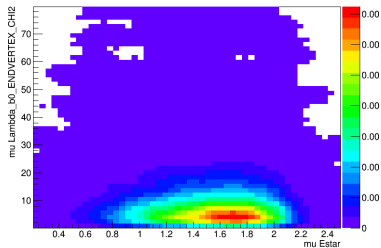
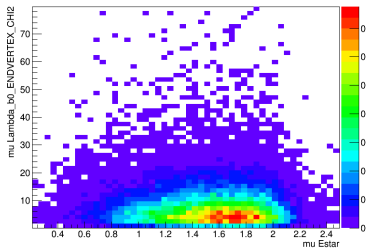
Generated



Sanity checks 2, $\Lambda_b \rightarrow \Lambda_c(2625)\mu\nu$

Original

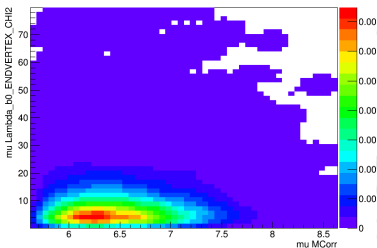
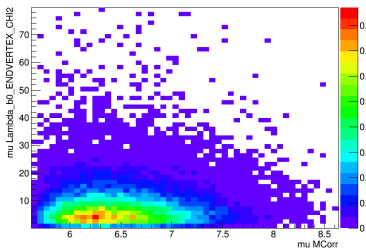
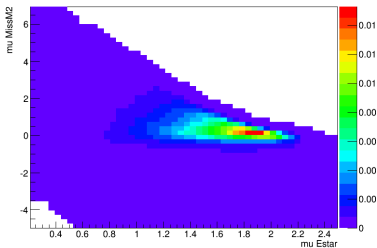
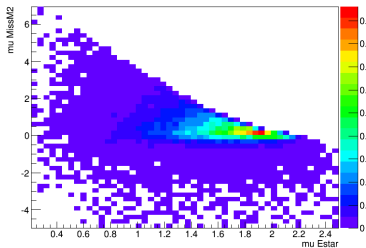
Generated



Sanity checks 2, $\Lambda_b \rightarrow \Lambda_c(2625)\mu\nu$

Original

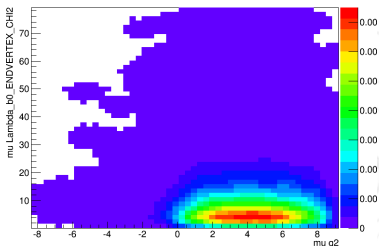
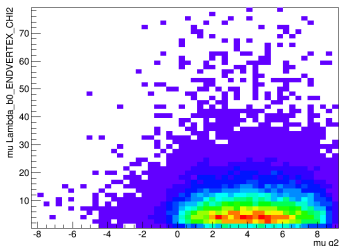
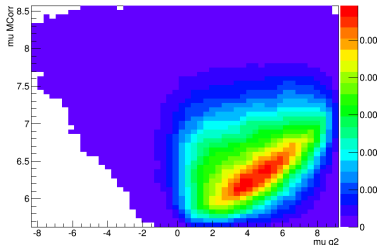
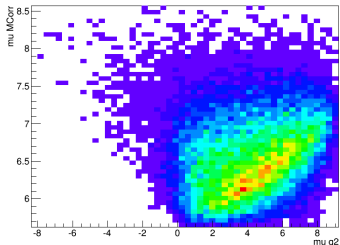
Generated



