

2018 SISG Bayesian Statistics for Genetics

R Notes: Multinomial Sampling

Jon Wakefield

Departments of Statistics and Biostatistics, University of
Washington

2018-07-24

Hardy-Weinberg via Fisher's exact test

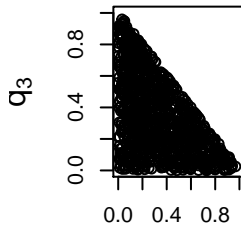
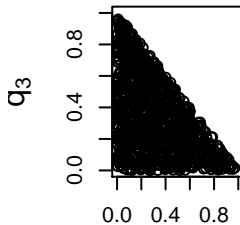
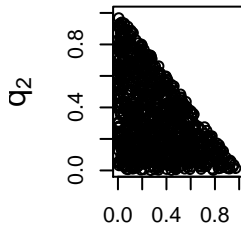
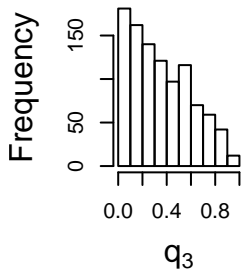
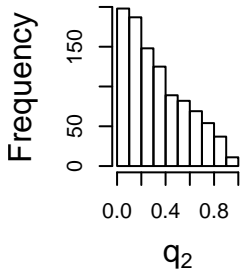
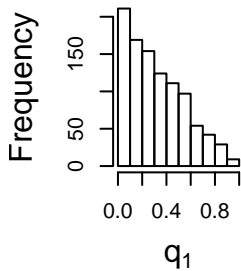
```
library(hwde)
n1 <- 88
n2 <- 10
n3 <- 2
exact <- hwexact(n1, n2, n3)
exact
## [1] 0.06544427
```

We obtain a p-value of 0.07

Displaying samples from a dirichlet(1,1,1)

```
library(VGAM) # To access the rdiric function
nsim <- 1000
q <- rdiric(nsim, c(1, 1, 1))
# Univariate marginal representations
par(mfrow = c(2, 3))
hist(q[, 1], xlab = expression(q[1]), main = "", cex.lab = 1.5,
      xlim = c(0, 1))
hist(q[, 2], xlab = expression(q[2]), main = "", cex.lab = 1.5,
      xlim = c(0, 1))
hist(q[, 3], xlab = expression(q[3]), main = "", cex.lab = 1.5,
      xlim = c(0, 1))
# Bivariate representations
plot(q[, 1], q[, 2], xlim = c(0, 1), ylim = c(0, 1),
      xlab = expression(q[1]), ylab = expression(q[2]),
      cex.lab = 1.5)
plot(q[, 1], q[, 3], xlim = c(0, 1), ylim = c(0, 1),
      xlab = expression(q[1]), ylab = expression(q[3]),
      cex.lab = 1.5)
plot(q[, 2], q[, 3], xlim = c(0, 1), ylim = c(0, 1),
      xlab = expression(q[2]), ylab = expression(q[3]),
      cex.lab = 1.5)
```

Displaying samples from a dirichlet(1,1,1)



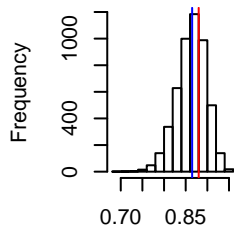
Bayes analysis of (88,10,2) data

```
n1 <- 88
n2 <- 10
n3 <- 2
p1 <- 88/100 + 0.5 * 10/100 # Estimated allele frequencies
p2 <- 2/100 + 0.5 * 10/100 # for A1 and A2
v1 <- v2 <- v3 <- 1
nsim <- 5000
q <- rdiric(nsim, c(n1 + v1, n2 + v2, n3 + v3)) # The posterior
par(mfrow = c(1, 3))
hist(q[, 1], xlab = expression(q[1]), main = expression(paste("Posterior for ",
  q[1])))
abline(v = n1/(n1 + n2 + n3), col = "red")
abline(v = p1^2, col = "blue")
hist(q[, 2], xlab = expression(q[2]), main = expression(paste("Posterior for ",
  q[2])))
abline(v = n2/(n1 + n2 + n3), col = "red")
abline(v = 2 * p1 * p2, col = "blue")
hist(q[, 3], xlab = expression(q[3]), main = expression(paste("Posterior for ",
  q[3])))
abline(v = n3/(n1 + n2 + n3), col = "red")
abline(v = p2^2, col = "blue")
```

