Poisson Modeling (continued)

> assume( lambda > 0 );
>
p := ( y, lambda ) -> lambda^y * exp(-lambda) / y!;

\[
p(y, \lambda) = \frac{\lambda^y \exp(-\lambda)}{y!}
\]

> simplify( sum( p( y, lambda ), y = 0 .. infinity ) );

1

> simplify( sum( y * p( y, lambda ), y = 0 .. infinity ) );

\[
\lambda
\]

> simplify( sum((y - lambda)^2 * p(y, lambda), y = 0 .. infinity) );

\[
\lambda
\]

Thus if \( Y \sim \text{Poisson}(\lambda) \), \( E(Y) = V(Y) = \lambda \), which people sometimes express by saying that the variance-to-mean ratio (VTMR) for the Poisson is 1.

R can be used to check informally whether the Poisson is a good fit to the LOS data:

rosalind 77> R

R : Copyright 2001, The R Development Core Team
Version 1.2.1 (2001-01-15)

> dpois( 0:7, mean( y ) )
[1] 0.126005645 0.261011693 0.270333539 0.186658872 0.096662630 0.040045947 0.013825386 0.004091186

> print( n <- length( y ) )

[1] 14

> table( y ) / n

<table>
<thead>
<tr>
<th>y</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0.07142857</td>
<td>0.35714286</td>
<td>0.28571429</td>
<td>0.14285714</td>
<td>0.07142857</td>
<td>0.07142857</td>
</tr>
</tbody>
</table>
Poisson Modeling (continued)

```r
> cbind( c( dpois( 0:6, mean( y ) ),
        1 - sum( dpois( 0:6, mean( y ) ) ) ),
        apply( outer( y, 0:7, ‘==’ ), 2, sum ) / n )
```

```
[,1] [,2]
[1,] 0.126005645 0.07142857
[2,] 0.261011693 0.35714286
[3,] 0.270335390 0.28571429
[4,] 0.186658872 0.14285714
[5,] 0.096662630 0.07142857
[6,] 0.040045947 0.00000000
[7,] 0.013825386 0.07142857
[8,] 0.005456286 0.00000000
```

The second column in the above table records the values of the **Poisson probabilities** for \( \lambda = 2.07 \), the mean of the \( y_i \), and the third column is the **empirical relative frequencies**; informally the fit is reasonably good.

Another **informal check** comes from the fact that the sample mean and variance are 2.07 and \( 1.542^2 \approx 2.38 \), which are reasonably close.

**Exchangeability.** As with the AMI mortality case study, before the data arrive I recognize that my uncertainty about the \( Y_i \) is exchangeable, and you would expect from a generalization of the binary-outcomes version of de Finetti’s Theorem that the structure of a **plausible Bayesian model** for the data would then be

\[
\theta \sim p(\theta) \quad \text{(prior)} \tag{59}
\]

\[
(Y_i|\theta) \overset{\text{iid}}{\sim} F(\theta) \quad \text{(likelihood),}
\]

where \( \theta \) is some parameter (vector) and \( F(\theta) \) is some **parametric family of distributions** on the non-negative integers indexed by \( \theta \).
Poisson Modeling (continued)

Thus, in view of the preliminary examination of the data above, a **plausible Bayesian model** for these data is

\[
\lambda \sim p(\lambda) \quad \text{(prior)} \quad (60)
\]

\[
(Y_i|\lambda) \overset{\text{IID}}{\sim} \text{Poisson}(\lambda) \quad \text{(likelihood)},
\]

where \( \lambda \) is a **positive real number**.

**NB** (1) This approach to model-building involves a form of **cheating**, because we’ve **used the data twice**: once to choose the model, and again to draw conclusions conditional on the chosen model.

The result in general can be a failure to **assess** and **propagate model uncertainty** (Draper 1995).

(2) **Frequentist** modeling often employs this **same kind of cheating** in specifying the likelihood function.

(3) There are two Bayesian ways out of this dilemma: **cross-validation** and **Bayesian non-parametric/semi-parametric** methods.

The **latter** is beyond the scope of this course; I’ll give examples of the **former** later.

