# Astrostatistics School: requirements.

*By S. Andreon*

Participants to the school are requested to install a few software packages and to write some reading/plotting routines prior to the beginning of the school. Failing to do so will significantly impair the learning outcomes.

Attendees should:

1. have their own computer (with an internet connection), and be acquainted with it;
2. have installed JAGS[1] which in turn may demand the installation of some additional libraries;
3. it may be useful to browse the table of distributions (e.g., page 30 of version 2 of the user manual) to refresh your memory about the mathematical expression of some famous functions;
4. be able to make plots and simple data manipulation using their preferred environment. In particular, attendees should have already written routines for:

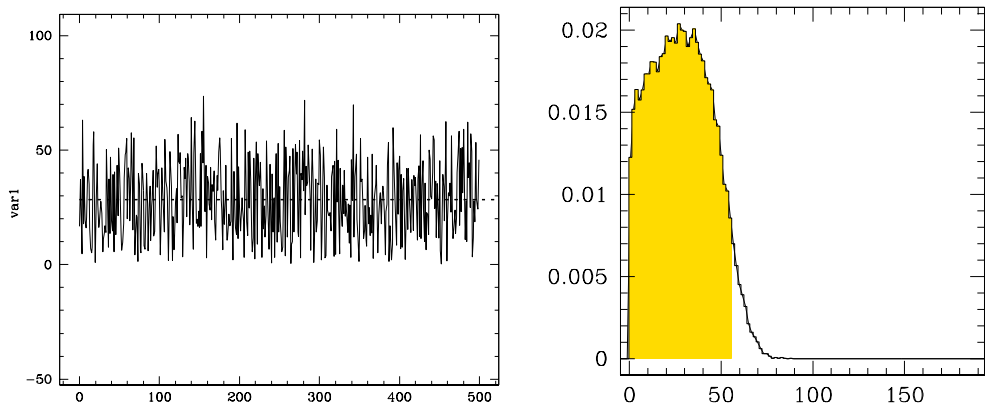


Figure 1: Left panel: Trace plot. Right panel: Marginal distribution (histogram)

- properly reading files in the CODA format: CODAindex.txt[2] describes the content of the CODAchain1.txt[3] file by listing the variable names, where they start and where

[1] https://sourceforge.net/projects/mcmc-jags/

[2] http://www.brera.mi.astro.it/%7Estefano.andreon/corso_metodi_bayesiani/CODAindex.txt If you experience difficulties in reaching this and other links, do not cut& paste them. Instead, type them from scratch.

[3] http://www.brera.mi.astro.it/%7Estefano.andreon/corso_metodi_bayesiani/CODAchain1.txt

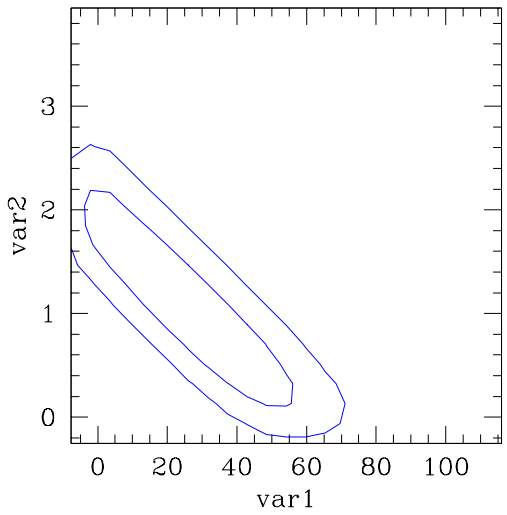


Figure 2: Contours. In this plot, contours are imprecisely determined close to $var1 = 0$ e $var2 = 0$ boundaries, and not corrected for smoothing effects. You are not asked to do better (but nothing precludes you from doing it).

they end. For example (inspect CODAindex.txt) the variable $s$ starts at line 1 and ends at 50,000 (and it is on the 2nd column) of CODAchain1.txt. The reading routine should work for any number of variables (e.g. 10) and of samples (e.g. 30,000).