Bayes analysis of (88,10,2) data

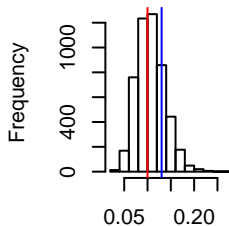
Univariate posterior distributions: blue lines are the MLEs in the full model, red lines under the HWE model

Posterior for q_1



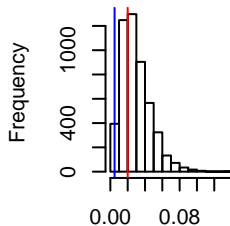
q_1

Posterior for q_2



q_2

Posterior for q_3



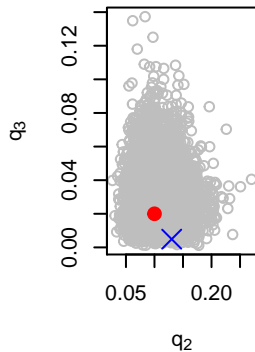
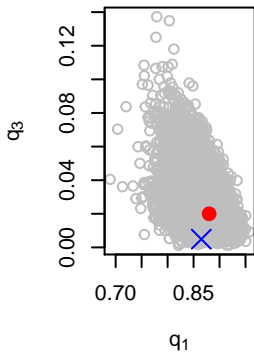
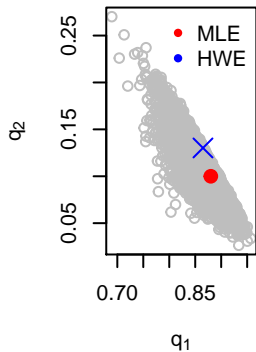
q_3

Bayes analysis of (88,10,2) data

```
par(mfrow = c(1, 3))
plot(q[, 2] ~ q[, 1], xlab = expression(q[1]), ylab = expression(q[2]),
     col = "grey")
points(n1/(n1 + n2 + n3), n2/(n1 + n2 + n3), col = "red",
       pch = 20, cex = 2)
points(p1^2, 2 * p1 * p2, col = "blue", pch = 4, cex = 2)
legend("topright", legend = c("MLE", "HWE"), col = c("red",
  "blue"), pch = c(20, 20), bty = "n")
plot(q[, 3] ~ q[, 1], xlab = expression(q[1]), ylab = expression(q[3]),
     col = "grey")
points(n1/(n1 + n2 + n3), n3/(n1 + n2 + n3), col = "red",
       pch = 20, cex = 2)
points(p1^2, p2^2, col = "blue", pch = 4, cex = 2)
plot(q[, 3] ~ q[, 2], xlab = expression(q[2]), ylab = expression(q[3]),
     col = "grey")
points(n2/(n1 + n2 + n3), n3/(n1 + n2 + n3), col = "red",
       pch = 20, cex = 2)
points(2 * p1 * p2, p2^2, col = "blue", pch = 4, cex = 2)
```

Bayes analysis of (88,10,2) data

Bivariate posterior distributions: blue lines are the MLEs in the full model, red lines under the HWE model



Functions of interest: implied priors

We assume a “dirichlet(1,1,1)” distribution

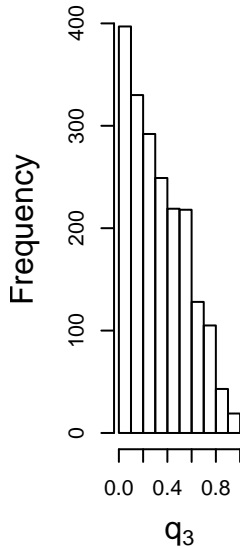
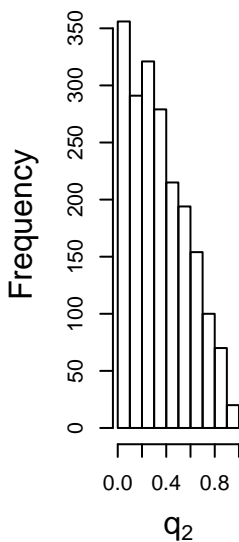
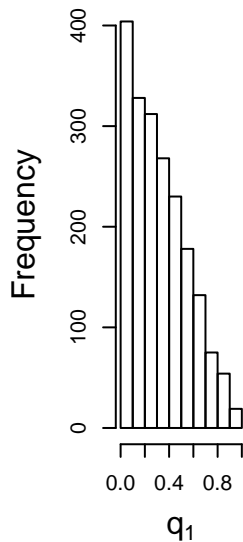
```
v1 <- v2 <- v3 <- 1
nsim <- 2000
samps <- rdiric(nsim, c(v1, v2, v3))
q1 <- samps[, 1]
q2 <- samps[, 2]
q3 <- samps[, 3]
p1 <- q1 + q2/2
p2 <- q3 + q2/2
f <- (q1 - p1^2)/(p1 * p2)
D <- q1 - p1^2
psi <- q2^2/(p1 * p2)
## Functions of interest
cat("Prior prob f>0: ", sum(f > 0)/nsim, "\n")
## Prior prob f>0: 0.6415
cat("Prior prob D>0: ", sum(D > 0)/nsim, "\n")
## Prior prob D>0: 0.6415
```