To get more practice with Bayesian calculations I’m going to **ignore the model uncertainty problem for now** and pretend that somehow we knew that the Poisson was a good choice.

The **likelihood function** in model (60) is

\[
l(\lambda|y) = c p_{Y_1,...,Y_n}(y_1, \ldots, y_n|\lambda)
= c \prod_{i=1}^{n} \frac{\lambda^{y_i} e^{-\lambda}}{y_i!}
= c \lambda^s e^{-n\lambda},
\]
The Conjugate Prior

where \( y = (y_1, \ldots, y_n) \) and \( s = \sum_{i=1}^{n} y_i \); here \( (\prod_{i=1}^{n} y_i!)^{-1} \) can be \textbf{absorbed} into the generic positive \( c \) because it doesn’t involve \( \lambda \).

Thus (as was true in the Bernoulli model) \( s = \sum_{i=1}^{n} y_i \) is \textbf{sufficient} for \( \lambda \) in the Poisson model, and we can write \( l(\lambda|s) \) instead of \( l(\lambda|y) \) if we want.

If a \textbf{conjugate} prior \( p(\lambda) \) for \( \lambda \) exists it must be such that the product \( p(\lambda) l(\lambda|s) \) has the same mathematical form as \( p(\lambda) \).

Examination of (61) reveals that the same trick works here as with Bernoulli data, namely taking the \textbf{prior to be of the same form as the likelihood}:

\[
p(\lambda) = c \lambda^{\alpha-1} e^{-\beta \lambda}
\]  

(62)

for some \( \alpha > 0, \beta > 0 \)—this is the \textbf{Gamma} distribution \( \lambda \sim \Gamma(\alpha, \beta) \) for \( \lambda > 0 \) (see Gelman et al., 2003, Appendix A).

As usual Maple can work out the \textbf{normalizing constant}:

rosalind 80> maple

|
> assume( lambda > 0, alpha > 0, beta > 0 );

> p1 := ( lambda, alpha, beta ) -> lambda^~( alpha - 1 ) * exp( - beta * lambda );

( alpha - 1 )

p1 := ( lambda, alpha, beta ) -> lambda exp(-beta lambda)

> simplify( integrate( p1( lambda, alpha, beta ),

lambda = 0 .. infinity ) );

(-alpha~)  

beta~ GAMMA(alpha~)
The Gamma Distribution

Thus $c^{-1} = \beta^{-\alpha} \Gamma(\alpha)$ and the proper definition of the Gamma distribution is

If $\lambda \sim \Gamma(\alpha, \beta)$ then

$$p(\lambda) = \frac{\beta^\alpha}{\Gamma(\alpha)} \lambda^{\alpha-1} e^{-\beta \lambda}$$

(63)

for $\alpha > 0, \beta > 0$.

As usual Maple can also be used to explore the behavior of this family of distributions as a function of its inputs $\alpha$ and $\beta$:

> p := ( lambda, alpha, beta ) -> beta^alpha * lambda^( alpha - 1 ) * exp( - beta * lambda ) / GAMMA( alpha );

$$\begin{array}{c}
\text{alpha} \\
\text{beta} \\
\text{lambda} \\
\exp(\text{-beta lambda}) \\
\text{GAMMA(\alpha)}
\end{array}$$

> plotsetup( x11 );

> plot( { p( lambda, 1, 1 ), p( lambda, 2, 1 ), p( lambda, 3, 1 ), p( lambda, 6, 1 ) }, lambda = 0 .. 14, color = black );

$\alpha$ evidently controls the shape of the Gamma family.
**Gamma Distribution (continued)**

When $\alpha = 1$ the Gamma distributions have a special form which you’ll probably recognize—they’re the **exponential** distributions $\mathcal{E}(\beta)$: for $\beta > 0$

If $\lambda \sim \mathcal{E}(\beta)$ then $p(\lambda) = \begin{cases} 
\beta e^{-\beta \lambda} & \text{for } \lambda > 0 \\
0 & \text{otherwise} 
\end{cases}$. \hspace{1cm} (64)

```r
> plot( { p( lambda, 2, 1 ), p( lambda, 2, 2 ), p( lambda, 2, 3 ) },
    lambda = 0 .. 7, color = black );
```

In the Gamma family the parameter $\beta$ controls the **spread** or **scale** of the distribution.

