# UMPU, Equal-Tailed, and Pratt Fuzzy Confidence Intervals

Charles J. Geyer　　Glen D. Meeden

August 28, 2005

This note provides a detailed example of three fuzzy confidence intervals proposed in Geyer and Meeden (2005) and in the comments on that paper (Agresti and Gottard, 2005; Brown, Cai and DasGupta, 2005). We use the binomial distribution with sample size $n = 10$ for our example.

- The *UMPU interval* proposed by Geyer and Meeden (2005), based on the uniformly most powerful unbiased (UMPU) test Lehmann (1959). Its membership function is the

$$\theta \mapsto 1 - \phi(x, \alpha, \theta) \tag{1}$$

  where $\phi$ is the critical function of the UMPU test (Geyer and Meeden, 2005, Section 1.4, especially 1.4.2).

- The *equal-tailed interval* proposed by Agresti and Gottard (2005), attributed by them to Stevens (1950), although, of course, the notion of a *fuzzy* confidence interval was not exactly what Stevens proposed. This is (1) where $\phi$ is the critical function of the equal-tailed randomized test.

- The *Pratt interval* proposed by Brown, Cai and DasGupta (2005), attributed by them to Pratt (1961), although, of course, the notion of a *fuzzy* confidence interval was not exactly what Pratt proposed. This is (1) where $\phi(\,\cdot\,, \alpha, \theta)$ is the critical function of the randomized likelihood ratio test with null hypothesis that the data are Binomial$(n, \theta)$ and alternative hypothesis that the data have the discrete uniform distribution on $\{0, \ldots, n\}$.

The UMPU critical function is programmed as follows, using the `ump` package (available from CRAN, `http://cran.r-project.org`).

```
> check.args <- function(x, n, alpha, theta) {
+     stopifnot(is.numeric(x))
+     stopifnot(is.numeric(n))
+     stopifnot(is.numeric(alpha))
+     stopifnot(is.numeric(theta))
+     stopifnot(length(x) == 1)
+     stopifnot(length(n) == 1)
+     stopifnot(length(alpha) == 1)
+     stopifnot(x == as.integer(x))
+     stopifnot(n == as.integer(n))
+     stopifnot(0 <= x & x <= n)
+     stopifnot(0 <= alpha & alpha <= 1)
+     stopifnot(all(0 <= theta & theta <= 1))
+ }
> library(ump)
> phi.umpu <- function(x, n, alpha, theta) {
+     check.args(x, n, alpha, theta)
+     umpu.binom(x, n, theta, alpha)
+ }
```

The equal-tailed critical function is programmed as follows.

```
> phi.eqtail <- function(x, n, alpha, theta) {
+     check.args(x, n, alpha, theta)
+     c1 <- qbinom(alpha/2, n, theta)
+     c2 <- qbinom(alpha/2, n, theta, lower.tail = FALSE)
+     P1 <- pbinom(c1 - 1, n, theta)
+     P2 <- pbinom(c2, n, theta, lower.tail = FALSE)
+     p1 <- dbinom(c1, n, theta)
+     p2 <- dbinom(c2, n, theta)
+     g1 <- (alpha/2 - P1)/p1
+     g2 <- (alpha/2 - P2)/p2
+     g1[c1 == c2] <- ((alpha - P1 - P2)/p1)[c1 ==
+         c2]
+     g2[c1 == c2] <- g1[c1 == c2]
+     phi <- rep(1, length(theta))
+     phi[c1 == x] <- g1[c1 == x]
+     phi[c2 == x] <- g2[c2 == x]
+     phi[c1 < x & x < c2] <- 0
+     phi
+ }
```