Functions of interest

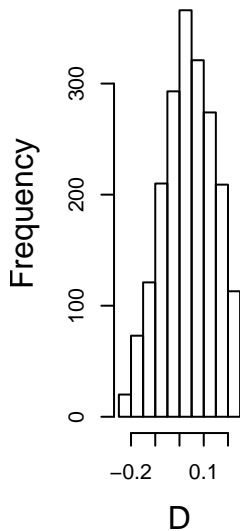
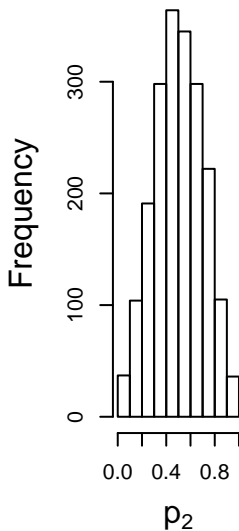
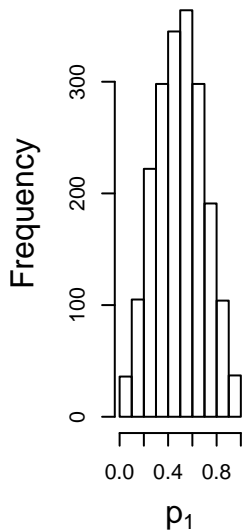
Examine prior summaries for different functions of interest.

```
par(mfrow = c(1, 3))
hist(q1, main = "", xlab = expression(q[1]), cex.lab = 1.5)
hist(q2, main = "", xlab = expression(q[2]), cex.lab = 1.5)
hist(q3, main = "", xlab = expression(q[3]), cex.lab = 1.5)
par(mfrow = c(1, 3))
hist(p1, main = "", xlab = expression(p[1]), cex.lab = 1.5)
hist(p2, main = "", xlab = expression(p[2]), cex.lab = 1.5)
hist(D, main = "", xlab = expression(D), cex.lab = 1.5)
par(mfrow = c(1, 2))
hist(f, main = "", xlab = "f", cex.lab = 1.5)
hist(psi, main = "", xlab = expression(psi), cex.lab = 1.5)
```

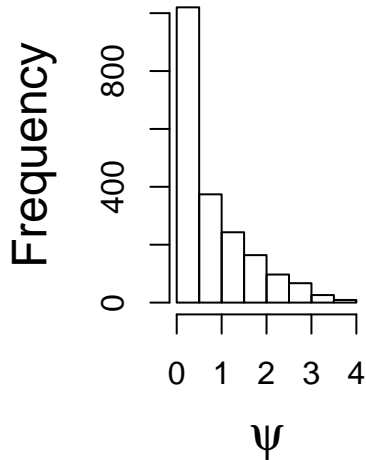
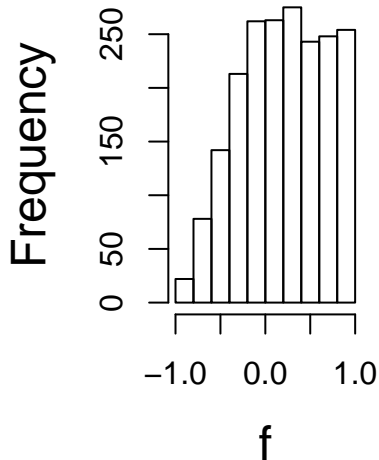
Functions of interest: prior margins on q_1, q_2, q_3 .



Functions of interest: priors on p_1, p_2, D



Functions of interest: priors on f, ψ .



Inference for f

The MLE is $\hat{f} = 0.23$ with asymptotic standard error 0.17.

