Astrostatistics

Danail Obreschkow

Harley Wood School of Astronomy, 6 July 2020

Prerequisites

This markdown document supports a 1-hour lecture on Astrostatistics focussed on Bayesian inference. The lecture was targeted to higher-degree research students in astronomy and cosmology.

You will need some familiarity with the programming language **R** to run this markdown document interactively. I recommend using **R** version 4.x with **RStudio** 1.3.x (or later). They are both available for free on all common operating systems. If you have not used **R** before, I recommend the introductory videos by Aaron Robotham available online (https://www.youtube.com/watch?v=EDhDwz4uoJ8).

In order to compile this document requires a series of packages listed below. Normally these packages are installed automatically upon opening the markdown document in **Rstudio**. Otherwise you can install them manually using the **install.packages** command:

```
install.packages(c('magicaxis','pracma','ellipse','hyper.fit'))
```

The only exception is the **cooltools** package, which is not yet on CRAN (the "Central R Archive Network"), but can be downloaded from github using the **devtools** package:

```
install.packages('devtools')
library(devtools)
install_github('obreschkow/cooltools')
```

Load required libraries:

```
library(magicaxis)
library(pracma)
suppressMessages(library(cooltools))
suppressMessages(library(ellipse))
suppressMessages(library(hyper.fit))
```

Foreword: "Astrostatistics" – is this really a thing?

Astrostatistics is not a new type of statistics, unlike, say, statistical physics. The term "astrostatistics" rather denotes the set of statistical tools regularly involved in the analysis of astronomical and cosmological data. These tools include spatial statistics, time series analysis, multivariate regression and, especially, Bayesian inference. The reason that astrostatistics has nonetheless become a recognized term is that statistics appears to be (even) more important in astronomy and cosmology than in most other fields of science. There are at least three good reasons for this:

- Astronomical data are generally very expensive in terms of the time, expertise and financial resources required to gather them. Because of this high price, it is often economical to invest a considerable effort into statistical analysis tools that allow us to extract the last bit of information contained in the data.
- Astronomical evidence must rely on *observations* of the universe as it is, rather than on simplified control *experiments* normally conducted in other natural sciences. This makes astronomical data inevitably contaminated by unintended effects, such as noise from foreground/background objects,

the cosmic environment of the objects studied, etc. Powerful statistical methods are normally needed to extract the relevant information from all this natural contamination.

- Astronomical data are fundamentally limited by the finite size and lifetime of the observable universe. Regardless of how strong our observational tools are, we will always be limited by this horizon. For instance, we only ever get to see one realization of the cosmic microwave background (CMB); it is impossible to add more large-scale modes by observing the CMB longer or better. Extracting the maximum information from the finite number of modes/objects is therefore often our only hope to constrain fundamental theories.

In appreciation of these motivations, I decided to dedicate this lecture on "Astrostatistics" to the much more narrow topic of statistical *inference* in astronomy, which, in the most general sense, is the art of finding a theoretical model describing a finite set of empirical observations. More specifically, statistical inference is the mathematical way to distinguish between good and bad models and to determine the free parameters of a model, given some data.

It is important to caution that statistical inference in (astro)physics is generally an *inductive* process as opposed to a *deductive* one, since it relies on a *finite* sample of observations to determine a model that can normally describe *infinitely* many cases. This is particularly true for the construction of physical laws, which strive to be universally applicable, but always rely on a limited set of observations, taken over a finite duration. For the inference problem to become solvable, it is therefore inevitable to invoke some *metaphysical* principles, such as the idea that the Universe obeys some *simplicity* (see Occam's razor), that it is built upon special *symmetries* (e.g. homogeneity and isotropy of space-time) and/or that it follows a mathematical *beauty*.

A powerful mathematical framework for inference should, however, be able to distinguish between models that make different predictions regarding the actually observed data; and it should be able to account for prior knowledge about the 'true' model. The mathematical framework that most directly incorporates these features is the 'Bayesian' framework, as it allows us to assign *probabilities* to models/model parameters, given the data. This lecture is therefore focussed on Bayesian inference and associated tools, such as maximum likelihood estimation, maximum posterior estimates, and Laplace approximation.

