Hands-on: basf2

Logan Benninghoff (UMiss)

Alex Gale (UC)

Belle II Summer Workshop 2025

at Virginia Tech

Overview

- Preview of tasks, two difficulty levels
- Steering file anatomy reminder
- basf2 setup
- Returning to the tasks and hands-on work time

Goal for Hands-on Session

Produce a steering script for a sample physics analysis.

Two levels of difficulty/complexity, beginner and intermediate.

We will connect to KEKCC, set up command line basf2, give you some work time, then walk through writing the steering file.

Beginner

Input mdst file: /group/belle2/users2022/alexgale/B2SW_example_files/udst/sub00/udst_000001_prod00029739_task10020000001.root

- Reconstruct $D^0 \to K^-\pi^+$

- Make some reasonable cuts on K+ and pi+

- Save InvM and dM

- Extra ~ external to basf2: Plot distributions of variables.

Intermediate

Input mdst file: /group/belle2/users2022/alexgale/B2SW_example_files/udst/sub00/udst_000001_prod00029739_task10020000001.root

- Reconstruct $D^0 \to K^- K^+$ and $D^0 \to K^- \pi^+$

- Reconstruct $\bar{B}_0 \to (D^{*+} \to D^0 \pi^+) \pi^-$

- Make reasonable cuts

- Vertex fit with a best candidate selection on chiProb

- Save InvM, dM, Mbc, deltaE for the B0, and create an alias for InvM of the D0 in the B0 list

Steering File Anatomy (Incomplete)

import basf2 python module (different usage of “module” terminology)

Create the path

Read in data file(s)

Make final state particle list

Form composite particles, truth match

Save variables to an output file

Process the path

Steering File Anatomy (Analysis)

import basf2 python module (different usage of "module" terminology) + other python modules

Create the path (and set other initial condition info)

Read in data file(s)

Make final state particle list (with cuts)

Form composite particles, truth match (with cuts)

Save variables to an output file (with alias names)

Process the path (print statistics)

Setup

You are free to use code editor of your choice, VSCode, in command line with VIM, TextEdit, etc. (Some of these are nicer than others)

Even if you don’t run the file, you can follow along and produce the python steering file without any extra setup.

Connect to KEKCC

Run in your terminal:

ssh your_username@sshcc1.kek.jp

Then put in your password. Wait to be logged on and then run:

ssh your_username@login.cc.kek.jp

Then put in your password. You should be logged into kekcc :)

Setup

When you have logged into KEKCC run:

source /cvmfs/belle.cern.ch/tools/b2setup

b2help-releases

b2setup (the output of the previous command)

Or…

Setup

Go to https://software.belle2.org/ and choose one of the recommended releases. Then you can run:

source /cvmfs/belle.cern.ch/tools/b2setup

b2setup (your chosen release)

If you already know the release you want to use, you can do this in one line. Run,

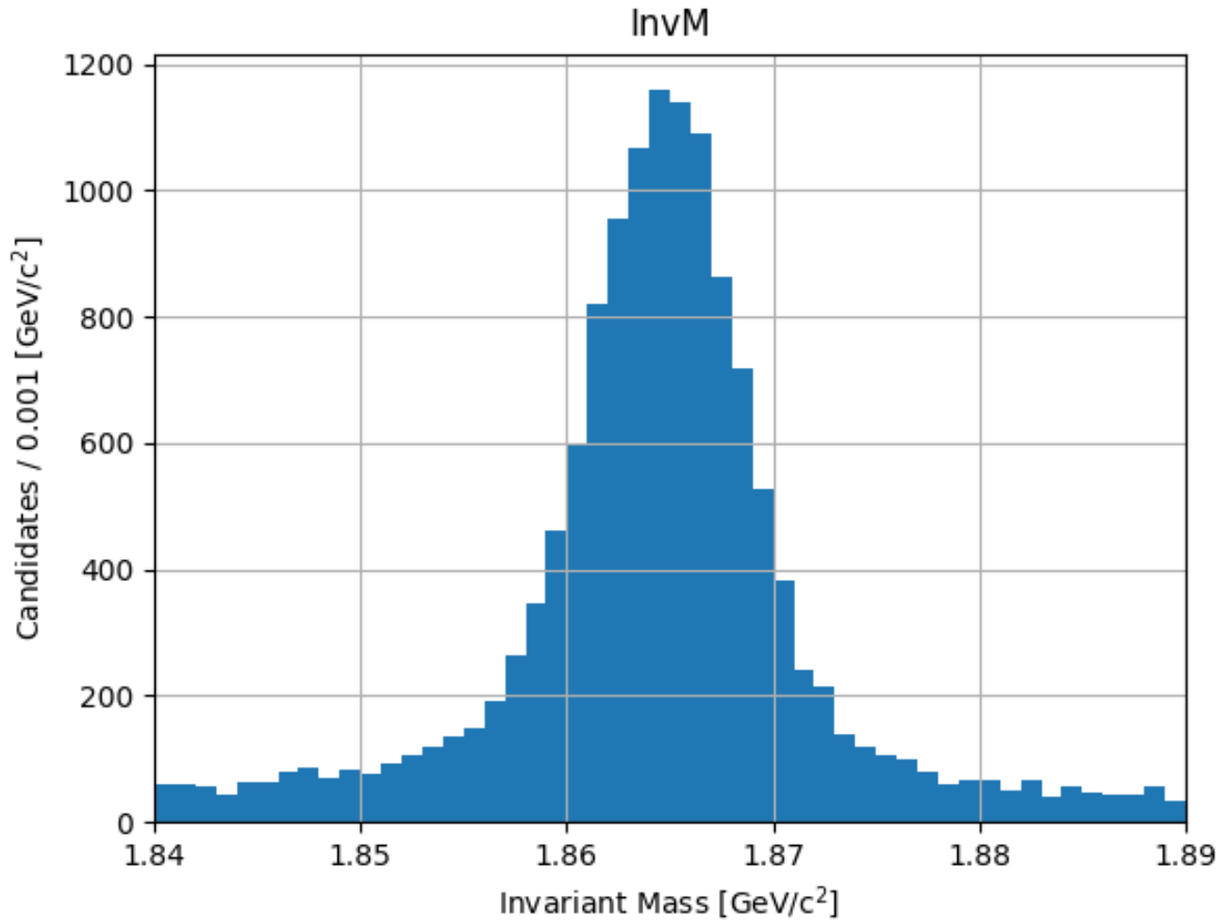
source /cvmfs/belle.cern.ch/tools/b2setup (the release you want to run)

