

Demo: Average-case performance of interval estimation (draft version)

Paul Gustafson

July 24, 2025

This R markdown demo resides at github.com/paulgstf/Bayes-demos

The beginning

Our first task is to reconstruct a plot from the previous demo in this series.

Say the joint distribution of observable binary variables (X_1, X_2, Y) is parameterized by $p_{ij} = Pr(X_1 = i, X_2 = j)$ and $q_{ij} = Pr(Y = 1 | X_1 = i, X_2 = j)$. The scalar parameter of interest is taken to be $\psi = q_{10} - q_{00}$, the risk difference between the high and low levels of X_1 , when $X_2 = 0$.

We generate an ensemble of possible “states of the world” by sampling *iid* draws from “Nature’s prior,” which we set as $p \sim \text{Dirichlet}(1, 1, 1, 1)$, $q_{ij} \sim \text{Unif}(0, 1)$, all mutually independent. A function to generate one draw from Nature’s prior is as follows:

```
truth <- function() {  
  p <- matrix(rdirichlet(1, rep(1,4)), 2,2)  
  q <- matrix(runif(4), 2, 2)  
  
  trgt <- q[2,1] - q[1,1]  
  
  list(p=p, q=q, trgt=trgt)  
}
```

Next up is a function that, given parameter vector (or “truth,” or “state-of-world”), (p, q) , generates a dataset of n observations. We use $n = 250$.

```
datagen <- function(truth, n=250) {  
  
  x.ct <- as.vector(rmultinom(1, size=n, prob=truth$p))  
  y.ct <- rbinom(4, size=x.ct, prob=truth$q)
```

```
list(x.ct=x.ct, y.ct=y.ct)
}
```

Finally, we need a function to determine the posterior distribution of the parameters given a dataset. The following function accomplishes this task by producing (a large number of) Monte Carlo realizations from the posterior distribution.

```
posterior.MC <- function(dta, hyp.a=rep(1,4), hyp.b=rep(1,4), m=10000) {
  # posterior distribution of q
  # represented as m iid draws (rows)
  q.pst <- t(replicate(m,
    rbeta(4, shape1=hyp.a + dta$y.ct,
          shape2=hyp.b + dta$x.ct-dta$y.ct)))

  # omitting the posterior distribution of p,
  # not needed under present circumstances

  # return posterior summaries for (just) the target parameter
  list(mn=mean(q.pst[,2]-q.pst[,1]),
        sd=sd(q.pst[,2]-q.pst[,1]),
        cred.90=quantile(q.pst[,2]-q.pst[,1], c(0.05,0.95)))
}
```

Note that some *hyperparameters* are inputs to this function. These encode the investigator's prior specification that $q_{ij} \sim \text{beta}(a_{ij}, b_{ij})$. The default values $(a_{ij}, b_{ij}) = (1, 1)$ correspond to uniform priors on the four elements of $Pr(Y = 1|X_1, X_2)$.

Note also that the function above returns the mean and standard deviation of the posterior distribution of ψ , as well as the 5-th and 95-th percentiles of this distribution. More precisely, the function outputs are *numerical approximations* to these quantities, with the numerical error diminishing as m , the number of Monte Carlo draws from the posterior distribution, increases. When using Monte Carlo methods to implement Bayesian analysis, it is very unwieldy to *always* remind the reader that we are reporting numerical approximations of posterior quantities, rather than the quantities themselves. Vigilance is required, however, to keep this point in mind. In particular, it behooves us to choose m large enough so that the numerical error is negligible for practical purposes.

And note that the pair of posterior quantiles is referred to as a 90% posterior *credible interval* for ψ , with the interpretation of 90% probability that the interval contains the true value of the target, given the observed data. In fact a more fulsome description is that the 0.05 and 0.95 posterior quantiles comprise the 90% *equal-tailed* posterior credible interval. Discarding equal amounts of probability at either end of the posterior distribution is an intuitive way to construct a set with a desired (high) probability of containing the target. There are, however, other compelling schemes for achieving this objective.

We are now in a position to form an ensemble of parameter values (generated from Nature's prior), corresponding datasets, and posterior inferences arising from each of these datasets.

```
ensmbl.sz <- 4000

### generate an ensemble of parameter values
truth.ensemble <- replicate(n=ensmbl.sz, truth(), simplify=F)

### generate the corresponding datasets
data.ensemble <- lapply(truth.ensemble, datagen)

### generate the corresponding inferences
infrnc.ensemble <- lapply(data.ensemble, posterior.MC)
```

To make this feel less abstract, for the first element of the ensemble, we have the underlying truth:

```
truth.ensemble[[1]]

## $p
##      [,1] [,2]
## [1,] 0.351 0.4334
## [2,] 0.116 0.0995
##
## $q
##      [,1] [,2]
## [1,] 0.681 0.547
## [2,] 0.137 0.678
##
## $trgt
## [1] -0.543
```

Followed by the data arising from this truth:

```
data.ensemble[[1]]

## $x.ct
## [1] 72 37 113 28
##
## $y.ct
## [1] 52 5 49 17
```

Or, more intuitively,

```
kable(cbind(c(0,0,1,1), c(0,1,0,1),
            data.ensemble[[1]]$x.ct - data.ensemble[[1]]$y.ct,
            data.ensemble[[1]]$y.ct),
      col.names=c("X1", "X2", "Y=0", "Y=1"))
```

X1	X2	Y=0	Y=1
0	0	20	52
0	1	32	5
1	0	64	49
1	1	11	17

This is further followed by the inference arising from these data:

```
infrnc.ensemble[[1]]
```