The Pratt critical function is programmed as follows.

```
> phi.pratt.aux <- function(n, alpha, theta) {
+     x <- seq(0, n)
+     phi <- rep(1, length(x))
+     if (alpha == 1) {
+         return(phi)
+     }
+     if (alpha == 0) {
+         return(0 * phi)
+     }
+     if (theta == 0) {
+         phi[x == 0] <- alpha
+         return(phi)
+     }
+     if (theta == 1) {
+         phi[x == n] <- alpha
+         return(phi)
+     }
+     pnull <- dbinom(x, n, theta)
+     porder <- rev(order(1/pnull))
+     corder <- cumsum(pnull[porder])
+     outies <- corder < alpha
+     phi[porder[!outies]] <- 0
+     P <- sum(pnull[porder[outies]])
+     irand <- porder[!outies][1]
+     phi[irand] <- (alpha - P)/pnull[irand]
+     return(phi)
+ }
> phi.pratt <- function(x, n, alpha, theta) {
+     check.args(x, n, alpha, theta)
+     phi <- rep(1, length(theta))
+     for (i in 1:length(theta)) {
+         foo <- phi.pratt.aux(n, alpha, theta[i])
+         phi[i] <- foo[seq(0, n) == x]
+     }
+     phi
+ }

> x <- 0
> theta <- seq(0, 1, length = 1001)
```

Now we are ready to look at one case. We start with $x = 0$ (and for all cases $n = 10$ and $\alpha = 0.05$).

Figure 1 is produced by the following code

```
> fci1 <- 1 - phi.umpu(x, n, alpha, theta)
> fci2 <- 1 - phi.eqtail(x, n, alpha, theta)
> fci3 <- 1 - phi.pratt(x, n, alpha, theta)
> fred <- theta[fci1 > 0 | fci2 > 0 | fci3 > 0]
> par(mar = c(5, 4, 0, 0) + 0.1)
> plot(theta, fci1, xlim = c(0, 1), ylim = c(0, 1),
+     type = "l", col = "red", xlab = "success probability",
+     ylab = "degree of membership", cex = 1.5, cex.axis = 1.5,
+     cex.lab = 1.5, lwd = 1.5)
> lines(theta, fci2, col = "green")
> lines(theta, fci3, col = "blue")
```

and appears on p. 5.

```
> x <- 1
```

Now we do $x = 1$. These fuzzy intervals are shown in Figure 2 on p. 6.

```
> x <- 2
```

Now we do $x = 2$. These fuzzy intervals are shown in Figure 3 on p. 7.

```
> x <- 3
```

Now we do $x = 3$. These fuzzy intervals are shown in Figure 4 on p. 8.

```
> x <- 4
```

Now we do $x = 4$. These fuzzy intervals are shown in Figure 5 on p. 9.

```
> x <- 5
```

Now we do $x = 5$. These fuzzy intervals are shown in Figure 6 on p. 10.

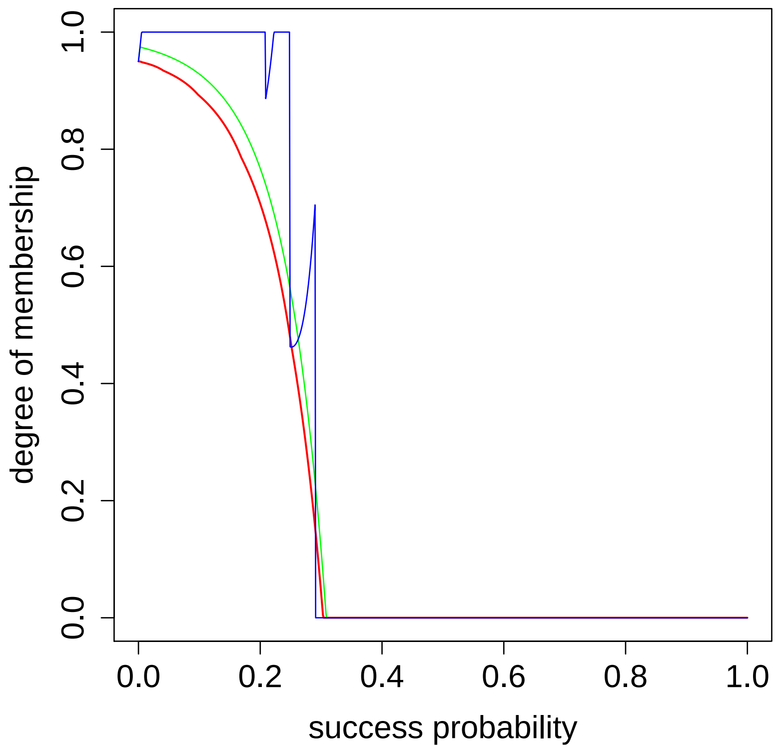


Figure 1: Fuzzy Confidence Intervals. 95% fuzzy confidence intervals for the binomial distribution, $x = 0$, $n = 10$. Red is UMPU (Geyer-Meeden), green is equal-tailed (Agresti-Gotard), blue is Pratt (Brown-Cai-DasGupta).