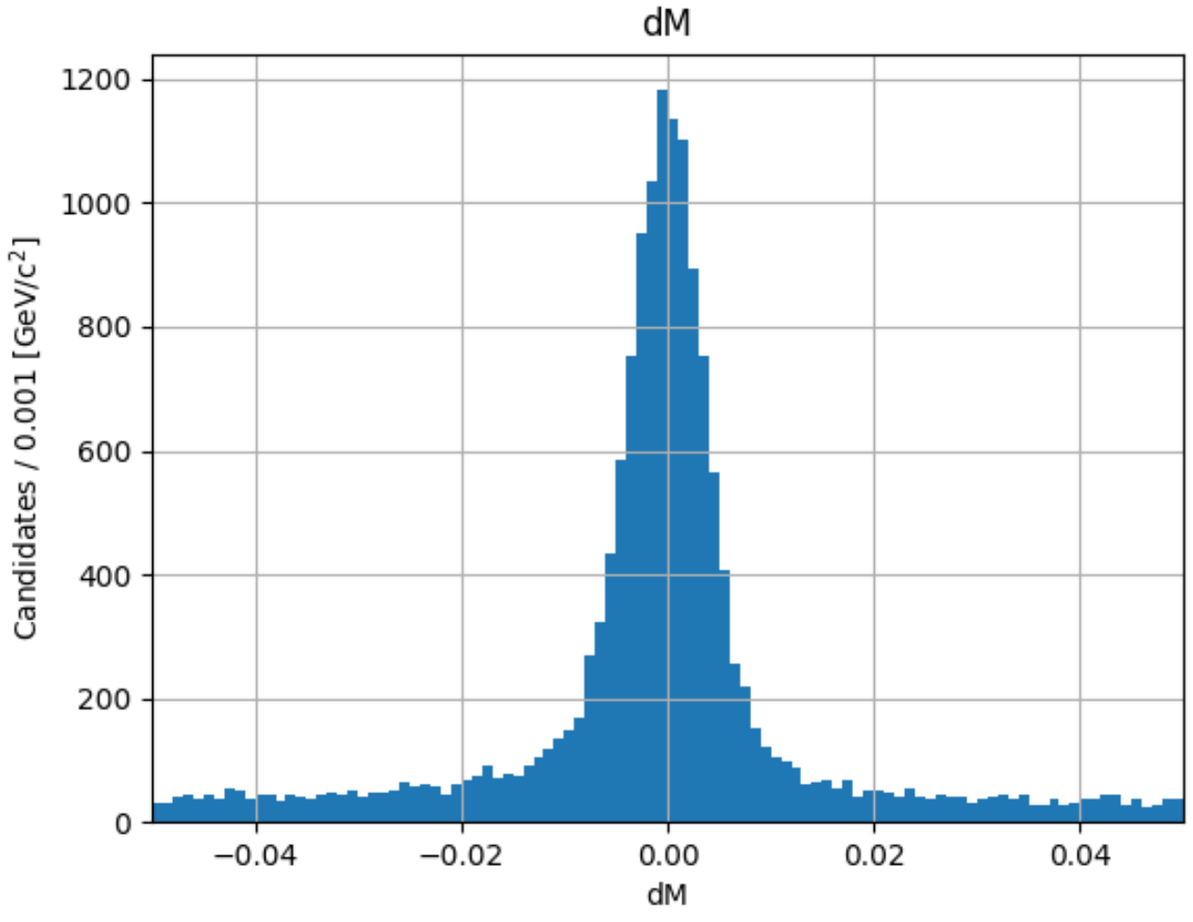
Work Time

Backup

Example 1 Plots:

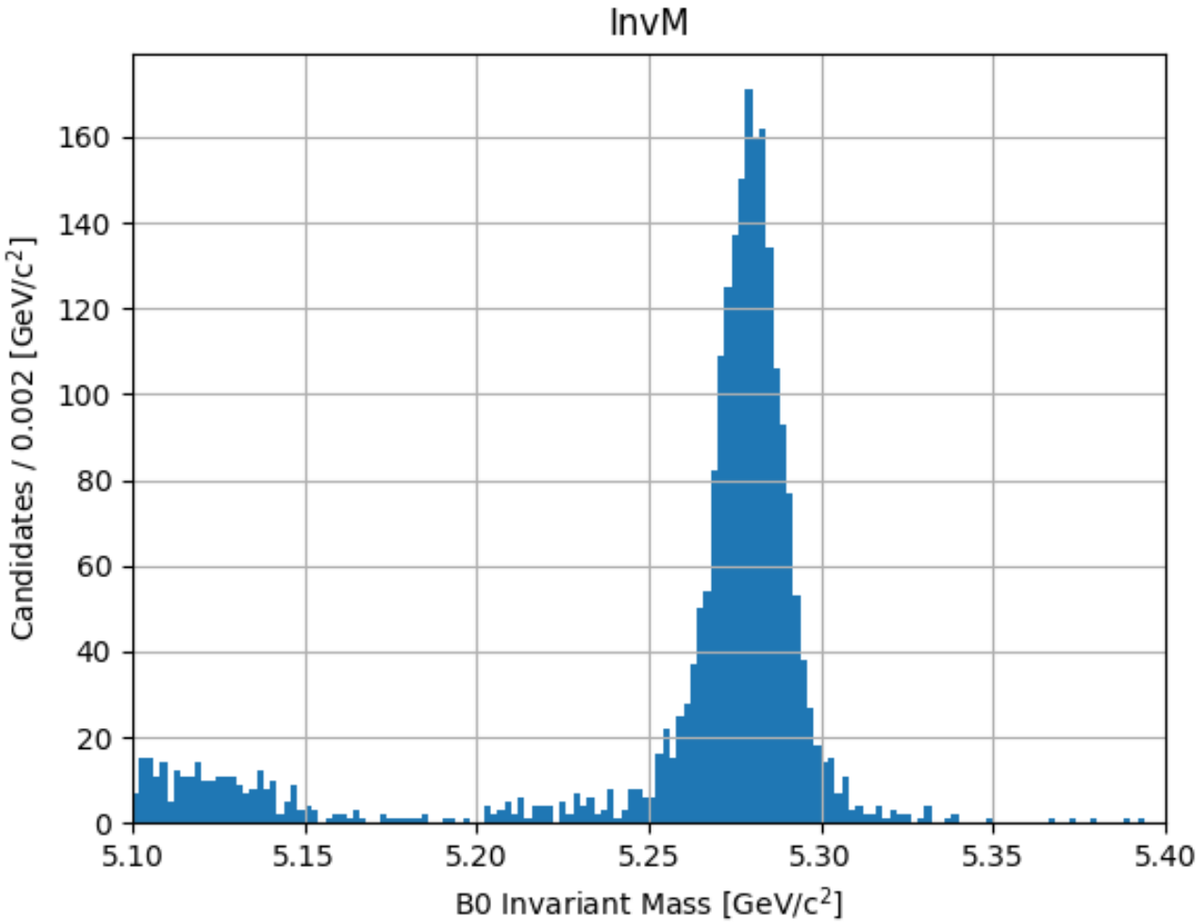
D^0 mass defined in Belle2 Monte Carlo 1.86484 GeV

