Parameter restrictions for the sake of identification: Example 3

Paul Gustafson

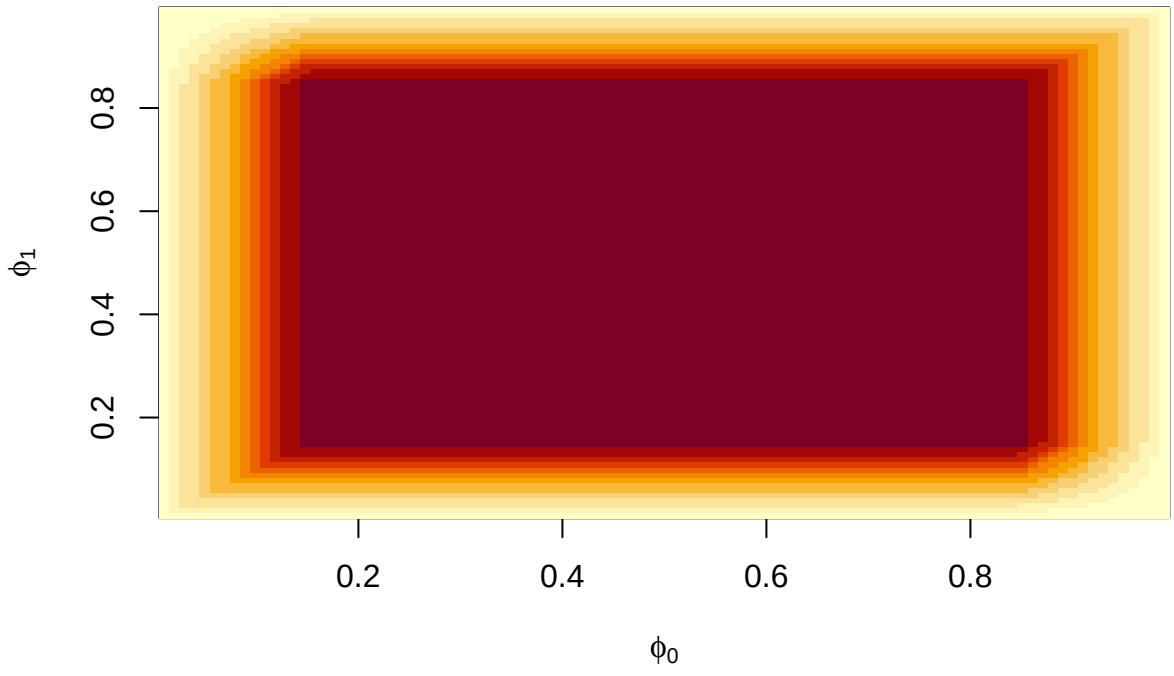
September 30, 2022

```
### marginal prior density for phi under Model 0
margpridens <- function(phi, a=c(.85, .85)) {
  b <- c(max(a[1],1-phi), max(a[2],phi))

  (log(b[1]) + log(b[2]) - log(sum(b)-1)) / ((1-a[1])*(1-a[2]))
}
```

Take a look at this (bivariate) prior density

```
phival <- expand.grid((1:100)/101, (1:100)/101)
densval <- apply(phival,1,margpridens)
image(x=(1:100)/101, y=(1:100)/101, z=matrix(densval, nrow=100),
      xlab=expression(phi[0]),ylab=expression(phi[1]))
```




```
### limiting posterior mean of risk difference under Model 0
### determined by numerical integration
```

```
limpstmn <- function(phi, a=c(.85, .85)) {
  b <- c(max(a[1],1-phi), max(a[2],phi))
  tmp <- expand.grid(sp=b[1]+(1-b[1])*((1:50)/51),
                     sn=b[2]+(1-b[2])*((1:50)/51))
  dens <- 1/((tmp$sn+tmp$sp-1)^2)
  trg <- (phi[2]-phi[1])/(tmp$sn+tmp$sp-1)

  sum(trg*dens)/sum(dens)
}
```

