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

Paul Gustafson

2024-02-16

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

The beginning

Consider a statistical problem involving three binary variables, (X_1, X_2, Y). The true "state of the world," or "state of nature," is the joint distribution on this triplet. We characterize this with *parameters* (p, q) defined as $p_{ij} = Pr(X_1 = i, X_2 = j)$, and $q_{ij} = Pr(Y = 1|X_1 = i, X_2 = j)$. In real, applied work we regard p and q as *fixed but unknown* population quantities we wish to estimate using appropriate data.

Typically, some parameters are of more scientific interest than others. For instance, say that in a biostatistical context Y is an outcome variable (coded as zero/one for better/worse), X_1 is an exposure variable, and X_2 is a confounding variable. And say the goal is to infer

$$\begin{aligned}\psi &= Pr(Y = 1|X_1 = 1, X_2 = 0) - Pr(Y = 1|X_1 = 0, X_2 = 0) \\ &= q_{10} - q_{00},\end{aligned}$$

the risk difference for the exposure-disease relationship when the confounder takes on the first of its two levels.

Th real-world problem is to estimate the (fixed but unknown) ψ using data. However, to study the *operating characteristics* of estimation procedures, we invoke some artistic license concerning how the true values of (p, q) (and consequently ψ) came to be. More precisely, we pretend that Mother Nature (henceforth just Nature) uses a random draw from *all possible* truths to arrive at *the* truth. And we term the distribution over (p, q) which governs this draw to be *Nature's prior distribution* (or sometimes just *Nature's prior*, for brevity).

Scenario A

To fix a scenario, say Nature's prior is encoded as:

```
truth.A <- 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)
}
```

Here Nature sets the bivariate distribution of the exposure and the confounder by drawing $p \sim \text{Dir}(1, 1, 1, 1)$, while the outcome prevalences arise as independent draws, $q_{ij} \sim \text{Unif}(0, 1)$. We can generate, and examine, an instance of this:

```
truth.A.instance <- truth.A()

truth.A.instance
```

```
## $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
```

Next we emulate how Nature subsequently generates the observable data, given the underlying state of the world:

```
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)
}
```

While the code does this in an aggregate way, we can equally well think that for each of $n = 250$ study subjects, first X is generated to be one of $\{(0, 0), (0, 1), (1, 0), (1, 1)\}$ (with corresponding chances p). Then, given X, the outcome $Y = 1$ arises with probability q_X.

We instantiate this for the present truth:

```
data.A.instance <- datagen(truth.A.instance)
```

The resulting dataset is characterized by the following frequencies for Y given X:

```
kable(cbind( c(0,1,0,1), c(0,0,1,1),
              data.A.instance$x.ct-data.A.instance$y.ct,
              data.A.instance$y.ct),
      col.names=c("$X_1$","$X_2$","$Y=0$","$Y=1$"))
```

$X_1$	$X_2$	$Y = 0$	$Y = 1$
0	0	27	73
1	0	24	4
0	1	53	54
1	1	7	8

We want to use this dataset (or indeed whatever dataset is presented to us) to infer, as best we can, the state of the world. The following function takes a dataset and a prior distribution as input, providing a summary of the *posterior distribution* on the target parameter as output:

```
posterior.gen <- function(dta, hyp.a0, hyp.a1, m=10000) {

  # posterior distribution of q
  # represented as m iid draws (rows)
  q.pst <- t(replicate(m,
              rbeta(4, shape1=hyp.a0 + dta$y.ct,
                       shape2=hyp.a1 +dta$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.95=quantile(q.pst[,2]-q.pst[,1], c(0.025,0.975)))
}
```

The math under the hood here is, fortunately, straightforward. For each (i, j) we assign a beta(a_{ij0}, a_{ij1}) prior distribution to the outcome prevalence q_{ij}. Then the posterior distribution for q has four independent components, each of the form $(q_{ij}|\text{data}) \sim \text{beta}(a_{ij0} + d_{ij0}, a_{ij1} + d_{ij1})$, where d_{ijk} is the frequency of $(X_1 = i, X_2 = j, Y = k)$ in the data (as seen in the data table above). Broadly, a situation like this, where the prior and posterior distributions are different members of the same tractable family of distributions, is referred to as involving *conjugate updating.* In fact there are cleaner ways to proceed when conjugate updating applies, but for the sake of future generality, note that we numerically represent the posterior distribution of q by generating a large number (m) of draws from this distribution. The hope and intent is that m is sufficiently large such that the numerical error in the output is inconsequential. (As a thought experiment here, the difference between "knowing" a distribution versus having m independent observations drawn from it vanishes, as m goes to infinity.)