**Definition** Given a random quantity $y$ whose density $p(y|\sigma)$ depends on a parameter $\sigma > 0$, if it’s possible to express $p(y|\sigma)$ in the form $\frac{1}{\sigma} f\left(\frac{y}{\sigma}\right)$, where $f(\cdot)$ is a function which does not depend on $y$ or $\sigma$, then $\sigma$ is called a **scale** parameter for the parametric family $p$. 
Gamma Distribution (continued)

Letting \( f(t) = e^{-t} \) and taking \( \sigma = \frac{1}{\beta} \), you can see that the Gamma family can be expressed in this way, so \( \frac{1}{\beta} \) is a scale parameter for the Gamma distribution.

As usual Maple can also work out the mean and variance of this family:

\[
> \text{simplify} (\text{integrate}( p(\lambda, \alpha, \beta), \lambda = 0 .. \infty ));
\]

\[
1
\]

\[
> \text{simplify} (\text{integrate}(\lambda \cdot p(\lambda, \alpha, \beta), \lambda = 0 .. \infty ));
\]

\[
\frac{\alpha}{\beta}
\]

\[
> \text{simplify} (\text{integrate}((\lambda - \alpha / \beta)^2 \cdot p(\lambda, \alpha, \beta), \lambda = 0 .. \infty ));
\]

\[
\frac{\alpha}{\beta^2}
\]

Thus if \( \lambda \sim \Gamma(\alpha, \beta) \) then \( E(\lambda) = \frac{\alpha}{\beta} \) and \( V(\lambda) = \frac{\alpha}{\beta^2} \).

**Conjugate updating** is now straightforward: with \( y = (y_1, \ldots, y_n) \) and \( s = \sum_{i=1}^{n} y_i \), by Bayes’ Theorem

\[
p(\lambda|y) = c p(\lambda) l(\lambda|y) = c \left( c \lambda^{\alpha-1} e^{-\beta \lambda} \right) \left( c \lambda^s e^{-n \lambda} \right)
\]

\[
= c \lambda^{\alpha+s-1} e^{-(\beta+n) \lambda},
\]

and the resulting distribution is just \( \Gamma(\alpha+s, \beta+n) \).
Conjugate Poisson Analysis

This can be summarized as follows:

\[
\begin{align*}
&\{(\lambda|\alpha, \beta) \sim \Gamma(\alpha, \beta)\} \\
&\{(Y_i|\lambda) \overset{\text{IID}}{\sim} \text{Poisson}(\lambda), \quad i = 1, \ldots, n\} \\
&\quad \rightarrow (\lambda|s) \sim \Gamma(\alpha^*, \beta^*),
\end{align*}
\]

where \((\alpha^*, \beta^*) = (\alpha + s, \beta + n)\) and \(s = \sum_{i=1}^{n} y_i\) is a sufficient statistic for \(\lambda\) in this model.

The posterior mean of \(\lambda\) here is evidently \(\frac{\alpha^*}{\beta^*} = \frac{\alpha + s}{\beta + n}\), and the prior and data means are \(\frac{\alpha}{\beta}\) and \(\bar{y} = \frac{s}{n}\), so (as was the case in the Bernoulli model) the posterior mean can be written as a weighted average of the prior and data means:

\[
\frac{\alpha + s}{\beta + n} = \left(\frac{\beta}{\beta + n}\right) \left(\frac{\alpha}{\beta}\right) + \left(\frac{n}{\beta + n}\right) \left(\frac{s}{n}\right).\]

Thus the prior sample size \(n_0\) in this model is just \(\beta\) (which makes sense given that \(\frac{1}{\beta}\) is the scale parameter for the Gamma distribution), and the prior acts like a dataset consisting of \(\beta\) observations with mean \(\frac{\alpha}{\beta}\).