Sanity checks 2, $\Lambda_b \rightarrow \Lambda_c(2595)\mu\nu$

Original

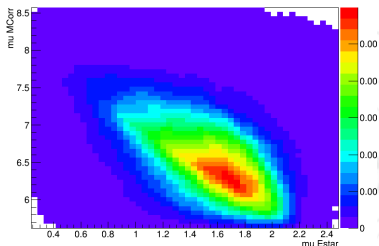
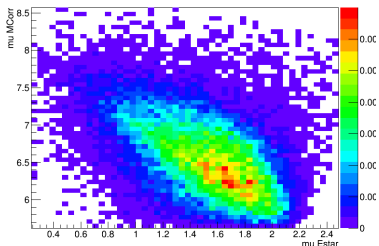
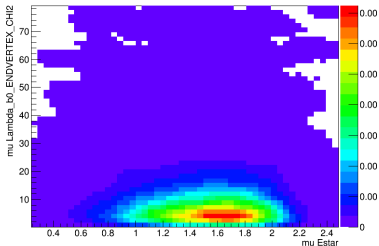
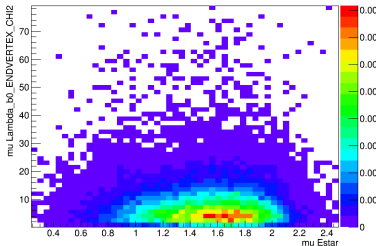
Generated



Sanity checks 2, $\Lambda_b \rightarrow \Lambda_c(2595)\mu\nu$

Original

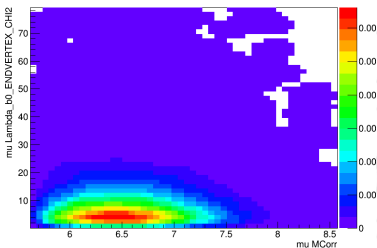
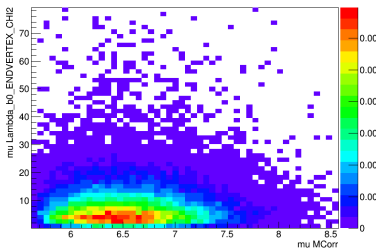
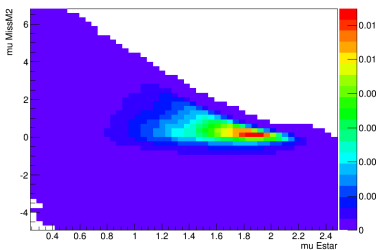
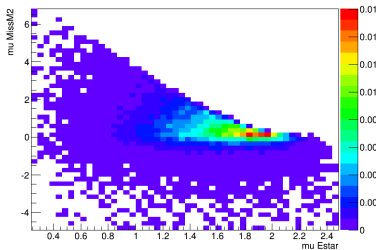
Generated



Sanity checks 2, $\Lambda_b \rightarrow \Lambda_c(2595)\mu\nu$

Original

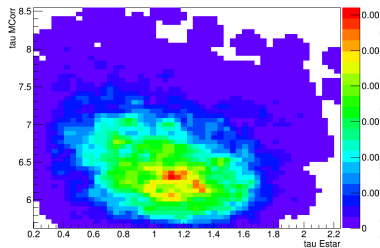
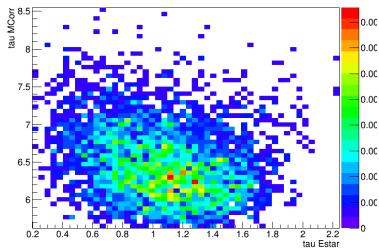
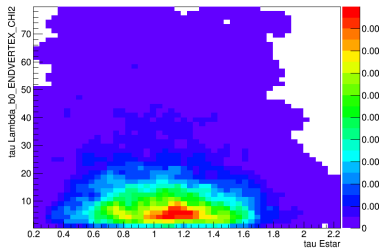
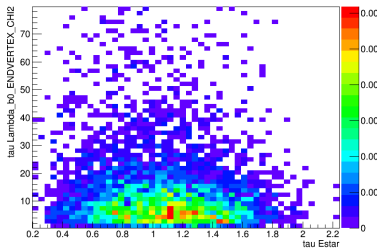
Generated