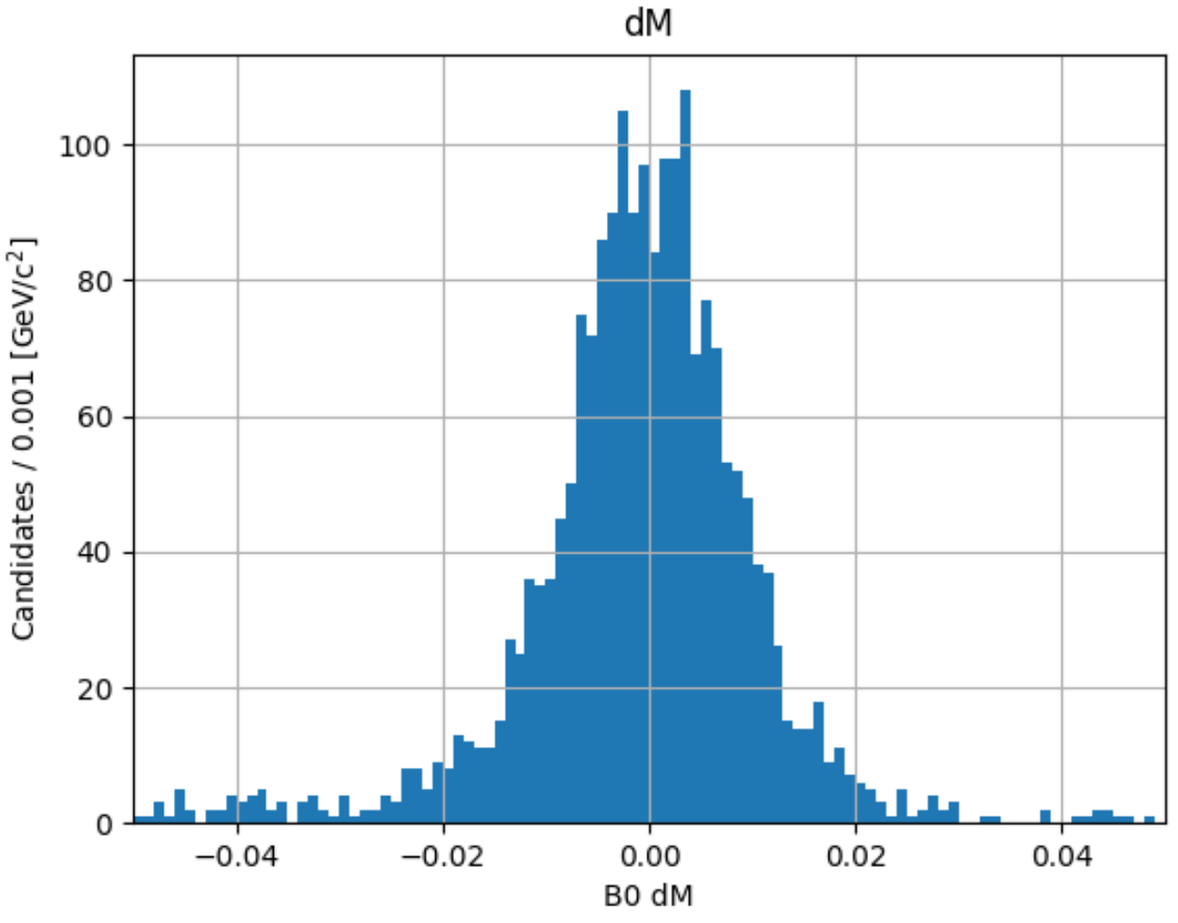




Example 2 Plots:

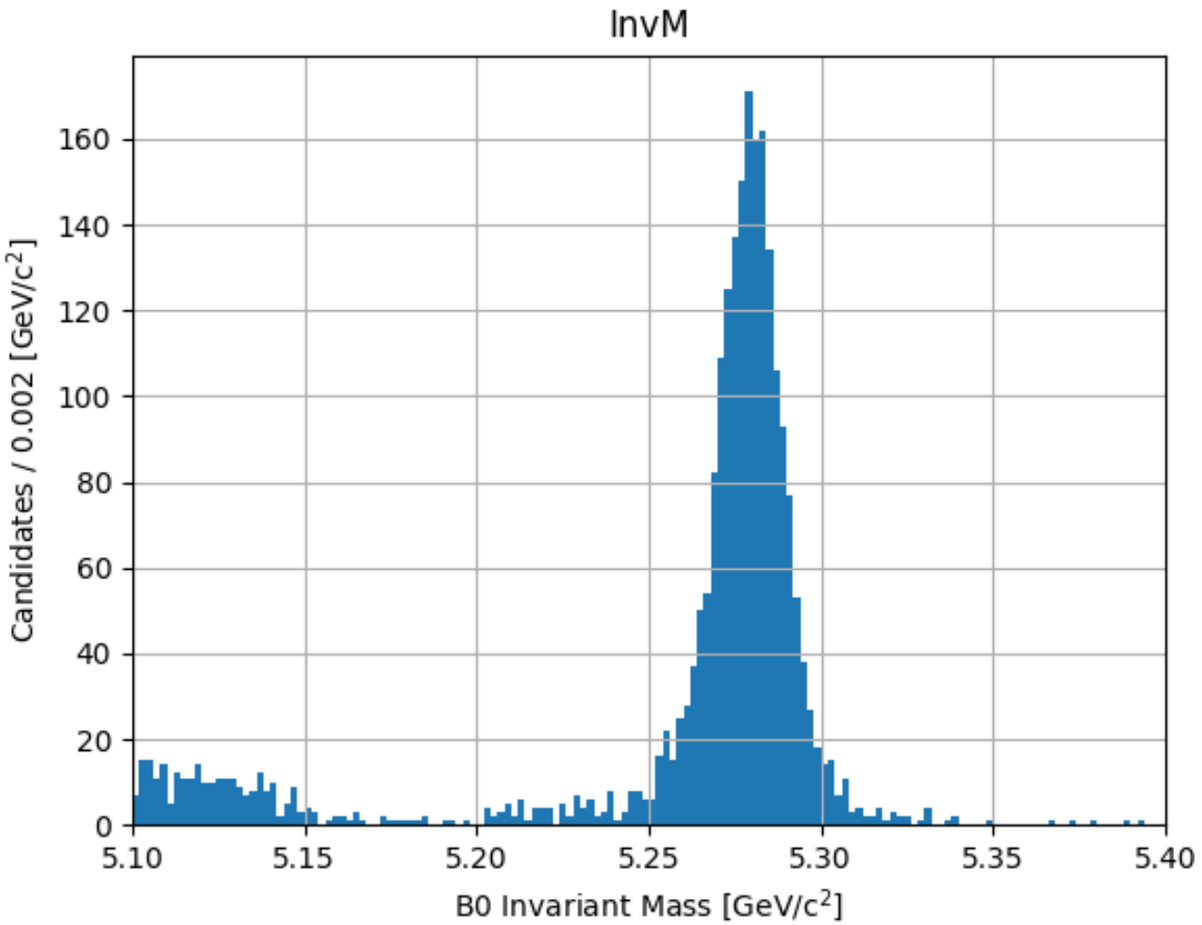
B^0 mass defined in Belle2 Monte Carlo 5.27958 GeV