**LOS data analysis.** Suppose that, before the current data set is scheduled to arrive, I know little about the mean length of hospital stay of women giving birth to premature babies.

Then for my prior on \(\lambda\) I’d like to specify a member of the \(\Gamma(\alpha, \beta)\) family which is relatively flat in the region in which the likelihood function is appreciable.
The $\Gamma(\epsilon, \epsilon)$ Prior

A convenient and fairly all-purpose default choice of this type is $\Gamma(\epsilon, \epsilon)$ for some small $\epsilon$ like 0.001.

When used as a prior this distribution has prior sample size $\epsilon$; it also has mean 1, but that usually doesn’t matter when $\epsilon$ is tiny.

```r
> plot( p( lambda, 0.001, 0.001 ), lambda = 0 .. 4, color = black );
```

With the LOS data $s = 29$ and $n = 14$, so the likelihood for $\lambda$ is like a $\Gamma(30, 14)$ density, which has mean $\frac{30}{14} \approx 2.14$ and

$$SD \sqrt{\frac{30}{14^2}} \approx 0.39.$$

Thus by the Empirical Rule the likelihood is appreciable in the range (mean $\pm$ 3 SD) $\approx (2.14 \pm 1.17) \approx (1.0, 3.3)$, and you can see from the plot above that the prior is indeed relatively flat in this region.

From the Bayesian updating in (66), with a $\Gamma(0.001, 0.001)$ prior the posterior is $\Gamma(29.001, 14.001)$. 
LOS Data Analysis

It's useful, in summarizing the **updating** from prior through likelihood to posterior, to make a table that records measures of **center** and **spread** at each point along the way.

For example, the $\Gamma(0.001, 0.001)$ **prior**, when regarded (as usual) as a **density** for $\lambda$, has mean $1.000$ and SD $\sqrt{1000} \approx 31.6$ (i.e., informally, as far as we’re concerned, before the data arrive $\lambda$ could be **anywhere between 0 and (say) 100**).

And the $\Gamma(29.001, 14.001)$ **posterior** has mean $\frac{29.001}{14.001} \approx 2.071$ and SD $\sqrt{\frac{29.001}{14.001^2}} \approx 0.385$, so after the data have arrived we know **quite a bit more than before**.

There are two main ways to summarize the **likelihood**—Fisher’s approach based on **maximizing** it, and the Bayesian approach based on regarding it as a density and **integrating** it—and it’s instructive to compute them both and **compare**.

The **likelihood-integrating** approach (which, at least in one-parameter problems, is essentially equivalent to Fisher’s (1935) attempt at so-called **fiducial** inference) treats the $\Gamma(30, 14)$ likelihood as a density for $\lambda$, with mean $\frac{30}{14} \approx 2.143$ and SD $\sqrt{\frac{30}{14^2}} \approx 0.391$.

As for the **likelihood-maximizing** approach, from (61) the log likelihood function is

$$ll(\lambda|y) = ll(\lambda|s) = \log (c \lambda^s e^{-n\lambda}) = c + s \log \lambda - n\lambda,$$

and this is **maximized** as usual (check that it’s the max) by setting the **derivative** equal to 0 and solving:

$$\frac{\partial}{\partial \lambda} ll(\lambda|s) = \frac{s}{\lambda} - n = 0 \quad \text{iff} \quad \lambda = \hat{\lambda}_{\text{MLE}} = \frac{s}{n} = \bar{y}.$$
LOS Analysis (continued)

Since the MLE $\hat{\lambda}_{\text{MLE}}$ turns out to be our old friend the sample mean $\bar{y}$, you might be tempted to conclude immediately that $\widehat{SE}(\hat{\lambda}_{\text{MLE}}) = \hat{\sigma}/\sqrt{n}$, where $\hat{\sigma} = 1.54$ is the sample SD, and indeed it’s true in repeated sampling that $V(\bar{Y}) = \frac{V(Y_1)}{n}$; but the Poisson distribution has variance $V(Y_1) = \lambda$, so that $\sqrt{V(\bar{Y})} = \frac{\sqrt{\lambda}}{\sqrt{n}}$, and there’s no guarantee in the Poisson model that the best way to estimate $\sqrt{\lambda}$ in this standard error calculation is with the sample SD $\hat{\sigma}$ (in fact we have a strong hint from the above MLE calculation that the sample variance is irrelevant to the estimation of $\lambda$ in the Poisson model).

