

SISG Bayes: Session 9 demo

Ken Rice

2026-05-27

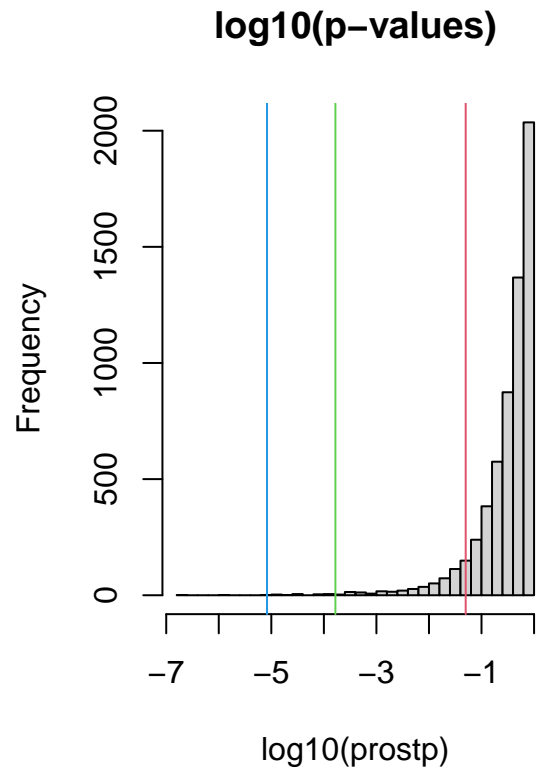
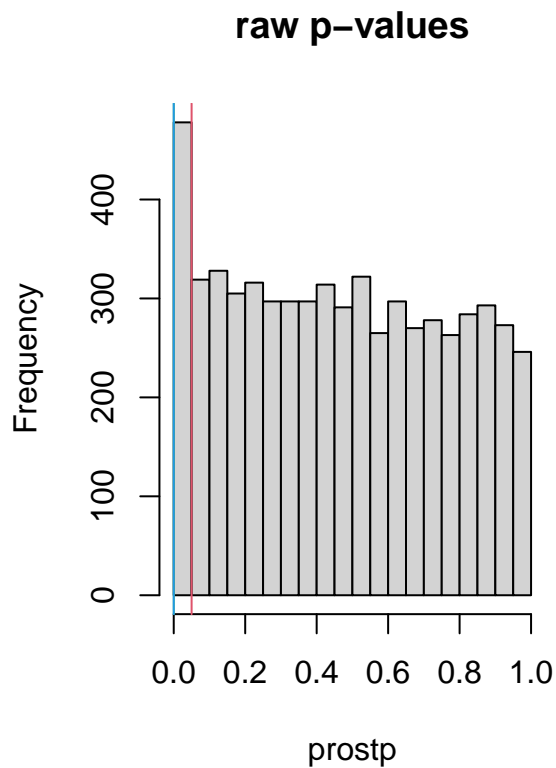
Reading in prostate cancer expression levels: columns 1:50 are controls, 51:102 are cancer cases

```
prost <- read.csv("http://faculty.washington.edu/kenrice/sisgbayes/prostmat.csv")
# plain old p-value
prostp <- apply(prost, 1, function(x){t.test(x[1:50], x[51:102])$p.value})
# linear regression slope of expression levels on case/control status
prostbeta <- apply(prost, 1, function(x){mean(x[1:50]) - mean(x[51:102])})
# estimated variance of the slope estimates
prostv <- apply(prost, 1, function(x){vcov(lm(x~rep(1:0, times=c(50, 52))))[2,2]})
```

P-value based methods

Plots of p-values, and suggested thresholds: (note these will change little if we used weakly informative priors and Bayesian sign-decision methods)

```
par(mfrow=c(1,2))
hist(prostp, n=30, main="raw p-values")
abline(v=c(0.05, 1/6033, 0.05/6033), col=2:4)
hist(log10(prostp), n=30, main="log10(p-values)")
abline(v=log10(c(0.05, 1/6033, 0.05/6033)), col=2:4)
```