Sanity checks 2, $\Lambda_b \rightarrow \Lambda_c(2625)\tau\nu$

Original

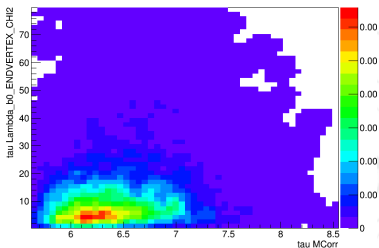
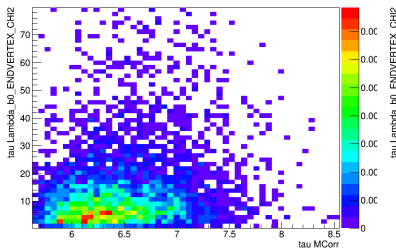
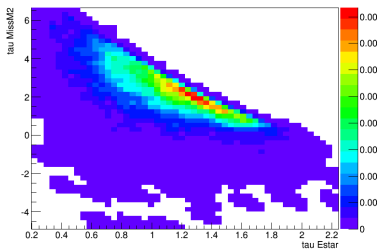
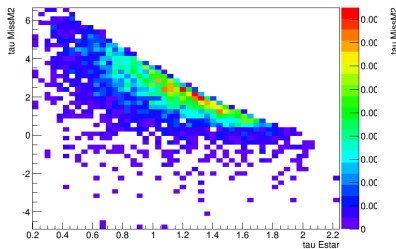
Generated



Sanity checks 2, $\Lambda_b \rightarrow \Lambda_c(2625)\tau\nu$

Original

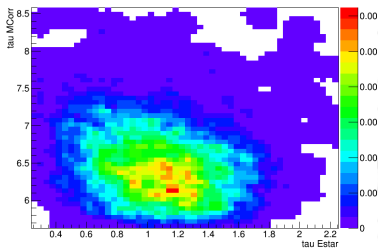
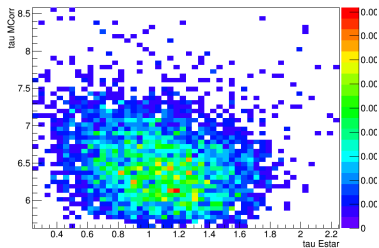
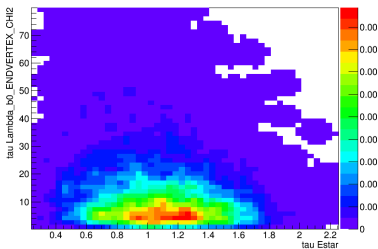
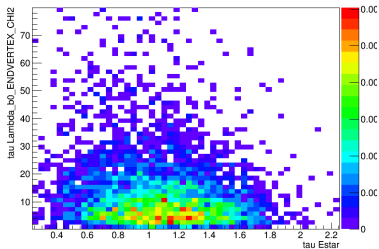
Generated



Sanity checks 2, $\Lambda_b \rightarrow \Lambda_c(2595)\tau\nu$

Original

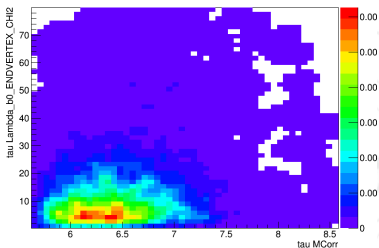
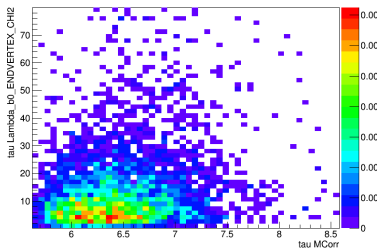
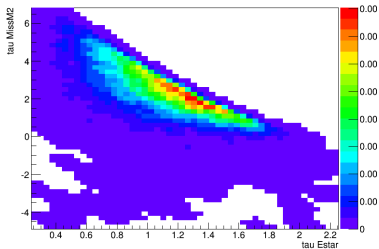
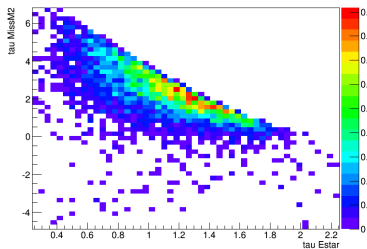
Generated



Sanity checks 2, $\Lambda_b \rightarrow \Lambda_c(2595)\tau\nu$

Original

Generated



Conclusions

- ⇒ The toys work perfectly!
- ⇒ Produced 300M of each type
- ⇒ First BDT algorithms are already being trained.
- ⇒ Looks like the trees are stable but need a bit more tests to be sure, should be ready by the end of the week.

Backup