The right (large-sample) likelihood-based standard error for $\hat{\lambda}_{\text{MLE}}$, using the Fisher information logic we examined earlier, is obtained from the following calculation:

$$\frac{\partial^2}{\partial \lambda^2} \log l(\lambda|y) = -\frac{s}{\lambda^2}, \quad \text{so}$$

$$\hat{I}(\hat{\lambda}_{\text{MLE}}) = \left[-\frac{\partial^2}{\partial \lambda^2} \log l(\lambda|y)\right]_{\lambda=\hat{\lambda}_{\text{MLE}}}$$

$$= \left(\frac{s}{\lambda^2}\right)_{\lambda=\bar{y}} = \frac{s}{\bar{y}^2} = \frac{s}{\bar{y}^2} = \frac{n}{\bar{y}^2}$$

$$= \frac{n}{\bar{y}^2}, \quad \text{and}$$

$$\hat{V}(\hat{\lambda}_{\text{MLE}}) = \hat{I}^{-1}(\hat{\lambda}_{\text{MLE}}) = \frac{\bar{y}}{n} = \frac{\lambda_{\text{MLE}}}{n}.$$

So in this case study Fisher’s likelihood-maximizing approach would estimate $\lambda$ by $\hat{\lambda}_{\text{MLE}} = \bar{y} = \frac{29}{14} \approx 2.071$, with a give-or-take of $\widehat{SE}(\hat{\lambda}_{\text{MLE}}) = \sqrt{\frac{\lambda_{\text{MLE}}}{\sqrt{n}}} = \frac{1.44}{\sqrt{14}} \approx 0.385$. 

71
LOS Analysis (continued)

All of this may be summarized in the following table:

<table>
<thead>
<tr>
<th></th>
<th>Prior</th>
<th>Maximizing</th>
<th>Integrating</th>
<th>Posterior</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean/Estimate</td>
<td>1.00</td>
<td>2.071</td>
<td>2.143</td>
<td>2.071</td>
</tr>
<tr>
<td>SD/SE</td>
<td>31.6</td>
<td>0.385</td>
<td>0.391</td>
<td>0.385</td>
</tr>
</tbody>
</table>

The discrepancies between the likelihood-maximizing and likelihood-integrating columns in this table would be smaller with a larger sample size and would tend to 0 as $n \to \infty$.

The prior-likelihood-posterior plot comes out like this:

```r
> plot( { p( lambda, 0.001, 0.001 ), p( lambda, 30, 14 ), p( lambda, 29.001, 14.001 ) }, lambda = 0 .. 5, color = black );
```
LOS Analysis (continued)

For interval estimation in the maximum-likelihood approach the best we could do, using the technology I’ve described to you so far, would be to appeal to the CLT (even though \( n \) is only 14) and use \( \hat{\lambda}_{\text{MLE}} \pm 1.96 \frac{SE(\hat{\lambda}_{\text{MLE}})}{\hat{\lambda}_{\text{MLE}}} \approx 2.071 \pm (1.96)(0.385) \approx (1.316, 2.826) \) as an approximate 95% confidence interval for \( \lambda \).

You can see from the previous plot that the likelihood function is skewed, so a more careful method (e.g., the bootstrap; Efron 1979) would be needed to create a better interval estimate from the likelihood point of view.

