Demo: Sequential Data Collection and Optional Stopping (draft version)

Paul Gustafson

2024-06-12

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

The beginning

Consider a statistical problem where the datapoints $Y_1, Y_2, \ldots$ arise sequentially in time. We presume these are *iid* observations of the form $Y_i \sim N(\theta, 1)$. We also presume a scientific context with legitimate *a priori* uncertainty about whether or not there might be "an effect," i.e., uncertainty about whether or not $\theta = 0$. For instance, say Y_i is a difference between post-intervention and pre-intervention measurements, for the i-th study subject. Then $\theta = 0$ represents absence of a systematic effect of the intervention.

To acknowledge that $\theta = 0$ is plausible, we let binary parameter M indicate presence of an effect, and specify a joint prior distribution on (M, θ). Specifically, $M \sim \text{Bernoulli}(p)$ *a priori*, while $(\theta | M = 1) \sim N(\mu, \tau^2)$. (And, by construction, $M = 0$ implies $\theta = 0$.)

This prior is indexed by *hyperparameters* (p, μ, τ): p describes the evidence in favor of there being an effect, μ is the best guess of the effect's size (presuming the effect exists), and τ describes variation around this guess.

This is a *conjugate* prior, meaning that the joint posterior distribution of $(M, \theta | Y_{1:n} = y_{1:n})$ has the same form as the prior, but with updated hyperparameters. The impact of acquiring data $Y_{1:n} = y_{1:n}$ is to move the hyperparameters from (p, μ, τ) to (p_n, μ_n, τ_n), where some algebraic manipulations with normal density functions reveal that:

$$\begin{aligned}
\text{logit}(p_n) &= \text{logit}(p) \\
&\quad + \log \phi \left\{ n^{-1/2} \left(1 + \tau^2\right)^{1/2} (\bar{y}_n - \mu) \right\} \\
&\quad - \log \phi \left\{ t^{1/2} \bar{y}_n \right\}, \\
\tau_n^{-2} &= \tau^{-2} + n, \\
\mu_n &= \frac{\tau^2 \mu + t \bar{y}_n}{\tau^2 + n}.
\end{aligned}$$

Here $\bar{y}_n = n^{-1}(y_1 + \ldots + y_n)$ denotes the mean of $y_{1:n}$, while $\phi()$ is the standard normal density function.

We implement this conjugate updating as follows:

```
updater <- function(hyp, y) {

  n <- length(y); y.bar <- mean(y)

  list(
    p = expit(logit(hyp$p) +
              dnorm(y.bar, mean=hyp$mu, sd=sqrt(hyp$tau^2+1/n), log=T) -
              dnorm(y.bar, mean=0, sd=sqrt(1/n), log=T)),

    tau = sqrt(1/ ( 1/hyp$tau^2 + n) ),

    mu = (y.bar + hyp$mu/(n*hyp$tau^2)) / (1+1/(n*hyp$tau^2))
  )
}
```

As a sanity check, we get to the same evidence about parameters whether we update "all in one go," or "one datapoint at a time," with the datapoint order being irrelevant in the latter case:

```
y.test <- c(2.1, 1.8, 0.5)
hyp.0 <- list(p=0.5, mu=0, tau=2)
```

```
unlist(updater(hyp.0, y.test))
```

```
##     p   tau    mu
## 0.845 0.555 1.354
```

```
unlist(updater(updater(updater(hyp.0, y.test[1]), y.test[2]), y.test[3]))
```

```
##     p   tau    mu
## 0.845 0.555 1.354
```

```
unlist(updater(updater(updater(hyp.0, y.test[2]), y.test[3]), y.test[1]))
```

```
##     p   tau    mu
## 0.845 0.555 1.354
```

As a word on interpretation here, our hyperparameter specification of $p = 0.5$ is neutral on whether $\theta \neq 0$. After observing the three data points, however, we lean more toward there being an effect, with $p_3 = Pr(M = 1|Y_{1:3} = y_{1:3}) = 0.845$.

Moving on, we focus on the sequential nature of the data collection. We could visualize a single data trajectory by plotting $\bar{y}_n$ against n. In fact we do this for multiple data trajectories, each of which arises under its own value of (M, θ). This style of data simulation is described at length in an earlier vignette in this series, and in Chapter 2 of *Bayesian Statistical Inference: Concepts, Hallmarks, and Operating Characteristics.* Of note, we refer to generating an *ensemble* of parameter and data pairs, with the distribution of parameter values across the ensemble referred to as *Nature's* prior distribution. For our demonstration, we set this as follows.

```
hyp.nature <- list(p=0.15, mu=0, tau=.5)
```

This setting of $p_0 = 0.15$ is a bit skeptical about the existence of an effect. Such skepticism is realistic for many scientific narratives. For instance, most genes are probably *not* differentially expressed across disease status, and most chemical compounds are probably *not* efficacious as treatments for a given disease.

