Decision Qualities of Bayesian and Frequentist Hypothesis Tests applied to Binomial Experiments with Predetermined Sample Sizes

Joukje Willemsen (4257634), Bachelor Student

Utrecht University

Author Note

A special thank you to my supervisor Dave Hessen who provided me with this interesting research subject, who always made time to schedule in a feedback session whenever I requested it and who vary patiently read and commented on my (lengthy!) previous versions.

Abstract

There is plenty of literature about the theoretical and philosophical differences between frequentist and Bayesian approaches. However, surprisingly little research has been conducted with regard to comparing these methods in performance. This thesis aims to provide a comparison of the discriminatory performances of Bayesian and frequentist methods, confined to making inferences about proportions and binomial data. The results of the simulation study show that the ROC curves of the p-value and Bayes factor coincide when M1 is specified as a symmetric probability distribution. When this is not the case, the decision qualities of the Bayesian approach is as good as the prior that was used. By matching the "rejection regions" of Bayesian and frequentist tests, the tests would yield equal decision qualities. It would be better practice to combine both procedures by calibrating Bayesian decision rules into frequentist decision rules (or vice versa) to ensure effective control of Type I error probabilities while simultaneously take advantage of the Bayesian benefits.

Decision Qualities of Bayesian and Frequentist Hypothesis Tests applied to Binomial Experiments with Predetermined Sample Sizes

According to García & Puga (2018) "*it seems that for most researchers, statistical practice is becoming an automated procedure in which most researchers tend to blindly use statistical approaches to make decisions based on their data*". For over the last 75 years, the predominant option and mainstream statistical analysis is characterized by using the null hypothesis significance testing (NHST) procedure with the p-value being a tool to decide about the null hypothesis (García & Puga, 2018; Kruschke & Liddell, 2018b). The term "null hypothesis testing" in this article refers to point-value hypothesis testing for which the hypothesis being tested is a specific value of a parameter: the null hypothesis value. The p-value represents the number of times we would observe a sample statistic, or more extreme one, in case the null hypothesis is true in the population sampled from, and an experiment with a set sample size is repeated several times under the same conditions (García & Puga, 2018).

Shift to estimation with uncertainty

Many articles and books propose that a better option would be to discard the p-value and encourage a shift to estimation with uncertainty (Cumming, 2014). For example by not only including effect sizes and their 95% CIs in analyses, but also focusing the attention on these values in order to emphasize the importance and precision of the estimated effect size (Halsey, Curran-everett, Vowler, & Drummond, 2015; Kruschke & Liddell, 2018b). Effect sizes and their 95% CIs can be used to make threshold-based decisions about statistical significance in the same way as the p-value can be applied (their decisions will always correspond), but they provide more information than the p-value. In addition, the effect size and 95% CIs allow findings from

several experiments to be combined with meta-analysis to obtain more accurate effect-size estimates.

Confidence Intervals in combination with a specified "Region Of Practical Equivalence" (which will be referred to as the "CI+ROPE" approach) can be used when the experimenter is indifferent about a smaller effect size or difference from the null hypothesis value but does care about a larger difference (Barker, Rolka, Rolka, & Brown, 2001). With this approach it would be possible to find evidence that two populations are "practically equivalent".

Shift to Bayesian analyses

The p-value does not computes the probability of the observed data under the null hypothesis (H_0), but the probability of the observed data and more extreme data under H_0 (Hubbard & Lindsay, 2008). Therefore, the p-value denotes not only the probability of what was observed, but also the probabilities of the more extreme events that did not occur. Besides, the p-value and CIs depend on the stopping and testing intentions of the researcher (Kruschke & Liddell, 2018b). As a result, different stopping or testing intentions yield different p-values and CIs for any fixed set of observed data.

Secondly, the p-value cannot provide evidence for the null hypothesis (Dienes, 2016). Despite that, a non-significant result is often in practice taken as evidence for a null hypothesis. But "*absence of evidence is not evidence of absence*" (Altman, 1995).

According to Bayesians, these are major weaknesses regarding the usefulness of p-values (Hubbard & Lindsay, 2008; Kruschke & Liddell, 2018b). They charge that a valid measure of strength of evidence cannot be dependent on the probabilities of unobserved outcomes. Bayesian statistics combine prior beliefs about phenomena with observed data to update this belief according to the laws of probability theory (García & Puga, 2018; See & Cohen, 2007). One

Bayesian alternative to frequentist NHST is the Bayes factor (BF) approach (Kruschke, 2011). The Bayesian equivalent of the CI+ROPE approach is the High Density Interval & Region Of Practical Equivalence (which will be referred to as the "HDI+ROPE" decision rule).

However, Bayesian inference is not without controversies among statisticians (Gelman, 2008). Especially the use of priors is still object of hot discussions (García & Puga, 2018). Opponents of the Bayesian approach reason that scientist should be concerned with objective knowledge rather than subjective priors. In his article, Gelman (2008) quotes Andrew Ehrenbergh who wrote "*Bayesianism assumes: (a) Either a weak or uniform prior, in which case why bother?, (b) Or a strong prior, in which case why collect new data?, (c) Or more realistically, something in between, in which case Bayesianism always seems to duck the issue*".