Some trial and error with Maple can be used to find the lower and upper limits of the central 95% posterior interval for \( \lambda \):

\[
\begin{align*}
&> \text{evalf( Int( p( lambda, 29.001, 14.001 ), lambda = 0 .. 1.316 ) ));} \\
&\quad \quad .01365067305 \\
&> \text{evalf( Int( p( lambda, 29.001, 14.001 ), lambda = 0 .. 1.4 ) ));} \\
&\quad \quad .02764660367 \\
&> \text{evalf( Int( p( lambda, 29.001, 14.001 ), lambda = 0 .. 1.387 ) ));} \\
&\quad \quad .02495470339 \\
&> \text{evalf( Int( p( lambda, 29.001, 14.001 ), lambda = 2.826 .. infinity ) ));} \\
&\quad \quad .03403487851 \\
&> \text{evalf( Int( p( lambda, 29.001, 14.001 ), lambda = 2.890 .. 5 ) ));} \\
&\quad \quad .02505306648 \\
&> \text{evalf( Int( p( lambda, 29.001, 14.001 ), lambda = 2.890 .. infinity ) ));} \\
&\quad \quad .02505307631
\end{align*}
\]
LOS Analysis (continued)

Thus a **95% (central) posterior interval** for \( \lambda \), given a diffuse prior, runs from **1.387 to 2.890**, and is (correctly) **asymmetric** around the posterior mean of 2.071.

\( R \) can be used to work out the **limits of this interval** even more readily:

> help( qgamma )

```
GammaDist package:base R Documentation

The Gamma Distribution

Description:

Density, distribution function, quantile function and random generation for the Gamma distribution with parameters ‘shape’ and ‘scale’.

Usage:

dgamma(x, shape, scale=1, log = FALSE)
pgamma(q, shape, scale=1, lower.tail = TRUE, log.p = FALSE)
qgamma(p, shape, scale=1, lower.tail = TRUE, log.p = FALSE)
rgamma(n, shape, scale=1)
```

Arguments:

- **x, q**: vector of quantiles.
- **p**: vector of probabilities.
- **n**: number of observations.
- **shape, scale**: shape and scale parameters.
- **log, log.p**: logical; if TRUE, probabilities p are given as log(p).
- **lower.tail**: logical; if TRUE (default), probabilities are \( P[X \leq x] \), otherwise \( P[X > x] \).
Details:

If ‘scale’ is omitted, it assumes the default value of ‘1’.

The Gamma distribution with parameters ‘shape’ = a and ‘scale’ = s has density

\[ f(x) = \frac{1}{s^a \Gamma(a)} x^{(a-1)} e^{-(x/s)} \]

for \( x > 0 \), \( a > 0 \) and \( s > 0 \). The mean and variance are \( E(X) = a s \) and \( \text{Var}(X) = a s^2 \).

Value:

‘dgamma’ gives the density, ‘pgamma’ gives the distribution function ‘qgamma’ gives the quantile function, and ‘rgamma’ generates random deviates.

Note:

The cumulative hazard \( H(t) = -\log(1 - F(t)) \) is ‘-pgamma(t, ..., lower = FALSE, log = TRUE)’.

See Also:

‘gamma’ for the Gamma function, ‘dbeta’ for the Beta distribution and ‘dchisq’ for the chi-squared distribution which is a special case of the Gamma distribution.

Examples:

\[ -\log(dgamma(1:4, shape=1)) \]
\[ p <- (1:9)/10 \]
\[ pgamma(qgamma(p, shape=2), shape=2) \]
\[ 1 - 1/exp(qgamma(p, shape=1)) \]

\[ > qgamma( 0.025, 29.001, 1 / 14.001 ) \]
\[ [1] 1.387228 \]

\[ > qgamma( 0.975, 29.001, 1 / 14.001 ) \]
\[ [1] 2.890435 \]
LOS Analysis (continued)

Maple or R can also be used to obtain the probability content, according to the posterior distribution, of the approximate 95% (large-sample) likelihood-based interval:

```r
> evalf( int( p( lambda, 29.001, 14.001 ), lambda = 1.316 .. 2.826 ) );

.9523144484
```

So the maximization approach has led to decent approximations here (later I’ll give examples where maximum likelihood doesn’t do so well in small samples).