For our setting of Nature's prior, we proceed to generate an ensemble of parameters and data trajectories.

```
num.traj <- 100   ### number of trajectories to simulate
N <- 50           ### number of datapoints per trajectory

### generate the ensemble of "true" parameter values
M.tr <- rbinom(num.traj, size=1, prob=hyp.nature$p)
theta.tr <- rep(0, num.traj)
theta.tr[M.tr==1] <- rnorm(sum(M.tr), mean=hyp.nature$mu, sd=hyp.nature$tau)

### generate the ensemble of data trajectories, one per each parameter setting
dta <- t(sapply(theta.tr, rnorm, n=N, sd=1))
```

As a sanity check, by construction we should have a mix of zero and non-zero θ values in our ensemble:

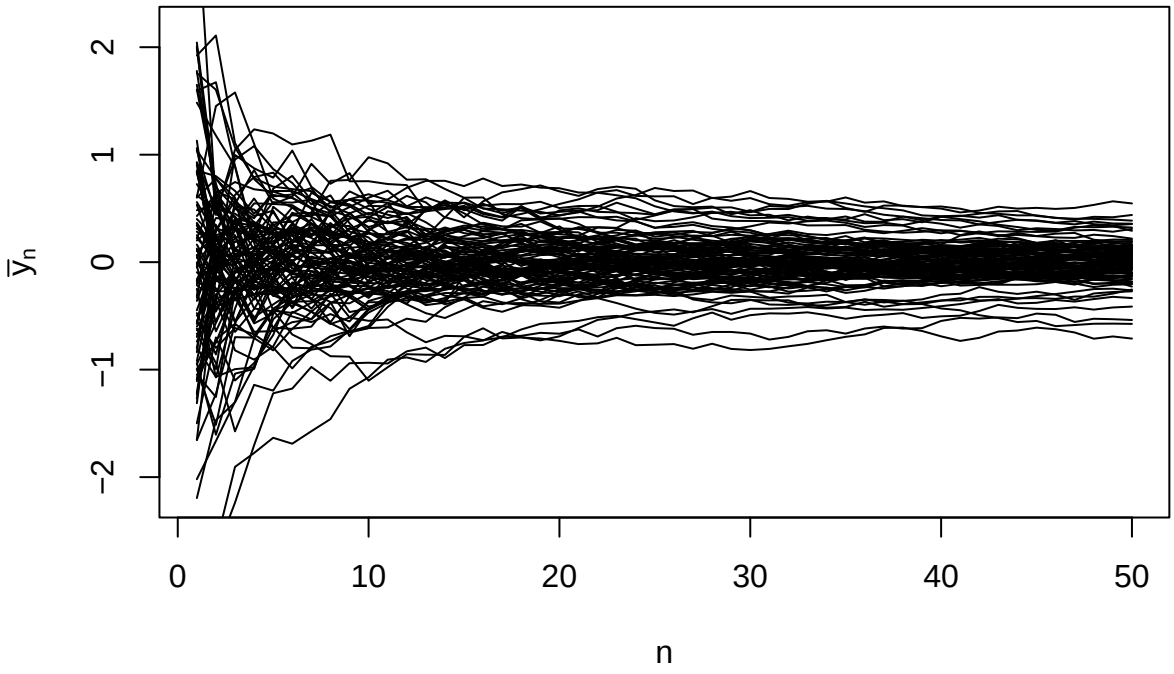
```
head(cbind(theta.tr, dta[,c(1:3, N)]), 10)
```

```
##       theta.tr
##  [1,]    0.000 -0.574  1.1504  1.1438 -0.1126
##  [2,]    0.000  0.853  0.7473  0.3895 -0.9850
##  [3,]    0.000  0.284  0.0295  0.7997  1.2812
##  [4,]    0.000 -1.091  0.2523 -0.1409  0.0284
##  [5,]    0.137 -0.569  0.7714  1.3054  1.7207
```

```
##  [6,]    0.000 -1.092  0.4769  0.3349 -1.8713
##  [7,]    0.000  0.126 -1.0106  0.3813  0.2377
##  [8,]    0.000  0.102  0.3473  0.5386  0.3114
##  [9,]    0.378  1.041  0.5179  0.0609 -0.0487
## [10,]    0.000 -0.696  0.5071  1.3884  1.0907
```

Next we visualize all the datastreams in terms of the evolution of the sample mean.

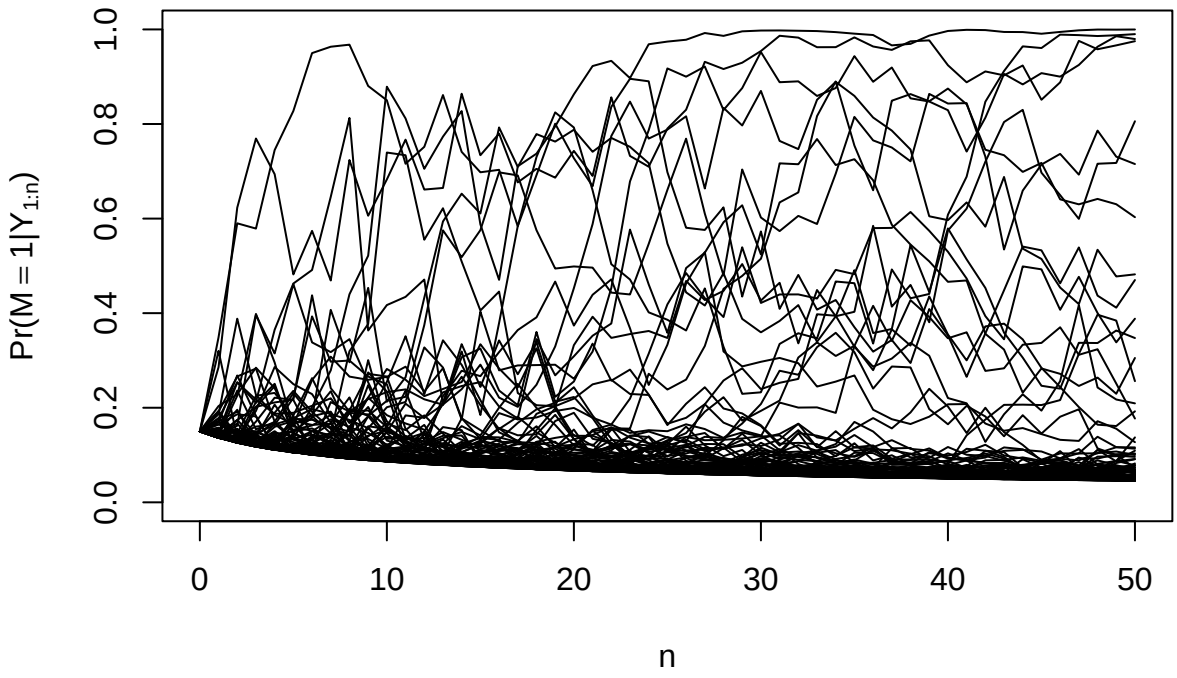
```
dta.cummean <- t(apply(dta, 1, FUN=GMCM:::cummean))

plot(0, 0, type="n",
  xlim=c(1,N), xlab="n",
  ylim=2.2*c(-1,1), ylab=expression(bar(y)[n]))

apply(dta.cummean, 1, FUN=points, x=1:N, type="l")
```