Besides, Bayesian analysis ignores error rates as they do not take into account results that were not observed (Kruschke & Liddell, 2018b). Therefore, Bayesian analysis cannot (directly) control them.

Previous Research Performance of Bayesian and Frequentist approaches

There is plenty of literature about the theoretical and philosophical differences between frequentist and Bayesian approaches (Bolstad & Curran, 2016; Kruschke & Liddell, 2018b; Ortega & Navarrete, 2017). However, surprisingly little research has been conducted regarding the comparison of these methods in terms of performance (Jeon & De Boeck, 2017). In diagnostic tests with dichotomous outcomes (positive/negative test results), the conventional approach of diagnostic test evaluation uses sensitivity or "proportion true positives" (the probability of obtaining a positive result when there is indeed an effect) and specificity or "one minus the proportion false positives" (the probability of obtaining a positive result when there is no effect present) as measures of accuracy (Hajian-Tilaki, 2013).

The HDI+ROPE decision rule is relatively new and was first described by Kruschke in 2010. Beside theoretical comparisons, the decision quality of the CI+ROPE approach and HDI+ROPE approach have not been compared in a (simulation) study in terms of decision qualities.

Only a few studies have investigated the proportion true positives and the proportion false positives of the p-value and the BF separately (García & Puga, 2018; Jeon & De Boeck, 2017). However, these studies did not address the relationship between sensitivity and specificity when assessing the decision qualities of the respective tests. In this light, it should be noted that neither the proportion true positives as the proportion false positives in itself validly represent the tests' discriminatory performance as high sensitivity may be accompanied by low specificity (Glas, Lijmer, Prins, Bonsel, & Bossuyt, 2003). After all, the sensitivity is inversely related with specificity (Hajian-Tilaki, 2013).

Measuring Decision Quality

A more informative measure would be to combine the proportion true positives and the proportion false positives into a single performance index, being the positive likelihood ratio (LR+) (Florkowski, 2008; Hajian-Tilaki, 2013). The positive likelihood ratio of a test is the ratio of the probability of obtaining a positive result when there is indeed an effect and the probability of obtaining a positive result when there is no effect. For Frequentist NHST, the LR+ is the ratio of the power to the significance threshold (Bayarri et all. 2016).

Because sensitivity and specificity of the four methods depend on their decision thresholds which are chosen arbitrarily, assessing the proportion true positives and proportion false positives for one cut-off as a measure of accuracy is not particularly informative (Florkowski, 2008; Hajian-Tilaki, 2013). Rather, by assessing the sensitivity and specificity for a range of different thresholds, the proportion true positives can be plotted against the proportion false positives, yielding a more informative comparison measure called receiver operating characteristic (ROC) curve (Florkowski, 2008). In other words, the ROC curve graphically displays the trade-off between sensitivity and one minus the specificity among different thresholds. For every proportion true positives and every proportion false positives, the LR+ can be obtained from the ROC curve: the slope of a ROC curve at any point is equal to the LR+.

The ROC curve plays a central role in evaluating the diagnostic ability of tests to discriminate the true state of subjects and comparing two alternative diagnostic tasks when each task is performed on the same subject (Hajian-Tilaki, 2013). A ROC curve lying on the diagonal line reflects the performance of a diagnostic test that is no better than chance level. A ROC curve that is closer to the upper left-hand corner corresponds to a greater discriminant capacity. In contrast to single measures of sensitivity, specificity and the LR+, the ROC curve is not affected by decision criteria.

Aim study

Because hypothesis testing is very well-established in science (Bolstad & Curran, 2016), it would be interesting to know whether Bayesian tests outperform frequentist tests (or vice versa) in decision accuracy when testing a hypothesis. This thesis aims to provide a comparison of the discriminatory performances of Bayesian and frequentist methods, confined to making inferences about proportions and binomial data, using the example of assessing whether a coin is

'fair'. In this study, the p-value will be compared to a proposed Bayesian alternative: the Bayes factor. Besides, the CI+ROPE decision rule will be compared to its Bayesian equivalent: the HDI+ROPE decision rule. To do so, a simulation study is conducted that calculates the proportion true positives and false positives among different prior settings, different effect sizes and different sample sizes. The decision thresholds of the p-value and BF will be varied to obtain (an estimation of) their ROC curves. The simulation study is explorative in nature; no explicit hypotheses were stated beforehand.

Before discussing the simulation study and the result, each method will be introduced in the next chapter to obtain a deeper understanding of the logic behind each method, how the methods are conducted and how inferences are made as knowing the differences and similarities will be essential for interpreting the results.

Methods

Frequentist methods

When testing whether a coin is fair or not, the question asked by frequentists would be "*What is the probability of obtaining a sample proportion that is as extreme as, or more extreme than the obtained sample proportion, given that the null hypothesis is true and if we were to collect data the same way we collect data in the actual research?*" (See & Cohen, 2007). If this probability is small enough, the null hypothesis is rejected. This approach to hypothesis testing is closely related to Popper's theory in which scientific theories are never accepted as being true, but instead are only subjected to increasingly severe tests (Johnson, 2013).