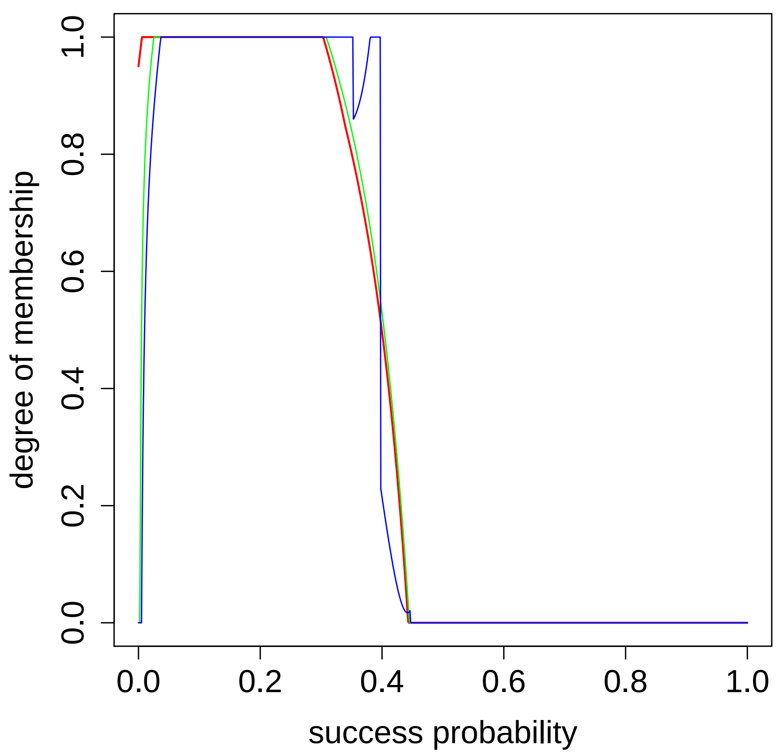


Figure 2: Fuzzy Confidence Intervals. 95% fuzzy confidence intervals for the binomial distribution, $x = 1$, $n = 10$. Red is UMPU (Geyer-Meeden), green is equal-tailed (Agresti-Gotard), blue is Pratt (Brown-Cai-DasGupta).