Hence, a 95% confidence interval is

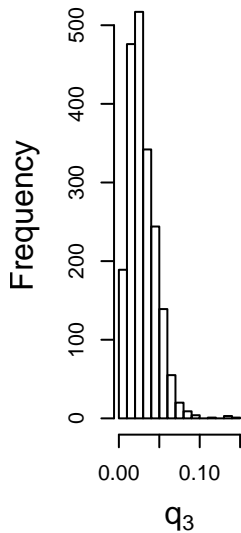
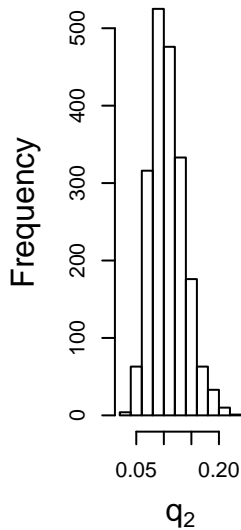
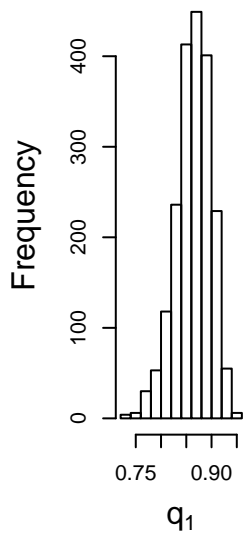
$$(0.23 - 1.96 \times 0.17, 0.23 + 1.96 \times 0.17) = (-0.1032, 0.5632).$$

The posterior median and 95% credible interval are given below.

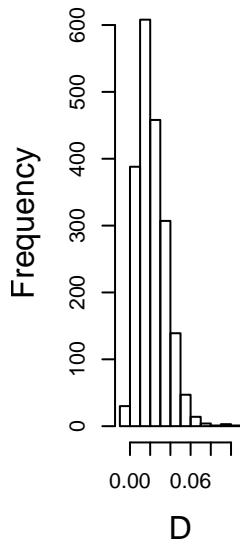
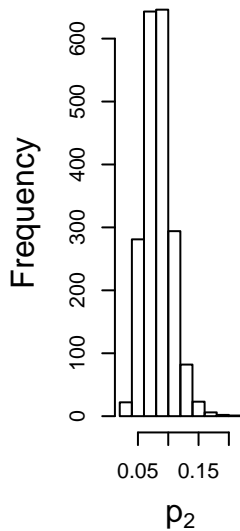
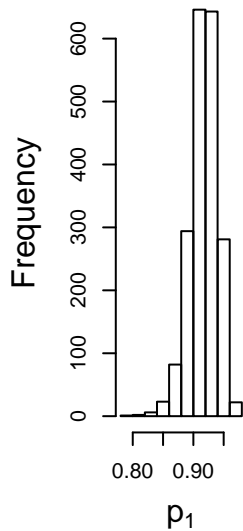
```
# Bayesian posterior quantiles are  
quantile(f, c(0.025, 0.5, 0.975))  
##          2.5%          50%          97.5%  
## -0.7039137  0.2151779  0.9586740
```

Subsequent figures give posterior distributions on functions of interest.

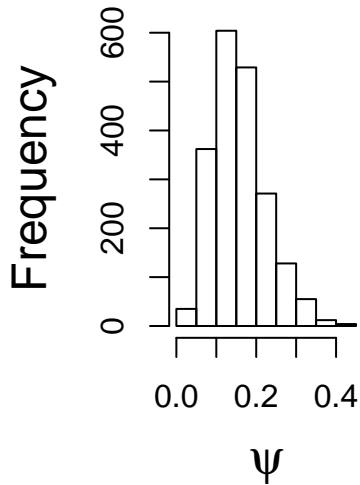
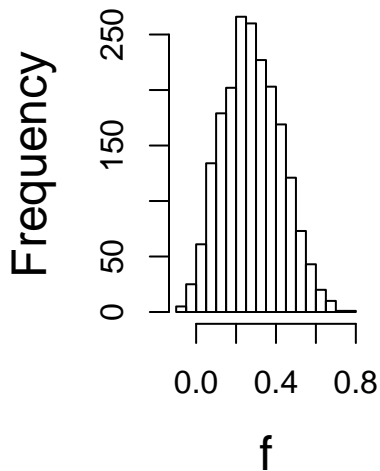
Dirichlet Posterior Distribution