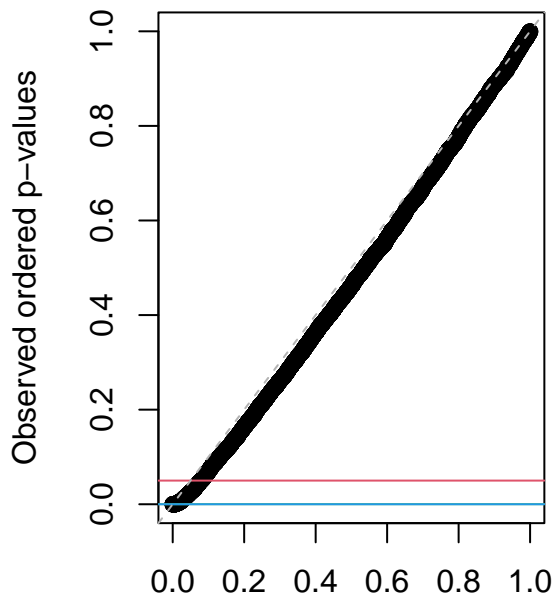


```
table(cut(prostp,c(0,0.05/6033,1/6033, 0.05, 1)))
```

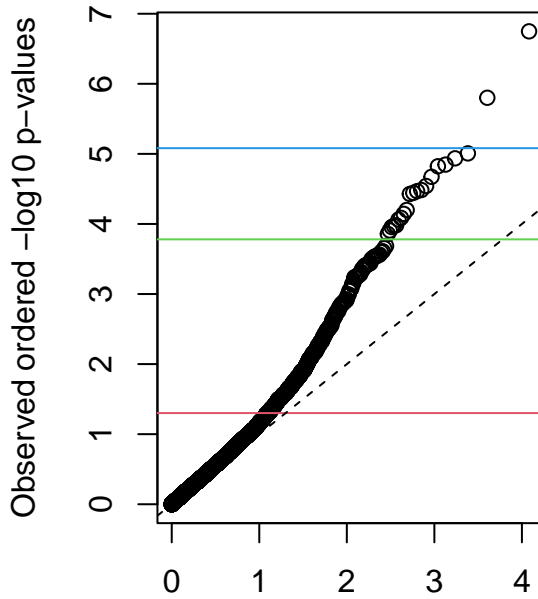
```
##
##      (0,8.29e-06] (8.29e-06,0.000166] (0.000166,0.05] (0.05,1]
##              2              19              457              5555
```

QQ plots compare the quantiles of the observed sample, i.e. the

```
par(mfrow=c(1,2))
plot(y=sort(prostp), x=ppoints(6033), xlab="Ordered p-values, expected", ylab="Observed ordered p-values",
      abline(h=c(0.05, 1/6033, 0.05/6033), col=2:4)
      abline(a=0, b=1, lty=2, col="grey70")
plot(y=-log10(sort(prostp)), x=-log10(ppoints(6033)), xlab="Ordered -log10 p-values, expected", ylab="Observed ordered -log10 p-values",
      abline(h=-log10(c(0.05, 1/6033, 0.05/6033)), col=2:4)
      abline(a=0, b=1, lty=2)
```



Ordered p-values, expected



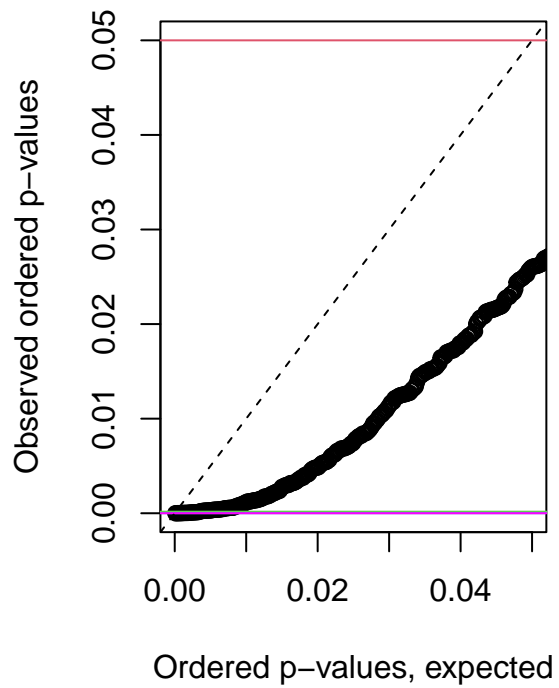
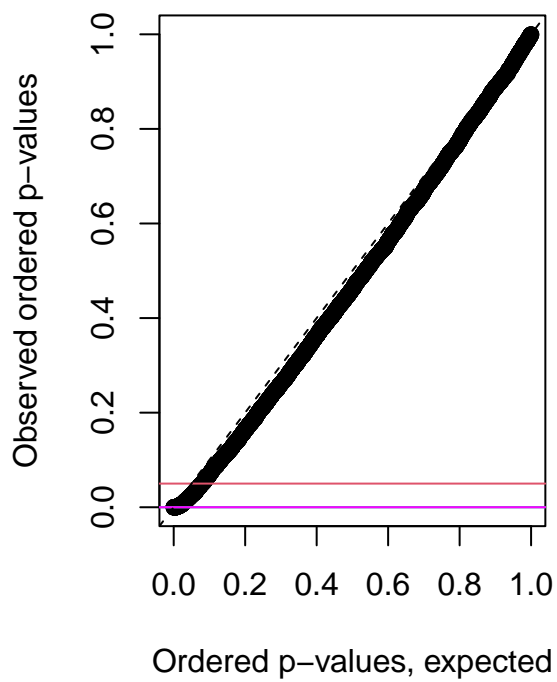
Ordered $-\log_{10}$ p-values, expected