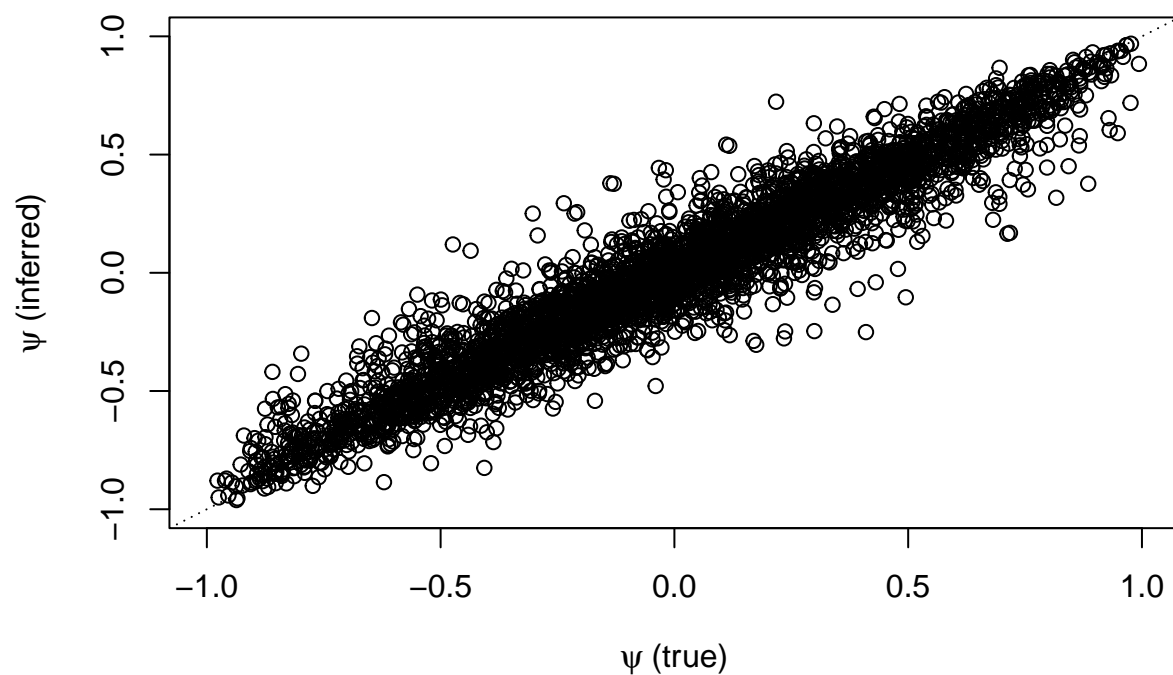
```
## $mn
## [1] -0.562
##
## $sd
## [1] 0.0757
##
## $cred.90
##      5%      95%
## -0.678 -0.430
```

Coming back to the ensemble, to show aggregate point-estimator performance we recreate a plot from the previous demo:

```
Truth <- sapply(truth.ensemble, function(z){z$trgt})

Estimate <- sapply(infrnc.ensemble, function(z){z$mn})

plot(Truth, Estimate, xlim=c(-1,1),ylim=c(-1,1),
     xlab=expression(paste(psi," (true)")),
     ylab=expression(paste(psi," (inferred)")))
)
abline(c(0,1), lty=3)
```



Note here that the vertical distance of points from the identity line represents estimation error across the ensemble.

