

Estimating a Gamma distribution

Thomas P. Minka

2002

Abstract

This note derives a fast algorithm for maximum-likelihood estimation of both parameters of a Gamma distribution or negative-binomial distribution.

1 Introduction

We have observed n independent data points $X = [x_1 \dots x_n]$ from the same density θ . We restrict θ to the class of Gamma densities, i.e. $\theta = (a, b)$:

$$p(x|a, b) = \text{Ga}(x; a, b) = \frac{x^{a-1}}{\Gamma(a)b^a} \exp\left(-\frac{x}{b}\right)$$

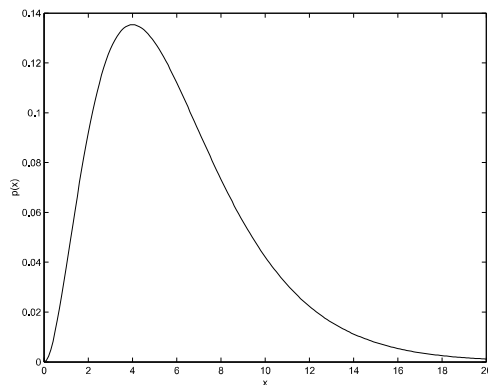


Figure 1: The $\text{Ga}(3, 2)$ density function.

Figure 1 plots a typical Gamma density. In general, the mean is ab and the mode is $(a - 1)b$.

2 Maximum likelihood

The log-likelihood is

$$\log p(D|a, b) = (a - 1) \sum_i \log x_i - n \log \Gamma(a) - na \log b - \frac{1}{b} \sum_i x_i \quad (1)$$

$$= n(a - 1) \overline{\log x} - n \log \Gamma(a) - na \log b - n\bar{x}/b \quad (2)$$

The maximum for b is easily found to be

$$\hat{b} = \bar{x}/a \quad (3)$$

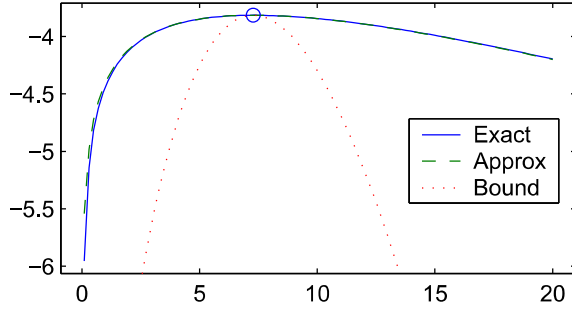


Figure 2: The log-likelihood (4) versus the Gamma-type approximation (9) and the bound (6) at convergence. The approximation is nearly identical to the true likelihood. The dataset was 100 points sampled from $\text{Ga}(7.3, 4.5)$.

Substituting this into (1) gives

$$\log p(D|a, \hat{b}) = n(a-1)\overline{\log x} - n \log \Gamma(a) - na \log \bar{x} + na \log a - na \quad (4)$$

We will describe two algorithms for maximizing this function.

The first method will iteratively maximize a lower bound. Because $a \log a$ is convex, we can use a linear lower bound:

$$a \log a \geq (1 + \log a_0)(a - a_0) + a_0 \log a_0 \quad (5)$$

$$\log p(D|a, \hat{b}) \geq n(a-1)\overline{\log x} - n \log \Gamma(a) - na \log \bar{x} + n(1 + \log a_0)(a - a_0) + na_0 \log a_0 - na \quad (6)$$

The maximum is at

$$0 = n\overline{\log x} - n\Psi(a) - n \log \bar{x} + n(1 + \log a_0) - n \quad (7)$$

$$\Psi(\hat{a}) = \overline{\log x} - \log \bar{x} + \log a_0 \quad (8)$$

where Ψ is the digamma function. The iteration proceeds by setting a_0 to the current \hat{a} , then inverting the Ψ function to get a new \hat{a} . Because the log-likelihood is concave, this iteration must converge to the (unique) global maximum. Unfortunately, it can be quite slow, requiring around 250 iterations if $a = 10$, less for smaller a , and more for larger a .

The second algorithm is much faster, and is obtained via ‘generalized Newton’ [1]. Using an approximation of the form,

$$\log p(D|a, \hat{b}) \approx c_0 + c_1 a + c_2 \log(a) \quad (9)$$

the update is

$$\frac{1}{a^{new}} = \frac{1}{a} + \frac{\overline{\log x} - \log \bar{x} + \log a - \Psi(a)}{a^2(1/a - \Psi'(a))} \quad (10)$$

This converges in about four iterations. Figure 2 shows that this approximation is very close to the true log-likelihood, which explains the good performance.

A good starting point for the iteration is obtained via the approximation

$$\log \Gamma(a) \approx a \log(a) - a - \frac{1}{2} \log a + \text{const.} \quad (\text{Stirling}) \quad (11)$$

$$\Psi(a) \approx \log(a) - \frac{1}{2a} \quad (12)$$

$$\hat{a} \approx \frac{0.5}{\log \bar{x} - \log x} \quad (13)$$

(Note that $\log \bar{x} \geq \overline{\log x}$ by Jensen's inequality.)

2.1 Negative binomial

The maximum-likelihood problem for the negative binomial distribution is quite similar to that for the Gamma. This is because the negative binomial is a mixture of Poissons, with Gamma mixing distribution:

$$p(x|a, b) = \int_{\lambda} \text{Po}(x; \lambda) \text{Ga}(\lambda; a, b) d\lambda = \int_{\lambda} \frac{\lambda^x}{x!} e^{-\lambda} \frac{\lambda^{a-1}}{\Gamma(a)b^a} e^{-\lambda/b} d\lambda \quad (14)$$

$$= \binom{a+x-1}{x} \left(\frac{b}{b+1} \right)^x \left(1 - \frac{b}{b+1} \right)^a \quad (15)$$

Let's consider a slightly generalized negative binomial, where the 'waiting time' for x is given by t :

$$p(x|t, a, b) = \int_{\lambda} \text{Po}(x; \lambda t) \text{Ga}(\lambda; a, b) d\lambda = \int_{\lambda} \frac{(\lambda t)^x}{x!} e^{-\lambda t} \frac{\lambda^{a-1}}{\Gamma(a)b^a} e^{-\lambda/b} d\lambda \quad (16)$$

$$= \binom{a+x-1}{x} \left(\frac{bt}{bt+1} \right)^x \left(1 - \frac{bt}{bt+1} \right)^a \quad (17)$$

Given a data set $D = \{(x_i, t_i)\}$, we want to estimate (a, b) . One approach is to use EM, where the E-step infers the hidden variable λ_i :

$$E[\lambda_i] = (x_i + a) \frac{b}{bt_i + 1} \quad (18)$$

$$E[\log \lambda_i] = \Psi(x_i + a) + \log \frac{b}{bt_i + 1} \quad (19)$$

The M-step then maximizes

$$(a-1) \sum_i E[\log \lambda_i] - n \log \Gamma(a) - na \log b - \frac{1}{b} \sum_i E[\lambda_i] \quad (20)$$

which is a Gamma maximum-likelihood problem.

References

- [1] Thomas P. Minka. Beyond newton's method. research.microsoft.com/~minka/papers/newton.html, 2000.