Let's try to illustrate the Benjamini-Hochberg algorithm, that steps up the p-values and rejects until the largest one where

$$p_{[j]} < \alpha j/m$$

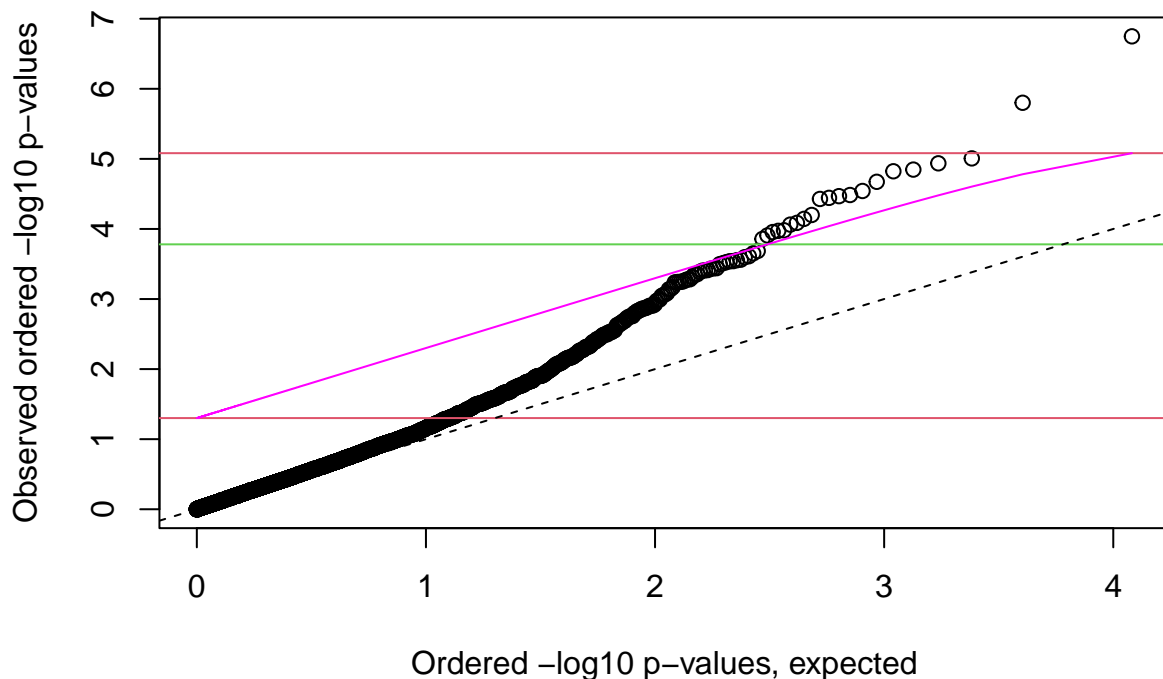
In other words: When do the points go above the magenta line? (It's not easy to tell, even if we zoom into the area of interest)

```
par(mfrow=c(1,2))
plot(y=sort(prostp), x=ppoints(6033), xlab="Ordered p-values, expected", ylab="Observed ordered p-values",
abline(h=c(0.05, 1/6033, 0.05/6033), col=2:4)
abline(a=0, b=0.05/6033, col="magenta")
abline(a=0, b=1, lty=2)
plot(y=sort(prostp), x=ppoints(6033), xlab="Ordered p-values, expected", ylab="Observed ordered p-values",
abline(h=c(0.05, 1/6033, 0.05/6033), col=2:4)
abline(a=0, b=0.05/6033, col="magenta")
abline(a=0, b=1, lty=2)
```



