BayesTesting.jl - Bayesian Hypothesis Testing without Tears

> *Theoretical satisfaction and practical implementation are the twin ideals of coherent statistics.*
> Dennis Lindley, 1980.

Bayestesting.jl is a Julia package that provides objective Bayesian hypothesis testing procedures that do not suffer from the problems inherent in both the standard Bayesian and frequentist approaches, and works well in practice:

1. The Jeffreys-Lindley-Bartlett paradox does not occur.
2. Any prior can be employed, including uninformative and reference priors, so the same prior employed for inference can be used for testing, and objective Bayesian posteriors can be used for testing.
3. In standard problems when the posterior distribution matches (numerically) the frequentist sampling distribution or likelihood, there is a one-to-one correspondence with the frequentist test.
4. Provides posterior odds against the null hypothesis that are easy to interpret (unlike *p*-values), do not violate the likelihood principle, and result from minimizing a linear combination of type I and II errors rather than fixing the type I error before testing.
5. The testing procedure satisfies the Neyman-Pearson lemma, so tests are uniformly most powerful, and satisfies the most general Bayesian robustness theorem.

Functions are provided for a variety of standard testing situations, along with more generic modular functions to easily allow the testing procedure to be employed in novel situations. For example, given any Monte Carlo or MCMC posterior sample for an unknown quantity, θ, the generic BayesTesting.jl function `mcodds` can be used to test hypotheses concerning θ, often with one or two lines of code. A number of examples with detailed explanations are provided in the package documentation to ensure ease of use.

Functions currently available (package is under development)

Hypothesis testing:

Optional parameter in following functions: h0= value in hull hypothesis (default is h0 = 0)

Bayesian_ttest(x) = posterior t-test of precise null hypothesis for mean of data x. Returns posterior odds, Bayesian p-value, 0.99 and 0.95 HPDIs, posterior density plot

correlation_ttest(x,y) = posterior t-test for correlation coefficient. Returns posterior odds, Bayesian p-value, 0.99 and 0.95 HPDIs, posterior plot

todds(theta_hat,theta_hat_se,v) = returns Student-t posterior odds for theta

mcodds(theta_draws) = returns posterior odds given MC sample for theta (any distribution).

bayespval(theta_draws) = returns "Bayesian *p*-value" (tail areas) given MC sample for theta

compare_means(x,y) = x, y are two samples, computes difference in means test results.
Returns: diff_mean, draws_m1, draws_m2, qs, tst , plt
diff_mean = MC sample from posterior of difference in means

draws_m1, draws_m2s = MC sample from posterior for each mean
diff_mean = MC sample from posterior for difference in means
qs = HPD quantile intervals from diff_mean
tst = posterior odds against zero difference and area in tail to one side of zero.
plt = plot of posterior densities

Alternatively
compare_means(m1, m2, s1, s2, n1, n2) provides same results using:
m1, m2, are sample means, s1, s2, are sample SDs, n1, n2 are sample number of obs.

Optional arguments:
M = MC sample size (larger for improvement numerical accuracy)
lbl = ["mu 1 label" "mu 1 label"], use lbl = ["" ""] to eliminate labels
lgd = :topright (choose legend position, :topleft to move to top left of figure, etc.)

compare_proportions(x,y) or **compare_proportions**(s1, s2, n1, n2) = comparison of proportions for data from Bernoulli trials, where x and y are binary (1 = "success", 0 = "failure") or s1 and s2 are number of successes out of n1 and n2 trials.

Optional arguments:
M = MC sample size (larger for improvement numerical accuracy)
lbl = ["mu 1 label" "mu 1 label"], use lbl = ["" ""] to eliminate labels
lgd = :topright (choose legend position, :topleft to move to top left of figure, etc.)
a = Beta prior shape1 parameter (default = 1 for uniform prior)
b = Beta prior shape2 parameter (default = 1 for uniform prior)

equiv_test(mc_sample, tolerance_interval) = returns posterior odds against equivalence, probabilities for regions adjacent to equivalence region, plot of posterior with equivalence/nonequivalence regions. The MC sample is from the posterior for difference in means or proportions obtained from the compare_means or compare_proportions functions.