Frequentist NHST using p-values

In the example of a coin, the null hypothesis would be that we have a fair coin; the true parameter value of the coin (θ) is 0.5. The alternative hypothesis is that the coin is not fair. This is denoted as H_0: $\theta = 0.5$ and H_1: $\theta \neq 0.5$. To test whether the coin is fair, we count the number of heads when we flip the coin N times. Let the random variable Y be the observed number of heads in a sample of N Bernoulli trials and let S be the sample space of y when the sample size is n; $S = \{0, 1, 2, \dots, n\}$. The conditional probability function for y, given that $\theta = 0.5$ is given by:

$$f(Y = y \mid \theta = 0.5) = \binom{n}{y} 0.5^y \, (1 - 0.5)^{n-y} \text{ for y } = 0, 1, \dots, n \qquad (1)$$

(Bolstad, 2016).

A p-value is used to describe the probability of getting the actual outcome, or an outcome more extreme when the null hypothesis would be true (Kruschke, 2015). For a one-tailed test that corresponds to the hypothesis $H_1 : \theta > 0.5$, this can be obtained from equation (1) by adding all the probabilities of obtaining the observed y (y_{obs}) and the more extreme y's such that $p_{1-tail} = \sum_{y=y_{obs}}^{n} f(Y = y \mid \theta = 0.5)$ (See & Cohen, 2007). To calculate the two-sided p-value that

corresponds to H_1: $\theta \neq 0.5$, we need to consider the more extreme values at both the high and the low end of the sampling distribution. The two-sided p-value is given by:

$$p_{2-tail} = \begin{cases} 2 \sum_{\substack{y=y_{obs} \\ n-y_{obs}}}^{n} f(Y = y \mid \theta = 0.5) & \text{if } y_{obs} \geq \frac{n}{2} \\ 2 \sum_{y=0}^{} f(Y = y \mid \theta = 0.5) & \text{if } y_{obs} \leq \frac{n}{2} \end{cases} \quad (2)$$

Figure 1 visualizes how the two-sided p-value is calculated for an example where $y_{obs} = 7$ and N = 10. The p-value is the summed probability density of the blue bars; the probability of obtaining 7 or more heads *plus* the probability of obtaining 3 heads or less.

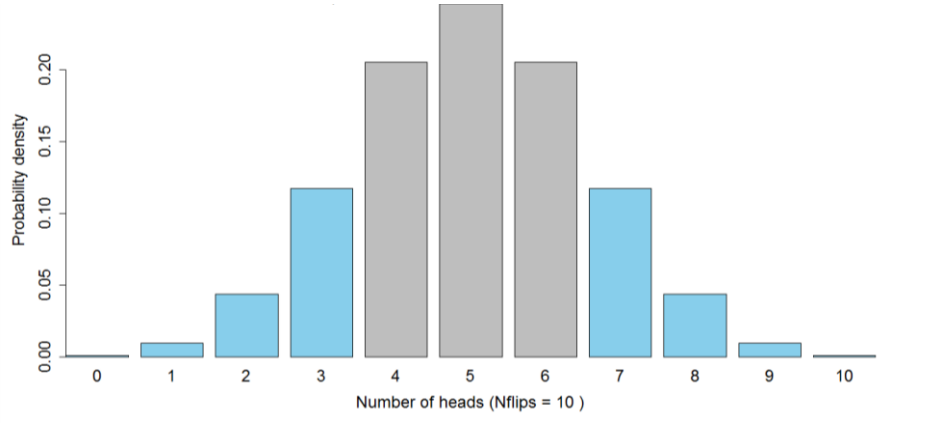


Figure 1 The p-value for y_{obs} = 7 and N=10 is the summed probability density of the blue bars; the probability of obtaining 7 or more heads: $p(y = 7) + p(y = 8) + \ ... \ + \ p(y = 10) \approx 0.1718$ plus the probability of obtaining $(N - 7)$=3 heads or less :$p(y = 3) + p(y = 2) + \ ... \ + \ p(y = 0) \approx \ 0.1718$.

If the underlying assumptions used to calculate the p-value hold, a smaller p-value means greater statistical incompatibility of the data with the null hypothesis and therefore providing more evidence to reject the null hypothesis (ASA; Wasserstein & Lazar, 2016).

Making an inference based on the p-value

In frequentist analysis, the decision rule is established by keeping the overall false alarm rate (a.k.a., Type I error rate) limited to a specified α (Kruschke & Liddell, 2018b). In NHST, this is achieved by specifying the significance threshold α (Bayarri et al., 2016). The significance

threshold is the probability under the null hypothesis that the test statistic falls in the rejection region (the Type I error probability). In practice, the significance threshold is fixed, typically at α = 0.05. A p-value smaller or equal to α indicates there is enough evidence to reject the null and a p-value larger than α indicates there is not enough evidence in the data to reject the null (Mulder & Wagenmakers, 2016).

$$p \leq \alpha : H_1$$

$$p > \alpha : H_0$$

Proportion false and true positives p-value

Setting the significant level at α (typically α = 0.05) assumes that rejecting the null will result in a $\alpha \times 100\%$ Type I error (Jeon & De Boeck, 2017). Therefore, the proportion false positives are given by $P(p \leq \alpha \mid H_0) = \alpha$.

Setting the significance level at α pins down the probability of obtaining a true-positive result, given by $P(p \leq \alpha \mid H_1)$ (Bayarri et al., 2016).