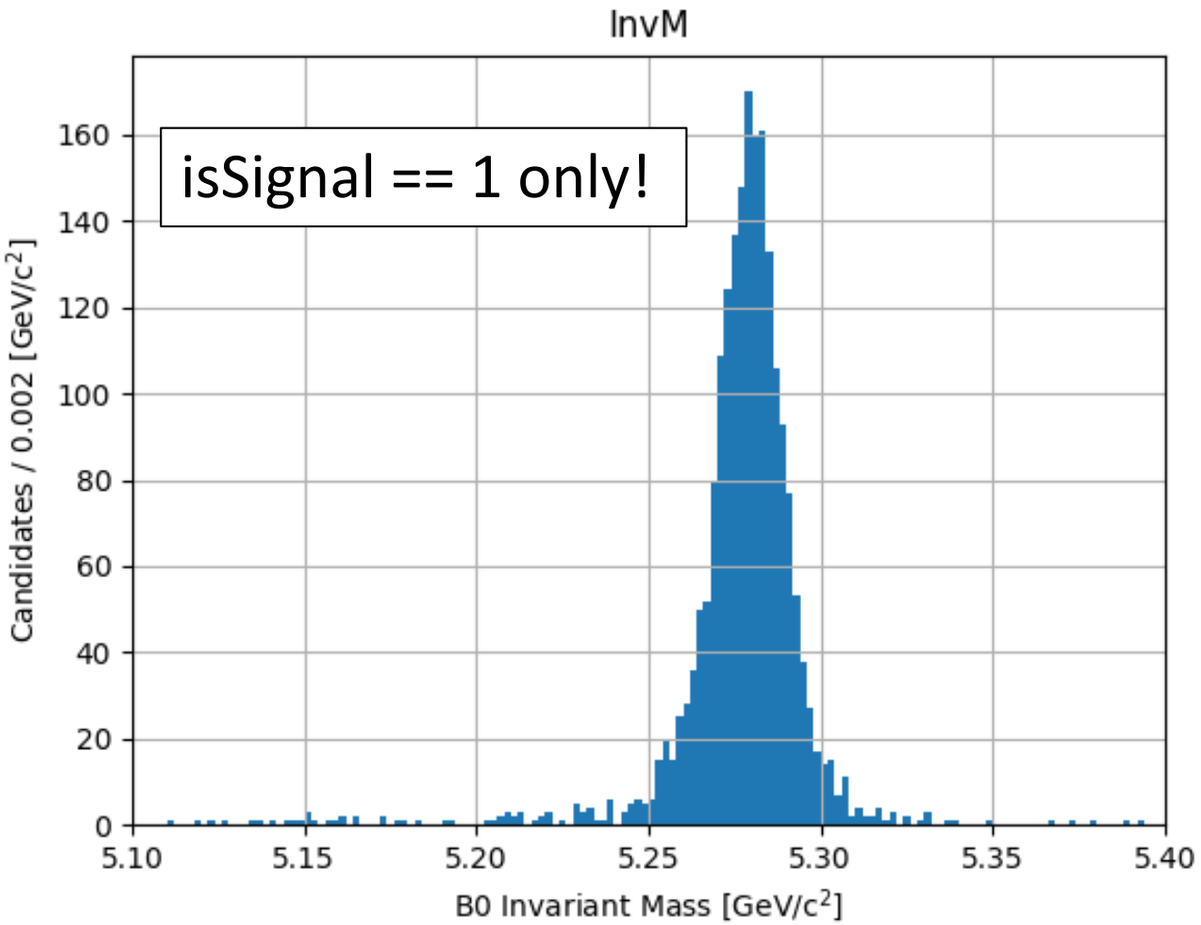




Example 2 Plots:

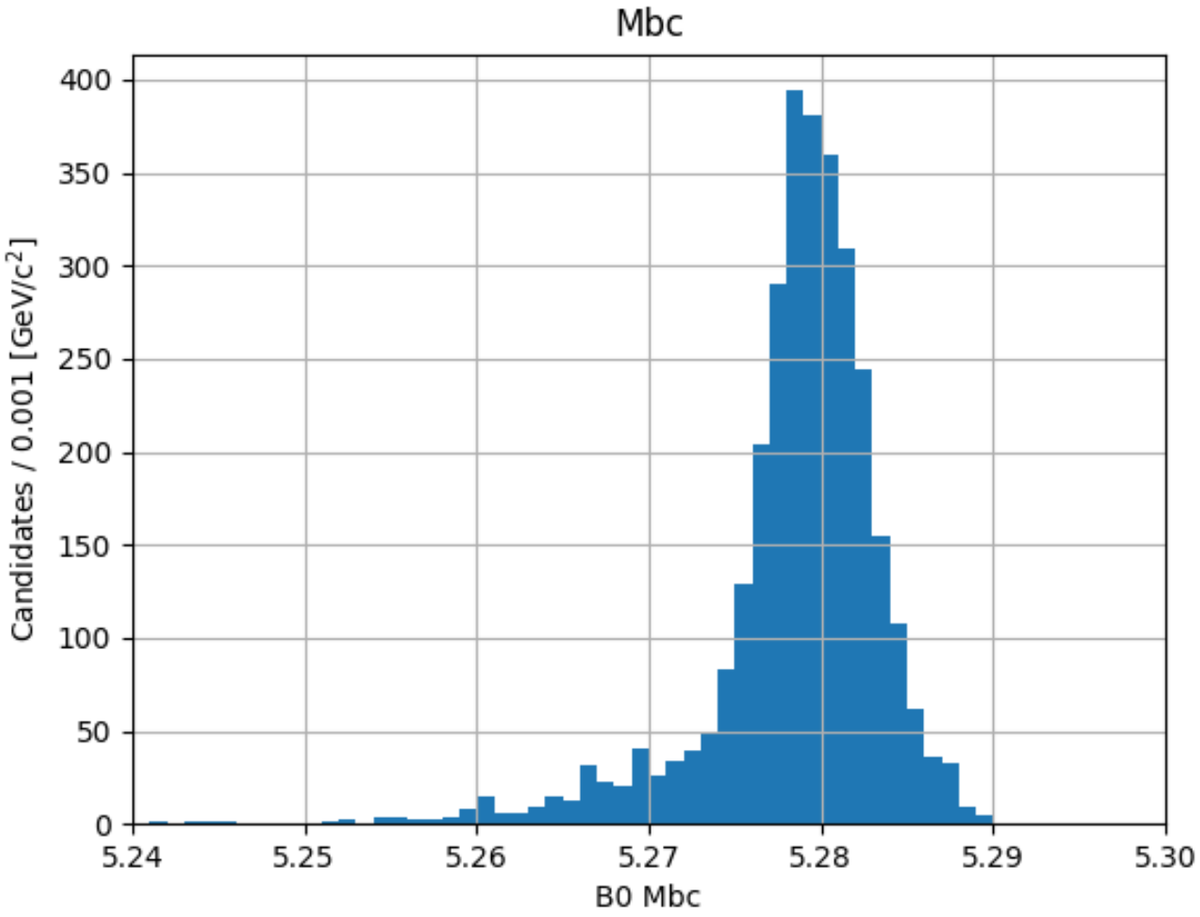
B^0 mass defined in Belle2 Monte Carlo 5.27958 GeV