But instead on the log10 scale, when do the points stay below the blue line?

```
plot(y=-log10(sort(prostp)), x=-log10(ppoints(6033)), xlab="Ordered -log10 p-values, expected", ylab="Observed -log10 p-values")
abline(h=-log10(c(0.05, 1/6033, 0.05/6033)), col=2:3)
abline(a=0, b=1, lty=2)
lines(x=-log10(ppoints(6033)), y=-log10(0.05*(1:6033)/6033), col="magenta" )
```



```
max(which(sort(prostp) < 0.05*(1:6033)/6033))
```

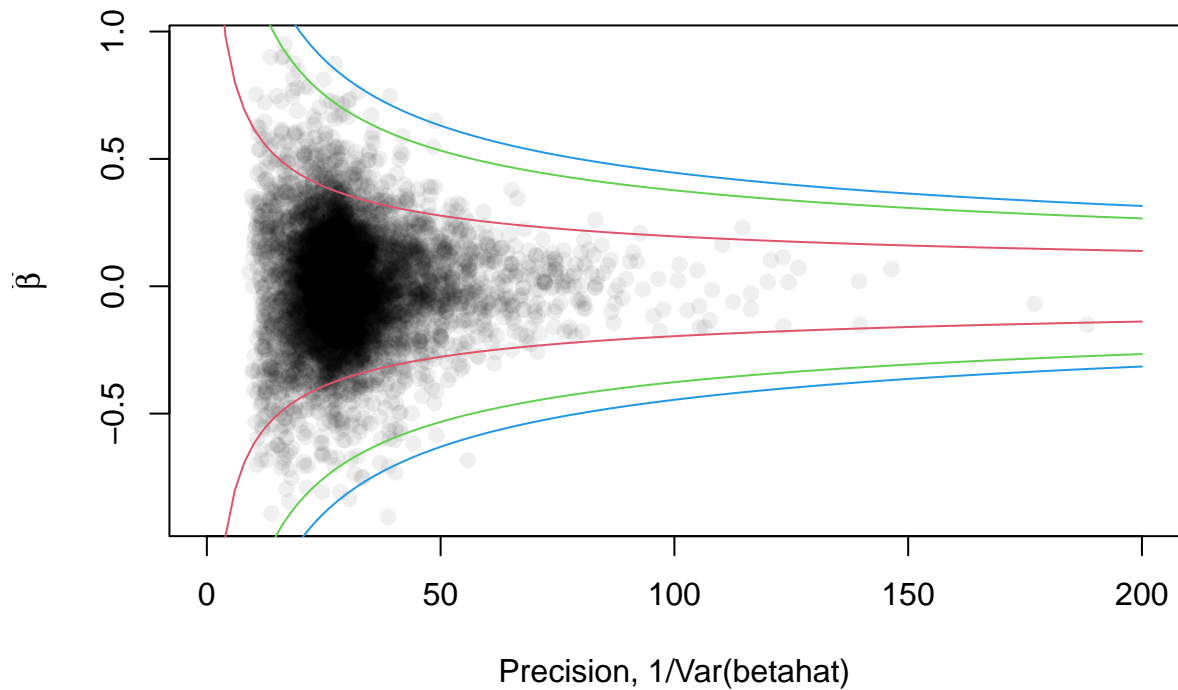
```
## [1] 21
```

BH 'lining up' with 1 EFP is not completely coincidence: we found 21 signals and expect 1 false positive, so our *realized* false discovery rate should be around 5%.