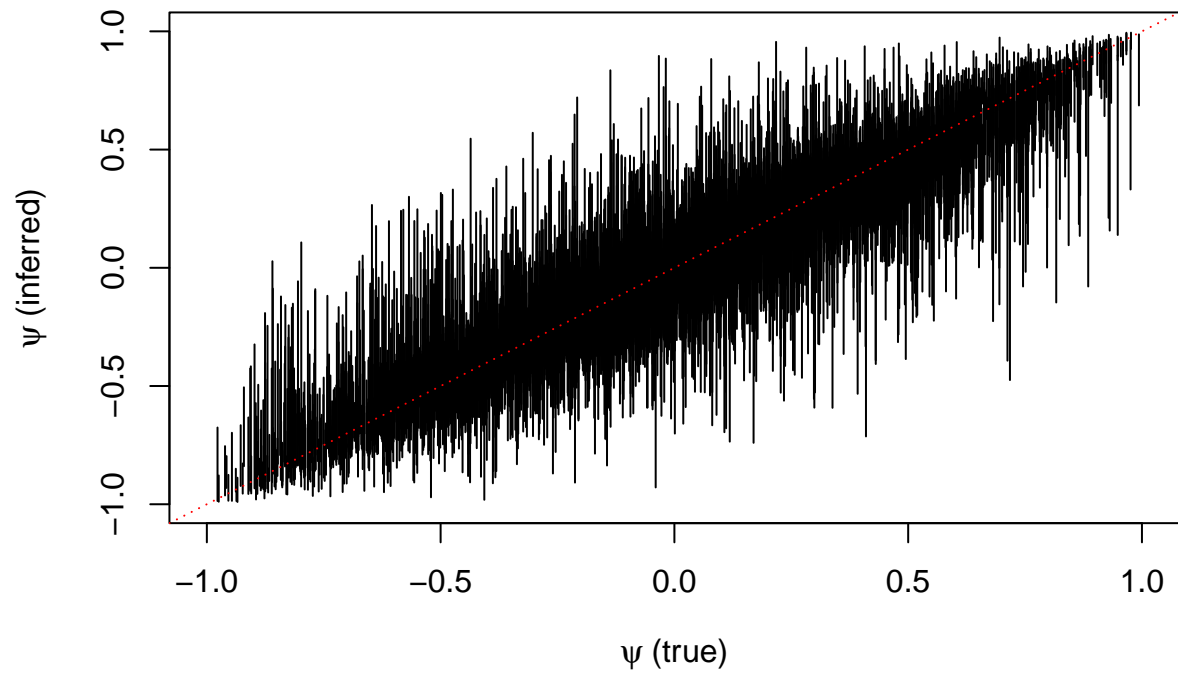
To extend this, we now consider the 90% credible intervals in the same way:

```
### interval estimate
Estimate <- sapply(infrnc.ensemble, function(z){z$cred.90})

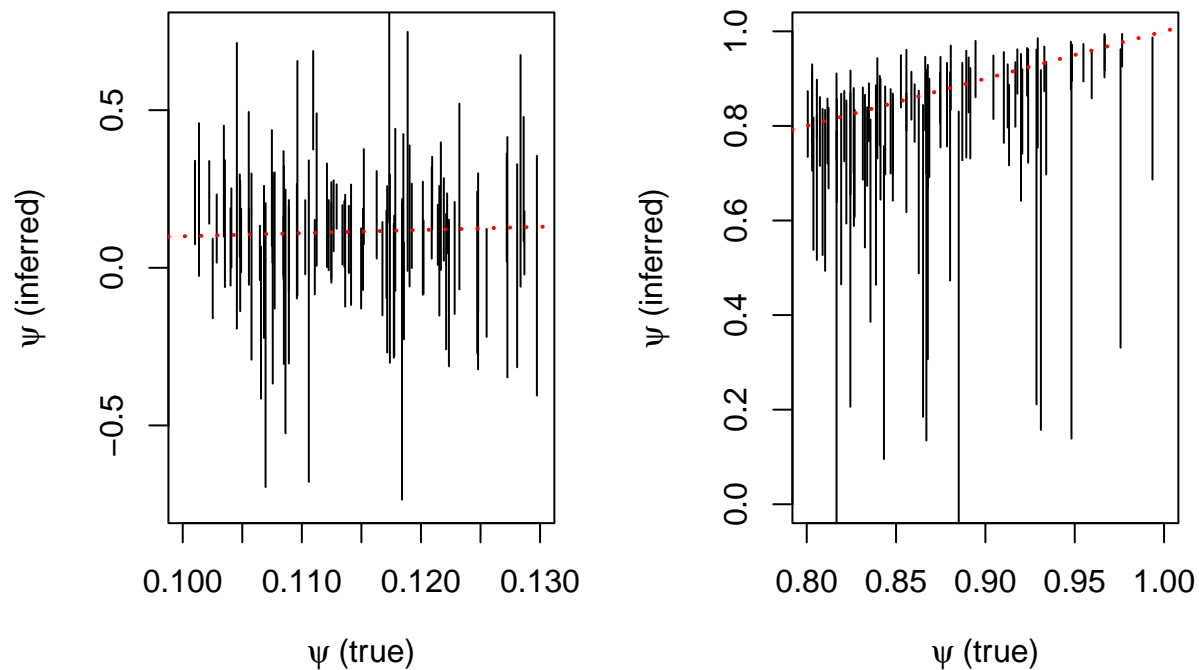
plot(-2, -2,
     xlim=c(-1,1), ylim=c(-1,1),
     xlab=expression(paste(psi," (true)")),
     ylab=expression(paste(psi," (inferred)")))

for (i in 1:ensmbl.sz) {
  points(rep(Truth[i],2), Estimate[,i], type="l")
}

abline(c(0,1), lty=3, col="red")
```



Since this plot is crowded, we also provide a couple of zoom-ins to sub-intervals of the true ψ values:



We see that for most ensemble members, the 90% credible interval contains the true value of ψ . To make this more quantitative, we report the proportion of instances in which this happens:

```
emp.cov <- mean((Estimate[1,]<Truth) & (Truth<Estimate[2,]))
emp.cov
```

```
## [1] 0.898
```

This result is in line with what we would hope, i.e., the 90% credible interval meets its objective in 90% of instances. In fact, we can be more quantitatively rigorous here. Viewing the 4000 generated scenarios as a random sample from all possible scenarios, we can associate a Monte Carlo standard error of 0.01 with the reported coverage of 0.898. Thus our simulated result is *consistent with* the probability of obtaining an interval which “hits” its target being exactly 90%. Bear in mind here that this is a probability statement about the *joint* distribution the parameter vector (p, q) and the observable data D .

As a further fun fact, we did not actually need an empirical simulation to verify that, across the ensemble, 90% of the 90% credible intervals will contain their target parameter. It is a mathematical certainty. To pontificate a bit, we are starting to stock up on related mathematical certainties. In the previous demo, we mentioned that the posterior mean of the

target parameter achieves the best possible average-case performance of *any* point estimator, *provided that the distribution of parameter values used to form the average (Nature’s prior) and the distribution the investigator uses to conduct the analysis (investigator’s prior) coincide*. Now we add a calibration guarantee to sit alongside: When these two distributions coincide, the fraction of resulting credible intervals which cover the true value of the target parameter is exactly the advertised level of the intervals. There are no caveats here, i.e., no regularity conditions or appeals to large-sample theory. This calibration guarantee is discussed at greater length in Chapter 3 of *Bayesian Statistical Inference: Concepts, Hallmarks, and Operating Characteristics*.

Of course, equating Nature’s prior and the investigator’s prior is in the realm of a thought experiment. In reality, you (as the investigator) must choose a prior distribution for the inference you plan to perform. What we know is that *if* you are willing to regard the pending dataset as arising from first drawing the true parameter value from this prior and then having the data ensue, *then* your credible interval is calibrated, with respect to the ensemble of parameter-data pairings that could arise via this mechanism.