Posterior inference:

update_mean(m1,m0,s1,s0,n1,n0) = For Gaussian posterior sample 1 (or prior) with mean = m0, sd = s0, number of obs. =n0, and Gaussian likelihood or posterior for sample 2 with mean = m1, SD = s1, number of obs. = n1, returns tuple of combined sample posterior mean = m2, SD = s2, number of obs. = n2

marginal_posterior_mu(m,s, n, M) = returns M draws from Student-t marginal posterior density with mean = m, SD = s, number of obs. = n. M is an optional argument (default is M = 10000).

blinreg(y,x) = returns regression results including posterior odds testing coeff. = 0, where y = vector of obs. for dependent variable, x = matrix of explanatory variables (not including the intercept).

gsreg(y,X) = Gibbs sampler for linear regression with default uninformative prior, X must contain vector of ones to include intercept.

gsreg(y,X, M=m, tau=t, b0=priorb, iB0 = invpriorcovb , d0=b, a0=a) = Gibbs sampler with NIG prior.
Note: iB0 = prior precision matrix = inv(prior variance matrix)
b0 must be a column vector,
a0 and b0 are prior parameters for tau ~ Gamma(a,b)

A few illustrative examples

Example 1: Testing a sample mean, $H_0: \mu = \mu_0$, with unknown variance.

Sample 50 obs. From a $N(0,1)$, compute the *t*-statistic and degrees of freedom, then compute the posterior odds vs. $H_0: \mu = 0$:

```
# generate example data
x = randn(50)  # or rand(Normal(0,1),50)

# compute odds, p-value, 0.99 & 0.95 HPDS, median and posterior plot:
odds, pvalue, quants, plot1 = Bayesian_ttest(x)

# Alternatively, compute posterior odds using todds function:
t_hat = mean(x)/(std(x)/sqrt(length(x)-1)
v = length(x) - 1
todds(that,v)
```

Example 2: Correlation coefficient test

For a correlation coefficient $\rho = corr(x,y) = cov(x,y)/\sqrt{var(x)\ var(y)}$, to test the null hypothesis $H_0: \rho = 0$ given data x and y:

```
# generate example data
x = randn(50)
y = 1.0 .+ 0.2.*x .+ randn(50)

# Test hypothesis:
results = correlation_ttest(y,x)

results[1]     # posterior odds
results[2]     # posterior p-value
results[3]     # 0.99, 0.95 HPD intervals and median
results[4]     # posterior density plot
```

Example 3: Comparison of means

Comparing two means from two different distributions (means from *any* distributions can be compared) is straightforward (using method (2) described in the methodology section below, or method (3) if conditional posteriors are available). To test the hypothesis $H_0: \mu_1 = \mu_2$, the procedure is:

1. Obtain a MC sample of size M from each of $p(\mu_i|D), i = 1,2$.
2. Compute the difference, $d^{(j)} = \mu_1^{(j)} - \mu_2^{(j)},\ \ j = 1, \ldots, M$.
3. Compute the posterior odds for the MC sample vector *d* using `mcodds(d)`.

Example 3 code

```
using Distributions, Plots, StatPlots, BayesTesting
```

```
# generate n observations of pseudo-data for two variables
srand(1299)
n1 = 50
n2 = 60  # unbalanced samples are fine.
x1 = rand(Normal(0.5,1),n1)
x2 = rand(Normal(1.0,0.5),n2)

### Now assume only have two samples, x1 and x2

# using compare_means function

# Either:
results = compare_means(x,y)
results[3]     # posterior odds and Bayesian p-value
results[4]     # plot of posterior densities for each mean and for
               # difference in means

# or:
diff_mean, draws_m1, draws_m2, qs, tst, plt3 = compare_means(x,y)

# results[1] contains diff_mean, results[2] contains draws_m1, etc.


# A do-it-yourself approach:
# Assuming Normally distributed variables (allowing for unequal
# variance), the marginal posteriors are both Student-t
# Draws from each t-distribution:
M = 100000
mu1_draws = mean(x1) .+ (std(x1)/sqrt(n1-1)).*rand(TDist(v),M)
mu2_draws = mean(x2) .+ (std(x2)/sqrt(n2-1)).*rand(TDist(v),M)

# compute posterior density of the difference in means
d_draws = mu2_draws - mu1_draws

# plot the posterior difference
plot(d_draws, st=:density,linewidth=2,label="Posterior for difference")

# compute odds of difference from zero
mcodds(d_draws)

# compute Bayesian one-sided p-value
one_sided_pval(d_draws)

# two-sided p-value assuming symmetric distribution
bayespval(d_draws)
```

Example 3: Regression parameters

For the normal linear model, the marginal posterior is Student-*t* distributed (as with the mean of a normal distribution in example 1), so the posterior odds are exactly the same formula as in example 1, with the t-value $= |\bar{\beta} - \beta_0|/SD(\bar{\beta})$, and $v = n - k$, $k =$ number of regression parameters.