With the posterior distribution being a consequence of the prior distribution and the data, some comment on the former is warranted. We have already use the language of "a prior" in the useful, but fictional, sense of how Nature chooses the state of the world. Now we have a more tangible situation. The person conducting the analysis of an actual dataset must specify an actual distribution over the unknown parameters. To keep things straight, we refer to this as the *investigator's* prior distribution, manifested here as a specific choice of a_{ijk}'s. In common Bayesian parlance, a is a set of *hyperparameters* to be user-specified (in contrast to the unknown *parameters* we seek to estimate).

Moving ahead, consider the specification $a_{ijk} = 1$ (for all i, j, k). More intuitively, this prescribes a Unif(0,1) prior to each of the four $(Y|X)$ outcome prevalences. It also happens to correspond to what we used for Nature's prior in our unfolding Scenario A. To make the ensuing code easier to follow, we create a function to carry out Bayesian inference *for this specific choice of the investigator's prior*:

```r
posterior.A <- function(dta) {
  posterior.gen(dta, hyp.a0=rep(1,4), hyp.a1=rep(1,4))
}
```

Now we are ready to perform, and examine, Bayesian inference for the dataset tabulated above:

```r
inference.A.instance <- posterior.A(data.A.instance)

inference.A.instance
```

```
## $mn
## [1] -0.557
##
## $sd
## [1] 0.0806
##
## $cred.95
##   2.5%  97.5%
## -0.700 -0.381
```

We can interpret the various summaries of the posterior distribution of ψ. The *posterior mean* of the target parameter, $\hat{\psi} = E(\psi|\text{data}) = -0.557$ is a best-guess for ψ (i.e., a point estimate in common statistical parlance). As a sanity check, $\hat{\psi}$ is close to the "intuitive" estimate of ψ based on sample proportions, namely $(4/28) - (73/100) = -0.587$.

The *posterior standard deviation*, $SD(\psi|\text{data}) = 0.081$, reflects the level of uncertainty in this guess. And to further describe uncertainty, the 95% credible interval running from -0.7 to -0.381 is a range of values we think likely to contain the true value. By definition, this interval contains the middle 95% of the posterior probability, and is technically regarded as an "equal-tailed" credible interval.

Into the Sandbox

When used in a real scientific problem, much of the story ends here. We can report out these (or other, or additional) posterior summaries to describe our knowledge about the truth, having seen the data. But we are in a made-up sandbox to test things in, so we can compare our inference to the truth that produced the data. For instance, we see an estimation error of $\hat{\psi} - \psi = -0.557 - -0.543 = -0.014$. Note that this magnitude of estimation error is compatible with the posterior standard deviation reported above.

Having examined performance of Bayesian inference in a single-use, we now turn attention to the average case. We simply repeat the above steps a large number (say 500) of times. This yields an ensemble of 500 possible true states of the world, each accompanied by a dataset, and then subsequently accompanied by an inference:

```
truth.ensemble.A <- replicate(n=1000, truth.A(), simplify=F)
```