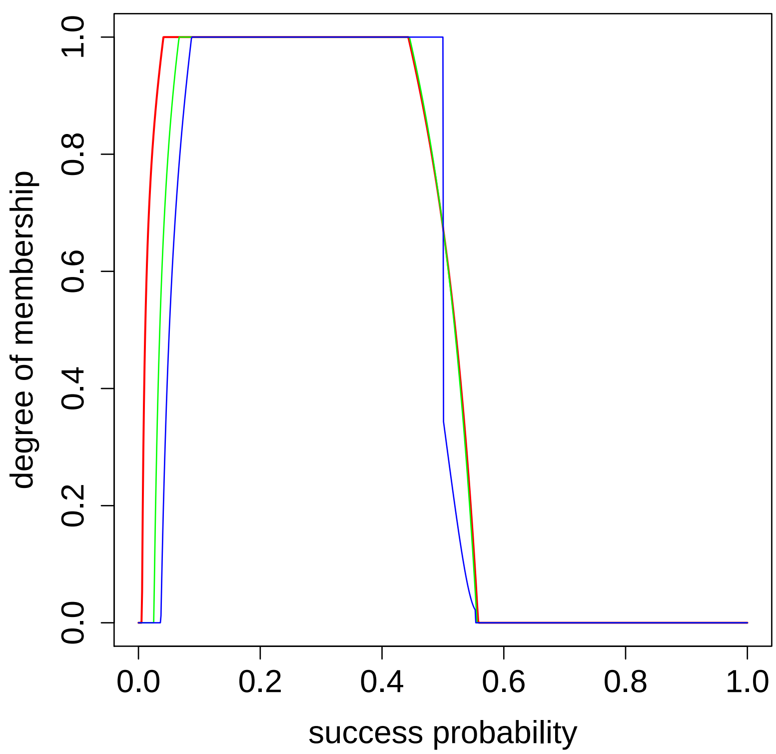


Figure 3: Fuzzy Confidence Intervals. 95% fuzzy confidence intervals for the binomial distribution, $x = 2$, $n = 10$. Red is UMPU (Geyer-Meeden), green is equal-tailed (Agresti-Gotard), blue is Pratt (Brown-Cai-DasGupta).

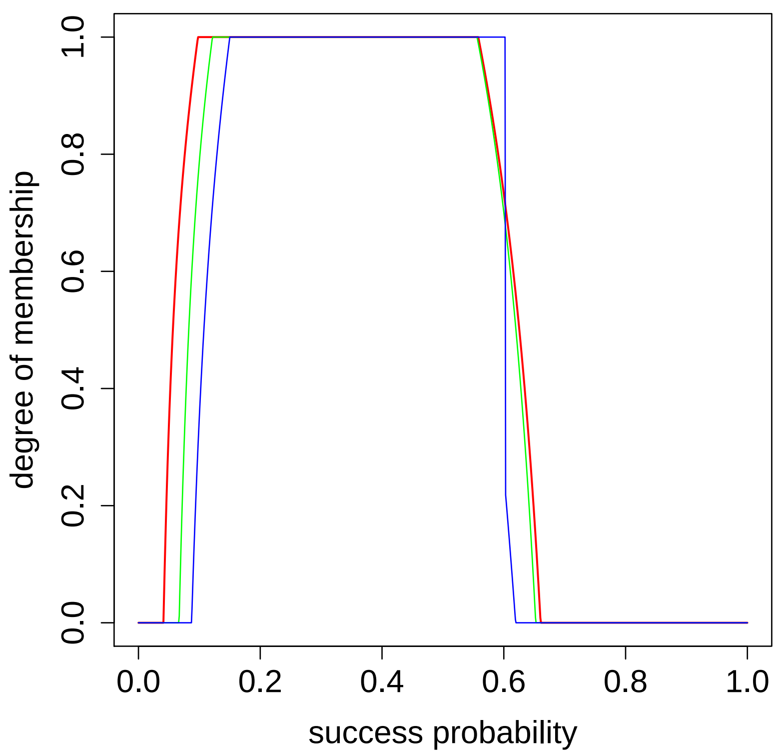


Figure 4: Fuzzy Confidence Intervals. 95% fuzzy confidence intervals for the binomial distribution, $x = 3$, $n = 10$. Red is UMPU (Geyer-Meeden), green is equal-tailed (Agresti-Gotard), blue is Pratt (Brown-Cai-DasGupta).