Other methods

First a funnel plot (see also volcano plots of $-\log(p)$ versus $\hat{\beta}$)

```
plot( y=prostbeta, x=1/prostv, xlab="Precision, 1/Var(betahat)", ylab=expression(hat(beta)), pch=19, col="black",
      curve( sqrt(1/x)*qnorm(0.05/2, lower=FALSE), col=2, add=TRUE)
      curve( sqrt(1/x)*qnorm(0.05/2, lower=TRUE ), col=2, add=TRUE)
      curve( sqrt(1/x)*qnorm(1/2/6033, lower=FALSE), col=3, add=TRUE)
      curve( sqrt(1/x)*qnorm(1/2/6033, lower=TRUE ), col=3, add=TRUE)
      curve( sqrt(1/x)*qnorm(0.05/2/6033, lower=FALSE), col=4, add=TRUE)
      curve( sqrt(1/x)*qnorm(0.05/2/6033, lower=TRUE ), col=4, add=TRUE)
```

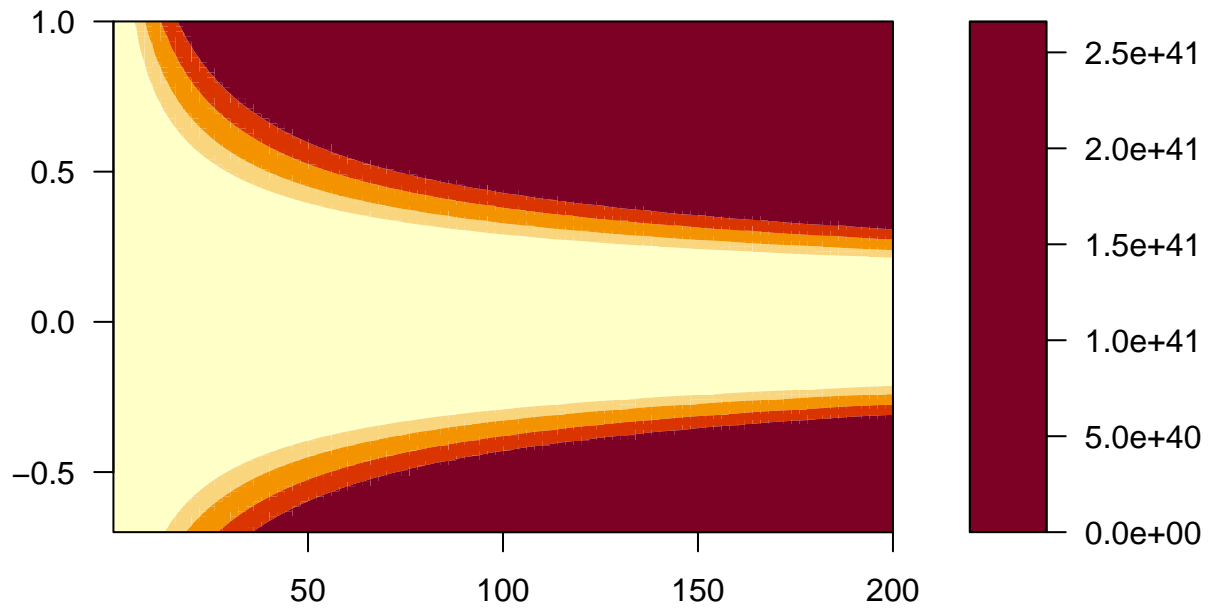


Finally, we'll show the approximate Bayes Factor values, for a few priors. The contours show the thresholds (BF=1, 3.2, 50, 150) where the evidence in favor of the alternative is borderline between Negative, Not worth more than a bare mention, Positive, Strong, and Very strong

```
w <- c(1, 50)
precvals <- seq(0.1,200, length=101)
betavals <- seq(-0.7, 1, length=101)
gg <- expand.grid(prec=precvals, beta=betavals)
gg$se <- with(gg, sqrt(1/prec))
gg$z <- with(gg, beta/se)
gg$v <- with(gg, 1/prec)
gg$bf1 <- with(gg, sqrt((v+w[1])/v) * exp(-z^2/2*w[1]/(w[1]+v)))
gg$bf2 <- with(gg, sqrt((v+w[2])/v) * exp(-z^2/2*w[2]/(w[2]+v)))

filled.contour(x=precvals, y=betavals, z=matrix(1/gg$bf1, 101,101), levels=c(0,1,3.2, 20, 150,max(1/gg$bf2, 1000)),
  xlab="", ylab="", main=paste("Prior variance W =",w[1]))
```


Prior variance $W = 50$



It's not a perfect match, but insisting on a fold-change of 2, i.e. $\hat{\beta} = \log(2) = 0.69$ as *well* as a significant p -value captures most of the BF-based results.