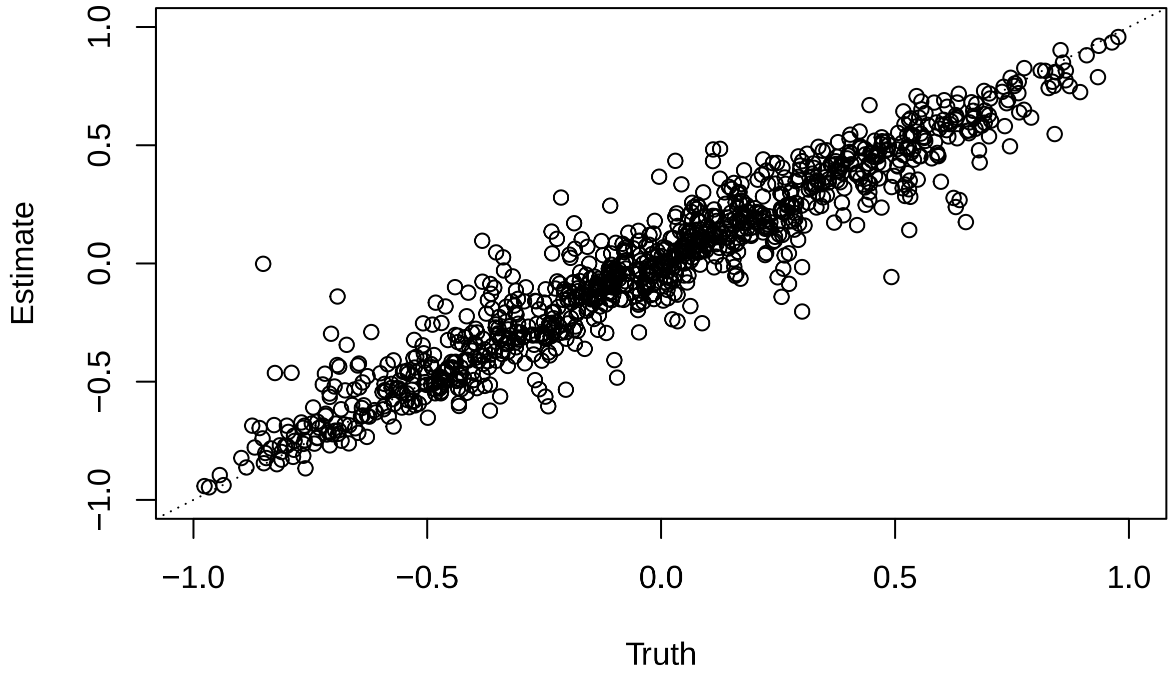
```
data.ensemble.A <- lapply(truth.ensemble.A, datagen)
```

```
inference.ensemble.AA <- lapply(data.ensemble.A, posterior.A)
```

Here we make a quick mention of variable names in the code. The ".A" suffix is deliberately appended for the ensemble of truths, simply because later we will compare this scenario (A) with some others (B, C, D). This suffix naturally carries over to the ensemble of datasets, since each dataset was generated under one of these truths. It might seem odd then that the double ".AA" is used to label the ensemble of inferences. This is deliberate, however. We have really used the scenario A settings *twice* in forming the collection of inferences. They are used once in terms of how the ensemble of truths is generated (which thereby influences how the datasets are are generated). Then they are used *again* in terms of the investigator's prior distribution of q that is specified each and every time we form the posterior distribution based on one of the datasets. We return to this point presently.

To examine our ensembles, we view the joint distribution of truth ψ and estimate $\hat{\psi}$, as encapsulated in our 500 draws of $(\psi, \hat{\psi})$ pairs:

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

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

plot(Truth, Estimate, xlim=c(-1,1),ylim=c(-1,1))
abline(c(0,1), lty=3)
```



This reassures, in that the two entities are quite strongly, and positively, correlated. We also note (but than leave hanging for now) the following. We can, by eye, discern a tendency of more under-estimation of ψ when ψ is large (and commensurately more over-estimation when ψ is small).

To more formally speak about average-case performance for $\hat{\psi}$ estimating ψ, a first thought is to average the ensemble of squared estimation errors. We do this, looking at the square-root of the average squared error (RASE) for interpretability:

```
RASE.AA <- sqrt(mean((Estimate-Truth)^2))

RASE.AA
```

```
## [1] 0.126
```

It's a bit hard to contextualize whether this is "good" or "bad" average-case performance. But certainly 0.126 is a typical magnitude of estimation error, $|\hat{\psi} - \psi|$, when Nature's prior and the investigator's prior are as specified above, and the available data are 250 *iid* observations of (X, Y).

Scenario B

Now we move to average-case performance in a made-up world B that differs from made-up world A. In Scenario B, nature is more likely to commit to smaller values of $Pr(Y = 1|X_1, X_2)$. Or, more colloquially, Scenario B is marked by bad outcomes being rarer. Particularly, we swap out our earlier "truth generator" in favor of:

```
truth.B <- function() {
  p <- matrix(rdirichlet(1, rep(1,4)), 2, 2)
  q <- matrix(rbeta(4, shape1=1, shape2=4), 2, 2)

  trgt <- q[2]-q[1]

  list(p=p, q=q, trgt=trgt)
}
```

With nature now generating each of the four conditional outcome prevalences from a beta$(1, 4)$ distribution (which has a mean of 0.2) we will indeed tend to encounter lower prevalences than in Scenario A.

In line with what we did previously in Scenario A, for the sake of readability we again specialize our inference algorithm down to the investigator's prior being that of scenario B:

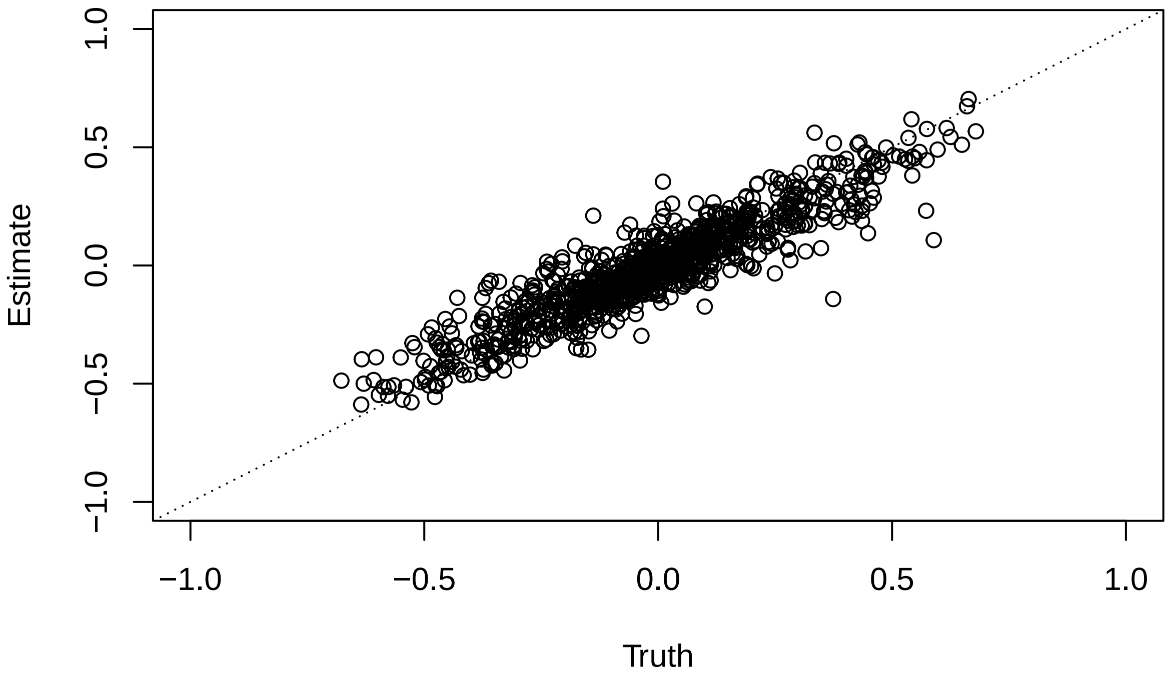
```
posterior.B <- function(dta) {
  posterior.gen(dta, hyp.a0=rep(1,4), hyp.a1=rep(4,4))
}
```

Now we can make ensembles, using the same workflow as before:

```
truth.ensemble.B <- replicate(n=1000, truth.B(), simplify=F)

data.ensemble.B <- lapply(truth.ensemble.B, datagen)

inference.ensemble.BB <- lapply(data.ensemble.B, posterior.B)
```

And again we can both see, and quantify, the estimation error incurred across this ensemble of possible truths:

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

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

plot(Truth, Estimate, xlim=c(-1,1),ylim=c(-1,1))
abline(c(0,1), lty=3)
```



```
RASE.BB <- sqrt(mean((Estimate-Truth)^2))

RASE.BB
```

```
## [1] 0.0936
```

We see the RASE for Scenario B is 25% smaller than seen in Scenario A. Very roughly, this makes intuitive sense. If the investigator knows more about the situations in which data will be analyzed, e.g., knows that lower outcome prevalences are likely, versus having no idea about outcome prevalences, then we expect the investigator to do better, on average.