- compute mean and standard deviation (check that $s$ has mean 28.5 and standard deviation 16.5)
- compute the shortest interval including x % of the samples (check that the 95% interval of $s$ is [0,56]). To compute it, you may, for example, start from the peak of the pdf and move down until they get x% of the samples.
- produce a trace plot, i.e., a plot that gives the variable value as a function of its rank (or step in the chain), as in Fig. 1. This plotting routine should work also if CODAindex.txt contains, say, 10 variables.
- produce normalized histograms, as in Fig. 1, right panel (note that the integral must be unity and independently of bin size).
- draw contours. The routine should work for non-elliptical contours, for example when one has two separate "islands". The contours should include about 68% and about 95% of the samplings. A small margin of error is allowed (i.e. 70% in place of 68% is fine). It is instead not allowed to draw contours at pre-defined thresholds (e.g, taking the peak value and dividing by a "magic number"). Check your contours against those in Fig. 2 with the sampling in CODAchain1.txt. The latter contours are somewhat approximated (and nothing better than this is required!).
- compute the mean $y$ for each (small) bin of $x$, plot it at the mean $x$ of the bin, using the sample generated by JAGS available at this URL[4]. Its CODAindex is here[5]. The found points should be roughly aligned on the line $y = x/2$. When done, make the reverse: compute instead the mean $x$ per each (small) bin of $y$ and plot mean values.

[4] http://www.brera.mi.astro.it/%7Estefano.andreon/corso_metodi_bayesiani/CODAchain_fakesampleregr.txt
[5] http://www.brera.mi.astro.it/~stefano.andreon/corso_metodi_bayesiani/CODAindex_fakesampleregr.txt

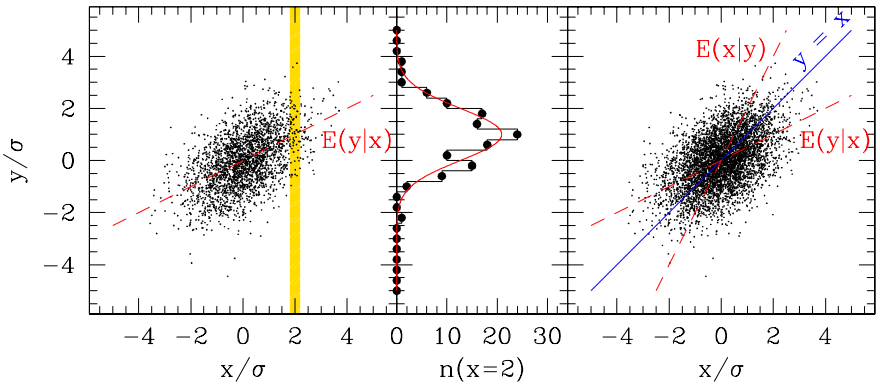


Figure 3: Left panel: 500 points drawn from a bivariate Gaussian, overlaid by the line showing the expected value of $y$ given $x$. The yellow vertical stripe captures those $y$ for which $x$ is close to 2. Central panel: Distribution of the $y$ values for $x$ values in a narrow band of $x$ centred on 2, as shade d in the left panel. Right panel: as the left panel, but we also add the lines joining the expected $x$ values at a given $y$, and the $x = y$ line.

The found points should be roughly aligned on the line $y = 2x$. Figure 3 is a (nicer) illustration of the idea (the red lines connect the computed points). You are not asked to exactly reproduce this figure, plotting the means (and checking them against the lines) suffices.

N.B. If you use Python or R, you got for free much of the above (and much more, e.g., corner.py and pyjags), but this may imply to learn a new plotting language.

## Checking JAGS istallation

In order to check having properly installed JAGS, first, save the file below as model.bug and make you sure that it only contain ASCII

```
# Bayesian Methods for the Physical Science. Learning from
# Examples in Astronomy and Physics. By S. Andreon and B. Weaver.
model {
for (i in 1:length(nrec)) {
 nrec[i] ~ dbin(eff[i],ninj[i])
 nrec.rep[i] ~ dbin(eff[i],ninj[i])
 eff[i] <- A + (B-A)*phi((E[i]-mu)/sigma)
}
A~dunif(0,1)
B~dunif(0,1)
mu~dunif(0,100)
sigma~dunif(0,100)
}
```

Second, save the file below as model.cmd and check that it only contain ASCII

```
model in model.bug
data in data.dat.R
compile,nchains(1)
initialize
update 3000
monitor set A, thin(10)
monitor set B, thin(10)
monitor set mu, thin(10)
update 100000
coda *
data to testata
samplers to testsamplers
exit
```

Now, using the data listed at this URL[6] (save the file as data.dat.R), run JAGS redirecting the standard input from the file model.cmd. Under linux the command to execute is

```
jags < model.cmd
```

provided that the jags executable is in the path. If a file CODAchain1.txt is produced, then JAGS has been properly installed.

[6] http://www.brera.mi.astro.it/%7Estefano.andreon/BayesianMethodsForThePhysicalSciences/data8.2.dat.R