Bayes theorem

The core idea of the Bayesian framework is to infer a *model* M (or testing a *hypothesis* H) from some *data* D (also known as *evidence* E) by identifying the model and data with A and B, respectively, in Bayes' theorem, $P(A|B)P(B) = P(B|A)P(A)$. We can thus write

$$P(M|D) = \frac{P(D|M)P(M)}{P(D)},$$

where

- $P(D|M)$ is the conditional probability of the data given the model, known as the *likelihood*;
- $P(M)$ is the *prior probability* (often just called the *prior*), encoding our knowledge about the model before considering the data;
- $P(M|D)$ is the *posterior probability* (often just called the *posterior*), i.e. the probability of the model, accounting for the data *and* prior knowledge;
- $P(D)$ is called the *marginal likelihood*, because it is obtained by integrating ('marginalising') $P(D|M)$ over all models.

Since $P(D)$ does not depend on the model, it has no bearing on the *relative* probabilities of different models. It only acts as a normalisation function and is often dropped in calculations.

Example 1: Rare Disease

As an example, let us consider the case of a rare disease, which has infected 0.1% of the population. A test is available that is 99.9% (99%) accurate at correctly determining positive (negative) patients. Your

test is positive. What is the probability that you have the disease? Using Bayes theorem this probability can be computed as follows

$$P(\text{sick}|\text{test}+) = \frac{P(\text{test}+|\text{sick})P(\text{sick})}{P(\text{test}+)} = \frac{P(\text{test}+|\text{sick})P(\text{sick})}{P(\text{test}+|\text{sick})P(\text{sick}) + P(\text{test}+|\text{healthy})P(\text{healthy})}$$

$$P(\text{sick}|\text{test}+) = \frac{0.999 \cdot 0.001}{0.999 \cdot 0.001 + 0.01 \cdot 0.999} = \frac{1}{11}$$

Thus, even though you tested positively, it is quite unlikely that you have the disease. This might seem counter-intuitive given the high accuracy of the test, but the small value of the prior dominates over this accuracy.

Parameterized models

Often the model M to be inferred from the data D belongs to a discrete or continuous family of models, specified by one or multiple discrete/continuous parameters. In this case it is common to rewrite the likelihood as

$$\mathcal{L}(\theta; D) \equiv P(D|\theta)$$

and consider $\mathcal{L}$ a function of θ at fixed D rather than the other way around. As a consequence, $\mathcal{L}$ is not generally normalised, i.e. $\int \mathcal{L}\, d\theta \neq 1$, unlike standard probability density functions (PDFs); hence the symbol $\mathcal{L}$ instead of P.

The likelihood function *is the probability of the data D assuming that the model is specified by the parameters θ*, which is *not* to be confused with the probability of θ given D (i.e. the *posterior*). This distinction is the reason that we use the semi-colon notation $(\theta; D)$ for the likelihood instead of the vertical bar $(\theta|D)$, normally reserved for conditional probabilities. Of course, in the scientific literature, all sorts of notations are used, so be prepared to get confused! It is also quite common to drop the data vector and simply write $\mathcal{L}(\theta)$.

Following Bayes' theorem, the posterior is related to the likelihood via

$$P(\theta|D) \propto \mathcal{L}(\theta; D)P(\theta).$$

Note that this equation uses a proportionality symbol instead of an equal sign, because we have dropped the θ-independent marginalized likelihood $P(D)$. It is understood that $P(\theta|D)$ has to be normalized, such that $\int P(\theta|D)\, d\theta = 1$.

Point estimators

Often one is interested in a *point estimator*, that is a single model parameter $\hat{\theta}$ which optimally describes the data (given a certain form of the model). Following the parameterized notation of Bayes' theorem, the most natural point estimator is the parameter θ, which maximises the posterior probability distribution $P(\theta|D)$. This parameter is known as the *maximum a posteriori (MAP)* estimator,

$$\hat{\theta}_{\text{MAP}} = \arg\max_{\theta} \mathcal{L}(\theta; D)P(\theta). \tag{1}$$

Quite frequently, one has no a priori idea about the value of θ (apart, perhaps, from certain limits). In this case, the prior $P(\theta)$ might be taken to be a constant (also known as a *flat* prior) and the MAP solution can be found by maximising the likelihood. This parameter solution is known as the *maximum likelihood estimator (MLE)*,

$$\hat{\theta}_{\text{MLE}} = \arg\max_{\theta} \mathcal{L}(\theta|D). \tag{2}$$

Note that it is generally convenient to work with the natural logarithm of the likelihood, called the *log-likelihood*,

$$\ell(\theta; D) \equiv \ln \mathcal{L}(\theta; D). \tag{3}$$

The logarithm conveniently transforms the product of probabilities into sums of log-probabilities.