More Scenarios (C and D)

The variations on the theme above are endless. We could quantify average-case estimation performance based on any distribution on q we so desire. For this demo we content ourselves with two further specifications, both of which are instances of the following joint distribution on q specified by this density function, up to a constant of proportionality:

$$\begin{aligned} f(q_{00}, q_{01}, q_{10}, q_{11}) \propto \quad & q_{00}^{a_{000}-1}(1-q_{00})^{a_{001}-1} \times \\ & q_{10}^{a_{100}-1}(1-q_{10})^{a_{101}-1} \times \\ & q_{01}^{a_{010}-1}(1-q_{01})^{a_{011}-1} \times \\ & q_{11}^{a_{110}-1}(1-q_{11})^{a_{111}-1} \times \\ & \exp\{-\lambda(q_{00}+q_{11}-q_{01}-q_{10})^2\}. \end{aligned}$$

If we specify $\lambda = 0$, this simplifies to independent beta distributions for the four prevalences. (For instance, $a_{\cdot\cdot 0} = (1, 1, 1, 1)$, $a_{\cdot\cdot 1} = (1, 1, 1, 1)$ yields Scenario A, while $a_{\cdot\cdot 0} = (1, 1, 1, 1)$, $a_{\cdot\cdot 1} = (4, 4, 4, 4)$ results in Scenario B.) However, specifying a positive value for λ induces a dependence amongst the four outcome prevalences. The nature of this dependence is to promote sets of prevalences that involve *less interaction*, on a risk difference scale. There is a rabbit hole to go down here in terms of interaction on one scale versus another, but we refrain. We simply note that with $\lambda > 0$ this distribution encourages $Pr(Y = 1|X_1 = 1, X_2) - Pr(Y = 1|X_1 = 0, X_2)$ to vary *less* across the two levels of X_2. (Or symnmetrically, it encourages the risk difference for X_2 to vary less across the two levels of X_1.)

There is another rabbit hole to go down in terms of how to implement Monte Carlo draws from this distribution. We mostly refrain, except to say that (i), *rejection sampling* can do the job, and (ii), this short function implements rejection sampling for this family of distributions:

```r
### sample a single draw from joint distribution of four probabilities
### arising by penalizing interaction of the risk-difference scale

rbeta4.pen <- function(shp1, shp2, lambda) {

  flg <- F
  while (!flg) {
    drw <- rbeta(4, shape1=shp1, shape2=shp2)
    flg <- runif(1) < exp(-lambda*(drw[1]+drw[4]-drw[2]-drw[3])^2)
  }
  drw
}
```

We can use this to instantiate Scenario C:

```r
truth.C <- function() {
  p <- rdirichlet(1, rep(1,4))
  q <- rbeta4.pen(shp1=rep(1,4), shp2=rep(1,4), lambda=2.5)

  trgt <- q[2]-q[1]

  list(p=p, q=q, trgt=trgt)
}
```

And also Scenario D:

```r
truth.D <- function() {
  p <- rdirichlet(1, rep(1,4))
  q <- rbeta4.pen(shp1=rep(1,4), shp2=rep(4,4), lambda=2.5)

  trgt <- q[2]-q[1]

  list(p=p, q=q, trgt=trgt)
}
```

Crudely then, Scenario C modifies Scenario A by adding penalization of interaction, and Scenario D modifies Scenario B in exactly the same way.

We wave our hands a tad, claiming that we have conjugate updating when we analyze data using the "penalized beta" distribution above as the investigator's prior. The hyperparameter λ will be unchanged, i.e., the same for the posterior as for the prior. But the data will "move" the other hyperparameters, according to $a_{ijk} \to a_{ijk} + d_{ijk}$, just as we saw for the simpler form of prior used in Scenarios A and B.