**Predictive distributions** in this model can be computed by Maple in the usual way: for instance, to compute $p(y_{n+1}|y)$ for $y = (y_1, \ldots, y_n)$ we want to evaluate

$$
p(y_{n+1}|y) = \int_0^\infty p(y_{n+1}, \lambda|y) d\lambda
$$

$$
= \int_0^\infty p(y_{n+1}|\lambda, y) p(\lambda|y) d\lambda
$$

$$
= \int_0^\infty p(y_{n+1}|\lambda) p(\lambda|y) d\lambda
$$

$$
= \int_0^\infty \lambda^{y_{n+1}} e^{-\lambda} \frac{(\beta^*)^{\alpha^*}}{y_{n+1}!} \frac{(\alpha^*)^{\alpha^* - 1} e^{-\beta^* \lambda}}{\Gamma(\alpha^*)} d\lambda,
$$

$$
= \frac{(\beta^*)^{\alpha^*}}{\Gamma(\alpha^*) y_{n+1}!} \int_0^\infty \lambda^{(\alpha^* + y_{n+1}) - 1} e^{-(\beta^* + 1) \lambda} d\lambda,
$$

where $\alpha^* = \alpha + s$ and $\beta^* = \beta + n$; in these expressions $y_{n+1}$ is a non-negative integer.

```r
> assume( astar > 0, bstar > 0, yf > 0 );

> simplify( bstar^astar * int( lambda^( astar + yf - 1 ) *
     exp( - ( bstar + 1 ) * lambda ), lambda = 0 .. infinity ) /
     ( GAMMA( astar ) * yf! ) );

astar^- ( -astar^- - yf^- )
bstar^- ( bstar^- + 1 )
-------------------------------------
GAMMA( astar^- ) GAMMA( yf^- + 1 )
```

76
Predictive Distributions

A bit of rearranging then gives that for $y_{n+1} = 0, 1, \ldots,$

$$p(y_{n+1}|y) = \frac{\Gamma(\alpha^* + y_{n+1})}{\Gamma(\alpha^*) \Gamma(y_{n+1} + 1)} \left( \frac{\beta^*}{\beta^* + 1} \right)^{\alpha^*} \left( \frac{1}{\beta^* + 1} \right)^{y_{n+1}}. \quad (72)$$

This is called the Poisson-Gamma distribution, because (71) is asking us to take a mixture (weighted average) of Poisson distributions, using probabilities from a Gamma distribution as the mixing weights.

(72) is a generalization of the negative binomial distribution (e.g., Johnson and Kotz 1994), which you may have encountered in your earlier probability study.