For each datastream, we can similarly view how the posterior probability in favor of an effect, p_n, evolves.

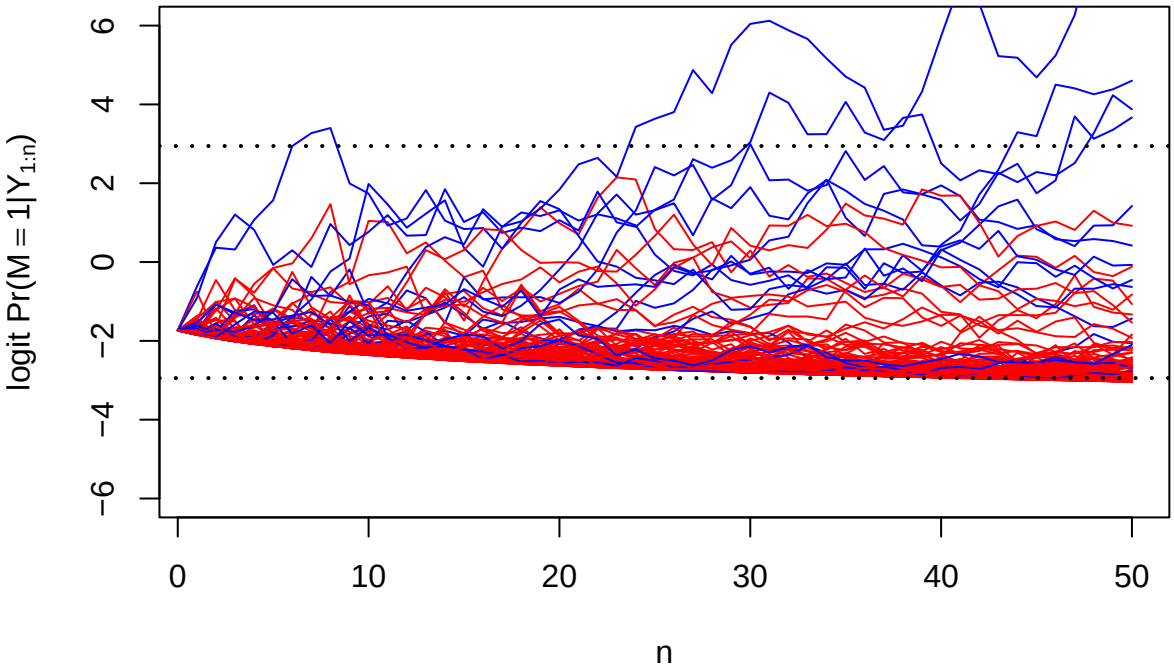
```r
dta.postprob <- matrix(NA, num.traj, N)
### (i,j) entry will be p_j for the i-th datastream

for (i in 1:num.traj) {

  hyp <- hyp.nature

  for (t in 1:N) {
    hyp <- updater(hyp, dta[i,t])
    dta.postprob[i,t] <- hyp$p
  }

}
```

```r
plot(0,0, type="n", xlim=c(0,N), ylim=c(0,1),
  ylab=expression(paste("Pr(", M==1, "|", Y[1:n],")")),
  xlab="n")

apply(cbind(hyp.nature$p,dta.postprob), 1, FUN=points, x=0:N, type="l")
```



Next we modify this plot in a few ways, to better understand what is going on:

- We plot $logit(p_n)$ instead of p_n, to better see the behavior of extreme posterior probabilities.
- We add two horizontal reference lines at values with some "heft" in terms of knowledge about the presence or absence of an effect. We place these at logit(0.05) and logit(0.95).
- We take advantage of the fact that we created these ficticious datastreams, so we know which of them arose from $M = 0$ (so $\theta = 0$), versus from $M = 1$ (so $\theta \neq 0$). Hence we color-code as such (red/blue for the former/latter).