Simulate draws from prior to compute RAMSE

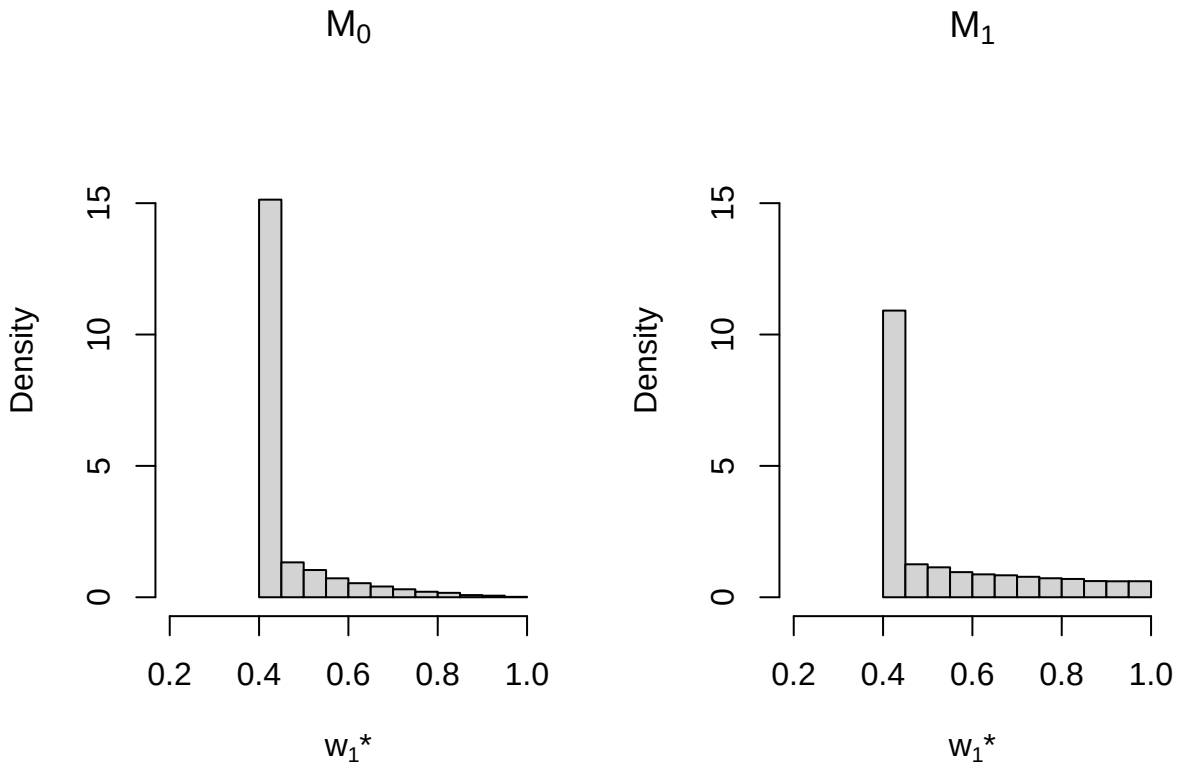
```
NREP <- 20000
r.sim <- cbind(runif(NREP),runif(NREP))
sn.sim <- runif(NREP, .85,1)
sp.sim <- runif(NREP, .85,1)
phi.sim <- cbind((1-sp.sim) + (sn.sim+sp.sim-1)*r.sim[,1],
                 (1-sp.sim) + (sn.sim+sp.sim-1)*r.sim[,2])
trg.sim <- r.sim[,2]-r.sim[,1]

### nature using M0
limbf.0 <- apply(phi.sim, 1, margpridens)
### two possibilities for analyst
limpst.00 <- apply(phi.sim, 1, limpstmn)
limpst.01 <- phi.sim[,2]-phi.sim[,1]

### nature using M1
limbf.1 <- apply(r.sim, 1, margpridens)
### two possibilities for analyst
limpst.10 <- apply(r.sim, 1, limpstmn)
limpst.11 <- trg.sim
```

Limiting posterior weight on M1, for ensemble of param values from M0, ensemble of param values from M1

```
par(mfrow=c(1,2))
hist(1/(1+limbf.0), prob=T,xlim=c(.2,1),ylim=c(0,18), breaks=seq(from=.4,to=1,by=.05),
     xlab=expression(paste(w[1],"*")), main=expression(M[0]))
hist(1/(1+limbf.1), prob=T,xlim=c(.2,1),ylim=c(0,18), breaks=seq(from=.4,to=1,by=.05),
     xlab=expression(paste(w[1],"*")), main=expression(M[1]))
```