Posterior summaries



Posterior summaries



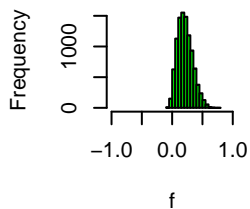
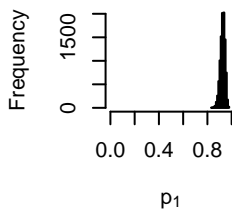
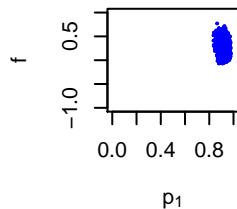
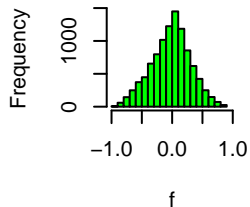
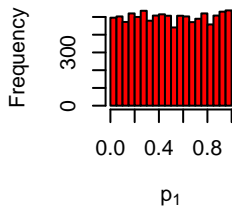
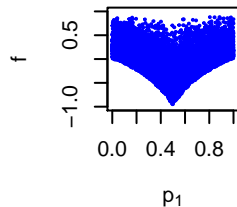
Non-conjugate analysis

```
library(HWEBayes)
nsim <- 50000
diricvec <- c(1, 1)
Prior <- LambdaOptim(nsim, diricvec, 0, 0.5, 0.5, 0.95,
  init = c(-1, log(1)))
## lambda mu and lambda sd = -1.248811 1.041571
nvec <- c(88, 10, 2)
nsamp <- 10000
Check <- SinglefPrior(nsamp, diricvec, Prior$lambda_mu,
  Prior$lambda_sd)
quantile(Check$f, p = c(0.5, 0.95))
##          50%          95%
## 0.001851612 0.498537395
```

Non-conjugate analysis

```
Out <- SinglefReject(nsamp, diricvec, Prior$lambda_mu,
  Prior$lambda_sd, nvec)
## Allele Marginal prob:  1 0.93
## Allele Marginal prob:  2 0.07
## Convergence = 0 (0 is successful convergence)
## Probs and f at max and fmin:
##  0.9299904 0.07000956 0.2318971 -0.07527987
## nsim norm constant (se) 95% interval:
## 10000 0.001023555 ( 7.341976e-06 ) 0.001009164 0.001037945
quantile(Out$fsamp)
##           0%           25%           50%           75%           100%
## -0.06969818  0.11886512  0.20179632  0.29519500  0.77060093
sum(Out$fsamp > 0)/nsamp
## [1] 0.9841
quantile(Out$fsamp)
##           0%           25%           50%           75%           100%
## -0.06969818  0.11886512  0.20179632  0.29519500  0.77060093
```

Non-conjugate analysis



Non-conjugate analysis

```
bvec0 <- c(1, 1)
bvec1 <- c(1, 1, 1)
PrnH0 <- DirichNormHWE(nvec, bvec0)
PrnH1sat <- DirichNormSat(nvec, bvec1)
BFHOH1sat <- PrnH0/PrnH1sat
cat("H0 norm = ", PrnH0, "\n")
## H0 norm = 0.0002993684
cat("H1 (sat) norm = ", PrnH1sat, "\n")
## H1 (sat) norm = 0.0001941371
cat("Conjugate Bayes factor = ", BFHOH1sat, "\n")
## Conjugate Bayes factor = 1.542047
BFHOH1NonConj <- PrnH0/Out$PrnH1
cat("Non-Conjugate Bayes factor = ", BFHOH1NonConj,
    "\n")
## Non-Conjugate Bayes factor = 0.2924792
```

Non-conjugate analysis: Increasing Sample Sizes by 2

```
exact2 <- hwexact(2 * n1, 2 * n2, 2 * n3)
exact2
## [1] 0.008886848
nvec2 <- 2 * nvec
PrnH02 <- DirichNormHWE(nvec2, bvec0)
PrnH1sat2 <- DirichNormSat(nvec2, bvec1)
BFH0H1sat2 <- PrnH02/PrnH1sat2
cat("2x Conjugate Bayes factor = ", BFH0H1sat2, "\n")
## 2x Conjugate Bayes factor = 0.4033526
Out2 <- SinglefReject(nsamp, diricvec, Prior$lambdamu,
  Prior$lambdasd, nvec2)
## Allele Marginal prob: 1 0.93
## Allele Marginal prob: 2 0.07
## Convergence = 0 (0 is successful convergence)
## Probs and f at max and fmin:
## 0.9299904 0.07000956 0.2318971 -0.07527987
## nsim norm constant (se) 95% interval:
## 10000 0.0002859723 ( 2.033496e-06 ) 0.0002819866 0.0002899579
sum(Out2$fsamp > 0)/nsamp
## [1] 0.998
BFH0H1NonConj2 <- PrnH02/Out2$PrnH1
cat("2x Non-Conjugate Bayes factor = ", BFH0H1NonConj2,
  "\n")
## 2x Non-Conjugate Bayes factor = 0.06947739
```