Since the logarithm is a monotonic function, a maximum of $\mathcal{L}$ is also a maximum of ℓ and vice versa. Hence, $\hat{\theta}_{\text{MLE}}$ can be obtained either through maximising $\mathcal{L}$ or ℓ.

Laplace approximation

We now turn to the problem of approximating the statistical uncertainty of the MAP/MLE. The following formalism is specific to the MLE, but it suffices to replace the likelihood for the product of the likelihood and the prior to write the analogous equations for the MAP.

In general, $\ell(\theta;\mathbf{x})$ can be a complicated function of θ. However, it is generally true that the lowest-order non-vanishing approximation of its shape around the absolute maximum $\hat{\theta}_{\rm MLE}$ is the second order term in the Taylor expansion. If θ is a single parameter,

$$\ell(\theta;\mathbf{x}) = \ell(\hat{\theta}_{\rm MLE};\mathbf{x}) - \tfrac{1}{2}\sigma^{-2}(\theta-\hat{\theta}_{\rm MLE})^2 + \mathcal{O}\big((\theta-\hat{\theta}_{\rm MLE})^3\big), \tag{4}$$

where, for the moment, $-\sigma^{-2}$ is just a funny way of writing the second-order Taylor coefficient, computed as

$$-\sigma^{-2} = \left.\frac{d^2\ell(\theta;\mathbf{x})}{d\theta^2}\right|_{\theta=\hat{\theta}_{\rm MLE}} \tag{5}$$

Since the second derivative at the maximum point is always negative, σ is a well-defined positive real. This parabolic approximation of ℓ corresponds to the Gaussian approximation

$$\mathcal{L}(\theta;\mathbf{x}) \approx \mathcal{L}(\hat{\theta}_{\rm MLE};\mathbf{x}) \exp\left(-\frac{(\theta-\hat{\theta}_{\rm MLE})^2}{2\sigma^2}\right) \tag{6}$$

with standard deviation σ (hence use of the symbol $-\sigma^{-2}$ in Eq. 4). By normalising this Gaussian, we obtain the normal parameter PDF (assuming a flat prior $P(\theta) = \text{constant}$),

$$P(\theta|\mathbf{x}) \approx \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(\theta-\hat{\theta}_{\rm MLE})^2}{2\sigma^2}\right). \tag{7}$$

This approximation of the model uncertainty is known as the *Laplace approximation*. This normal approximation of $\mathcal{L}$ often works surprisingly well in real examples as a consequence of the CLT.

The extension of the Laplace approximation to multivariate likelihood functions, i.e. where θ is a vector of parameters, is straightforward. In this case the covariance of the model parameters at the MLE solution is given by

$$\Sigma = -\mathcal{H}(\hat{\theta}_{\rm MLE})^{-1}, \tag{8}$$

where $\mathcal{H}$ is the *Hessian matrix*,

$$\mathcal{H}_{ij}(\theta) = \frac{\partial^2\ell(\theta;\mathbf{x})}{\partial\theta_i\partial\theta_j}. \tag{9}$$

Since $\mathcal{H}(\hat{\theta}_{\rm MLE})$ is a symmetric negative-definite matrix, the covariance exists and is a symmetric and positive-definite matrix, as it must be.

Example 2: Mass measurement

You need to determine the mass M of an astrophysical objects by counting the number of photons of a certain wavelength observed in a fixed exposure time. This is a fairly standard situation. Think, for instance, of the case of inferring the neutral atomic hydrogen (HI) mass of a galaxy by measuring its rest-frame 21cm emission of the hyperfine hydrogen atom in the electron ground state.

You know three things:

- On average, a mass of $10^9\mathrm{M}_\odot$ produces *one* photon count during the exposure time.
- You have counted 20 photons.
- The mass function of the object is $\phi(M) \propto M^{-1.8}$.

The last point means that the mean number of objects per comoving volume and per unit log-mass $x = \log_{10}(M/\mathrm{M}_\odot)$ is known to decline with increasing mass as $M^{-1.8}$ (roughly the mass function of dark matter haloes below the characteristic break).