For example, for data on presidential candidates' height difference and difference in vote popularity, a linear model is estimated. The model is,

$$pop_i = \alpha + \beta ht_i + u_i, u_i \sim N(0, \sigma^2),$$

where ht_i is the height ratio of the winner to the opposing presidential candidate, and pop_i is the proportion of the vote received by the winner.

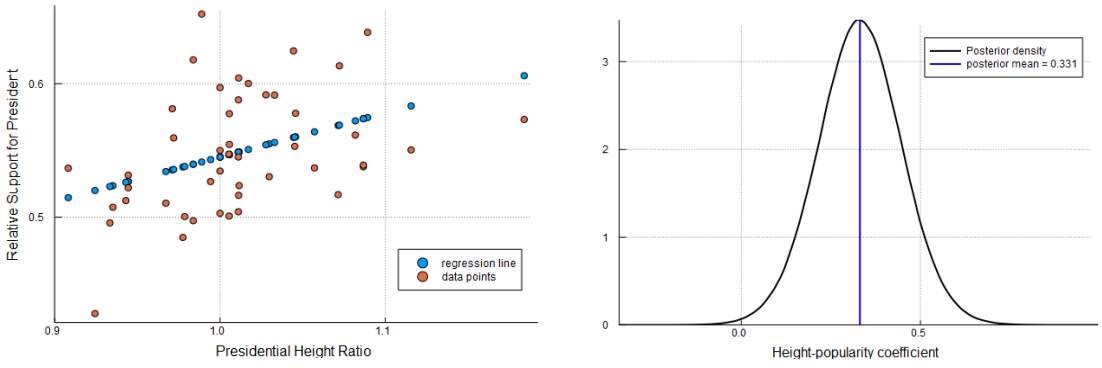
Example 3 code

```
using CSV, BayesTesting, Plots, StatPlots
dat = CSV.read("Presidents.csv")  ## data available on github
n = length(dat[:,2])
ht = Array(dat[2])
pop = Array(dat[3])
# using blinreg.jl function:
bhat, seb, odds, s, R2 = blinreg(pop,ht)
   #            Intercept slope
    # coeffs = [0.214   0.331]
    # s.e's = [0.116   0.114]
    # odds = [5.384  51.119]
    # p-val = [0.0712 0.0058]
    # s^2 (eqn. variance) = 0.00171, R-squared = 0.155

# using blinreg.jl function:
todds(bhat[2],seb[2],(n-2))

# plotting data, regression line, and posterior for beta
p1 = plot(ht, pop, st=:scatter, label="data
points",legend=:bottomright,xlabel="Actual and fitted values")
yhat = bhat[1] .+ ht.*bhat[2]
plot!(ht,yhat,st=:scatter,label="regression line" )

# posterior simulation
b2_draws = marginal_posterior_mu(bhat[2],seb[2],n,M=8000000)
p2 = plot(b2_draws,st=:density,label="", linecolor=:blue,xlims = [-0.25,
   1.00],xlabel="Posterior for height coeff.")
vline!([0.331], linecolor=:black, linewidth=0.4, label="mean = 0.331")
mcodds(b2_draws)
   # Results:
```




```
# HPD intervals:
quantile(bdraws, [0.005, 0.025, 0.975, 0.995])
#              = [0.0245, 0.101, 0.561, 0.638]
```

```
# Test of H0:β = 0.0
todds(0.331,42)  # = 51.119
```

The *p*-value of 0.0058, objective odds of 51.12:1 against the null, and 0.99 HPD interval of (0.024,0.638),which excludes zero, are all in agreement on the strength of the evidence against the null hypothesis of no effect, $H_0: \beta = 0$, whereas Wagenmakers *et al.* (2017) report a standard Bayes factor of only 6.33:1 against the null hypothesis.

Methodology: Objective Bayesian hypothesis testing

The hypothesis testing procedure developed in Mills (2017) has been applied in a number of settings including: comparison of means (Strawn *et al.*, 2018*a*), ANOVA testing (Mills and Namavari, 2016), meta-analysis (Strawn *et al.* 2018*b*), unit root and cointegration testing (Mills, 2013, Mills and Namavari, 2016), Granger causality testing (Mills *et al.*, 2017), and predictive model comparison (Cornwall and Mills, 2017). In all of these applications the procedure performs as well or better than frequentist methods, and does not exhibit any of the shortcomings of standard Bayes factors.