Frequentist confidence intervals

For those familiar with the construct of a *frequentist confidence interval* (FCI), conceptualizing a prior distribution for the investigator, let alone for Nature, may seem like tilting at windmills. Turning to a problem where FCIs are trivially available, say we will receive data in the form of n *iid* observations $Y_i \sim N(\theta, 1)$, with the dataset summarized by its sample mean, $\bar{Y} \sim N(\theta, n^{-1})$. A $(1 - \alpha)$ level confidence interval is given as $\bar{Y} \pm n^{-1/2} z_{\alpha/2}$, where $z_{\alpha/2}$ is the $(1 - \alpha/2)$ quantile of the standard normal distribution. For *any* underlying value of θ , the probability that the interval contains θ is $1 - \alpha$.

To contrast with a Bayesian analysis, say the investigator uses the prior distribution $\theta \sim N(\mu, \kappa_V^2)$. In aid of interpretation, it can be useful to think of prior precision rather than prior variance. More pointedly, we call this precision the *prior effective sample size*, denoted as $n_V^* = \kappa_V^{-2}$. The rationale of this phrasing becomes evident upon noting that the posterior distribution of θ is normal, with

$$E(\theta | \bar{Y} = \bar{y}) = \frac{n_V^* \mu + n \bar{y}}{n_V^* + n}, \quad (1)$$

$$\text{Var}(\theta | \bar{Y} = \bar{y}) = \frac{1}{n_V^* + n}. \quad (2)$$

We see overtly that the prior distribution contributes n^* units of information to the analysis, relative to the n provided by the data. As a practical matter, since the posterior distribution of θ is the normal distribution with moments (1) and (2), the equal-tailed level $(1 - \alpha)$ posterior credible interval for θ , takes the form of $E(\theta | Y = y) \pm SD(\theta | Y = y) z_{\alpha/2}$.

From the form of the posterior credible interval, computing its frequentist coverage is an exercise in algebraic manipulations. We encode the result as:


```

### Frequentist coverage

FC <- function(theta, n, n.eff, mu=0, lvl=0.9) {

  ### return frequentist coverage of the Bayesian interval
  ### n.eff is the (investigator) prior effective sample size

  r.eff <- n.eff/n; z <- qnorm(1-(1-lvl)/2)

  pnorm(r.eff*sqrt(n)*(theta-mu) + z*sqrt(1+r.eff)) -
  pnorm(r.eff*sqrt(n)*(theta-mu) - z*sqrt(1+r.eff))

}

```

To help intuit what goes on here, we consider the limit of $n_V^* \downarrow 0$ (or equivalently $\kappa_V \uparrow \infty$), whilst other inputs remain fixed. In one sense, this is not a “realizable” limit, i.e., there is no such thing as a normal prior distribution with infinite variance. More pragmatically, however, in this limit the endpoints of the credible interval tend to the endpoints of the frequentist confidence interval. Unsurprisingly, then, we can “read off” that the function will return $FC(\theta) = 1 - \alpha$ for all θ , when the input is $n_V^* = 0$. We can also establish that for $n_V^* > 0$, the first derivative of the coverage with respect to θ is zero when $\theta = \mu$, and indeed that $FC()$ is maximized at $\theta = \mu$. This backs up the intuitive notion that the credible interval will be particularly good at capturing truth when the truth is particularly consistent with the prior distribution. That is, the frequentist coverage will be highest when the true value of θ coincides with the mean of the investigator’s prior.

As a demonstration, we visualize $FC(\theta)$ when $n = 50$, $\alpha = 0.9$, and (i), $n_V^* = 1$, $\mu = 0$; (ii), $n_V^* = 4$, $\mu = 0$; (iii), $n_V^* = 16$, $\mu = 1.5$.

```

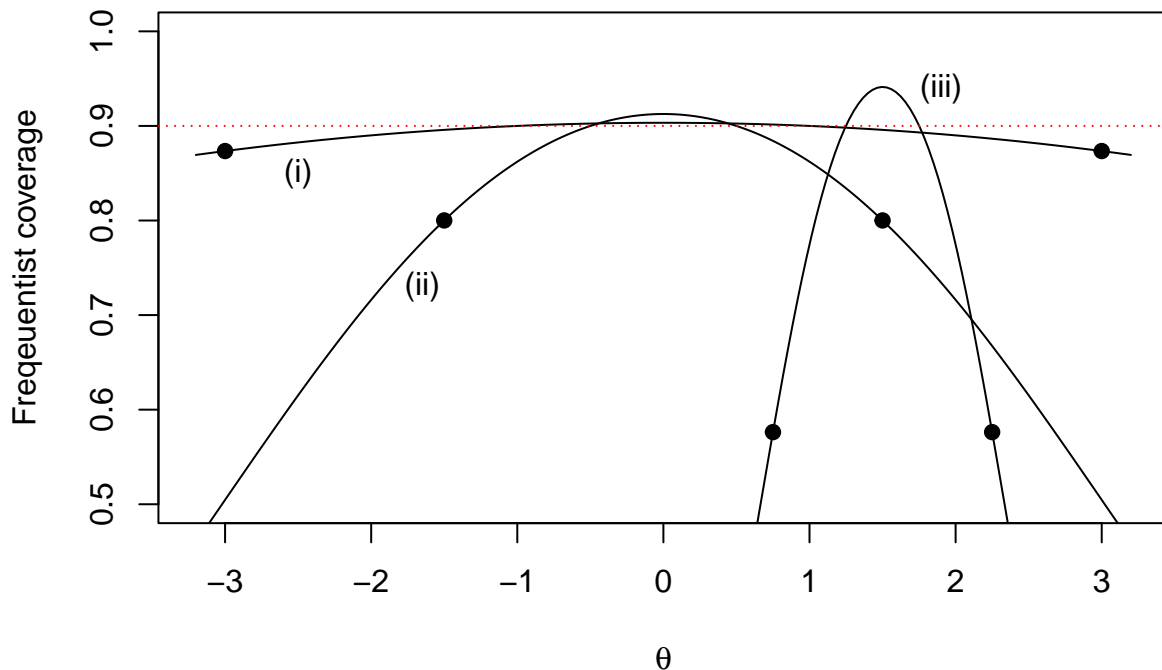
th.grd <- seq(from=-3.2, to=3.2, length=200)
plot(th.grd, FC(th.grd, n=50, n.eff=1, mu=0), type="l",
     xlab=expression(theta), ylab="Frequentist coverage", ylim=c(0.5,1))
points(c(-3,3), FC(c(-3,3), n=50, n.eff=1, mu=0), pch=19)

points(th.grd, FC(th.grd, n=50, n.eff=4, mu=0), type="l")
points(0.5*c(-3,3), FC(0.5*c(-3,3), n=50, n.eff=4, mu=0), pch=19)

points(th.grd, FC(th.grd, n=50, n.eff=16, mu=1.5), type="l")
points(1.5+0.25*c(-3,3), FC(1.5+0.25*c(-3,3), n=50, n.eff=16, mu=1.5), pch=19)

abline(h=.9, lty=3, col="red")
text(-2.5, 0.85, "(i)"); text(-1.65, 0.73, "(ii)"); text(1.9, 0.94, "(iii)")