What is the mass of the object you are observing?

One might think that 20 photons, each corresponding to a mass of $10^9\mathrm{M}_\odot$, simply implies a mass of $2\cdot10^{10}\mathrm{M}_\odot$ as the most probable answer. This is in fact the maximum likelihood estimator: the probability of counting $k = 20$ photons given $\lambda = M/10^9\mathrm{M}_\odot$ mass units follows a Poisson distribution,

$$\mathcal{L}(M) = P(M|k) = \frac{\lambda^k e^{-\lambda}}{k!}.$$

The maximum of this likelihood occurs indeed at $M = 2\cdot10^{10}\mathrm{M}_\odot$, i.e. $\lambda = 20$.

However, this solution ignores our prior knowledge. The proper Bayesian approach models the probability $P(M|D)$ that the object has a mass M, given the data and the prior:

$$P(M|D) \propto \mathcal{L}(M)\cdot P(M) \propto \frac{\lambda^k e^{-\lambda}}{k!}\cdot M^{-1.8}.$$

Let's do this numerically:

```
likelihood = function(M) dpois(20,M/10^9)
prior = function(M) M^(-1.8) # (not normalized)
marglikelihood = integrate(function(M) likelihood(M)*prior(M),1e8,1e11)$value
posterior = function(M) likelihood(M)*prior(M)/marglikelihood
```

Let's check if the posterior is normalised:

```
integrate(posterior,1e8,1e12)
```

```
## 1 with absolute error < 4.8e-07
```

Looks good. So let's now plot the posterior and determine the MAP solution

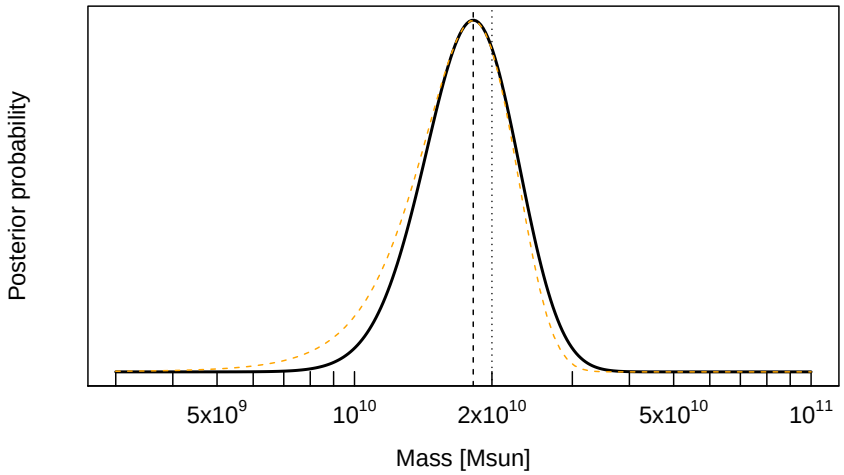
```
mass.mle = optimise(likelihood,c(3e9,1e11),maximum=TRUE)$maximum
mass.map = optimise(posterior,c(3e9,1e11),maximum=TRUE)$maximum
sigma = sqrt(-1/fderiv(function(m) log(posterior(m)),mass.map,2,h=1e5))
laplace = function(m) posterior(mass.map)*exp(-(m-mass.map)^2/(2*sigma^2))
sprintf('MLE solution: M = %.2e Msun',mass.mle)
```

```
## [1] "MLE solution: M = 2.00e+10 Msun"
```

```
sprintf('MAP solution: M = %.2e+-%.2e Msun',mass.map,sigma)
```

```
## [1] "MAP solution: M = 1.82e+10+-4.27e+09 Msun"
```

```
magcurve(posterior,3e9,1e11,n=300,log='x',lwd=2,yaxt='n',
         xlab='Mass [Msun]',ylab='Posterior probability')
abline(v=c(mass.map,mass.mle),lty=c(2,3))
curve(laplace,add=T,col='orange',lty=2) # Laplace approximation
```



Interestingly, most probable mass, i.e. the maximum (dashed line) of the posterior probability (solid line), lies below the mass that is normally expected to emit the number of photons received (dotted line)! This is an interesting ramification of the steepness of the mass function, known as *Eddington bias*.