Non-conjugate analysis: Increasing Sample Sizes by 5

```
exact5 <- hwexact(5 * n1, 5 * n2, 5 * n3)
exact5
## [1] 3.583063e-05
nvec5 <- 5 * nvec
PrnH05 <- DirichNormHWE(nvec5, bvec0)
PrnH1sat5 <- DirichNormSat(nvec5, bvec1)
BFH0H1sat5 <- PrnH05/PrnH1sat5
cat("5x Conjugate Bayes factor = ", BFH0H1sat5, "\n")
## 5x Conjugate Bayes factor = 0.003945016
Out5 <- SinglefReject(nsamp, diricvec, Prior$lambdamu,
  Prior$lambdasd, nvec5)
## Allele Marginal prob: 1 0.93
## Allele Marginal prob: 2 0.07
## Convergence = 0 (0 is successful convergence)
## Probs and f at max and fmin:
## 0.9299904 0.07000956 0.2318971 -0.07527987
## nsim norm constant (se) 95% interval:
## 10000 4.925704e-05 ( 3.517281e-07 ) 4.856765e-05 4.994643e-05
sum(Out5$fsamp > 0)/nsamp
## [1] 1
BFH0H1NonConj5 <- PrnH05/Out5$PrnH1
cat("5x Non-Conjugate Bayes factor = ", BFH0H1NonConj5,
  "\n")
## 5x Non-Conjugate Bayes factor = 0.0006368967
```

Non-conjugate analysis: Increasing Sample Sizes by 10

```
exact10 <- hwexact(10 * n1, 10 * n2, 10 * n3)
exact10
## [1] 5.305573e-09
nvec10 <- 10 * nvec
PrnH010 <- DirichNormHWE(nvec10, bvec0)
PrnH1sat10 <- DirichNormSat(nvec10, bvec1)
BFHOH1sat10 <- PrnH010/PrnH1sat10
cat("10x Conjugate Bayes factor = ", BFHOH1sat10, "\n")
## 10x Conjugate Bayes factor = 1.154053e-06
Out10 <- SinglefReject(nsamp, diricvec, Prior$lambda_mu,
  Prior$lambda_sd, nvec10)
## Allele Marginal prob: 1 0.93
## Allele Marginal prob: 2 0.07
## Convergence = 0 (0 is successful convergence)
## Probs and f at max and fmin:
## 0.9299904 0.07000956 0.2318971 -0.07527987
## nsim norm constant (se) 95% interval:
## 10000 1.245895e-05 ( 8.870687e-08 ) 1.228508e-05 1.263281e-05
sum(Out10$fsamp > 0)/nsamp
## [1] 1
BFHOH1NonConj10 <- PrnH010/Out10$PrnH1
cat("10x Non-Conjugate Bayes factor = ", BFHOH1NonConj10,
  "\n")
## 10x Non-Conjugate Bayes factor = 1.847024e-07
```


HWE analysis via Stan

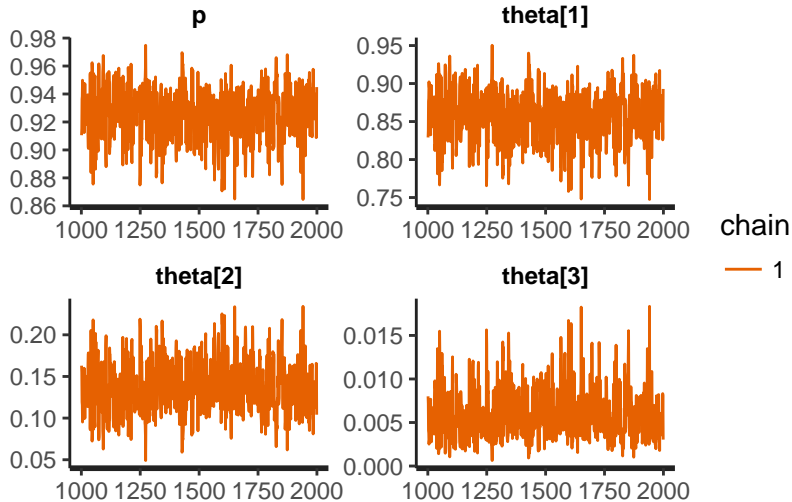
```
library(rstan)
stanexample <- stan("HWEexample.stan", data = list(y = c(88,
  10, 2)), iter = 2000, chains = 1, seed = 1234)
```

