

## Exercises on Parameter Estimation

**Exercise 1:** This exercise concerns maximum-likelihood fitting with the minimization program MINUIT, using either its python implementation `iminuit` or the root/C++ version `TMinuit`. The programs provided generate a data sample of 200 values from a pdf that is a mixture of an exponential and a Gaussian:

$$f(x; \theta, \xi) = \theta \frac{1}{\sqrt{2\pi}\sigma} e^{-(x-\mu)^2/2\sigma^2} + (1-\theta) \frac{1}{\xi} e^{-x/\xi}, \quad (1)$$

The pdf is modified so as to be truncated on the interval  $0 \leq x \leq x_{\max}$ . The program `Minuit` is used to find the MLEs for the parameters  $\theta$  and  $\xi$ , with the other parameters treated here as fixed. You can think of  $\theta$  as representing the fraction of signal events in the sample (the Gaussian component), and the parameter  $\xi$  characterizes the shape of the background (exponential) component.

To use python, you will need to install the package `iminuit` (should just work with “pip install `iminuit`”). See:

<https://pypi.org/project/iminuit/>

Then download and run the program `mlFit.py` or the jupyter notebook `mlFit.ipynb` from

<http://www.pp.rhul.ac.uk/~cowan/stat/python/iminuit/>

To use C++/ROOT, download the files from

<http://www.pp.rhul.ac.uk/~cowan/stat/root/tminuit/>

to your work directory and build the executable program by typing `make` and run by typing `./simpleMinuit`. This uses the class `TMinuit`, which is described here:

<https://root.cern.ch/doc/master/classTMinuit.html>

To use python, start with the program

<http://www.pp.rhul.ac.uk/~cowan/stat/python/mlFit/mlFit.py>

To use C++/ROOT, use the files from

<http://www.pp.rhul.ac.uk/~cowan/stat/root/mlFit/>

The instructions below refer to the python version; the corresponding steps for the C++/ROOT program are similar.

**1(a)** By default the program `mlFit.py` fixes the parameters  $\mu$  and  $\sigma$ , and treats only  $\theta$  and  $\xi$  as free. By running the program, obtain the following plots:

- the fitted pdf with the data;

- a “scan” plot of  $-\ln L$  versus  $\theta$ ;
- a contour of  $\ln L = \ln L_{\max} - 1/2$  in the  $(\theta, \xi)$  plane.

From the graph of  $-\ln L$  versus  $\theta$ , show that the standard deviation of  $\hat{\theta}$  is the same as the value printed out by the program.

From the graph of  $\ln L = \ln L_{\max} - 1/2$ , show that the distances from the MLEs to the tangent lines to the contour give the same standard deviations  $\sigma_{\hat{\theta}}$  and  $\sigma_{\hat{\xi}}$  as printed out by the program.

**1(b)** Recall that the inverse of the covariance matrix variance of the maximum-likelihood estimators  $V_{ij} = \text{cov}[\hat{\theta}_i, \hat{\theta}_j]$  can be approximated in the large sample limit by

$$V_{ij}^{-1} = -E \left[ \frac{\partial^2 \ln L}{\partial \theta_i \partial \theta_j} \right] = - \int \frac{\partial^2 \ln P(\mathbf{x}|\boldsymbol{\theta})}{\partial \theta_i \partial \theta_j} P(\mathbf{x}|\boldsymbol{\theta}) d\mathbf{x} , \quad (2)$$

where here  $\boldsymbol{\theta}$  represents the vector of all of the parameters. Show that  $V_{ij}^{-1}$  is proportional to the sample size  $n$  and thus show that the standard deviations of the MLEs of all of the parameters decrease as  $1/\sqrt{n}$ . (Hint: write down the general form of the likelihood for an i.i.d. sample:  $L(\boldsymbol{\theta}) = \prod_{i=1}^n f(x_i; \boldsymbol{\theta})$ . There is no need to use the specific  $f(x; \boldsymbol{\theta})$  for this problem.)

**1(c)** By modifying the line

```
numVal = 200
```

rerun the program for a sample size of  $n = 100, 400$  and  $800$  events, and find in each case the standard deviation of  $\hat{\theta}$ . Plot (or sketch)  $\sigma_{\hat{\theta}}$  versus  $n$  for  $n = 100, 200, 400, 800$  and comment on how this stands in relation to what you expect.

**1(d)** In python by modifying the line

```
parfix = [False, True, True, False]           # change these to fix/free parameters
```

or in C++/Root by using the TMinuit routines `FixParameter` and `Release`, find  $\hat{\theta}$  and its standard deviation  $\sigma_{\hat{\theta}}$  in the following four cases:

- $\theta$  free,  $\mu, \sigma, \xi$  fixed;
- $\theta$  and  $\xi$  free,  $\mu, \sigma$  fixed;
- $\theta, \xi$  and  $\sigma$  free,  $\mu$  fixed;
- $\theta, \xi, \mu$  and  $\sigma$  all free.

Comment on how the standard deviation  $\sigma_{\hat{\theta}}$  depends on the number of adjustable parameters in the fit.