Notes:

- It is slightly more standard and better in terms of the Laplace approximation to express the likelihood, posterior and uncertainties in log-mass units rather than linear masses.
- In principle, we could have skipped the normalisation by the marginal likelihood in this example, since it has no bearing on the maximum point.

Example 3: Black-body spectrum

Let us now consider an example of a multivariate inference problem, i.e. a problem where we aim to constrain multiple (here two) parameters simultaneously from the same data.

A thermal source of light is observed in three different colour filters. For simplicity, these filters are assumed to be perfect narrow band filters, which are fully transparent in a small wavelength range $\Delta\lambda = 0.1\text{nm}$, centred at a wavelength λ_i, and totally opaque at all other wavelengths. The data obtained through the three filters is summarised in Tab. 2.

Colour	Wavelength λ_i	Photon counts x_i
UV	281	9
blue	446	18
red	641	11

Table 1: Narrow band filter data used for black-body SED fitting.

Assuming a black-body, the energy radiated per unit wavelength over the duration of the measurement, the so-called 'spectral energy distribution' (SED), is given by the Planck law

$$F(\lambda; T, A) = \frac{10^{-40}\text{Jm}^4 A}{\lambda^5} \frac{1}{e^{\frac{hc}{\lambda k_\text{B} T}} - 1} \quad [\text{J/m}], \tag{10}$$

where h is the Planck constant, c is the speed of light and k_B is the Boltzmann constant. The model has two parameters: the temperature T and an amplitude A, which depends on the geometry and the quantum efficiency of the system, as well as on the duration of the measurement. We assume that A is independent of the wavelength. The constant 10^{-40}Jm^4 is an arbitrary scaling factor to make A dimensionless and avoid very small values, which could lead to some unnecessary numerical difficulties.

Our goal is to determine the most likely values of T and A and their uncertainties. We assume flat priors on both parameters, such that the MLE and MAP soluition are identical.

Let's first write down the likelihood for a single narrow band filter i: The probability P_i of observing x_i photons at wavelength $\lambda_i \pm \Delta\lambda/2$ is given by the Poisson statistics

$$\mathcal{L}_i(T, A; x_i) = P_i(x_i|T, A) = \frac{k_i^{x_i} e^{-k_i}}{x_i!},$$

where k_i is the expected number of photons (of energy $E = hc/\lambda_i$) seen through the filter centred at λ_i,

$$k_i = \Delta\lambda \cdot F(\lambda_i; T, A) \cdot \frac{\lambda_i}{hc},$$

which is dimensionless as required. We here assumed the spectrum of the thermal source to be constant over the small bandpass $\Delta\lambda$ of the filters – a slight approximation that can easily be justified given the large undertainties of the measurements (just a few photons).

Hence, the total log-likelihood function becomes

$$\ell(T, A) = \sum_{i=1}^{3} \left[x_i \ln(k_i) - k_i - \ln(x_i!)\right], \tag{11}$$

where the terms $\ln(x_i!)$ do not depend on the model parameters T and A. Therefore, the maximum point and the derivatives of ℓ are the same as those of the effective log-likelihood

$$\tilde{\ell}(T, A) = \sum_{i=1}^{3} \left[x_i \ln(k_i) - k_i\right]. \tag{12}$$

It is possible to make some analytical progress towards finding the maximum point $(\hat{T}, \hat{A})$ of $\tilde{\ell}(T, A)$. However, Eq. (12) lends itself to numerical optimisation. In **R**, this becomes