So generically inference can be realized via:

```r
posterior2.gen <- function(dta, hyp.a0, hyp.a1, lambda, m=10000) {
  # posterior on q
  # represented as m by 4 matrix
  q.pst <- t(replicate(m,
             rbeta4.pen(shp1=hyp.a0 + dta$y.ct,
                        shp2=hyp.a1 + dta$x.ct-dta$y.ct,
                        lambda=lambda)))



  ### posterior summaries
  list(mn=mean(q.pst[,2]-q.pst[,1]),
       sd=sd(q.pst[,2]-q.pst[,1]),
       cred.95=quantile(q.pst[,2]-q.pst[,1], c(0.025,0.975)))
}
```

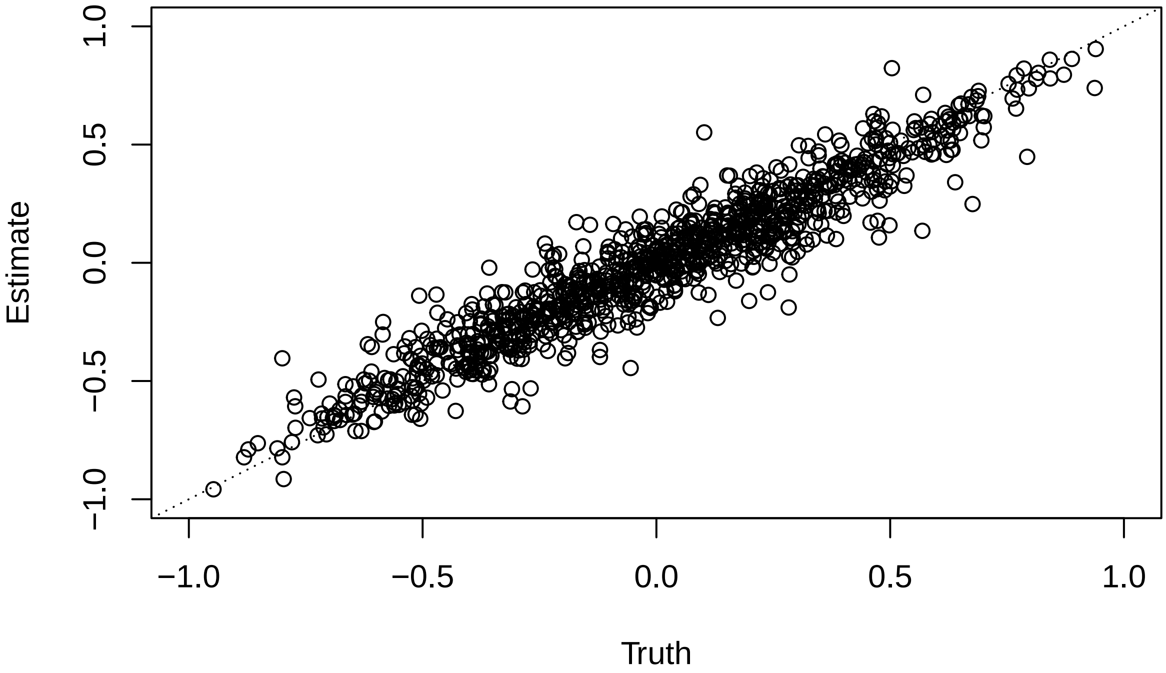
And we can create specialized versions to perform inference when the investigator takes the C or D scenario as the prior:

```
posterior.C <- function(dta) {
  posterior2.gen(dta, hyp.a0=rep(1,4), hyp.a1=rep(1,4), lambda=2.5)
}

posterior.D <- function(dta) {
  posterior2.gen(dta, hyp.a0=rep(1,4), hyp.a1=rep(4,4), lambda=2.5)
}
```

We won't display all the repetitive details of "ensembling" for these further scenarios (but you can see the code for yourself, if you wish). We will, however, show the results.