```



In each case, for θ near the prior mean, frequentist coverage exceeds the nominal level, but then falls below this level as θ moves away from μ . Moreover, the decline is more precipitous for narrower (i.e., larger n_V^*) prior distributions. Measured through the lens of frequentist coverage, a narrow prior is risky, if in fact the truth lies far from the prior mean. Looking at the marked points on the plot, which correspond to θ lying three prior standard deviations from the prior mean, we see that frequentist coverage when θ is in the tail of the investigator's prior is lower when the prior is narrower. Of course, this does not equate to a narrower prior being all around worse. A narrower prior brings tighter inference to the table, with the width of the credible interval scaling in proportion to $(1 + n_V^*/n)^{-1/2}$.

Bayesian Coverage, Revisited

We now turn attention to Bayesian coverage, i.e., with respect to the ensemble of parameter-data pairs generated, where Nature's prior governs the drawing of parameters, while the investigator's prior governs the posterior inference arising from each dataset. We already noted that when the two prior distributions coincide, the result is pleasingly simple and intuitive: The Bayesian coverage equals the advertised level of the credible interval. Now, however, we enter the weedy situation of a discrepancy between the investigator and Nature.

We stick with the normal data and prior situation from the previous section. The pertinent inputs are the size of the dataset, n , the hyperparameters which specify the investigator's prior, (μ_V, n_V^*) , and the hyperparameters which specify Nature's prior, (μ_N, n_N^*) . In fact we

simplify further to the case of $\mu_V = \mu_N = 0$. In doing so, we focus just on the widths of the two prior distributions, as expressed via the respective effective sample sizes.

We could take an overt path to an empirical demonstration, by repeatedly sampling a (θ, \bar{Y}) pair, computing the posterior credible interval based on each \bar{Y} , and comparing this to the value of θ that spawned said \bar{Y} . It turns out, however, that the frequentist coverage math implemented in the previous section is a helpful stepping stone to determining the Bayesian coverage. Extending slightly the previous notation, let $FC_n(\theta; n_V^*)$ be the frequentist coverage of the Bayesian interval, for sample size n , and investigator-specified prior effective sample size n_V^* . The mathematical shortcut, described at more length in Chapter 3 of *Bayesian Statistical Inference: Concepts, Hallmarks, and Operating Characteristics*, sees the Bayesian coverage of interest expressed as:

$$BC_n(\pi_N; \pi_V) = E_{\pi_N} \{FC_n(\theta; n_V^*)\}. \quad (3)$$

Quite literally, the Bayesian coverage, is the average frequentist coverage of the Bayesian procedure (based on the investigator's prior). The key here is that *average* is (i), across the parameter space, and (ii), weighted according to Nature's prior.

We demonstrate when the sample size is $n = 200$, using all combinations of n_N^* and n_V^* drawn from $\{1, 5, 25, 100\}$.

Bayesian coverage computed as an average of frequentist coverage

```
BC <-function(n, n.eff.inv, n.eff.nat) {
  th.grid <- qnorm((1:10000)/10001, mean=0, sd=sqrt(1/n.eff.nat))
  mean(FC(th.grid, n=n, n.eff=n.eff.inv))
}
```

```
n.grid <- c(1,5,25,100); ans <- matrix(NA,4,4)
```

```
for (i in 1:4) {for (j in 1:4) {
  ans[i,j] <- BC(n=200, n.eff.nat=n.grid[i], n.eff.inv=n.grid[j])
} }
```

Nature's prior ESS varies by row, investigator's prior ESS varies by column
kable(ans)

	1	5	25	100
1	0.900	0.884	0.610	0.222
5	0.901	0.900	0.829	0.456
25	0.901	0.903	0.900	0.755
100	0.901	0.904	0.914	0.900

As a first comment on the table, it was pre-ordained that the diagonal entries match the advertised coverage, since here Nature and the investigator use the same prior distribution.