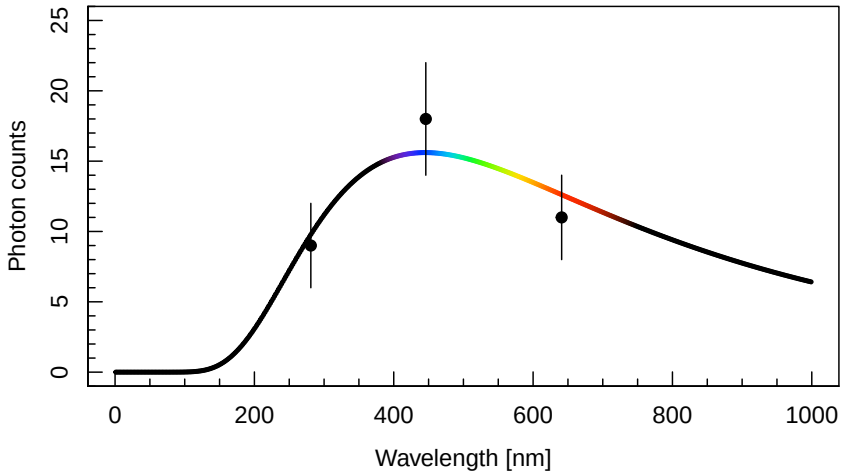
```
# initialise data and constants
data.wavelength = c(281,446,641) # [nm] values of lambda_i
data.counts = c(9,18,11) # values of x_i
c = 299792458 # [m/s] speed of light
h = 6.62607004e-34 # [m^2 kg/s] Planck's constant
k = 1.38064852e-23 # [m^2 kg/s^2/K] Boltzmann constant
dlambda = 1e-10 # [m] filter band width

# Planck law in energy/wavelength [J/m]
planck = function(lambda,temperature,A) {
  return(1e-40*max(0,A)*lambda^(-5)/(exp(h*c/(lambda*k*temperature))-1))
}

# Planck law in photons/Delta lambda
planck.photons = function(lambda,temperature,A) {
  dlambda*lambda/h/c*planck(lambda,temperature,A)
}

# likelihood
log.likelihood = function(p) { # p = (temperature,A)
  k = planck.photons(data.wavelength*1e-9,p[1],p[2])
  return(sum(data.counts*log(k+1e-99)-k)) # the constant 1e-99 avoids problems if k=0
}

# maximise likelihood
p.initial = c(1e4,10)
fit = optim(p.initial, log.likelihood, hessian=TRUE, control=list(fnscale=-1))
T.mle = fit$par[1]
A.mle = fit$par[2]
cat(sprintf('MLE solution: T = %.3eK, A = %.3e\n',T.mle,A.mle))
```

```
## MLE solution: T = 8.246e+03K, A = 6.011e+02
```

We have used the in-built **optim** function to maximise the likelihood. The argument *hessian=TRUE* forces this function to return the Hessian matrix at the maximum point, which we will need to compute the parameter covariance in the Laplace approximation.

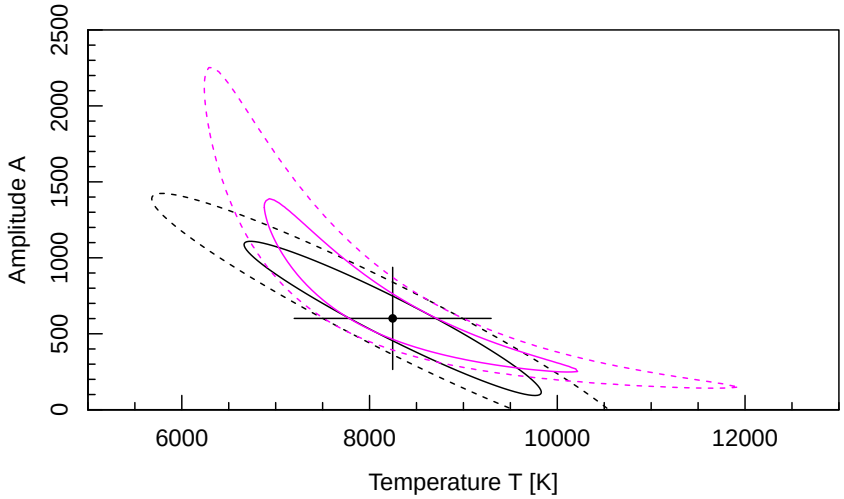
Let us now plot the most likely SED together with the data points and their Poisson uncertainties.

```
lambda.nm = seq(1,1000,by=2) # [nm]
magplot(lambda.nm,dlambda*lambda.nm*1e-9/c/h*planck(lambda.nm*1e-9,T.mle,A.mle),
        pch=20,cex=0.5,ylim=c(0,25),col=wavelength2col(lambda.nm),
        xlab='Wavelength [nm]', ylab='Photon counts')
points(data.wavelength,data.counts,pch=20,cex=1.5) # data points
segments(data.wavelength,
         y0=qpois(0.16,data.counts),y1=qpois(0.84,data.counts)) # 16%-84% uncertainties
```