For Scenario C:

```
plot(Truth, Estimate, xlim=c(-1,1),ylim=c(-1,1))
abline(c(0,1), lty=3)
```



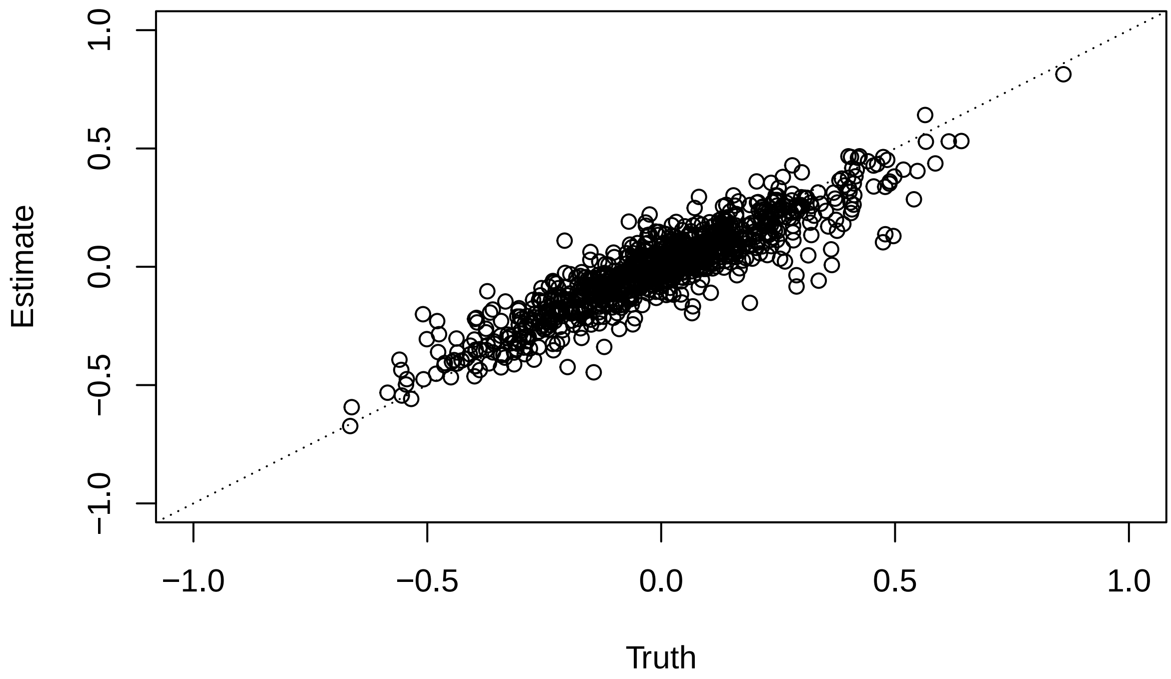
```
RASE.CC <- sqrt(mean((Estimate-Truth)^2))
```

```
RASE.CC
```

```
## [1] 0.114
```

For Scenario D:

```
plot(Truth, Estimate, xlim=c(-1,1),ylim=c(-1,1))
abline(c(0,1), lty=3)
```



```
RASE.DD <- sqrt(mean((Estimate-Truth)^2))
```

```
RASE.DD
```

```
## [1] 0.0864
```

Before moving on, let's take some stock of the situation. Using Scenarios A through D respectively, we see typical magnitudes of estimation error of $0.126, 0.094, 0.114, 0.086$. This makes some intuitive sense. In moving from A to B, or from A to C, we are concentrating the distribution of q, and moreover we get to know the concentrated distribution when we analyze a dataset. Having more information should indeed improve estimation, so it is unsurprising that RASE is reduced. And in thinking about Scenario D, it makes sense that having two pieces of (external-to-data) information — lower outcome prevalences *and* less interaction on the risk-difference scale — would improve estimation compared to having either piece in isolation.

Breaking the link between Nature and investigator

The RASE evaluations above reflect average performance in one sense, but are best possible in another sense. They average estimator performance across all the possible truths that could be the truth, *using the same weighting of these truths as the investigator specifies as an input to the data analysis*. That is, Nature's prior and the investigator's prior agree.