CI+ROPE Decision Rule

In the frequentist hypothesis testing framework, it is statistically impossible to support the hypothesis that a true effect size is exactly zero (Lakens, 2017). However, it is possible to assert that the true difference is unlikely to be outside a certain range by testing for equivalence (Jones, Jarvis, Lewis, & Ebbutt, 1996). Therefore, in contrast with NHST, the null hypothesis is not fixed at a specific point but set at a specified interval (l,u). Such that

$$H_0: l \leq \theta \leq u$$

$$H_1: l > \theta \text{ or } u < \theta$$

Combining a 'range of equivalence' with confidence intervals is a very simple equivalence testing approach.

Confidence intervals

According to Bolstad & Curran (2016), CIs are used by frequentists to find an interval that has a high probability of containing the true value of the parameter θ. In the long run, (1 − α) × 100% of CIs will include the true value of the parameter θ and an unidentified 5% will miss. The reasoning is that one CI most likely includes the true value of the parameter θ, but it might not.

Calculating CIs for a binomial parameter

When the sampling distribution of the estimator used is approximately normal, with mean equal to the true value, the CI for the estimation of the binomial parameter θ would be

$$CI_{(1-2\alpha)*100} = \hat{\theta} \pm z_{\alpha/2} \times \sqrt{\frac{\hat{\theta}(1-\hat{\theta})}{n}} \text{ where } \hat{\theta} = \frac{y}{n} \quad (3)$$

(W. M. Bolstad & Curran, 2016).

Specifying a ROPE

Based on what the researcher indicates as the smallest relevant difference between the hypothesized θ and the true θ, an upper (u) and lower (l) equivalence bound is specified (Lakens, 2017). The equivalence bounds can be based on either standardized differences (e.g. ± Cohen's d = 3) or raw scores on a scale point (e.g. ± 0.1). The interval between the two equivalence bounds knows many different names in literature: 'ROPE (region of practical equivalence)', 'null region', 'range of equivalence', 'equivalence interval', 'indifference zone', 'smallest effect size of interest' and 'good-enough belt' (Dienes, 2016; Jones et al., 1996; Kruschke & Liddell, 2018b).

In the example of a coin, we would ask ourselves when would we label the coin as being 'unfair'. If we want to assess whether the chance of getting 'heads' when flipping the coin is approximately 0.5, would we really bother when the chance of getting heads is 0.489 or 0.512? For example, we might judge these proportions as practically equivalent to 0.5 and define our region of practical equivalence as [0.45 0.55] meaning that we are only interested to see whether the chance of getting heads is either more than 55% or less than 45%.

Making an inference based on the CI+ROPE decision rule

When the CI is contained within the null region, the null region hypothesis can be accepted: we can conclude that θ falls within the equivalence bounds ($l \leq \theta \leq u$), meaning that the difference is not large enough to be relevant (Dienes, 2016). The H_1 can be accepted when the interval is entirely outside the null region ($l > \theta$ or $u < \theta$). If the interval falls neither fully inside or outside the null region, but spans both the null region and regions outside, the data do not discriminate between H_0 and H_1:

$$CI \subseteq ROPE : \mathrm{H}_0$$

$$CI \cap ROPE = \emptyset : \mathrm{H}_1$$

$$otherwise : \text{no decision}$$

Proportion false and true positives CI+ROPE

A false positive would be when the CI is outside the ROPE, even though the true θ is inside the null hypothesis interval $[l, u]$; $\alpha = P(CI_{(1-2\alpha)*100} \cap ROPE \mid l \leq \theta \leq u)$.
If a $(1 - 2\alpha) * 100\%$ CI is used to decide on equivalence, then the probability of the Type I error is α (Jones et al., 1996). So, for example, if a 95% interval is used, then α = 0.025.

A true positive would be when the CI is outside the ROPE when indeed the alternative hypothesis is true; given by $P(CI_{(1-2\alpha)*100} \cap ROPE | \, l > \theta$ or $u < \theta)$.

Bayesian methods

Bayesian approaches have gained attention as an alternative method to NHST (Jeon & De Boeck, 2017). The question asked by Bayesians when testing the coin would be "*Given the observed number of heads, which parameter values are most likely when we also take the prior information into account?*". The Bayes' rule derived from definitions of conditional probability is designed to answer this question (Kruschke, 2015). The factors of Bayes' rule are likelihood, prior and evidence:

$$p(\theta|data) = \frac{p(data|\theta) \times p(\theta)}{p(data)} \quad or \ posterior = \frac{likelihood * prior}{evidence} \tag{4}$$

Bayesians reason that the posterior gives an indication of how strongly we should have trust in the various parameter values, given the data. How the Bayes' rule is derived exactly, can be found in Appendix 3.

Bayesian model comparison

In Bayesian model comparison, the analyst sets up two competing models in the form of density functions which distribute the prior probabilities among all possible outcomes (Y) differently (Kruschke, 2011). When the data is obtained, the analyst calculates which model is more credible given the data.

