

Teilchenphysik II (Higgs-Physik) (SS 2016)

Institut für Experimentelle Teilchenphysik

Dr. R. Wolf, Dr. S. Wayand

<http://www-ekp.physik.uni-karlsruhe.de/~quast/>

Exercises Sheet 09
Release: Thu, 03.06.2016

Exercise 29: Monte Carlo Event Generation (computing exercise)

In this exercise you will install and use the event generator `pythia` to study distributions of signal and background processes. The environment to run this and other event generators is readily provided in the form of a “virtual machine” based on the operating system Linux (Ubuntu 14.04). Some documentation on the Monte Carlo event generator `pythia` will be provided together with this exercise.

a) Download the virtual machine from the web page:

<http://www-ekp.physik.uni-karlsruhe.de/~quast/VMroot/>

and follow the instructions in the file `VMroot.pdf`. You should now be able to start your virtual machine and to log in. This virtual machine already contains an installed version of the data analysis package `root` and some examples in the subdirectory

`~/root`.

The next step is the installation of the `pythia8` Monte Carlo event generator. From within the virtual machine, open the web browser and change to:

<http://www-ekp.physik.uni-karlsruhe.de/~quast/VMroot/addons/pythia/>

Download all packages and files in this directory:

- `pythia8.help`,
- `install_pythia8.tgz`,
- `userdata_pythia8.tgz`.

Follow the instructions in `pythia8.help` to install the `pythia8` Monte Carlo event generator in the directory

`/usr/local`

and the files needed to execute it in the user-directory

`~/generators`.

In order to set the paths to all libraries, execute the configuration file with the command

`source setpythia`.

b) In your virtual machine, change to the directory

```
~/generators/examples.
```

This directory contains some examples (`mainNN.cc`) and a `Makefile` to produce the executable files. Type

```
make main01.
```

This will compile the program `main01.cc`, link it against the pythia8 libraries to produce an executable file `main01.exe`. Running the program via

```
./main01.exe> out01
```

writes the output to the file `out01`. First, inspect the source code in `main01.cc` to understand:

- how pythia is set up;
- how a simple histogram is booked;
- how the event loop works;
- how variables in the generated event are accessed and filled into the histogram;
- how to obtain the output at the end.

Now have a look at the result file `out01`. It contains details about the run that has been performed, the parameter settings and shows a full event listing with all generated intermediate and final state particles. At the end, there is information on the calculated cross section of each relevant sub-process, and finally the histogram showing the distribution of the number of stable charged particles.

Inspect some more of the examples and run them. The second one shows the distribution of the transverse momentum of Z bosons produced in collisions of protons and anti-protons. **Example 03** uses a “configuration file” `main03.cmd` to configure pythia. The configuration file defines the beam parameters (=initial particles) and the sub-processes of interest. In the C++ code a number of histograms are booked and filled. **Example 03** is a good model of typical studies with pythia8 at “generator level”.

c) You are now ready for your first own exercise. The aim is to simulate a signal process, here the signal of *supersymmetric* Higgs bosons decaying to two muons, on top of a background of $\text{di-}\mu$ events originating from the decay $Z^{(*)}/\gamma^* \rightarrow \mu\mu$. The Higgs boson has an assumed mass of 150 GeV. Your task is to separate the signal from the background by looking at the distribution of a highly-discriminating variable, the invariant $\text{di-}\mu$ mass. To get started, download the templates from:

```
http://www-ekp.physik.uni-karlsruhe.de/~quast/vorlesung/TP2HiggsSS16/Uebungen/pythia-exercise.tgz
```

and install it under your account by unpacking the archive. Now change to the work directory:

```
cd generators/pythiaexercise
```

and type

```
source setpythia.sh
```

to set up the necessary paths to *root* and *pythia8*. In the directory you find the *Makefile* for the three example programs provided. Type

```
make
```

to produce the executables *runcmnd.exe*, *EventRecord.exe* and *NtupleProducer.exe*. The program *runcmnd.cc* is a variant of the program *main03.cc*, which accepts the name of a *pythia* configuration file as an argument on the command line and then starts *pythia*. Run it with the files *Higgs.cmd* and *DrellYan.cmd* as inputs and note down the cross sections of the two processes.

Now you have two choices: the analysis can be implemented completely inside the program *runcmnd.cc*, or you can use the program *NtupleProducer.cc* as an example. This program writes a *root* ntuple with relevant variables, which can then be analyzed using *root*. Depending on your choice, calculate the appropriate variables and store them in *pythia* histograms, or modify the source code *NtupleProducer.cc* to write out the variables of the event to an n-tuple, as indicated in the code. In practice, we mostly choose the latter option, as the n-tuple resembles the output format of the real experiment, which is passed to the same analysis program as the data. If you choose option two, you have to write a *root* macro that opens the n-tuples one after the other, calculates the invariant mass of muon pairs, and fills them into histograms. The first option is simpler, and also sufficient for this exercise.

Next, implement the detector acceptance, in this case given by a range in rapidity, $|y| < 2.4$, and a threshold of the transverse momentum, $p_T > 25$ GeV, for each muon. Fill the invariant mass distributions if both muons are within the acceptance. Looking at the problem more carefully, you will notice that you need to generate a large number of background events to sufficiently populate the region where you expect the signal to be. Find a suitable number of events to generate, modify the configuration file *DrellYan.cmd* and re-run the background simulation. In order to study the amount of signal expected on top of the Drell-Yan background, the invariant mass distributions of signal and background need to be scaled appropriately and displayed on the same graph. To do this correctly you need to scale both histograms to an identical integrated luminosity (as you did for **Exercise 28 d**), which is obtained from (i) the number of events that you have produced and; (ii) the cross sections of the processes you are looking at. For Drell-Yan production you can use the cross section that has been given for **Exercise 13 c**). For the signal process you can use a production cross section of 0.25 pb(!). This cross section corresponds to a real prediction of the minimal supersymmetric SM **hMSSM scenario** for the model parameters $m_A = 150$ GeV (corresponding to the mass of the new Higgs boson “A”) and $\tan\beta = 15$ (corresponding to the ratio of the vacuum expectation values for the two Higgs boson doublets in this model). From the common graph estimate the expected signal significance for an integrated luminosity of 300 fb^{-1} . To estimate the significance define your signal region within two standard deviations around the center of the signal peak at 150 GeV.