3.6 Project F2: Make Accurate Measurements of Particle Masses

3.6.1 Goals of project

You will use LHCb data on the invariant mass of particle candidates that you were introduced to during a checkpoint. You will analyse this in a much more sophisticated way, closer to the actual analysis performed leading to its publication. You will use the maximum likelihood process to fit different mass model shapes to the data. From this you will determine the parameters of the mass model for the signal peaks, and their errors. You will start with a very simple Gaussian mass model. You will then improve this and use a more sophisticated model.

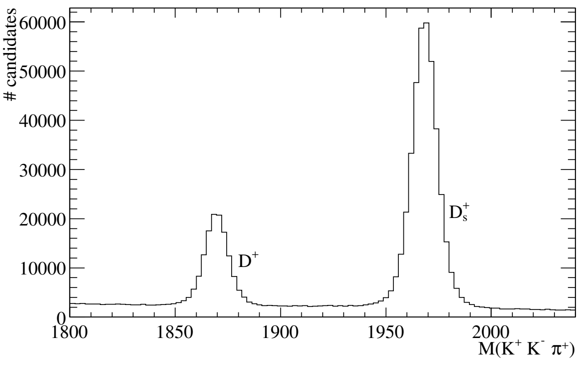
The projects have an open-ended aspect and are an opportunity where you can show your own initiative and demonstrate your experimental and computational skills.

3.6.2 Equipment for project F2

This is a data handling exercise and the Raspberry Pi is not required, but can still be used. You may find it faster to use the CPlab computers — there are several available in the DAH laboratory. Alternatively you could use a personal laptop, but you will need to be able to install Python and Python packages on your own.

3.6.3 Detailed project description

You were previously introduced to the LHCb Upsilon data. In this project you will explore another LHCb dataset collected in 2011 where a pair of oppositely charged kaons and a pion have been combined. Two clear peaks are observed in this mass spectrum corresponding to the D_s^+ (quark content $c\bar{s}$) and D^+ (quark content $c\bar{d}$) mesons (charge conjugation is implied), see figure below For more information see: DOI:10.1007/JHEP06(2013)065.



$K^+K^-\pi^+$ invariant mass distribution.

Download the files kkp.bin and kkpp.bin files from the DAH Dropbox, These files contain the data recorded by LHCb in 2011 for $D^+_{(s)}$ and D^0 decays respectively. Focus on the $D^+_{(s)}$ data file (kkp.bin): The files are written in binary format and contain seven observables

- invariant mass of $K^+K^-\pi^+$ candidate in MeV$/c^2$
- invariant mass of kaon pair in MeV$/c^2$
- transverse momentum of $K^+K^-\pi^+$ candidate in GeV$/c$
- rapidity η of $K^+K^-\pi^+$ candidate
- minimum tranverse momentum of the three tracks in the $K^+K^-\pi^+$ candidate
- electric charge of the candidate
- polarity of the LHCb magnetic field

Write a Python script that reads the data from this file, see below.

../scripts/project_F_b.py

```
import numpy as np

# import data
# xmass = np.loadtxt(sys.argv[1])
f = open("datafiles/kkp.bin","r")
datalist = np.fromfile(f,dtype=np.float32)

# number of events
nevent = len(datalist)/7
xdata = np.split(datalist,nevent)
print(xdata[0])

# make list of invariant mass of events
xmass = []
for i in range(0,nevent):
    xmass.append(xdata[i][0])
    if i < 10:
        print(xmass[i])
```

During checkpoint 6, you performed some very simple “peak finding.” In this project you are going to do the analysis much like it would actually be carried out in a particle physics experiment.

1. Consider first the D^+ peak which is the particle with the lowest mass, i.e. the left most peak in the plot. Construct a composite probability density function (PDF) for the invariant mass of the muon pairs, which contains two components:

    - A Gaussian shape to fit the D^+ mass peak;
    - A shallow falling exponential to fit the background shape of the mass spectrum underneath and around the peak.

2. Use this PDF in a Maximum Likelihood fit to determine the parameters of the PDF. Note that it is essential that the composite PDF remains normalised to 1 over the range of the fit.

    Determine the D^+ meson mass and yield, and all other parameters, and their errors.

    You should be able to obtain the parameter errors directly from the minimization engine of your choice (scipy.optimize.minimise, scipy.optimise.curve_fit, lmfit, see `https://lmfit.github.io/lmfit-py/` or Minuit). Depending on your choice you will be able to chose different minimising methods. It would be good to show that you understand these by obtaining them yourself from the parameters of the Gaussian signal fit — this is described in the data handling lectures.

    Plot the fitted signal shape on top of the data.

3. Now consider the entire mass range, and perform a simultaneous fit for both peaks, and the underlying background. Again you should always report the parameter values, and their errors. Plot the fitted signal shape on top of the data.

4. The results so far probably look quite good by eye, i.e. the signal shape plotted on top of the data probably looks like it fits well. However this can be misleading when performing a precision measurement. You should make a plot of what are called the "residuals." A residual is the difference between the data in the binned histogram and the best-fit mass model value for the centre of that bin. Describe what you see.

5. There are several ways to enhance the scope of the project.

    - If the single Gaussian mass model does not fit the data perfectly, one can try other mass models, i.e. a signal PDF that goes beyond a single Gaussian function. One example is a PDF comprising a function which is the sum of two Gaussian functions (i.e. one narrow and one wide Gaussian function to fit a single D meson peak). Alternatively try a Crystal Ball function, which incorporates a non-Gaussian tail at the lower end of the mass peak. The functional shape is described elsewhere, e.g. see: `https://en.wikipedia.org/wiki/Crystal_Ball_function`. You could implement each of these functions in your PDF and see how much better they are at describing the data.
    - Read the publication and see what is said about systematic errors. Make a reasonable attempt at determining some systematic errors on the masses.
    - Compare your results to the PDG and previous measurements.
    - Study how the mass and the width of the peak (the resolution) depends on the transverse momentum ($p_\perp$) and rapidity (η).

3.6.4 Project planning

The project descriptions are generally significantly less detailed than what was made available for the checkpoints. Any material covered during checkpoints including python code examples are assumed to be known. Only essential and new information is provided and you are expected to take care of the details. Python code snippets are provided where necessary, but you will have to understand yourself what they do. It is recommend that you google for information about your project on the web, including data sheets of components and python libraries, if applicable. Python scripts should be well structured, either using functions or classes.

The timeline will vary between different projects, but in general, it is recommended that you plan your work as follows:

- weeks 7, 8 & 9: Building your gadget and/or writing code for project;
- week 9, 10: Analysis of data or equivalent, prepare supplementary material;
- week 10, 11: Finish writing of project report and prepare submission.

Note that you are advised to start writing your report as the project progresses.

For guidance on report writing, how the projects will be assessed, plagiarism and the submission deadline, please consult the DAH course booklet and the DAH grade descriptors, available on Learn.