Suppose we have two models; M_1 and M_0. Application of the Bayes' rule (equation 4) yields that $P(M_1|D) = \frac{p(D|M_1)P(M_1)}{P(D)}$ and $P(M_0|D) = \frac{p(D|M_0)P(M_0)}{P(D)}$. By taking the ratio of the two expressions above, the common denominator $P(D)$ drops out, resulting in the next formula:

$$\frac{P(M_1|D)}{P(M_0|D)} = \frac{p(D|M_1)}{p(D|M_0)} \times \frac{P(M_1)}{P(M_0)} \tag{5}$$

This equation specifies that, after observing the data, the posterior odds in favor of M_1 versus M_0 ($\frac{p(M_1|D)}{p(M_0|D)}$) is the ratio of the probability of the data given one model, relative to the probability of the data given a second model ($\frac{p(D|M_1)}{p(D|M_0)}$; also denoted as likelihood ratio), times the ratio of the prior beliefs ($\frac{P(M_1)}{P(M_0)}$). The likelihood ratio is the Bayes factor. The Bayes factor is one of the most popular Bayesian methods in selecting which of the two hypotheses fits the data better (Garc & Chen, 2005). It is common to use subscript on Bayes factors to refer to the models being compared (Rouder, Morey, Speckman, & Province, 2012). The first subscript refers to the model in the numerator and the second subscript refers to the model in the denominator such that:

$$BF_{10} = \frac{p(D|M_1)}{p(D|M_0)} = \frac{1}{BF_{01}} \quad (6)$$

Conceptually, the Bayes factor is simple; it is the ratio of the probabilities of the observed data under the two hypotheses (Morey, Romeijn, & Rouder, 2016). However, it is important to realize that the Bayes factor gives an indication of which model is a better predictor for the obtained data; it does not depend on one of the models being true (See & Cohen, 2007).

Bayesian hypothesis test using Bayes factors

From a Bayesian perspective, the fact that the null hypothesis is unlikely is not a sufficient reason to reject the null hypothesis as the data may be even more unlikely under the alternative hypothesis (Morey et al., 2016). The question asked when testing the null hypothesis that the coin is fair using the Bayes factor would be "*Is the observed number of heads more likely to be obtained under the alternative hypothesis than under the null hypothesis?*".

In Bayesian hypothesis testing using Bayes factors, the Bayes factor indicates the credibility of a particular alternative hypothesis relative to the null hypothesis(Morey et al., 2016). In Bayesian hypothesis testing, the null hypothesis and alternative hypothesis correspond

to two models; M_0 and M_1 respectively. Therefore, unlike NHST, for a Bayesian hypothesis test, an alternative hypothesis must be specified in the form of a prior distribution on the parameter (Kruschke & Liddell, 2018b). For H_0, the corresponding prior probability density function would place all the credibility on $\theta = 0.5$ and zero on every other θ. The null hypothesis is compared against an alternative prior distribution that spreads prior credibility over other values of the parameter.

For binomial observations, the alternative prior is usually specified as a beta distribution because the beta distribution is conjugate to the binomial distribution, which makes calculations easier (Bolstad, 2016).

The marginal likelihood for M_0 (i.e. the probability of the data given M_0) is given by the conditional probability function for y given $\theta = 0.5$ (equation 1). The marginal likelihood for H_1 is more difficult to calculate, because in H_1 θ is not set to one value but is located within the interval [0 1] (Wagenmakers, Lodewyckx, Kuriyal, & Grasman, 2010). In general, the marginal likelihood for H_1 is obtained by: $p(D|M_1) = \int p(D|\theta, M_1)p(\theta|M_1)d\theta$ (Wagenmakers et al., 2010). However, when we choose the alternative hypothesis to be a probability density function that places equal probability to every value of θ over the parameter space Θ such that $p(\theta|M_1) \sim Beta(1,1)$, the marginal likelihood for M_1 famously simplifies to $p(D|M_1) = 1/(n+1)$. The marginal likelihoods for M_0 and M_1 for an example where N = 20 are plotted as respectively blue and grey vertical bars in figure 2. The ratio of the marginal likelihoods of M_1 and M_0 (BF_{10}) is plotted as red dots.

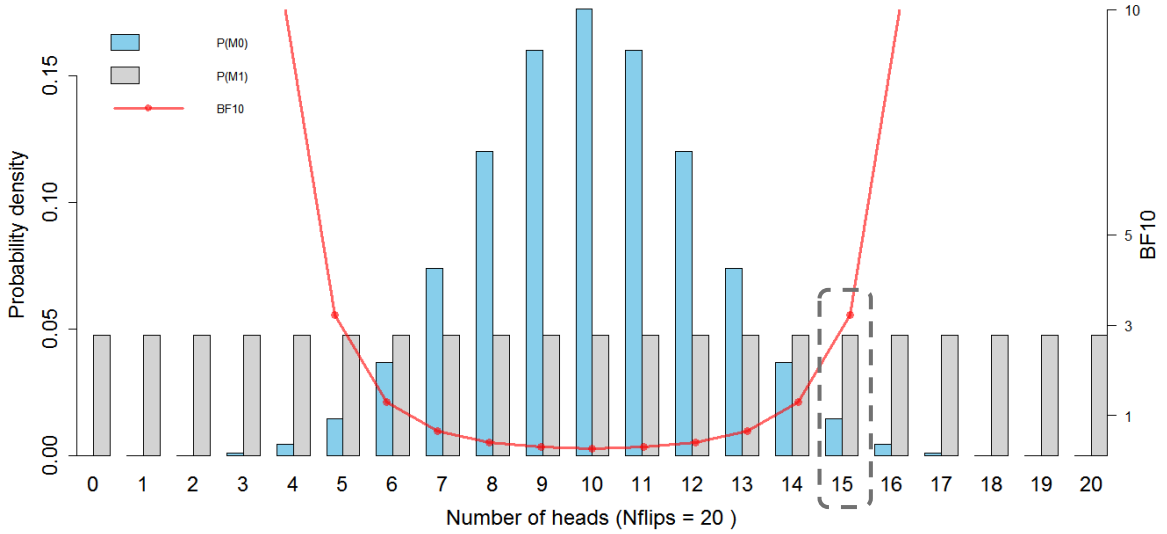


Figure 2 The marginal likelihoods for M_0 (blue bars) and M_1 (grey bars) for an example where N=20. The Bayes factor (BF10) , the ratio of the two marginal likelihoods is plotted in red