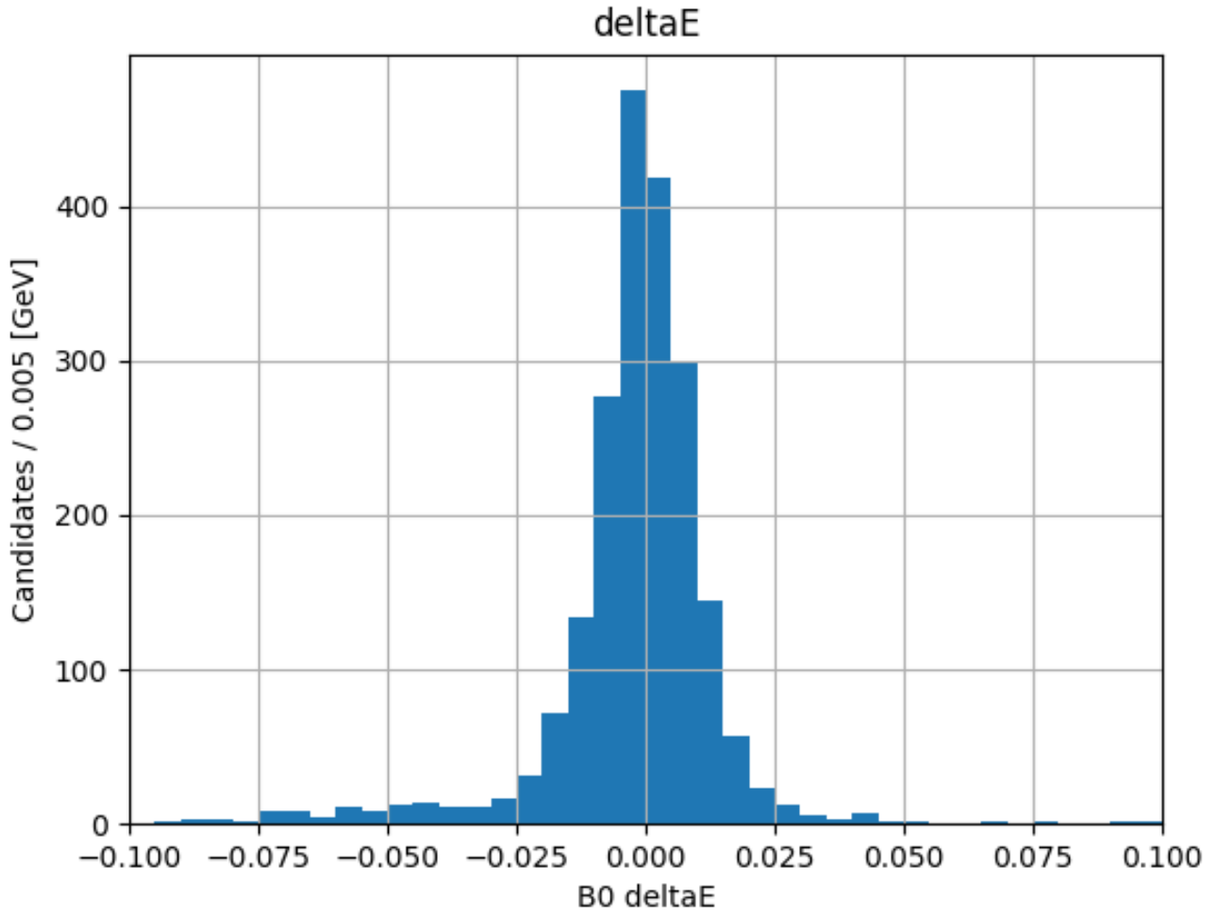


Example 2 Plots:

B^0 mass defined in Belle2 Monte Carlo 5.27958 GeV



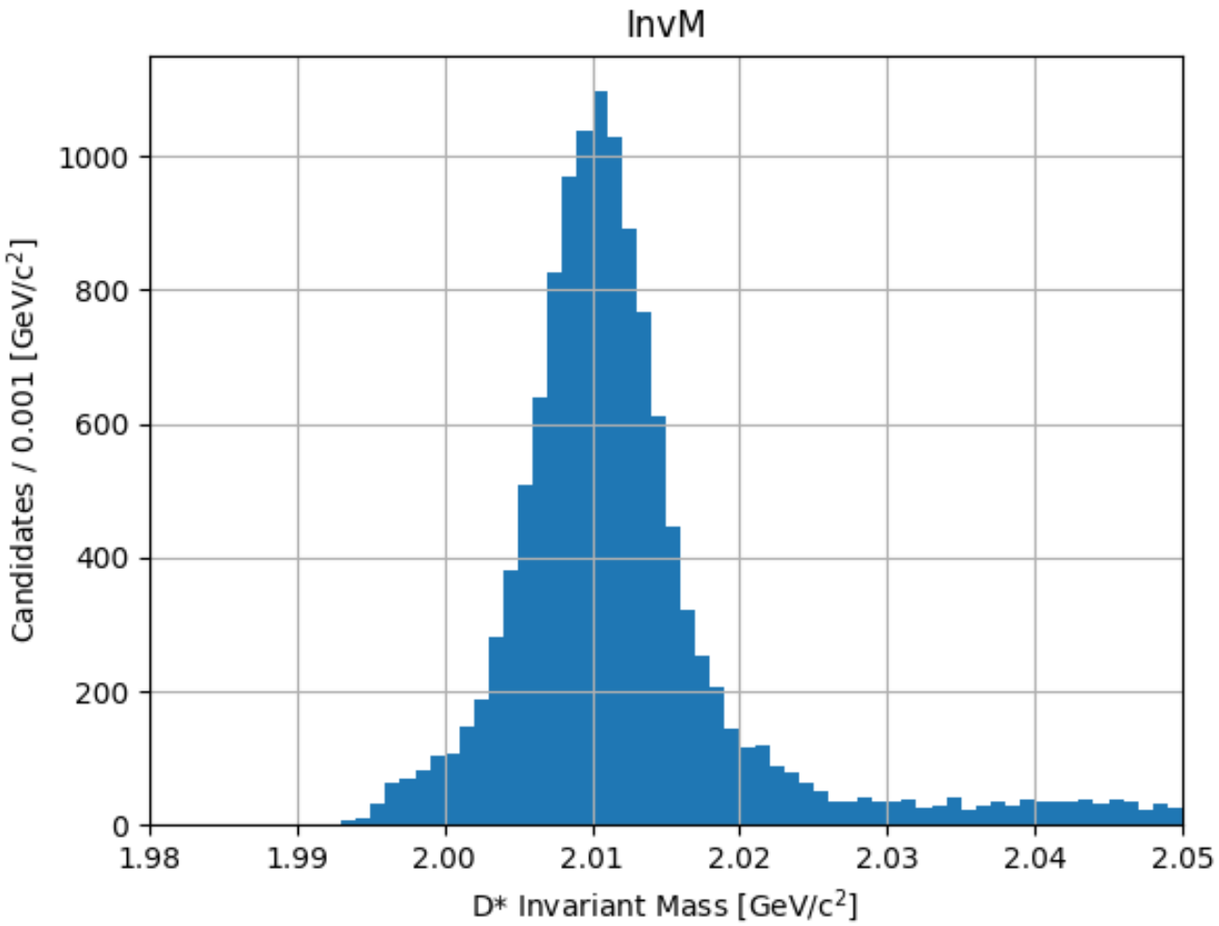
$$M_{bc} = \sqrt{E_{beam}{}^2 - |\overrightarrow{p_{B^0}}|^2}$$