(Bit more of a rabbit hole on this: Consider the plots of $FC(\theta)$ versus θ in the previous section. When we take the average height of the curve, weighted by π_V , we get an exact cancellation of the “above-advertised” and “below-advertised” portions of the curve.)

As a second comment, we see very mild over-coverage when the investigator’s prior is wider than that of Nature, whereas we can see dramatic under-coverage when the investigator’s prior is much narrower. This speaks to merit in tending to specify wide priors as a conservative strategy. It also fits in to the myriad of statistical settings in which overconfidence, writ large, has negative consequences. Though of course this conservatism comes with a cost of less tight inference. In the present instance, the interval width reduction as we move from the frequentist confidence interval (effectively the $n_V^* = 0$ case), through $n_V^* = 1, 5, 25, 100$ is as follows.

```
sqrt(1/(1 + c(0,n.grid/200)))
```

```
## [1] 1.000 0.998 0.988 0.943 0.816
```

When it is justifiable in terms of what is known about the scientific context at hand, a narrower prior distribution is your friend!

Now a partially identified problem

Say we wish we could observe *iid* realizations of binary variable Y , in order to make inference about $\theta = Pr(Y = 1)$. That is, we wish to infer the prevalence of a binary trait in a population. However, our “Y measuring machine” is imperfect: for a given study subject it actually returns Y^* . The machine is known to be perfectly *specific*, meaning that $Y = 0$ implies $Y^* = 0$. In terms of *sensitivity*, however, $\lambda = Pr(Y^* = 1|Y = 1)$ may fall below one. And, we are unsure about the extent of this problem. Say, for instance, we know only that λ exceeds some lower bound c . We exemplify using the choice $c = 0.8$ below.

Noting that $Pr(Y^* = 1) = \lambda\theta$, and letting S^* be the sum of the n binary measurements at hand, we encapsulate the situation with the statistical model $S^* \sim \text{Bin}(n, \lambda\theta)$, along with prior specification that $\theta \sim \text{Unif}(0, 1)$ independently of $\lambda \sim \text{Unif}(c, 1)$. This problem is readily appreciated to be nonstandard, since we can’t expect the observable S^* to be able to “separate” θ from λ , even though θ is our scientific parameter of interest. In more technical parlance, the problem is not statistically identified. We shall see implications of this weirdness in what follows.

To understand Bayesian inference in this problem, it is convenient to reparameterize from (θ, λ) to (ϕ, λ) , where $\phi = \lambda\theta$. The prior distribution transforms accordingly to:

$$\pi(\phi) = (1 - c)^{-1} \{-\log \max(1 - c, \phi)\}, \quad (4)$$

$$\pi(\lambda|\phi) = \{-\log \max(1 - c, \phi)\}^{-1} \frac{1}{\lambda} I\{\lambda > \max(1 - c, \phi)\}. \quad (5)$$

While not essential for what follows, note that this structure makes it easy to implement Monte Carlo draws from the posterior distribution, using importance sampling. So we work with (ϕ, λ) computationally, but transform back to (θ, λ) for scientific interpretation.

```
### Monte Carlo sample from the posterior of (phi,lambda)
### Uses importance sampling

pst.mc <- function(y,n,ctf, m=20000) {

  phi.mc <- rbeta(2*m, 1+y, 1+(n-y))

  wht <- -log(pmax(ctf,phi.mc))
  phi.mc <- sample(phi.mc, size=m, replace=T, prob=wht/sum(wht))

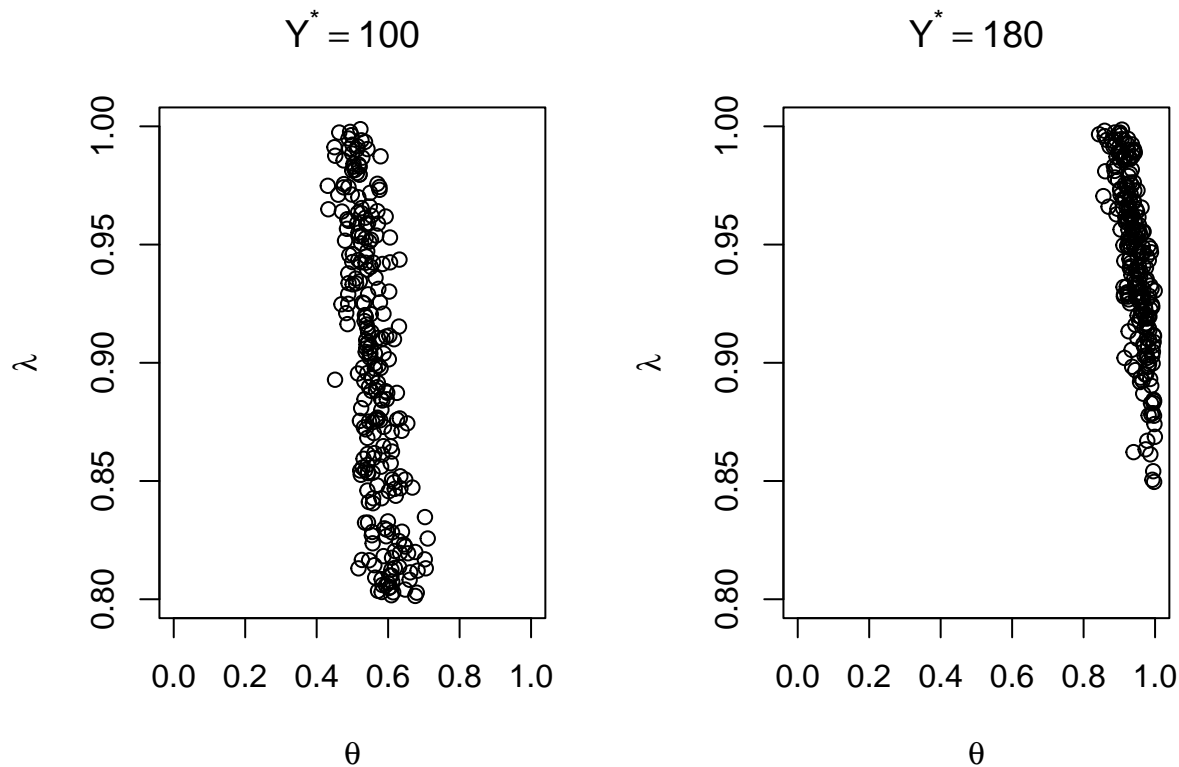
  lam.mc <- pmax(phi.mc, ctf)^(1-runif(m))

  list(th=phi.mc/lam.mc, lam=lam.mc)
}
```