To illustrate obtaining the Bayes Factor in the case of null hypothesis testing, suppose we have flipped a coin 20 times and we obtained 15 heads. The marginal likelihood for M_0 would be: $p(y = 15 \mid n = 20, M_0) = \binom{20}{15} 0.5^{15} (1 - 0.5)^{20-15} = 0.0147857$. The marginal likelihood for M_1 would be $p(y = 15 \mid n = 20, M_1) = \frac{1}{20+1} = 0.0476190$. The corresponding Bayes factor would be $BF_{10} = \frac{p(y=15|n=20,H_1)}{p(y=15|n=20,H_0)} \approx \frac{0.0147857}{0.0019991} \approx 3.2$. By looking at "number of heads = 15" at the x-axis in figure 2 (denoted with the dashed lining) we could have gotten a good estimation of the BF_{10} as the light grey bar a little bit more than 3 times as tall as the light blue bar. According to Bayesian inference, this indicates that the data are about 3.2 times as likely to have occurred under H_1 than under H_0 (See & Cohen, 2007).

Importantly, notice that the BF does not indicate the posterior odds of the two hypotheses as it ignores the prior odds ratio (i.e. how likely is H_1 versus how likely is H_0 before the data is observed, not to be confused with the prior probability density of the two models) (Kruschke & Liddell, 2018b). If the alternative hypothesis has a minuscule prior probability (for example we

know that 99% of the coins in circulation is fair and 1% is unfair), then BF_{10} must be enormous to compensate and a posterior probability that favors the null. The posterior probabilities of the models can be obtained by multiplying the BF by the prior odds ratio (see equation 5) (Kruschke, 2018b). Note that when the prior odds of the models are 50/50, then the Bayes factor numerically equals the posterior odds. Ultimately, Bayesians are interested in the posterior odds of the two models, but by reporting the BF, readers can use their own prior odds to determine the posterior odds.

From equation (5) follows that BF_{10} indicates how much odds of the models have shifted from prior to posterior (John K Kruschke, 2018b). Therefore, a second way of calculating the Bayes factor is by taking the ratio of the posterior odds to the prior odds.

Making an inference based on the BF

According to the common decision rule for Bayesian null-hypothesis, the Bayes factor is compared against a decision threshold BF_{crit} (Kruschke & Liddell, 2018b). When $BF_{01} > BF_{crit}$, the null hypothesis is accepted. Conversely, when $BF_{01} < 1/BF_{crit}$, the alternative hypothesis is accepted. The BF can also be defined with respect to the alternative hypothesis BF_{10}, which is simply the reciprocal of BF_{01}: $BF_{10} = \frac{1}{BF_{01}}$ (Kruschke, 2018b).

$$BF_{10} > BF_{crit} : H_1$$

$$BF_{10} < 1/BF_{crit} : H_0$$

$$otherwise : no\ decision$$

The choice of the decision threshold is set by practical considerations, just like the decision threshold of the p-value (Kruschke & Liddell, 2018a). A BF between 3 and 10 is supposed to indicate “moderate” evidence; a Bayes factor between 10 and 30 indicates “strong”

evidence and a Bayes factor greater than 30 indicates "very strong" evidence for the winning model.

False-alarm and true-positive rate

For the Bayes factor, the proportion false-alarms when making an inference would be $P(BF_{10} > BF_{krit} \mid H_0)$ which is equivalent to $P(BF_{01} < 1/BF_{krit} \mid H_0)$. In contrast to the frequentist NHST approach, where the false-positive rate is insensitive to sample size, the false-positive rate of the Bayes factor decreases when the sample size becomes higher (Jeon & De Boeck, 2017). Besides sample size, effect size and BF_{crit}, the false-positive rate also depends on how M_1 is specified (i.e. which outcomes are assigned to be more probable given that the alternative hypothesis is true) (Morey et al., 2016).

For the Bayes factor, the proportion true positives when making an inference would be $P(BF_{10} > BF_{crit} \mid H_1)$ which is equivalent to $P(BF_{01} < 1/BF_{crit} \mid H_1)$. The power increases in a monotonic way as a function of sample size, effect size and BF_{crit} and also depends on the prior probability distribution of the alternative hypothesis (Jeon & De Boeck, 2017; Morey et al., 2016).