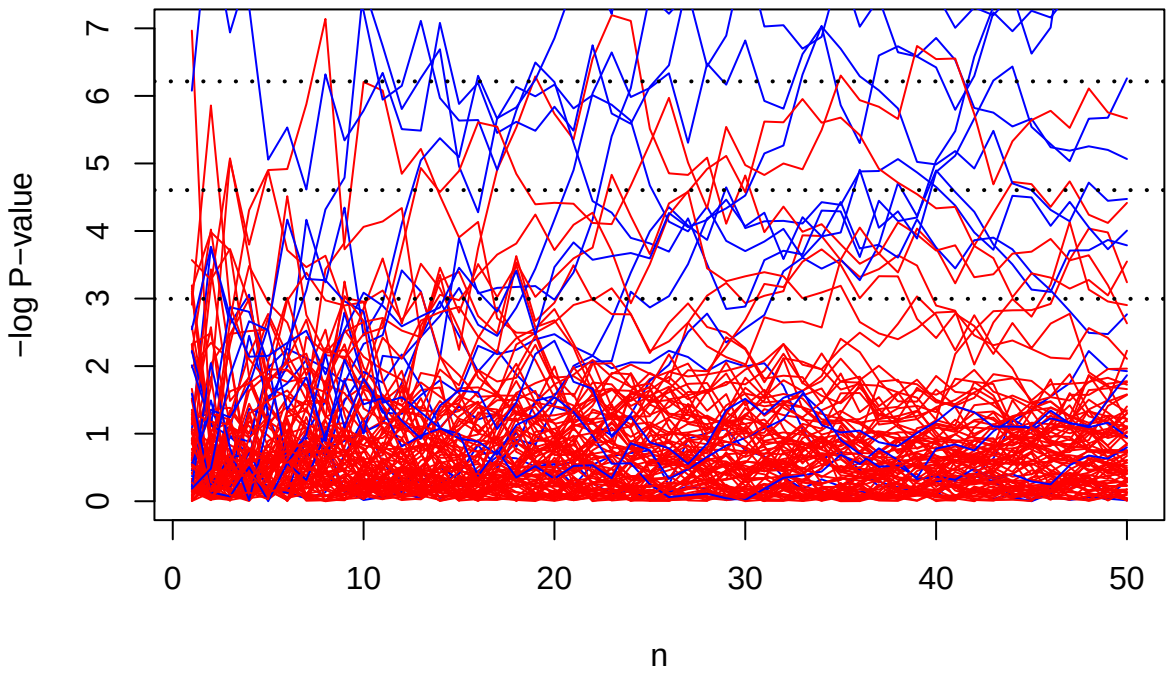


A portion of the (blue) trajectories arising from an effect do indeed yield posterior probabilities of an effect that (i), exceed 95% relatively early in the trajectory, and/or (ii), greatly exceed 95%. Similarly, some of the (red) trajectories arising in the absence of an effect do yield posterior probabilities of an effect below 5%. But this tends to manifest late in the time trajectory, and only just. So we see a fundamentally asymmetry in how evidence accrues for, versus against, the existence of an effect.

Having observed these patterns for the posterior probability of an effect, we can't help but be curious - what would the corresponding story be for frequentist P-values, with their attendant interpretation of a small P-value being evidence for an effect. To ease interpretation, we give this plot on the scale of $-\log P$, with hoizontal reference lines corresponding to P-values of 0.05, 0.01, and 0.002.

```
dta.pval <- matrix(NA, num.traj, N)

for (t in 1:N) {
  dta.pval[,t] <- 2*pnorm(-sqrt(t)*abs(dta.cummean[,t]))
}
```

```
plot(0,0, type="n",
  xlim=c(1,N), ylim=c(0,7),
  xlab="n", ylab="-log P-value")

for (i in 1:num.traj) {
  points(1:N, -log(dta.pval[i,]), type="l", col=c("red","blue")[1+M.tr[i]])
}

abline(h=-log(c(0.002, .01, .05)), lty=3, lwd=2)
```



Clearly a lot of the trajectories have sojourns into the realm of very small P-values. More pointedly, a lot of the *red* trajectories spend some time here, even though they are spawned in an absence of an effect. This comes as no surprise to those aware of *multiple comparison* challenges with frequentist hypothesis testing. The interesting contrast for our purposes is

that we *did not* commensurately see red trajectories with the posterior probability of an effect close to one.