HWE analysis via Stan

```
print(stanexample)
## Inference for Stan model: HWEexample.
## 1 chains, each with iter=2000; warmup=1000; thin=1;
## post-warmup draws per chain=1000, total post-warmup draws=1000.
##
##           mean se_mean   sd  2.5%   25%   50%   75%  97.5% n_eff Rhat
## p           0.93    0.00 0.02   0.89   0.91   0.93   0.94   0.96   326   1
## theta[1]    0.86    0.00 0.03   0.79   0.83   0.86   0.88   0.92   324   1
## theta[2]    0.14    0.00 0.03   0.08   0.12   0.13   0.16   0.20   322   1
## theta[3]    0.01    0.00 0.00   0.00   0.00   0.01   0.01   0.01   413   1
## lp__        -47.00    0.04 0.70 -48.97 -47.18 -46.74 -46.56 -46.50   391   1
##
## Samples were drawn using NUTS(diag_e) at Mon Jul 23 20:41:33 2018.
## For each parameter, n_eff is a crude measure of effective sample size,
## and Rhat is the potential scale reduction factor on split chains (at
## convergence, Rhat=1).
```

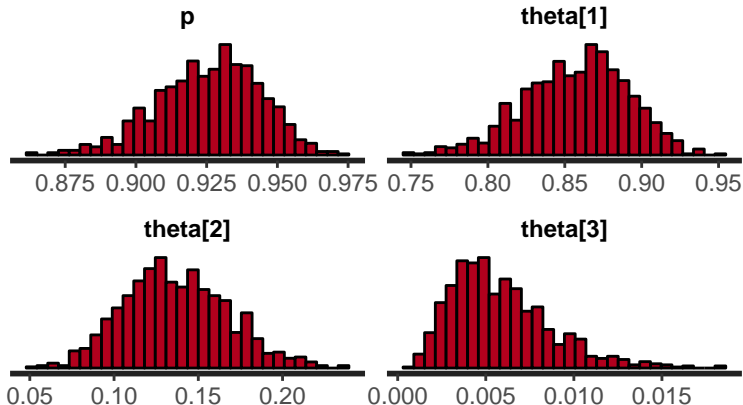
HWE analysis via Stan

```
traceplot(stanexample)
```



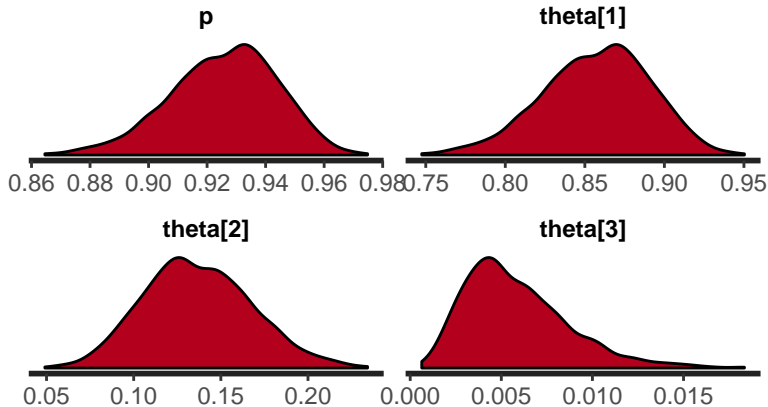
HWE analysis via Stan

```
stan_hist(stanexample)
```



