Stat 5421 Lecture Notes: Statistical Inference for the Poisson Distribution

Charles J. Geyer

October 05, 2021

1 License

2 Data

In our example the count is 17

```
x <- 17
```

3 Maximum Likelihood Estimator

The usual estimator of the parameter μ is $\hat{\mu} = x$. This turns out to also be the maximum likelihood estimator.

```
mu.hat <- x
```

4 Log Likelihood

We can plot the log likelihood function using the following code

```
logl <- function(mu) {
    result <- x * log(mu) - mu
    result[is.na(result)] <-  (- mu)
    return(result)
}
curve(logl, from = 0.0001, to = 50, n = 1001,
    xlab = expression(mu), ylab = "log likelihood",
    ylim = logl(mu.hat) + c(-10, 0))
abline(v = mu.hat, lty = "dashed")
```

The dashed vertical line shows where the MLE is, and it does appear to be where the log likelihood is maximized. The log likelihood goes to minus infinity as $\mu \to 0$ or $\mu \to \infty$. Except when $x = 0$ the log likelihood increases to 0 as $\mu \to 0$. Thus we cannot try to draw the curve from zero to infinity but rather from a little bit above zero to some big number. But that gives us a plot in which it is hard to see what is going on.

Hence the `ylim` optional argument to R function `curve`. It will turn out that the only interesting part of the log likelihood is the region near the maximum. Hence we include in our plot only the part of the curve in which the log likelihood is within 10 of the maximum. The 10 was pulled out of the air. We don't really want to get scientific about this yet (but do in the section on likelihood-based confidence intervals below).

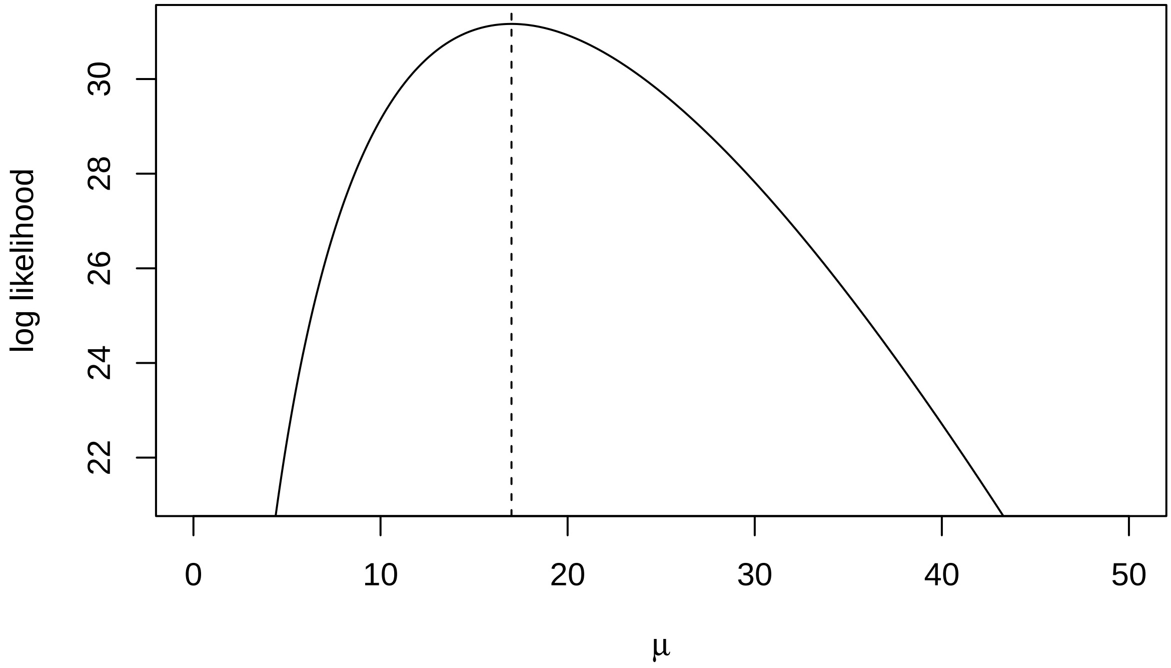

Figure 1: Log Likelihood

Also it really doesn't matter, since we are just using this plot to get some idea what is going on. We don't use this plot for statistical inference.

5 Confidence Intervals

5.1 Confidence Level

Set the confidence level. It is best programming practice to never hard code numbers like this, that is, the number 0.95 should only occur in your document once where it is used to initialize a variable. Then use that variable elsewhere.

If there is ever a need to change this (to use say 0.90 for the confidence level), then it only need be changed in one place and will be consistently used everywhere.

```
conf.level <- 0.95
```

5.2 Wald Confidence Interval

This interval is

$$\hat{\mu} \pm z_{\alpha/2} \cdot \sqrt{\hat{\mu}}$$

where $z_{\alpha/2}$ is the $1-\alpha/2$ quantile of the standard normal distribution, because the variance of the distribution is μ.

```
mu.hat + c(-1, 1) * qnorm((1 + conf.level) / 2) * sqrt(mu.hat)
```

```
## [1]  8.918861 25.081139
```

As with the binomial the Wald interval is zero width, hence ridiculous when $\hat{\mu} = 0$.

The Wald interval can be repaired by using a different procedure (Geyer, 2009, *Electronic Journal of Statistics*, **3**, 259–289). For the Poisson distribution, this recipe uses the interval $[0, -\log(\alpha)]$ for coverage $1-\alpha$. Thus when we observe $x = 0$ and want 95% confidence, the interval is

```
alpha <- 1 - conf.level
c(0, - log(alpha))
```

```
## [1] 0.000000 2.995732
```

This same fix can be used for any confidence interval, not just the Wald interval.