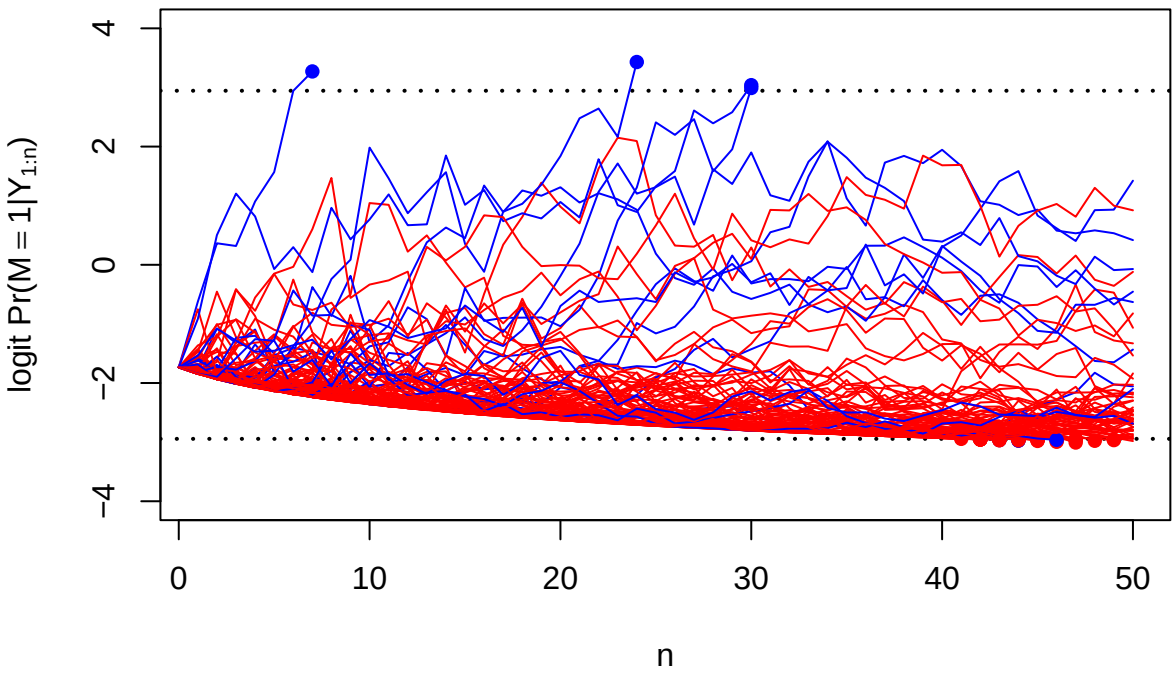
Optional Stopping

When data are collected sequentially in time, there are strong motivations to repeatedly update the analysis of the data after every new datapoint is acquired (or perhaps after every few are acquired). If one finds that the first n datapoints suffice to meet the inferential needs of the study, then resources can be conserved by not proceeding on to acquire the $(n+1)$-st datapoint. Additionally, there can be an ethical compulsion to stop. In a randomized trial comparing two medical interventions A and B, should the data from the first n patients provide strong evidence that one of the treatments is more efficacious than the other, then it would be very dubious to do any further randomization of patients. Rather, all subsequent patients could be offered the "winning" intervention.

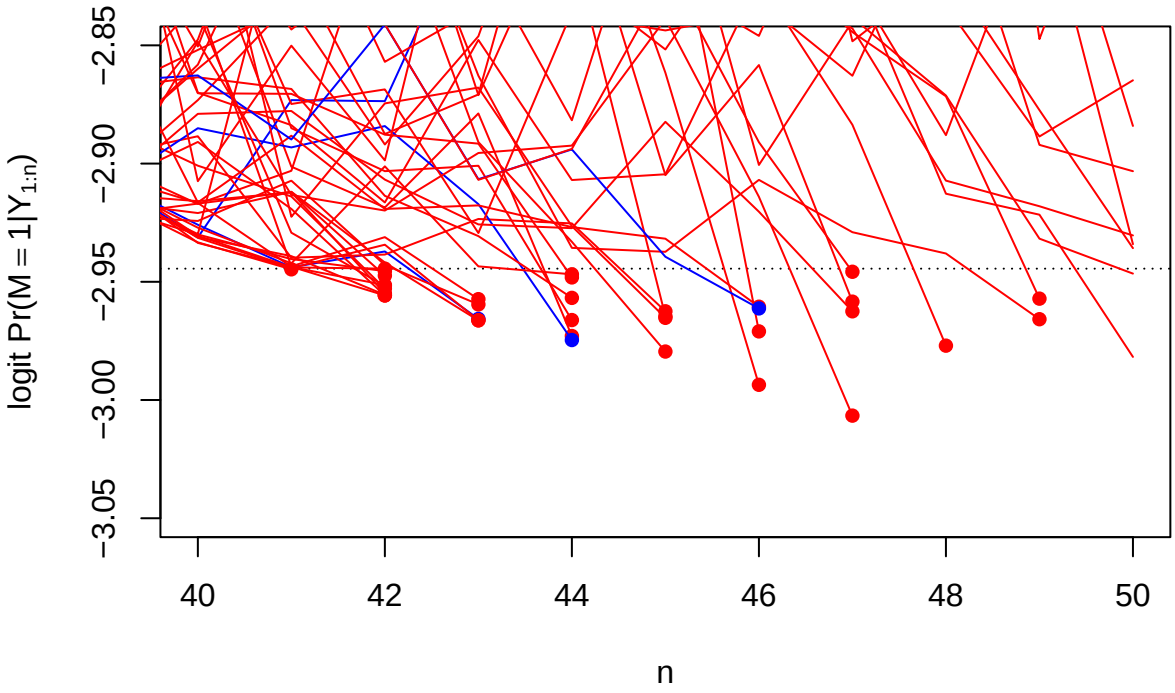
Above we examined the trajectories of $p_n = Pr(M = 1|Y_{1:n})$ where all the datastreams were acquired all the way to $n = 50$. What if, instead, we stopped data acquisition early if a stream reached either $p_n < 0.05$ or $p_n > 0.95$, before $n = 50$. Notably, in recent times some scientific journals, particularly in psychology, have recommended, or even required, this form of *stopping rule." We revisit the 100 simulated trajectories above, to see which would have stopped early.

```
### here the full trajectories (N=50) have already been simulated
### so we are looking back to see which would have stopped early, when

thrshld <- logit(0.95)

last.obs <- rep(N, num.traj); last.postprob <- dta.postprob[,N]

for (i in 1:num.traj) {
  if (max(abs(logit(dta.postprob[i,])))> thrshld) {
    last.obs[i] <- min((1:N)[abs(logit(dta.postprob[i,])) > thrshld])
    last.postprob[i] <- dta.postprob[i,last.obs[i]]
  }
}
```

The resulting milieu of p_n trajectories look as follows.