The new testing procedure is specifically designed to address the problems that arise when testing a precise null hypothesis compared to a composite alternative, $H_0: \Theta = \theta_0$ vs. $H_1: \Theta \neq \theta_0$, which is the most common testing problem in science. However, the same procedure is applicable to testing composite vs. composite and precise vs. precise hypotheses, so all testing situations are covered. The procedure is derived by partitioning the alternative hypothesis parameter space, forming the posterior odds ratio for each interval in the partition relative to the interval containing the null hypothesis, and computing the supremum of these ratio, giving the maximum posterior odds against the null hypothesis according to the data and any background information incorporated into the likelihood (model specification) and prior.

The generic testing procedure for testing $H_0: \Theta = \theta_0$ vs. $H_1: \Theta \neq \theta_0$, is as follows (see Mills, 2017, for details).

1. For posterior $p(\theta|D)$, conditional on data, D, compute the posterior odds ratio,

$$O = \frac{p(\bar{\theta}|D)}{p(\theta_0|D)}, \qquad \bar{\theta} = \underset{\theta}{\operatorname{argmax}}\, p(\theta|D).$$

2. Reject H_0 if $O \geq l(H_0|H_1)/\, l(H_1|H_0)$, where $l(H_i|H_j)$ is the loss or cost associated with choosing H_i when H_j is correct, so $l(H_1|H_0)$ is cost of a type I error, and $l(H_0|H_1)$ is the cost of a type II error.

The above steps only require evaluation of the posterior density at two points. The three main situations which arise are:

(1) The posterior is a known analytical form, so the pdf can be directly evaluated at the value in the null hypothesis and at its mode (using the Julia Distributions package).

(2) If a posterior simulation (MC) sample is available from $p(\theta|D)$, then evaluating a kernel density estimate of the posterior from the sample at the two points is carried out (using the Julia

KernelDensity package) with the BayesTesting.jl function `mcodds(MC_theta; h0=0)`, where MC_theta = the posterior simulation sample, and h0 is an optional argument to set to the value in the null hypothesis (set to zero by default).

(3) If other parameters are involved and an MCMC sample is generated for each parameter, then the marginal posterior can be evaluated by Rao-Blackwellizing over the conditional posterior using the BayesTesting.jl function rbodds(MC_theta, cond_params, cond_post), where cond_post = the kernel of conditional posterior density function, $p(\theta|\phi, D)$, MC_theta = the posterior simulation sample for the parameter being tested θ, cond_params = array of posterior simulation samples for each of the conditioning parameters, ϕ. This provides increased accuracy compared to (2). Note that the normalizing constant is not needed for the conditional posterior, only the kernel of the density, which is typically available whenever an MCMC sample is available since it is used in the Metropolis-Hastings step to generate the posterior draws.

Comparison with *p*-values

> "We have saddled ourselves with perversions of logic - *p*-values - and so we deserve our collective fate." (Berry, D., 2017)

Although the use of *p*-values has been heavily criticized in the statistics literature, a viable alternative has not been suggested that is broadly acceptable (Gelman and Carlin, 2017). The objective posterior odds implemented in BayesTesting.jl provides this missing alternative. The fact that the results are easily reproducible and match frequentist testing conclusions in standard problems, should allow a scientific consensus to develop on statistical hypothesis testing in practice.

While use of *p*-values is not recommended – objective posterior odds are designed to replace them - functions are also provided to compute "Bayesian *p*-values" to allow comparison with posterior odds and aid in the weaning process from "*p*-value-itis". The computed Bayesian *p*-values are based on the posterior density mass outside a highest posterior density (HPD) interval. A one-sided Bayesian *p*-value is computed as,

$$p\text{-value} = \begin{cases} \int_{\theta<\theta_0} p(\theta|D)d\theta, & \theta_0 \leq \bar{\theta}, \\ \int_{\theta>\theta_0} p(\theta|D)d\theta, & \theta_0 > \bar{\theta}. \end{cases}$$

A two-sided *p*-value is computed, assuming the posterior density is symmetric, by doubling the one-sided value. This will match the frequentist *p*-value when the posterior density is unimodal and symmetric.