Now compute AMSE values

```
amse <- matrix(NA,2,2)

### nature and analyst using M0
tmp <- (limpst.00-trg.sim)^2
amse[1,1] <- mean(tmp)

### nature M0, analyst mix
wt.0 <- 1/(1+limbf.0)  ### limiting weight on M1 (draws from M0)
tmp <- ((1-wt.0)*limpst.00 + wt.0*limpst.01 - trg.sim)^2
amse[1,2] <- mean(tmp)

### nature mix, analyst M0
tmp <- 0.5*(limpst.00-trg.sim)^2 + 0.5*(limpst.10 - trg.sim)^2
amse[2,1] <- mean(tmp)

### nature mix, analyst mix
wt.1 <- 1/(1+limbf.1)  ### limiting weight on M1 (draws from M1)
tmp <- 0.5*((1-wt.0)*limpst.00 + wt.0*limpst.01 - trg.sim)^2 +
       0.5*((1-wt.1)*limpst.10 + wt.1*limpst.11 - trg.sim)^2
amse[2,2] <- mean(tmp)


### ramse
round(sqrt(amse),4)
```

```
##        [,1]   [,2]
## [1,] 0.0270 0.0391
## [2,] 0.0462 0.0347
```

```
### relative increase when nature=M0
(sqrt(amse[1,2])-sqrt(amse[1,1]))/sqrt(amse[1,1])
```

```
## [1] 0.4481209
```

```
### relative decrease3 when nature=MIX
(sqrt(amse[2,1])-sqrt(amse[2,2]))/sqrt(amse[2,1])
```

```
## [1] 0.2487349
```

```
### bootstrap to check stability
bs.rep <- 200
r1.bs  <- r2.bs <- rep(NA,bs.rep)
for (i in 1:bs.rep) {

  smp.bs <- sample(1:NREP, size=NREP, replace=T)

  amse.bs <- matrix(NA,2,2)

  ### nature and analyst using M0
  tmp <- ((limpst.00-trg.sim)[smp.bs])^2
  amse.bs[1,1] <- mean(tmp)

  ### nature M0, analyst mix
  wt.0 <- 1/(1+limbf.0[smp.bs])  ### lmt wht on M1 (points from M0)
  tmp <- ((1-wt.0)*limpst.00[smp.bs] + wt.0*limpst.01[smp.bs] -
            trg.sim[smp.bs])^2
  amse.bs[1,2] <- mean(tmp)

  ### nature mix, analyst M0
  tmp <- 0.5*((limpst.00-trg.sim)[smp.bs])^2 +
         0.5*((limpst.10 - trg.sim)[smp.bs])^2
  amse.bs[2,1] <- mean(tmp)

  ### nature mix, analyst mix
  wt.1 <- 1/(1+limbf.1[smp.bs])  ### lmt wht on M1 (points from M1)
  tmp <- 0.5*((1-wt.0)*limpst.00[smp.bs] + wt.0*limpst.01[smp.bs] -
                trg.sim[smp.bs])^2 +
       0.5*((1-wt.1)*limpst.10[smp.bs] + wt.1*limpst.11[smp.bs] -
              trg.sim[smp.bs])^2
  amse.bs[2,2] <- mean(tmp)

  ### relative increase when nature=M0
  r1.bs[i] <- (sqrt(amse.bs[1,2]) - sqrt(amse.bs[1,1]))/
              sqrt(amse.bs[1,1])

  ### relative decrease when nature=MIX
  r2.bs[i] <- (sqrt(amse.bs[2,1]) - sqrt(amse.bs[2,2]))/
              sqrt(amse.bs[2,1])
}
```

```
summary(r1.bs)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.4246  0.4429  0.4480  0.4480  0.4529  0.4701
```

```
sqrt(var(r1.bs))  ### Monte Carlo SE
```

```
## [1] 0.008267927
```

```
summary(r2.bs)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.2428  0.2468  0.2487  0.2488  0.2508  0.2551
```

```
sqrt(var(r2.bs)) ### Monte Carlo SE
```

```
## [1] 0.002668611
```