And since it's a bit crowded, we also present a enlargement of the lower-right corner of the plot.



Overall, we see 41 of the 100 trajectories stopping early. The 4 that stop early because the posterior probability of an effect is high are indeed all generated with an effect present. While all but 3 of the 37 that stop early due to a low posterior probabilty of an effect are indeed generated in the absence of an effect.

Next we simulate a much larger number of trajectories, to better understand the operating characteristics of our procedure.

```
num.traj <- 5000

### in contrast to above, now we only simulate data until
### we stop (possibly early)

M.tr <- rbinom(num.traj, size=1, prob=hyp.nature$p)
theta.tr <- rep(0, num.traj)
theta.tr[M.tr==1] <- rnorm(sum(M.tr), mean=hyp.nature$mu, sd=hyp.nature$tau)

dta <- matrix(NA, num.traj, N)
dta.postprob <- rep(NA, num.traj)
dta.early <- rep(FALSE, num.traj)

for (i in 1:num.traj) {
```

```
  hyp <- hyp.nature

  j <- 1
  while (j <= N) {

    ### stop before j-th datapoint acquired?
    if (abs(logit(hyp$p)) > thrshld) {
      dta.early[i] <- TRUE
      break
    }

    ### acquire j-th datapoint
    dta[i,j] <- rnorm(1, mean=theta.tr[i], sd=1)
    hyp <- updater(hyp, dta[i,j])
    j <- j+1
  }

  dta.postprob[i] <- hyp$p
}
```

To frame this more as an inference problem, think of inferring absence of an effect if the terminal posterior probability of $M = 1$ is below 0.05, inferring presence if this probability exceeds 0.95, but otherwise calling the datastream inconclusive. With this lens, the following cross-classification of truth and inference arises for our 5000 simulated datastreams.

```
smry <- data.frame(
  truth=factor(c("M=0","M=1")[1+M.tr]),
  inference=cut(dta.postprob, breaks=c(0,0.05,0.95,1)),
  early=factor(c("full","early")[1+dta.early])
)
levels(smry$inference) <- c("infer M=0", "inconclusive", "infer M=1")
xtabs(~truth + inference, data=smry)
```

```
##       inference
## truth infer M=0 inconclusive infer M=1
##   M=0      1979         2241         5
##   M=1       110          406       259
```

Roughly, we have framed the stopping and inference procedures here to match: the inconclusive datastreams are those that do not stop early. As truth in advertising, however, and as per the above visualization, a small number of trajectories will cross the threshold *on* the acquisition of the 50-th datapoint. This happens to be the case for 108 out of the 2089 inferences in favour of $M = 0$, and 2 out of the 264 inferences in favour of $M = 1$.

Too much meddling?

Statisticians are well attuned to potential biases that can arise if we use the data already in hand to inform whether or not to collect more data, but then present inferences as if the amount of data to collect had been pre-specified. For instance, and harkening back to the earlier plot of P-value trajectories, what would happen if we stop data acquisition early if (and only if) the P-value falls below a threshold of say $P = 0.01$. The earlier plot suggests a fair proportion of the trajectories generated in absence of an effect would, in fact, stop early. Indeed a quick simulation reveals this proportion to be about 8.6%. Using this stopping rule, but then declaring an early-stopped datastream to contain $P = 0.01$ strength of evidence in favor of an effect, would be quite misleading.

Since a stopping rule based on P-values leads to this interpretive peril, it is reasonable to wonder about the analogous story if stopping is based on posterior probabilities. To investigate, we appeal first to a general sense in which Bayesian procedures are *calibrated*. As is stressed in earlier vignettes in this series, and in Chapters 2 and 3 of *Bayesian Statistical Inference: Concepts, Hallmarks, and Operating Characteristics*, by their very construction Bayesian inference procedures have an inherent calibration to them. If we generically write D to represent the observable data, then, with respect to the *ensemble of parameter-data pairs*, it is the case that

$$Pr\left\{M = 1 | Pr(M = 1|D) = p\right\} = p. \tag{1}$$

That is, amongst the subset of pairs for which D produces $Pr(M = 1|D) = p$, the proportion actually spawned when $M = 1$ is indeed p.

As can happen with equations in statistics, one could debate whether equation (1) is simple or profound, and whether a proof is required. If helpful, we offer this:

$$\begin{aligned} Pr\{M = 1|Pr(M = 1|D) = p\} &= E\left\{Pr(M = 1|D, Pr(M = 1|D) = p)|Pr(M = 1|D) = p\right\} \\ &= E\left\{Pr(M = 1|D)|Pr(M = 1|D) = p\right\} \\ &= p. \end{aligned}$$

Here the second equality follows because once we condition on the all the data D, further conditioning on some function of D is irrelevant.