Finally, we visualize the parameter uncertainties:

```
# Plot MLE solution
xlim = c(5e3,1.3e4) # needed again later
ylim = c(0,2.5e3) # needed again later
magplot(T.mle,A.mle,xlim=xlim,ylim=ylim,pch=20,xaxs='i',yaxs='i',
        xlab='Temperature T [K]', ylab='Amplitude A')

# Laplace approximation
covariance = -solve(fit$hessian)
lines(ellipse(covariance,centre=c(T.mle,A.mle),level=0.68)) # 1-sigma region
lines(ellipse(covariance,centre=c(T.mle,A.mle),level=0.95),lty=2) # 2-sigma region
T.sd = sqrt(covariance[1,1])
A.sd = sqrt(covariance[2,2])
segments(T.mle-T.sd, A.mle, T.mle+T.sd)
segments(T.mle, A.mle-A.sd, y1=A.mle+A.sd)

# Full posterior
n.grid = 150
x.range = seq(xlim[1],xlim[2],len=n.grid)
y.range = seq(ylim[1],ylim[2],len=n.grid)
z = array(NA,c(n.grid,n.grid))
for (i in 1:n.grid) {
  for (j in 1:n.grid) z[i,j] = log.likelihood(c(x.range[i],y.range[j]))
}
z = exp(z)
contour(x.range,y.range,z,levels=contourlevel(z,c(0.68,0.95)), lty=c(1,2),
        drawlabels=FALSE, add=TRUE, col='magenta')
```



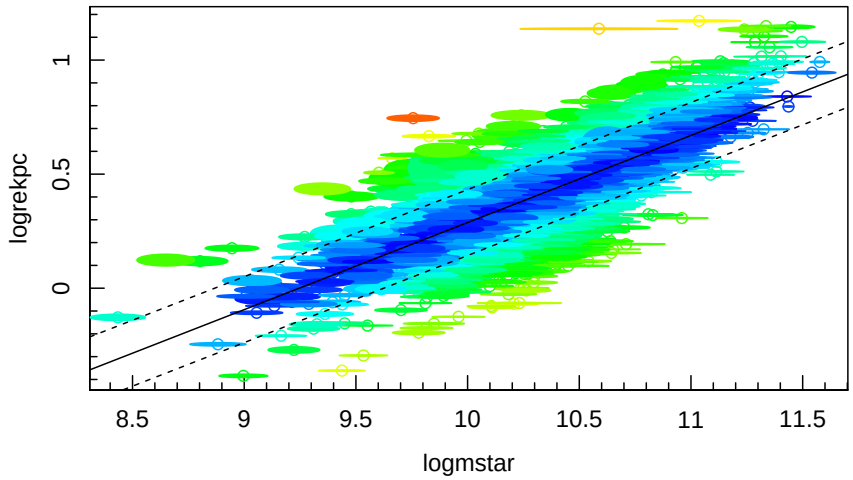
Note that the 2-σ ellipse of the Laplace approximation reaches to $A < 0$, which is clearly non-physical. For more accurate posteriors, we need to consider the full likelihood and identify the minimal regions that contain 68% and 95% of the probability mass, respectively. In this example, this can be done numerically on a grid (magenta). In general, higher-dimensional problems, more sophisticated techniques are required (e.g. MCMC), which usually produce a sample representing the posterior (see later in this course).

Example 4: Lines/planes/hyper-planes

This last example shows how to solve a three-parameter model: we like to fit the stellar mass-size relation of disk galaxies in the local universe via a power law (two parameters) and log-normal intrinsic scatter (3rd parameter), given data with heteroscedastic measurement uncertainties.

This and higher-dimensional linear fits can be handled by the **hyper.fit** package. Here, I only give one brief example and refer the reader to our paper "Hyper-Fit: Fitting Linear Models to Multidimensional Data with Multivariate Gaussian Uncertainties" (https://arxiv.org/abs/1508.02145).

```
data(GAMAsmVsize)
plot(hyper.fit(GAMAsmVsize[,c('logmstar', 'logrekpc')],
               vars=GAMAsmVsize[,c('logmstar_err','logrekpc_err')]^2,
               weights=GAMAsmVsize[,'weights']))
```



You can produce such fits online (https://hyperfit.icrar.org).