HDI+ROPE

In a Bayesian framework, a probability distribution across the space of parameter values is specified to represent the uncertainty in parameter values (Kruschke, 2018a). The Bayesian inference is merely the re-allocation of this uncertainty in parameter values, or "updating the probability distribution", according to the mathematics of conditional probability. The result is a posterior probability distribution across all possible parameter values. From the posterior distribution on the parameter we can assess what the 95% most credible values are (Kruschke & Liddell, 2018a). This interval is called the 95% HDI. Analogous to the frequentist method of CI+ROPE, Bayesians can make an inference about the null hypothesis based on the range of most credible parameter values in combination with a region of practical equivalence (ROPE). The question asked when testing the coin using the HDI+ROPE approach would be "*Given the observed number of heads, are the 95% most probable parameter values (when taking prior information into account) inside or outside the ROPE, or neither?*". The 95% HDI is obtained from the posterior distribution. The next paragraph will briefly explain how the posterior distribution is calculated.

Posterior distribution

The posterior distribution is a probability density among all the possible parameter values (Kruschke, 2018a). Recall from equation (4) that the posterior can be obtained from the likelihood $p(D|\theta)$, prior $p(\theta)$ and evidence $p(D)$.

The likelihood is closely linked to the conditional probability function for y given a fixed θ or the binomial likelihood function (see equation 1). If we look at this same relationship between θ and y, but we hold y fixed at the number of heads we obtained by flipping the coin N times, and if we let θ vary over its possible values, we have the likelihood function given by:

$$p(D|\theta) = f(y \mid n, \theta) = \binom{n}{y} \theta^{y} (1-\theta)^{n-y} \text{ for } 0 \leq \theta \leq 1 \quad (7)$$

(Bolstad, 2016)

The relationship between θ and y remains the same as in equation (1), but now y is fixed at the value that occurred and the subject of the formula has changed to the parameter. The binomial likelihood function as a function of θ (equation 7) has the same form as a *Beta*(a,b) distribution: a product of θ to a power times (1-θ) to another power (Bolstad, 2016).

The prior $p(\theta)$ can be specified as a *Beta*(a,b) distribution (Kruschke, 2010). For example, the a and b in the beta prior can be thought of as if they were previously observed data in which there were a heads and b tails. Four examples of different beta priors are visualized in figure 3.

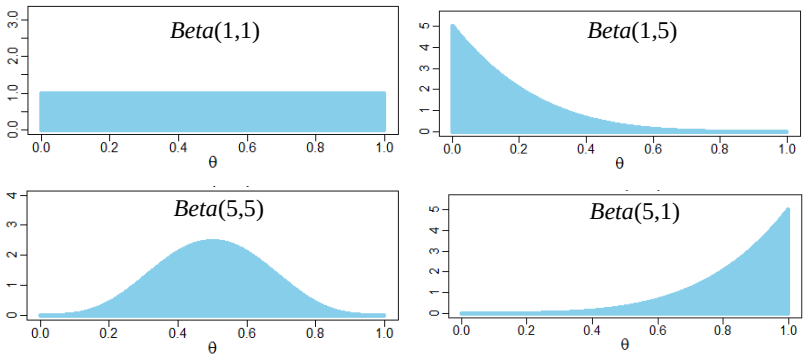


Figure 3: When we only know that the coin has a head side and a tail side, this would tantamount to having previously observed one head and one tail; corresponding to a Beta(1,1) prior. When we have previously observed 5 heads and 1 tail, this would result in a Beta(5,1) prior; placing more credibility on higher θ values. Conversely, a Beta(1,5) prior places more credibility on lower θ values. When we have previously observed 5 heads and 5 tails, the highest credibility is placed at θ = 0.5 while higher and lower values of θ are moderately probable too.

The beauty of using a *Beta*(a,b) distribution as a prior is that, when the beta prior is multiplied with the binomial likelihood, the exponents of θ and (1-θ) of the binomial likelihood can simply be added to the beta prior to obtain the beta posterior (Bolstad, 2016). There is no need for integration or complicated calculations to find the posterior. When the prior distribution

is *Beta*(θ, a, b), and the data have y heads in N flips, then the posterior distribution is *Beta*(θ; y + a, n-y + b). The math behind this simple rule can be found in Appendix 4.

When a and b would be the previously observed number of heads and tails respectively, y + a would be the number of heads of the two studies in total while n-y + b would be the number of tails of the two studies in total, thereby simply updating the previous knowledge with the new results. For this reason Kruschke (2010) reasons that "*consensually informed prior distributions permit cumulative scientific knowledge to rationally affect conclusions drawn from new observations*".

HDI

The 95% High Density Interval (HDI) is the collection of all the parameter values with the highest probability density in the posterior distribution, such that the total probability of values in the 95% HDI is 95% (Kruschke, 2018a). The width of the HDI indicates the uncertainty about the parameter value.

ROPE

The second key ingredient in the decision method is a range of parameter values that is good enough for practical purposes (Kruschke, 2018a). This procedure is equivalent to the procedure of specifying the region of practical equivalence as used by frequentist equivalence testing using CIs.