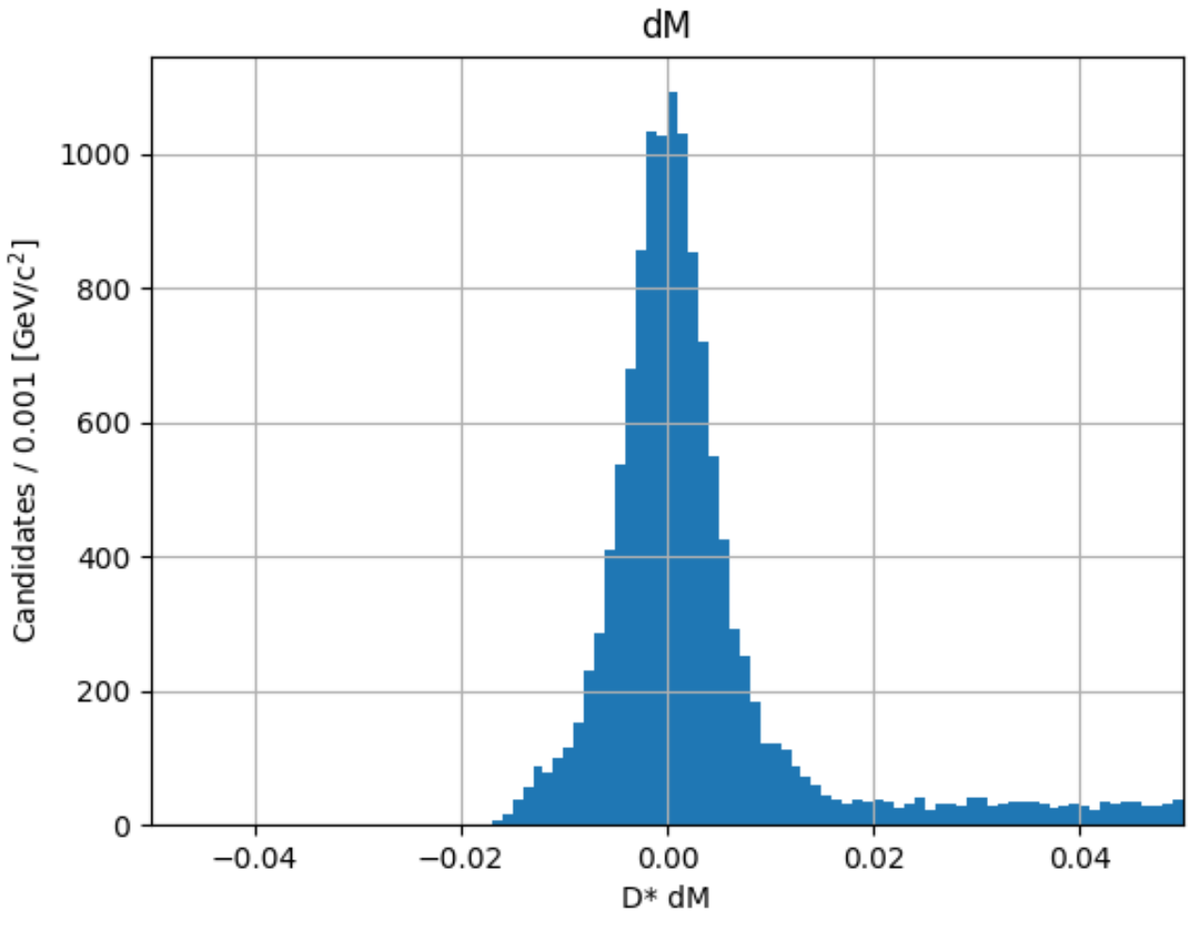


$$deltaE = E_{B^0} - \frac{E_{e^+e^-CM}}{2}$$

Example 2 Plots:

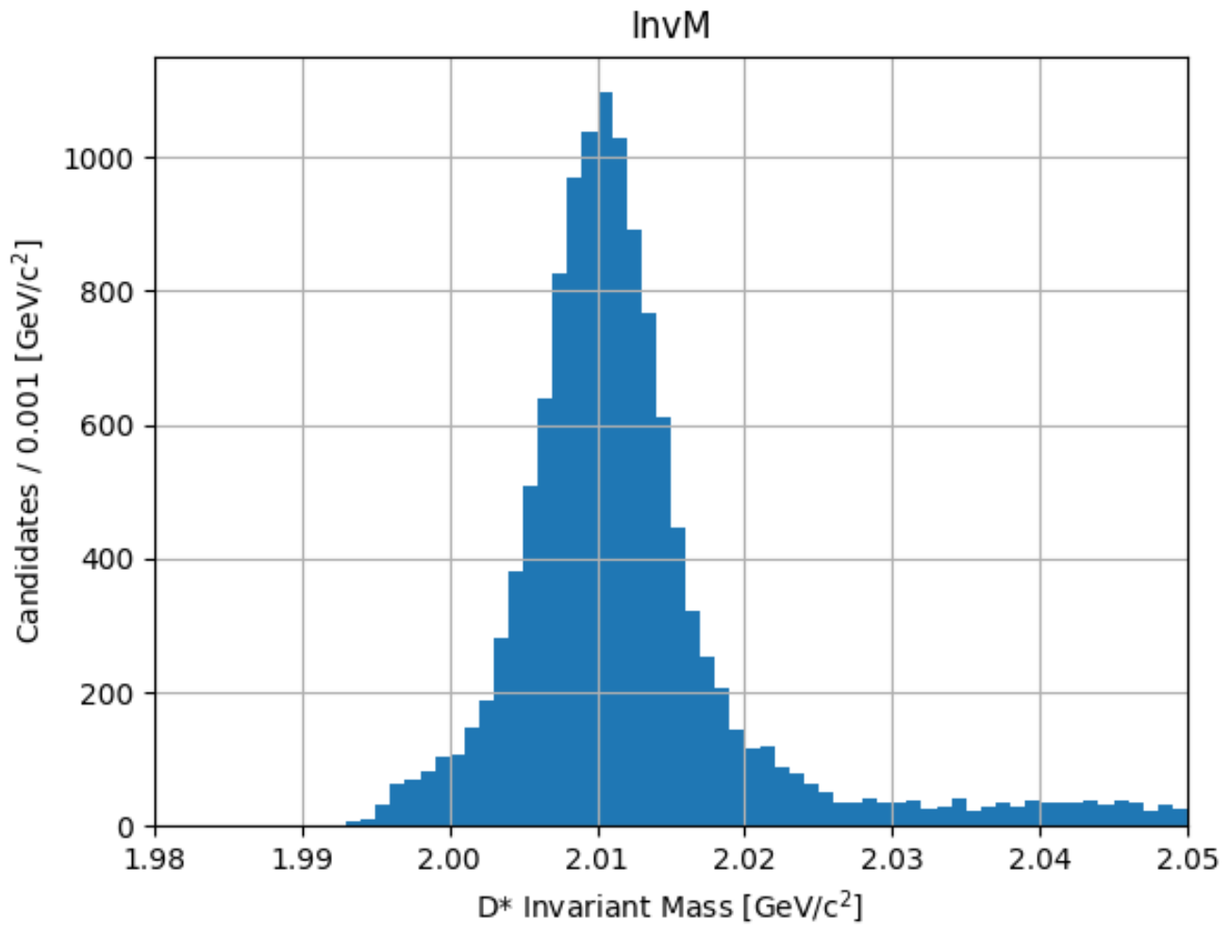
D^{*+} mass defined in Belle2 Monte Carlo 2.00685 GeV

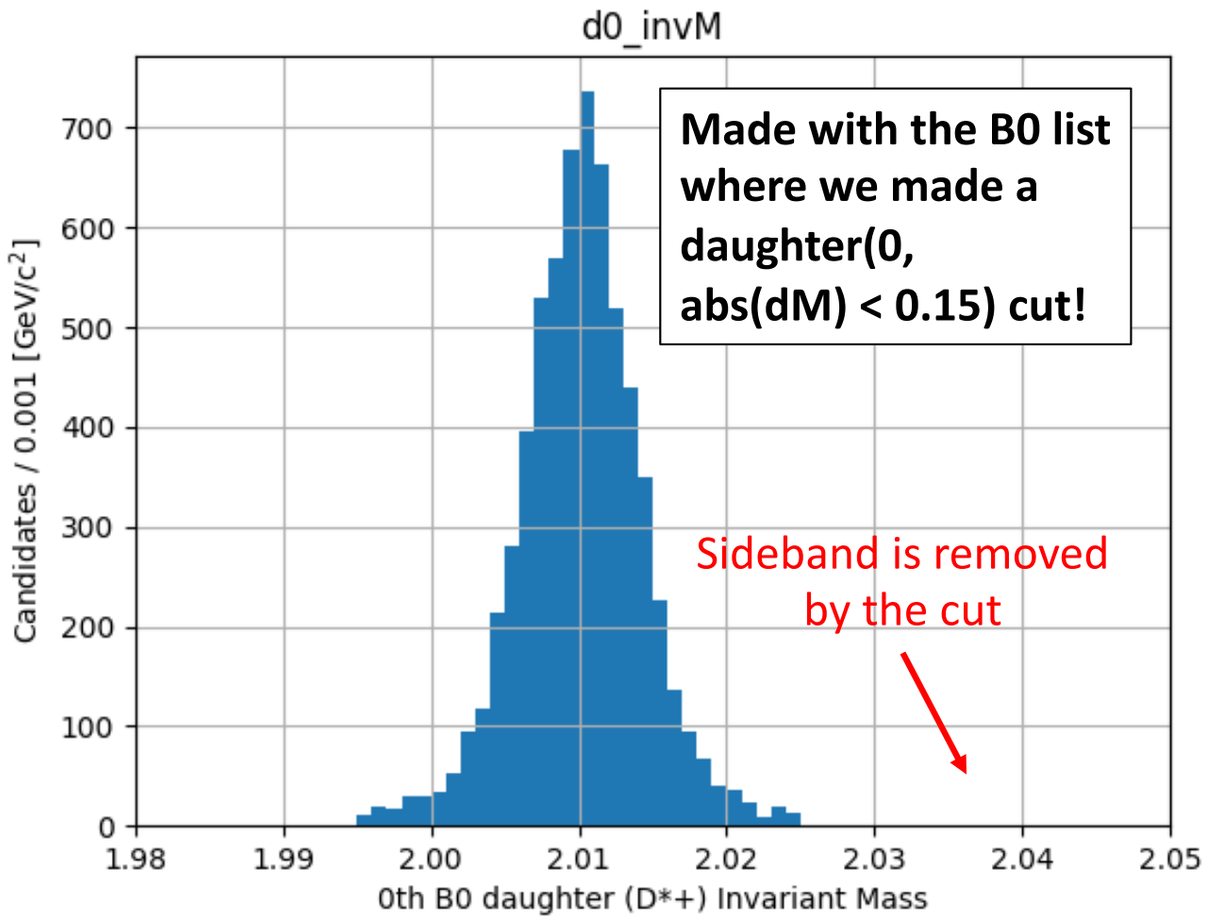




Example 2 Plots:

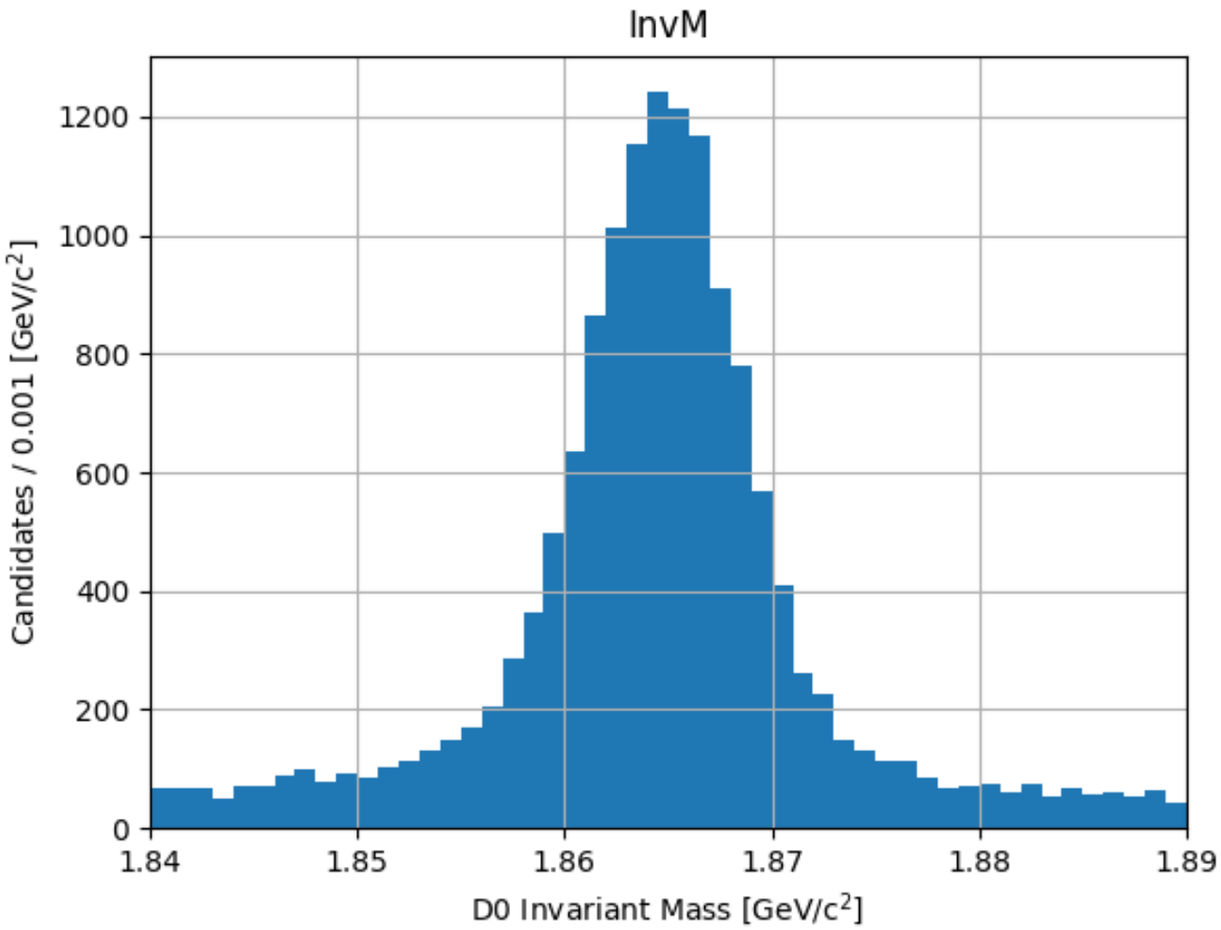
D^{*+} mass defined in Belle2 Monte Carlo 2.00685 GeV

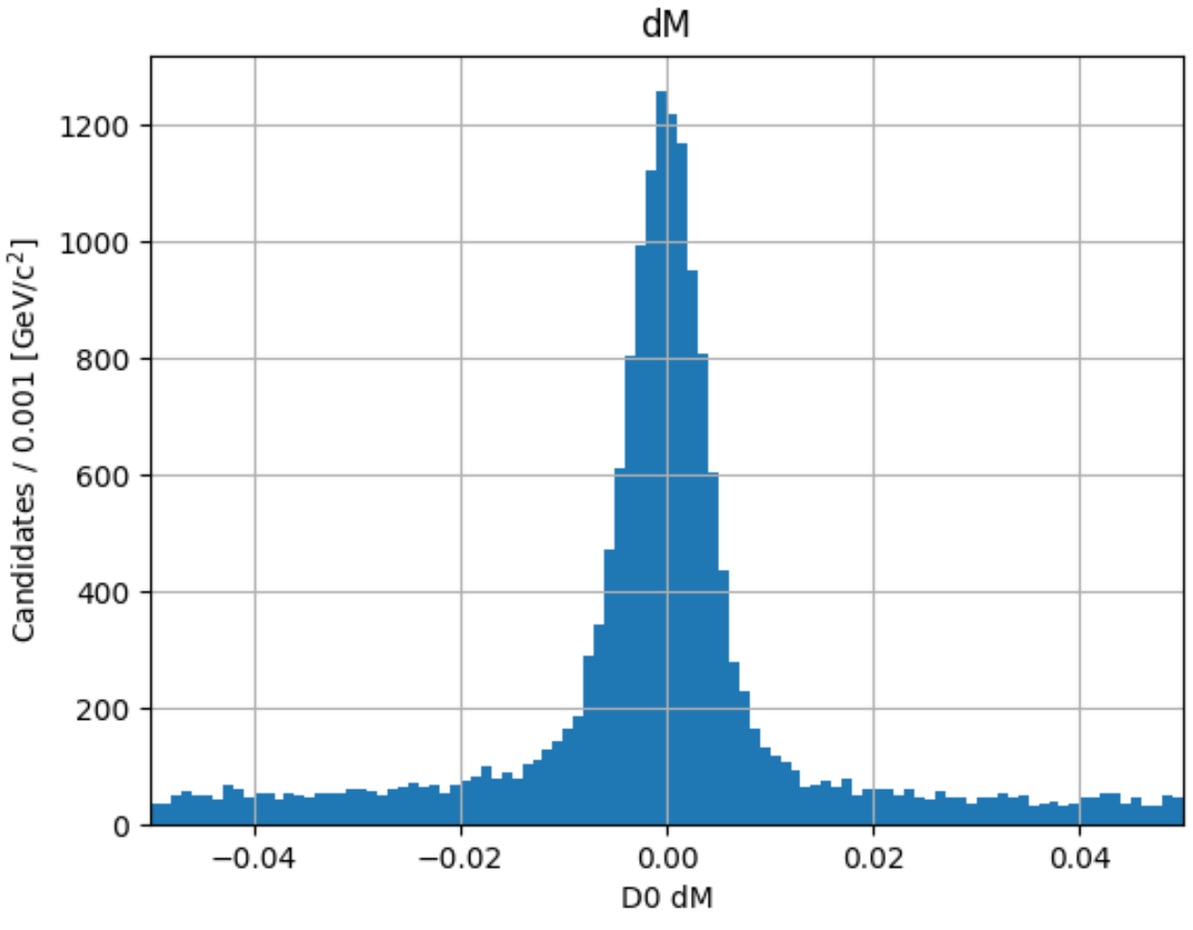




Example 2 Plots:

D^0 mass defined in Belle2 Monte Carlo 1.86484 GeV





Example 2 Plots:

D^0 mass defined in Belle2 Monte Carlo 1.86484 GeV