In the above, we have generically written D to represent "the observed data.'' However, when early-stopping is contemplated, we should be careful about what actually comprises the data. Let K_n be the set of all $y_{1:n}$ for which our algorithm dictates that we collect all of $y_{1:n}$ and proceed to collect y_{n+1}. Informally, this is the "keep going" subset of possible values of the first n observations. Now say we find ourselves having stopped (early), say after the collection of 7 datapoints. A complete description of what we have observed is: $Y_{1:6} \in K_6$, *and* $Y_{1:7} \notin K_7$, *and* $Y_{1:7} = y_{1:7}$. The appropriate evidence concerning M is therefore summarized as $Pr(M = 1|Y_{1:6} \in K_6, Y_{1:7} \notin K_7, Y_{1:7} = y_{1:7})$.

In general, however, for any sequence $y^*_{1:n}$ which satisfies $y^*_{1:(n-1)} \in K_{n-1}$ and $y^*_{1:n} \notin K_n$, we have

$$Pr(M = 1|Y_{1:(n-1)} \in K_{n-1}, Y_{1:n} \notin K_n, Y_{1:n} = y^*_{1:n}) = Pr(M = 1|Y_{1:n} = y^*_{1:n}), \tag{2}$$

simply because $Y_{1:n} = y^*_{1:n}$ *implies* that $Y_{1:(n-1)} \in K_{n-1}$ and $Y_{1:n} \notin K_n$.

Equation (2) is extremely liberating. We can use a stopping rule, such that the amount of data to be collected is *a priori* unknown. If you like, the sample size N is *a priori* random. However, when the data signal us to stop at $N = n$ (because $Y_{1:(n-1)} \in K_{n-1}$ but $Y_{1:n} \notin K_n$), then we can compute the posterior probability that $M = 1$ *as if* a sample size of n had been pre-specified.

To see the calibration described by equation (1) in action, we return to our 5000 simulated trajectories, and categorize them into 20 bins of width 0.05 (from 0 to 1) on the $Pr(M = 1|D)$ scale.‘ We then simply inspect and visualize the within-bin proportions of $M = 1$.

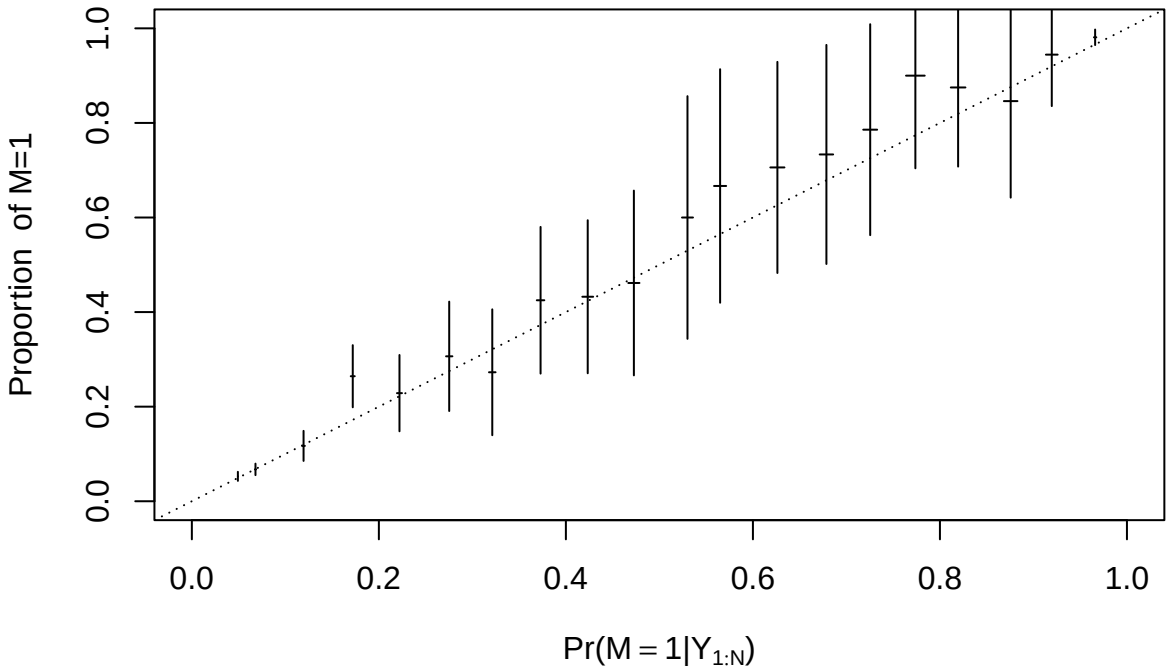
```
smry2 <- data.frame(
  M.tr=M.tr,
  postprob=dta.postprob,
  bin=cut(dta.postprob, breaks=seq(from=0, to=1, by=0.05))
)
```

```
### take a look at the size and composition of some of the bins
tmp <- aggregate(M.tr~bin, data=smry2,
  FUN=function(x) {c("  SIZE"=length(x), "FREQ(M=1)"=sum(x))} )
colnames(tmp)[2] <- ""

head(tmp)
```

```
##          bin   SIZE FREQ(M=1)
## 1   (0,0.05]   2089       110
## 2 (0.05,0.1]   1633       110
## 3 (0.1,0.15]    393        46
## 4 (0.15,0.2]    174        46
## 5 (0.2,0.25]    105        24
## 6 (0.25,0.3]     62        19
```

```
tail(tmp)
```

```
##           bin   SIZE FREQ(M=1)
## 15 (0.7,0.75]     14        11
## 16 (0.75,0.8]     10         9
## 17 (0.8,0.85]     16        14
## 18 (0.85,0.9]     13        11
## 19 (0.9,0.95]     18        17
## 20   (0.95,1]    264       259
```



As a small point here, note that our plot aims to reflect simulation uncertainty; the plotted points are within-bin means of $Pr(M = 1|D)$ and $I\{M = 1\}$ respectively, with confidence intervals given to represent uncertainty about these means (relative to their infinite-simulation counterparts). Some of the mid-range bins (in terms of $Pr(M = 1|D)$) contain few trajectories, so this uncertainty is substantial. Nonetheless, the results are completely commensurate with the claimed identity relationsip of (1).