5.3 Score Interval (also called Rao)

To derive this inverval we invert the score test. The test statistic is

$$\frac{x - \mu}{\sqrt{\mu}}$$

so the interval is the set of μ such that

$$\left| \frac{x - \mu}{\sqrt{\mu}} \right| < z_{\alpha/2}$$

or

$$x^2 - 2x\mu + \mu^2 < \mu z_{\alpha/2}^2$$

or

$$\mu^2 - (2x + z_{\alpha/2}^2)\mu + x^2 < 0$$

From the quadratic formula this interval has end points

$$\frac{2x + z_{\alpha/2}^2 \pm \sqrt{(2x + z_{\alpha/2}^2)^2 - 4x^2}}{2}$$

or

$$x + \frac{z_{\alpha/2}^2}{2} \pm z_{\alpha/2} \sqrt{x + \frac{z_{\alpha/2}^2}{4}}$$

```
z <- qnorm((1 + conf.level) / 2)
x + z^2 / 2 + c(-1, 1) * z * sqrt(x + z^2 / 4)
```

```
## [1] 10.61447 27.22699
```

5.4 Likelihood Interval (also called Wilks)

```
crit <- qchisq(conf.level, df = 1)
fred <- function(mu) 2 * (logl(mu.hat) - logl(mu)) - crit
tol <- sqrt(.Machine$double.eps)
if (mu.hat == 0) {
    low <- 0
} else {
    low <- uniroot(fred, lower = 0, upper = mu.hat, tol = tol)$root
}
hig <- uniroot(fred, lower = mu.hat, upper = 2 * mu.hat,
    extendInt = "upX", tol = tol)$root
c(low, hig)
```

```
## [1] 10.14533 26.40924
```

6 Hypothesis Tests

6.1 Hypotheses

6.1.1 Upper-Tailed

$$\begin{aligned} H_0 &: \mu = \mu_0 \\ H_1 &: \mu > \mu_0 \end{aligned}$$

6.1.2 Lower-Tailed

$$\begin{aligned} H_0 &: \mu = \mu_0 \\ H_1 &: \mu < \mu_0 \end{aligned}$$

6.1.3 Two-Tailed

$$\begin{aligned} H_0 &: \mu = \mu_0 \\ H_1 &: \mu \neq \mu_0 \end{aligned}$$

Suppose we have

```
mu.zero <- 13
```

6.2 Score Test (also called Rao)

The test statistic is (repeating what was said in the section on the score interval above)

$$t = \frac{x - \mu}{\sqrt{\mu}}$$

where in both places μ is now the value of μ hypothesized under the null hypothesis. Call that μ_0 and we have

$$t = \frac{x - \mu_0}{\sqrt{\mu_0}}$$

```
tstat <- (x - mu.zero) / sqrt(mu.zero)
tstat
```

```
## [1] 1.1094
```

```
# P-values
# upper-tailed
pnorm(tstat, lower.tail = FALSE)
```

```
## [1] 0.1336287
```

```
# lower-tailed
pnorm(tstat)
```

```
## [1] 0.8663713
```

```
# two-tailed
2 * pnorm(abs(tstat), lower.tail = FALSE)
```

```
## [1] 0.2672575
```

6.3 Likelihood Ratio Test (also called Wilks)

As always the test statistic for a two-tailed test is

$$t = 2[l(\hat{\mu}) - l(\mu_0)]$$

and has a chi-square distribution on one degree of freedom under the null hypothesis.

This can be converted into an asymptotically normal distribution by taking the square root and assigning the sign of $\hat{\mu} - \mu_0$ because the square of a standard normal random variable is chi-square with one degree of freedom.

And we can also use this signed likelihood ratio test statistic for the two-tailed test because its square is the original likelhood ratio test statistic.

```
tstat <- sign(mu.hat - mu.zero) * sqrt(2 * (logl(mu.hat) - logl(mu.zero)))
tstat
```

```
## [1] 1.058761
```

```
# P-values
# upper-tailed
pnorm(tstat, lower.tail = FALSE)
```

```
## [1] 0.1448542
```

```
# lower-tailed
pnorm(tstat)
```

```
## [1] 0.8551458
```

```r
# two-tailed
2 * pnorm(abs(tstat), lower.tail = FALSE)
```

```
## [1] 0.2897085
```

6.4 Wald Test

This is just like the score test except that the variance is estimated under the alternative rather than under the null.

$$t = \frac{x - \mu_0}{\sqrt{\hat{\mu}}}$$

and this has the problem of blowing up when $\hat{\mu} = 0$. It also doesn't make intuitive sense to assume μ_0 is the true unknown parameter value and then not use $\sqrt{\mu_0}$ for the standard deviation.

So nobody recommends this (for Poisson), and we won't even illustrate it.

6.5 Exact Test

Here the test statistic is just x.

```r
# P-values
# upper-tailed
ppois(x - 1, mu.zero, lower.tail = FALSE)
```

```
## [1] 0.1645069
```

```r
# lower-tailed
ppois(x, mu.zero)
```

```
## [1] 0.890465
```

The reason for the `x - 1` is the discreteness of the Poisson distribution (that's the way `lower.tail = FALSE` works).

There is no exact two-tailed because the exact (Poisson) distribution is not symmetric, so there is no reason to us $|X - \mu_0|$ as a test statistic. One could also say there are many possible two-tailed exact tests and no reason to prefer any one of them.

6.6 Fuzzy Test

There is a fuzzy test, but there is no computer implementation available. So we don't cover that either.