We give two brief examples of the posterior distribution arising when $n = 200$ and, respectively, $Y^* = 100$ and $Y^* = 180$.

```
xmp.1 <- pst.mc(100, 200, ctf=0.8, m=250)
xmp.2 <- pst.mc(180, 200, ctf=0.8, m=250)
```

We can “see” the posterior distribution of (θ, λ) in each case, via a scatterplot of the Monte Carlo draws from the distribution:



Recall here that the prior distribution on λ is uniform between 0.8 and 1. The first dataset doesn't seem to resolve any of this uncertainty, with the posterior weight still spread across this range. In contrast, the second dataset brings more to the table. By eye at least, a value of λ less than say 0.87 seems quite unlikely, with a value below 0.85 being very unlikely.

As mentioned above, We are drawn to this example, in part, for its weirdness. The lack of identification means that we can't expect to precisely learn how defective our Y^* measuring device is (as governed by λ), even if we were able to collect a massive amount of data. That said, we might learn a little bit about its fallibility, as per the second dataset above. Or we might learn almost nothing, as per the first dataset. However, the fundamental fact that Bayesian coverage is as advertised, when Nature and investigator share the same prior, is immutable. We can simulate to confirm:

```
nsmb1.sz <- 40000

### ensemble of parameter values
truth.nsmb1 <- list(th=runif(nsmb1.sz), lam=runif(nsmb1.sz, min=0.8, max=1))

### corresponding datasets
dat.nsmb1 <- rbinom(nsmb1.sz, size=200, prob=truth.nsmb1$th*truth.nsmb1$lam)

### corresponding 80% equal-tailed credible intervals for theta
```

```

credi.nsmbl <- matrix(NA, nsmb1.sz, 2)

for (i in 1:nsmb1.sz) {
  credi.nsmbl[i,] <- quantile(pst.mc(y=dat.nsmbl[i], n=200, ctf=0.8)$th,
                             c(0.1,0.9))
}

```

```

### Work out the Bayesian coverage of the Bayesian interval
cvg <- mean( (credi.nsmbl[,1] < truth.nsmbl$th) &
             (truth.nsmbl$th < credi.nsmbl[,2]) )

```

```

### confirm coverage as advertised
c(cvg, sqrt(cvg*(1-cvg)/nsmb1.sz))

```

```
## [1] 0.80315 0.00199
```

For interest, we also work out frequentist coverage of the Bayesian procedure, at the point $(\theta, \lambda) = (0.30, 0.85)$ in the parameter space.

```

sz.fq <- 10000
tru.A <- list(th=0.3, lam=0.85)
dat.A <- rbinom(sz.fq, size=200, prob=tru.A$th*tru.A$lam)

credi.fq.A <- matrix(NA, sz.fq, 2)

for (i in 1:sz.fq) {
  credi.fq.A[i,] <- quantile(pst.mc(y=dat.A[i], n=200, ctf=0.8)$th,
                             c(0.1,0.9))
}

```

```

### empirical coverage and Monte Carlo standard error
cvg.A <- mean( (credi.fq.A[,1] < tru.A$th) & (tru.A$th < credi.fq.A[,2]))

c(cvg.A, sqrt(cvg.A*(1-cvg.A)/sz.fq))

```

```
## [1] 0.81950 0.00385
```

And we repeat this at two other points in the parameter space:

```

tru.B <- list(th=0.9, lam=0.98)
tru.C <- list(th=0.9, lam=0.82)

```

For these we get frequentist coverage proportion of 0.8217 (with MCSE of 0.0038), and 0.6187 (with MCSE of 0.0033), respectively.

Here we see a coverage crash, At the third point considered in the parameter space, the frequentist coverage of the 80% Bayesian credible interval falls far below 80%.