To round out the story, we repeat the visualization, this time with the bins being vigntiles of p_N for the 5000 realized trajectories.

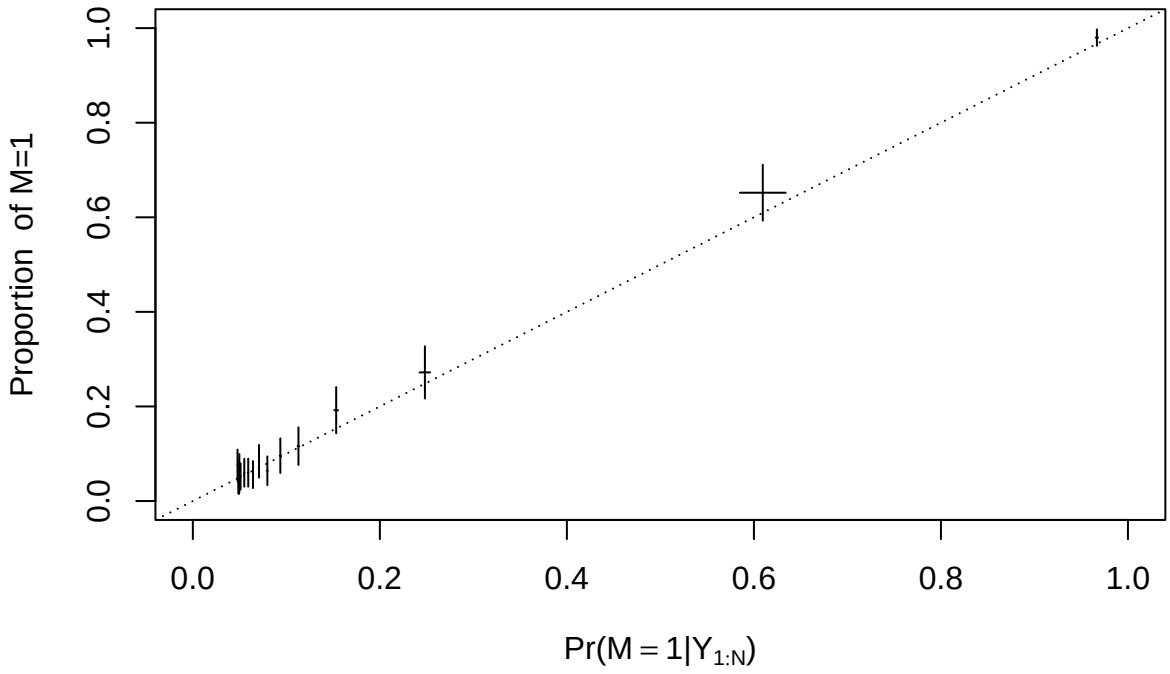
```
smry3 <- data.frame(
  M.tr=M.tr,
  postprob=dta.postprob,
  bin=cut(dta.postprob,
    breaks=c(0, quantile(dta.postprob, seq(from=0.05, to=0.95, by=0.05)), 1))
)

### take a look at the size and composition of some of the bins
tmp <- aggregate(M.tr~bin, data=smry3,
  FUN=function(x) {c("  SIZE"=length(x), "FREQ(M=1)"=sum(x))} )
colnames(tmp)[2] <- ""
```

```r
head(tmp)
```

```
##                  bin   SIZE FREQ(M=1)
## 1       (0,0.04841]    250        19
## 2 (0.04841,0.04894]    250        10
## 3 (0.04894,0.04917]    250        14
## 4 (0.04917,0.04946]    250        10
## 5 (0.04946,0.04953]    250        10
## 6 (0.04953,0.04967]    250        11
```

```r
tail(tmp)
```

```
##                  bin   SIZE FREQ(M=1)
## 15 (0.08527,0.1017]    250        24
## 16  (0.1017,0.1269]    250        29
## 17  (0.1269,0.1856]    250        48
## 18  (0.1856,0.3498]    250        68
## 19  (0.3498,0.9518]    250       163
## 20       (0.9518,1]    250       245
```



Now the simulation uncertainty is manifested by the mid-range bins being very wide, however compatibility with the identity relationship (1) is still evident.

This demonstration has only scratched the surface of what happens if we analyze data "in real time," and use the present analysis to decide on whether or not to continue with data accrual. Based on what we have seen, it might be tempting to conclude that we can't make such schemes work using "classical" statistical techniques, but can trivially do so with Bayesian techniques. There are shards of truth in this conclusion, but also a ton of missing nuance. A deeper dive on the topic is provided in Chapter 9 of *Bayesian Statistical Inference: Concepts, Hallmarks, and Operating Characteristics.*