Maple can try to get simple expressions for the mean and variance of this distribution:

```maple
> pg := (y, alpha, beta) -> GAMMA( alpha + y ) *
    ( beta / ( beta + 1 ) )^alpha * ( 1 / ( beta + 1 ) )^y /
    ( GAMMA( alpha ) * GAMMA( y + 1 ) );

GAMMA(alpha + y) |--------| |--------|
\beta + 1/ \beta + 1/
pg := (y, alpha, beta) -> ------------------------------------------

GAMMA(alpha) GAMMA(y + 1)

> simplify( sum( pg( y, alpha, beta ), y = 0 .. infinity ) );
1

> simplify( sum( y * pg( y, alpha, beta ), y = 0 .. infinity ) );

alpha
-----
beta

So the mean of the distribution in (72) is $E(y_{n+1}|y) = \frac{\alpha^*}{\beta^*}$. 
Inference and Prediction

> simplify( sum( ( y - alpha / beta )^2 * pg( y, alpha, beta ),
> y = 0 .. infinity ) );

\[
\begin{align*}
2 / \beta & \quad \alpha \quad \text{alpha - beta} \\
\alpha & \quad \text{alpha} \quad \text{alpha - beta} \\
\beta + 1 & \quad \text{beta} \quad \text{beta}
\end{align*}
\]

Maple has failed to realize that this expression may be considerably simplified: Bernardo and Smith (1994) note that the variance of the distribution in (72) is just

\[
V(y_{n+1}|y) = \frac{\alpha^*}{\beta^*} \left( 1 + \frac{1}{\beta^*} \right). \quad \text{(73)}
\]

This provides an interesting contrast between inference and prediction: we've already seen in this model that the posterior mean and variance of \( \lambda \) are \( \frac{\alpha^*}{\beta^*} = \frac{\alpha + s}{\beta + n} \) and \( \frac{\alpha^*}{(\beta^*)^2} = \frac{\alpha + s}{(\beta + n)^2} \), respectively.

<table>
<thead>
<tr>
<th>Quantity</th>
<th>Mean</th>
<th>Posterior</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \lambda )</td>
<td>( \frac{\alpha + s}{\beta + n} )</td>
<td>( \frac{\alpha + s}{(\beta + n)^2} = \frac{\alpha + s}{\beta + n} \left( 1 + \frac{1}{\beta + n} \right) )</td>
</tr>
<tr>
<td>( y_{n+1} )</td>
<td>( \frac{\alpha + s}{\beta + n} )</td>
<td>( \frac{\alpha + s}{\beta + n} \left( 1 + \frac{1}{\beta + n} \right) )</td>
</tr>
</tbody>
</table>

Thus \( \lambda \) (the inferential objective) and \( y_{n+1} \) (the predictive objective) have the same posterior mean, but the posterior variance of \( y_{n+1} \) is much larger, as can be seen by the following argument.

(1) Denoting by \( \mu \) the mean of the population from which the \( Y_i \) are thought of as (like) a random sample, when \( n \) is large \( \alpha \) and \( \beta \) will be small in relation to \( s \) and \( n \), respectively, and the ratio \( \bar{y} = \frac{s}{n} \) should more and more closely approach \( \mu \)—thus for large \( n \),

\[
E(\lambda|y) = E(y_{n+1}|y) \doteq \mu. \quad \text{(74)}
\]
Inference and Prediction

(2) For the Poisson distribution the (population) mean \( \mu \) and variance \( \sigma^2 \) are equal, meaning that for large \( n \) the ratio \( \frac{\alpha+s}{\beta+n} \) will be close both to \( \mu \) and to \( \sigma^2 \).

Thus for large \( n \),

\[
V(\lambda|y) = \frac{\sigma^2}{n} \quad \text{but} \quad V(y_{n+1}|y) = \sigma^2. \quad (75)
\]

An informal way to restate (75) is to say that accurate prediction of new data is an order of magnitude harder (in powers of \( n \)) than accurate inference about population parameters.

**Bayesian model-checking with predictive distributions.**

One way to check a model like (58) is as follows.

```r
for ( i in 1:n ) {

    Temporarily set aside observation \( y_i \), obtaining a new dataset \( y_{-i} = (y_1, \ldots, y_{i-1}, y_{i+1}, \ldots, y_n) \) with \( n-1 \) observations.

    Use the current Bayesian model applied to \( y_{-i} \) to predict \( y_i \), and summarize the extent to which the actual value of \( y_i \) is surprising in view of this predictive distribution.

}
```

One possible measure of surprise is predictive \( z \)-scores:

\[
z_i = \frac{y_i - E[y_i|y_{-i}]}{\sqrt{V[y_i|y_{-i}]}}. \quad (76)
\]

Compare the surprise measure with its expected behavior if the model had been “correct” (e.g., \( z = (z_1, \ldots, z_n) \) should have mean 0 and SD 1).
Predictive Model-Checking

Example: the LOS data. Here's some R code to carry out this program on the LOS data.

```r
rosalind 25> R

R : Copyright 2001, The R Development Core Team
Version 1.2.1 (2001-01-15)

> poisson.gamma <- function( y, alpha, beta ) {
    log.density <- lgamma( alpha + y ) + alpha *
    log( beta / ( beta + 1 ) ) + y * log( 1 / ( beta + 1 ) ) -
    lgamma( alpha ) - lgamma( y + 1 )

    return( exp( log.density ) )
}

> print( y <- sort( y ) )
[1] 0 1 1 1 1 1 2 2 2 2 3 3 4 6

> print( y.current <- y[ -1 ] )
[1] 1 1 1 1 1 2 2 2 2 3 3 4 6

> print( n.current <- length( y.current ) )
[1] 13

> alpha <- beta <- 0.001

> print( s.current <- sum( y.current ) )
[1] 29

> print( alpha.star <- alpha + s.current )
[1] 29.001

> print( beta.star <- beta + n.current )
[1] 13.001
```