Testing a Normal mean with unknown variance

For a sample from a Gaussian with unknown mean and variance, $x \sim N(\mu, \sigma^2)$, suppose we wish to test $H_0{:}\,\mu = \mu_0$ vs. $H_1{:}\,\mu \neq \mu_0$. For the uninformative (Jeffreys uniform, log-uniform) prior, $p(\mu, \sigma^2) \propto 1/\sigma^2$, the marginal posterior for μ is in the form of a (noncentral) Student-*t* density, $t(\bar{\mu}, \nu s^2, \nu)$ with posterior parameters, $\nu = n - 1$, $\bar{\mu} = \sum x_i/n$, and $s^2 = \sum_i (x_i - \bar{\mu})^2 / \nu$. The objective posterior odds are then,

$$O = \left(1 + \frac{t^2}{v}\right)^{\frac{v+1}{2}},$$

where $t = \sqrt{(\bar{\mu} - \mu_0)^2/s^2}$, the square of the usual *t*-statistic.

This is computed in the function `todds(t,v)`.

If instead we have *M* draws from the Student-*t* marginal posterior density, mc_sample, the odds, O, can be computed using `mcodds(mc_sample, h0=`μ_0`),` which uses a nonparametric kernel density approximation to the posterior to evaluate the odds.

Lastly, if an MCMC sample is available for both μ and σ^2, the odds can be computed from the conditional posterior density, $p(\mu|\sigma^2, D)$, which is a Gaussian, [function available soon, currently being implemented].

Summary

The examples in the documentation currently under development will illustrate a number of different ways to easily compute objective posterior odds ratios for both precise and composite hypotheses. The functions in BayesTesting.jl allow for flexible testing in a variety of settings. As mentioned earlier, these methods have been applied successfully to:

- Comparison of means (Strawn *et al.*, 2018*a*)
- ANOVA testing (Mills and Namavari, 2016)
- Meta-analysis (Strawn *et al.* 2018*b*)
- Binomial proportions equivalence testing (Mills, 2018)
- Unit root testing (Mills, 2013)
- Cointegration testing (Mills and Namavari, 2016)
- Granger causality testing (Mills *et al.*, 2017)
- Predictive model comparison (Cornwall and Mills, 2017)

References

Berry, D. (2017) A *p*-Value to Die For. *Journal of the American Statistical Association*, 112:519, 895-897

Cornwall, G. and Mills, J. (2017) Prediction Based Model Selection Criteria. University of Cincinnati.

Gelman, A. and Carlin, J. B. (2017) Some Natural Solutions to the *p*-Value Communication Problem—and Why They Won't Work. Journal of the American Statistical Association, 112:519, 899-901.

Lindley, D. V. (1980). Jeffreys's contribution to modern statistical thought. In Zellner, A. (Ed.) Bayesian analysis in econometrics and statistics: Essays in honor of Harold Jeffreys, 35–39.

Mills, J.A. (2015) Objective Bayesian Unit Root Testing. University of Cincinnati, DOI: 10.13140/RG.2.2.13158.32328

Mills, J.A. Cornwall, G, Sauley, B., Weng, H. (2018) Bayesian Predictive Granger Causality Testing.

University of Cincinnati.

Mills, J.A. and Namavari, H. (2016) Objective Bayesian ANOVA Testing. University of Cincinnati

Mills, J.A. and Namavari, H. (2017) Residual Based Objective Bayesian Cointegration Testing. University of Cincinnati.

Mills, J.A. (2017) Objective Bayesian Precise Hypothesis Testing. University of Cincinnati [original version: 2007]

Mills, J.A. and Strawn, J.R. (2018) Bayesian equivalence testing. University of Cincinnati.

Strawn J.R., Mills J.A., Cornwall G.J., Mossman, S.A., Varney, S.T., Keeshin B.R., Croarkin, P.E. (2018*a*) Buspirone in Children and Adolescents with Anxiety: A Review and Bayesian Analysis of Abandoned Randomized Controlled Trials. Journal of Child and Adolescent Psychopharmacology. 28:1, 2-9.

Strawn J.R., Mills, J.A., Sauley, B.A., Welge, J.A. (2018*b*) The Impact of Antidepressant Dose and Class on Treatment Response in Pedriatric Anxiety Disorders: A Meta-Analysis. Journal of the American Academy of Child and Adolescent Psychiatry, 57:4, 235-244.

Wagenmakers, E, Marsman, M, Jamil, T, Ly, A, Verhagen, J, Love, J, Seler, R, Gronau, QF, Smira, M, Epskamp, S, Matzke, D, Rouder, JN, Morey, RD (2017) Bayesian inference for psychology. Part I: Theoretical advantages and practical ramifications. Pychonomic Bulletin & Review, 25:1, 35-57. DOI 10.3758/s13423-017-1343-3