HWE analysis via Stan

```
stan_dens(stanexample)
```



HWE analysis via Stan

Now run a second example with

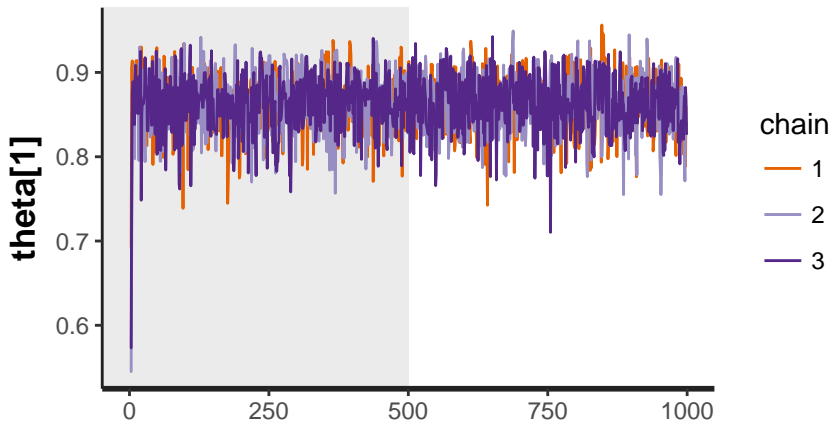
```
stanexample2 <- stan("HWEexampleSaturated.stan",  
  data = list(y = c(88, 10, 2), alpha = c(1,  
    1, 1)), iter = 1000, chains = 3,  
  seed = 1234)
```

HWE analysis via Stan

```
summary(stanexample2)
## $summary
##           mean      se_mean      sd      2.5%      25%
## theta[1]  0.86364137 0.0009698448 0.03204995  0.796408996  0.84297928
## theta[2]  0.10694565 0.0009071363 0.02914322  0.056216354  0.08509521
## theta[3]  0.02941297 0.0005063260 0.01634523  0.006538859  0.01735054
## lp__      -49.18671869 0.0367749137 0.93604437 -51.568059828 -49.59677209
##           50%      75%      97.5%    n_eff    Rhat
## theta[1]  0.86540537  0.88566043  0.91828361 1092.0693 1.000051
## theta[2]  0.10506044  0.12712657  0.16505920 1032.1197 1.000564
## theta[3]  0.02652115  0.03810645  0.06719045 1042.1300 1.000420
## lp__      -48.92713551 -48.51721491 -48.24191009  647.8725 1.001987
##
## $c_summary
## , , chains = chain:1
##
##           stats
## parameter      mean      sd      2.5%      25%      50%
## theta[1]  0.86373016 0.03125062  0.798139673  0.84400116  0.8643620
## theta[2]  0.10647039 0.02868388  0.054506645  0.08564891  0.1051546
## theta[3]  0.02979944 0.01607163  0.006897829  0.01757219  0.0272143
## lp__      -49.18929202 0.94430406 -51.494728045 -49.60043191 -48.9446966
##           stats
## parameter      75%      97.5%
## theta[1]  0.88540814  0.91779117
## theta[2]  0.12592753  0.16232566
## theta[3]  0.04002564  0.06457734
## lp__      -48.52867531 -48.23164561
##
## , , chains = chain:2
##
##           stats
## parameter      mean      sd      2.5%      25%      50%
## theta[1]  0.8641257 0.03230712  0.796570633  0.84297718  0.86719922
## theta[2]  0.1058380 0.02911062  0.057143836  0.08267446  0.10297417
## theta[3]  0.0300263 0.01606577  0.008006262  0.01757651  0.02699158
```

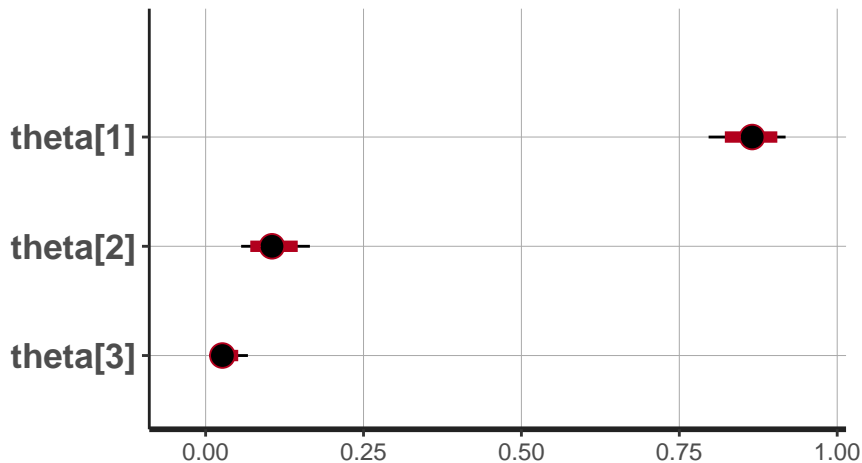
HWE analysis via Stan

```
traceplot(stanexample2, pars = c("theta[1]"), inc_warmup = TRUE)
```



HWE analysis via Stan

```
plot(stanexample2, color = "green")
```



HWE analysis via Stan

```
stan_scatter(stanexample2, pars = c("theta[1]", "theta[2]"),  
            color = "blue", size = 2)
```