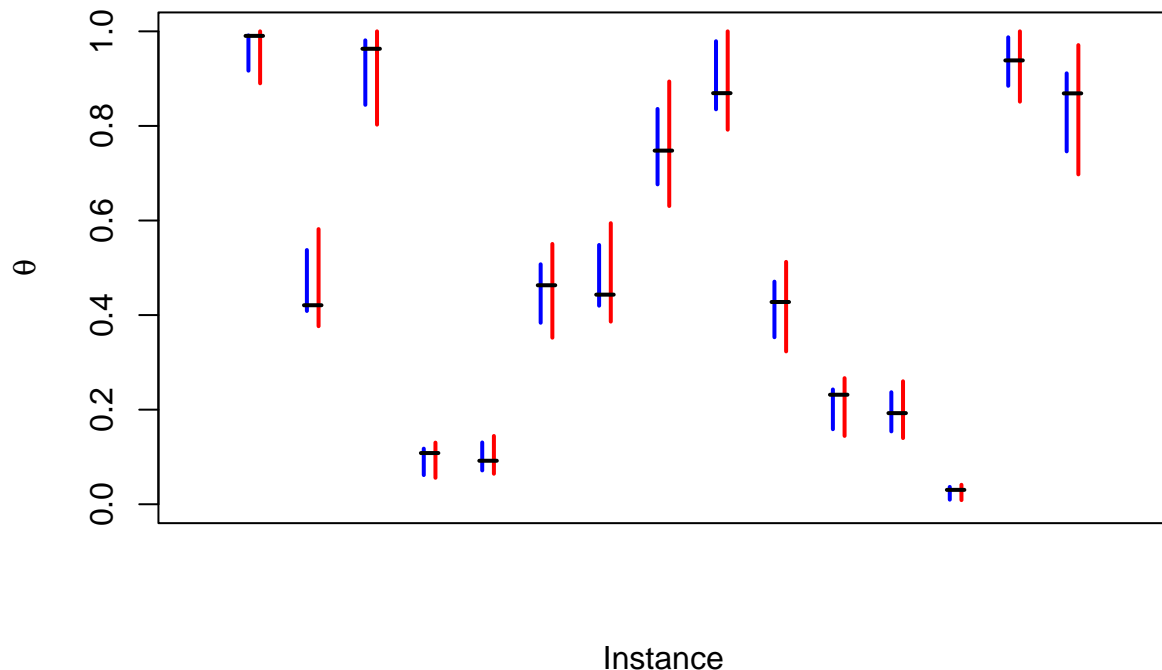
One way to guard against low frequentist coverage at some points in the parameter space would be as follows. Since the data have the form $Y \sim \text{Bin}(n, \phi)$, forming either a Bayesian credible interval or a frequentist confidence interval for ϕ would be a very “regular” problem. Moreover, our parameter of interest θ must lie in $(\phi, \min\{\phi/c, 1\})$. Intuitively then, if (a, b) is an interval estimate for ϕ , then $(a, \min\{b/c, 1\})$ is motivated as an interval estimate for θ . For instance, we can so morph the 80% equal-tailed credible interval for ϕ that arises from a uniform prior on ϕ :

```
credi.cnsv <- cbind(
  qbeta(0.1, shape1=1+dat.nsmbl, shape2=1+200-dat.nsmbl),
  pmin(qbeta(0.9, shape1=1+dat.nsmbl, shape2=1+200-dat.nsmbl)/0.8, 1))
```

To get a feeling for this, for the first 15 parameter-data pairs in the ensemble, we visualize the Bayesian interval, the conservative interval, and the true value of the target parameter:

```
plot(-1,-1, xlim=c(0,16),ylim=c(0,1), type="n", xaxt="n",
     xlab="Instance",ylab=expression(theta))

for (i in 1:15) {
  points(rep(i-.1,2), credi.nsmbl[i,], type="l",col="blue",lwd=2)
  points(rep(i+.1,2), credi.cnsv[i,], type="l",col="red",lwd=2)
  points(i+.15*c(-1,1), rep(truth.nsmbl$th[i],2), type="l", lwd=2)
}
```

Unsurprisingly, just by eye we can see the conservative procedure to indeed be safer, but at the cost of wider intervals.

We can similarly implement the conservative procedure for our “data-only” ensembles generated at three fixed parameter values. And we can summarize both coverage, and average interval width, as follows.

type	cvrg.Bayes	cvrg.cons	width.Bayes	width.cons
Ensemble	0.803	0.957	0.117	0.183
Freq.-A	0.820	0.956	0.100	0.153
Freq.-B	0.822	0.983	0.105	0.152
Freq.-C	0.619	0.964	0.166	0.269

We see a decided lack of a free lunch. Both across the parameter space, and at the three designated points, the actual coverage far exceeds the advertised 80%. This is achieved, however, with average interval widths more than 50% greater than those of 80% credible intervals for θ .

Wrap up

To conclude this vignette, consider, somewhat generically, three interval estimation procedures, each with say an 80% coverage advertisement. These are, respectively, an 80% posterior credible interval arising from prior distribution A over the parameter space, an 80% posterior credible interval arising from prior distribution B, and an 80% frequentist confidence interval. The coverage *guarantees* at hand are summarized thus:

	Nature	Bayes.A	Bayes.B	Conf..Int
distribution A		0.80	???	0.80
distribution B		???	0.80	0.80
fixed point C		???	???	0.80
fixed point D		???	???	0.80
...		???	???	0.80

The Bayesian procedure works (exactly!), albeit under quite specific circumstances: We count hits and misses with respect to joint sampling of parameter and data dyads, with the distribution of parameters (as per Nature) matching the prior chosen for data analysis (as per the investigator). On the other hand, the frequentist confidence interval works, full stop. By construction it works as advertised if we repeatedly sample data under any static value of the parameter vector. And consequently it works as advertised under any choice of Nature's prior distribution. In isolation, this tilts opinion in favour of the confidence interval. As we have seen, however, a frequentist confidence interval will generally be wider than a Bayesian interval. Moreover, it may not be possible to construct a frequentist confidence interval for a given problem. Recognizing these facts puts the Bayesian interval back into the conversation! For a deeper dive on this, see Chapter 3 of the forthcoming *Bayesian Statistical Inference: Concepts, Hallmarks, and Operating Characteristics*.