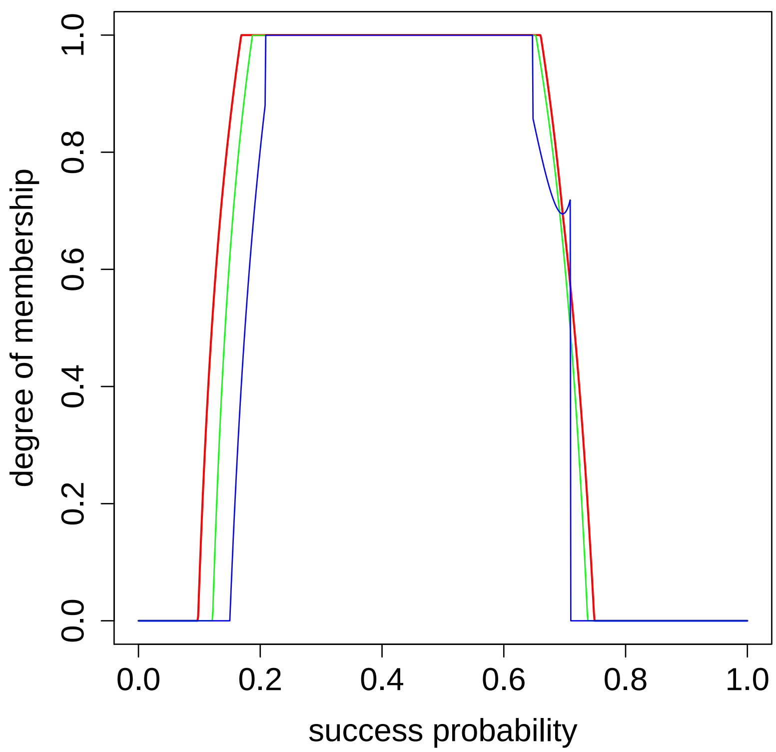


Figure 5: Fuzzy Confidence Intervals. 95% fuzzy confidence intervals for the binomial distribution, $x = 4$, $n = 10$. Red is UMPU (Geyer-Meeden), green is equal-tailed (Agresti-Gotard), blue is Pratt (Brown-Cai-DasGupta).

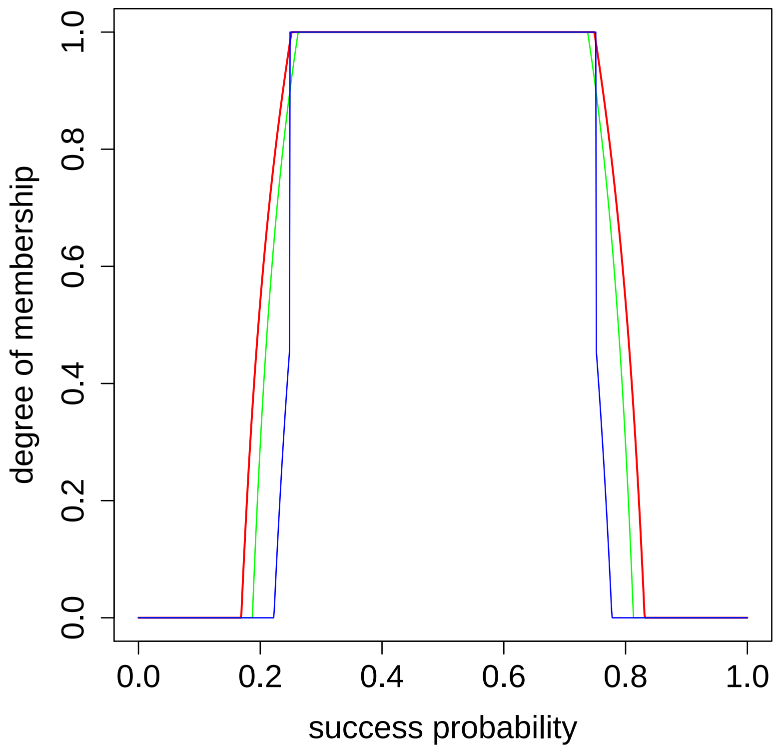


Figure 6: Fuzzy Confidence Intervals. 95% fuzzy confidence intervals for the binomial distribution, $x = 5$, $n = 10$. Red is UMPU (Geyer-Meeden), green is equal-tailed (Agresti-Gotard), blue is Pratt (Brown-Cai-DasGupta).

## References

Agresti, A. and Gottard, A. (2005). Randomized confidence intervals and the mid-P approach, comment on Geyer and Meeden (2005). To appear in *Statistical Science*.

Brown, L. D., Cai, T. T., and DasGupta, A. (2005). Comment on Geyer and Meeden (2005). To appear in *Statistical Science*.

Geyer, C. J. and Meeden, G. D. (2005). Fuzzy and randomized confidence intervals and p-values (with discussion). To appear in *Statistical Science*, `http://www.stat.umn.edu/geyer/fuzz`.

Lehmann, E. L. (1959). *Testing Statistical Hypotheses*. Wiley, New York. (2nd ed., Wiley, 1986; Springer, 1997).

Pratt, J. W. (1961). Length of confidence intervals. *Journal of the American Statistical Association*, 56, 549–567.

Stevens, W. L. (1950). Fiducial limits of the parameter of a discontinuous distribution. *Biometrika*, 37, 117–129.