To make this more explicit, let $\hat{\psi}_A(y) = E_A(\psi|Y = y)$ be the posterior mean of ψ when prior A is employed by the investigator, and when data $Y = y$ are observed. Let $\theta = (p, q)$ comprise all the parameters, and note we will write Θ rather than θ when we want to emphasize averaging over different parameter values. Then our first $RASE$ calculation above can be reagarded as an evaluation of:

$$RASE_{A,A} = E_A\left[\{\hat{\psi}_A(Y) - \psi(\Theta)\}^2\right].$$

Note here the A subscript on the expectation reminds us that we are averaging with respect to the joint distribution of (Θ, Y) arising from Θ being marginally distributed according to the Scenario A prior, while $(Y|\Theta)$ is conditionally distributed according to the data-generating mechanism at hand. (The data-generating mechanism is presumed to *not* vary across the four different scenarios.)

A natural and important question is what happens to the average-case performance when the investigator's prior differs from Nature's prior. For instance, how large is

$$RASE_{A,B} = E_A\left[\{\hat{\psi}_B(Y) - \psi(\Theta)\}^2\right].$$

We proceed to find out:

```
inference.ensemble.AB <- lapply(data.ensemble.A, posterior.B)
```

```
Truth <- sapply(truth.ensemble.A, function(z){z$trgt})
Estimate <- sapply(inference.ensemble.AB, function(z){z$mn})

RASE.AB <- sqrt(mean((Estimate-Truth)^2))

RASE.AB
```

```
## [1] 0.16
```

So the typical estimation error across possible truths as weighted by distribution A, when the analysis of each dataset takes distribution B as input, is 0.16. Note here that $RASE_{A,B}$ is inflated compared to $RASE_{A,A}$, by about 27%. Thinking about the qualitative difference between A and B scenarios, this reflects average-case performance across a wide range of possible truths worsening when the investigator forecasts a narrower range of possible truths (here in the form of favoring lower outcome prevalences).

The same workflow above can be applied to any combination of Nature and investigator priors, leading naturally to arranging the 16 RASE values in a matrix (or table, if you prefer). Letting Nature and investigator vary with rows and columns respectively, we see:

```r
row.names(RASE.mtx) <- c("Nature A","Nature B","Nature C","Nature D")
```

```r
kable(RASE.mtx,
        col.names=c("A","B","C","D"))
```

	A	B	C	D
Nature A	0.126	0.160	0.135	0.164
Nature B	0.112	0.094	0.105	0.095
Nature C	0.116	0.149	0.114	0.143
Nature D	0.115	0.087	0.104	0.086

As a sanity check, note we have already encountered the on-diagonal values of this RASE matrix above (and also the (1,2) entry, for that matter).

To whet our appetite for going further in our study of Bayesian inference, we raise four issues.

1. Just looking at the range of the 16 RASE values, there is nearly a factor of two variation. Average-case performance can vary substantially with (i), the actual nature of the scenarios being averaged across, and (ii), what the investigator presumes as the scenarios being averaged across.

2. If we focus on any specific choice of Nature's prior (i.e., pick any row of the matrix), the best performance is seen when the investigator's prior matches that of Nature. More concretely, the row-wise minimums of the RASE matrix all occur on the diagonal. (**Spoiler alert:** this is mathematically guaranteed to happen in *all* problems, rather than just by happenstance in *this* problem.)

3. In contrast to the second point, not all the column-wise minimums of the RASE matrix occur on the diagonal. And related to this, The matrix is far from symmetric. As an example, $RASE_{A,B}$ is substantially larger than $RASE_{B,A}$. It would be overly facile to simply say the average-case performance degrades based on the magnitude of discrepancy between Nature and investigator prior distributions.

4. While this is a pretty minimal working example, moving the investigator's prior from say A to B is emblematic of what might be possible in a biostatistical context, e.g., enough is known *a priori* to be confident that bad outcomes will be quite rare in the upcoming study. Similarly, moving from A to C is a very simple example of presuming some *smoothness* in an unknown function, with the using of smoothing being ubiquitious in statistical work.

These issues are amongst those investigated in Chapter 2 of *Bayesian Statistical Inference: Concepts, Hallmarks, and Operating Characteristics*.