Making an inference based on the HDI+ROPE decision rule

The decision rule itself is analogous to the decision rule of the frequentist equivalence testing method using CIs. If the 95% HDI of the parameter distribution falls completely outside the ROPE, the null value is rejected because the 95% most credible values of the parameter are all not practically equivalent to the null value (Kruschke, 2018a). The null value is accepted for practical purposes when the 95% HDI of the parameter distribution falls completely inside the ROPE, because the 95% most credible values of the parameter are all practically equivalent to the null value. Otherwise, when the 95% HDI is neither of both, some of the most credible values are practically equivalent to the null while other of the most credible values are not. In these cases no decision is made:

$$HDI \subseteq ROPE : H_0$$

$$HDI \cap ROPE = \emptyset : H_1$$

$$otherwise : \text{no decision}$$

False positives rate and true positives rate

A false positive would be when the HDI is entirely outside the ROPE even though the true θ is inside the null hypothesis interval $[l, u]$, which is given by $P(HDI \cap ROPE = \emptyset \mid l \leq \theta \leq u)$. Likewise, a true positive would be the probability that the HDI is outside the ROPE when indeed the alternative hypothesis is true, which is given by $P(HDI \cap ROPE = \emptyset \mid l > \theta \text{ or } u < \theta)$.

Simulation study

The goal of the simulation study is assessing the decision qualities of the two frequentist and Bayesian methods by comparing their ROC curves under different conditions (i.e. different effect sizes, different sample sizes and different (alternative) priors of the Bayesian methods).

Methods

Programs, packages, and functions used

The experiment was executed in the R software (version 3.4.3. for Windows, R Core Team, 2017), using RStudio (version 1.1.423 for Windows, RStudio Team, 2016). Besides using the native functions implemented in the R software, I downloaded the package Hmisc (version 4.1-1., F.E. Harrell, 2018) to calculate binomial confidence intervals and the function "HDIofICDF" to calculate the Bayesian High Density Intervals. The function HDIofICDF is provided by Kruschke on his website, in the DBDA2EPROGRAMS.ZIP file (Kruschke, 2016).

Computing the test statistics & making inferences

The R-codes for calculating a two-tailed p-value, the Bayes factor (BF_{10}) and High Density Interval are adopted from Kruschke (2015). The (Wilson) CIs are calculated using the "binconf" function in the Hmisc package (Harrell, 2018). For the p-value, the alternative hypothesis was accepted when $p \leq \alpha$. For the Bayes factor, the alternative hypothesis was accepted when $BF_{10} > BFcrit$. For the CI+ROPE method, the alternative hypothesis was accepted when the entire 95%CI was outside the specified ROPE interval. For the HDI+ROPE method, the alternative hypothesis was accepted when the entire 95%HDI was outside the specified ROPE interval.

Settings & Data generation

One thousand samples were generated for every condition (sample size * effect size) by using two big for-loops. Each sample consisted of 0's and 1's generated under the true parameter value, indicated by the effect size, with as many elements as indicated by the sample size. The different sample sizes computed were: 20, 30, 40, 50, 60, 70, 80, 90, 100, 150, 200, 250, 300 and 400. The different effect sizes (in Cohen's h) simulated were: 0, 0.2, 0.3, 0.4, 0.5 which corresponds with a 'true θ' of respectively 0.5000, 0.5990, 0.6478, 0.6947, 0.7397. See appendix 4 why Cohen's h is an appropriate effect size for the difference in proportions and how the Cohen's h is computed. In total, the experiment had (14*5) 70 different conditions and 70 thousand samples. The R script used for the experiments can be found in Appendix 6.

The ROPE was specified according to Kruschke's advice (2018a); set the limits by ±1/2 times what would be a small effect according to Cohen when there is no way to specify ROPE limits by their real-world consequences. Because we might say that an effect is practically equivalent to zero if it is less than half of a small effect (Kruschke, 2018a). According to Cohen, a small effect is Cohen's h = 0.2 (Cohen, 1988). Therefore, for the CI+ROPE and HDI+ROPE method, the ROPE limits were set corresponding to an effect size of ± Cohen's h = 0.1. This corresponds with an interval of [0.4501, 0.5499]. For the CI+ROPE and HDI+ROPE criteria, the test statistic and corresponding decision about the null hypothesis and alternative hypothesis based on the obtained sample was calculated.

For the p-value and Bayes factor, for every sample, the obtained test statistic for the p-value and Bayes factor was compared with various specified α and critical Bayes factors respectively. The corresponding decision for every threshold was saved temporarily. The different decision threshold for the p-value (α's) were: 0.05, 0.025, 0.02, 0.015, 0.01, 0.005,

0.001, 0.0005, 0.0002, 0.0001. The critical Bayes factors: 1.5, 2, 3, 4, 5, 6, 7, 8, 10, 15, 30, 50, 80, 100.

For every condition, the proportion false alarms (when Cohen's h = 0) or true positives (when Cohen's h > 0) for the CI+ROPE and HDI+ROPE criteria and the different decision thresholds for the p-value and Bayes factors were calculated and saved.

This entire procedure was executed 4 times with different prior distributions (M_1 for the Bayes factor) *Beta*(a,b) for the Bayes factor and HDI in order to assess the decision qualities for prior distributions that place higher credibility in different directions. In experiment 1, a uniform prior distribution was used; *Beta*(1,1). The settings for the second, third and fourth experiment were *Beta*(1,5), *Beta*(5,1) and *Beta*(5,5) respectively. These priors correspond with the four situations as previously described and visualized in figure 3: placing credibility equally among al θ's, placing more credibility among higher θ's, placing more credibility among lower θ's and placing more credibility among θ's around θ = 0.5 respectively.

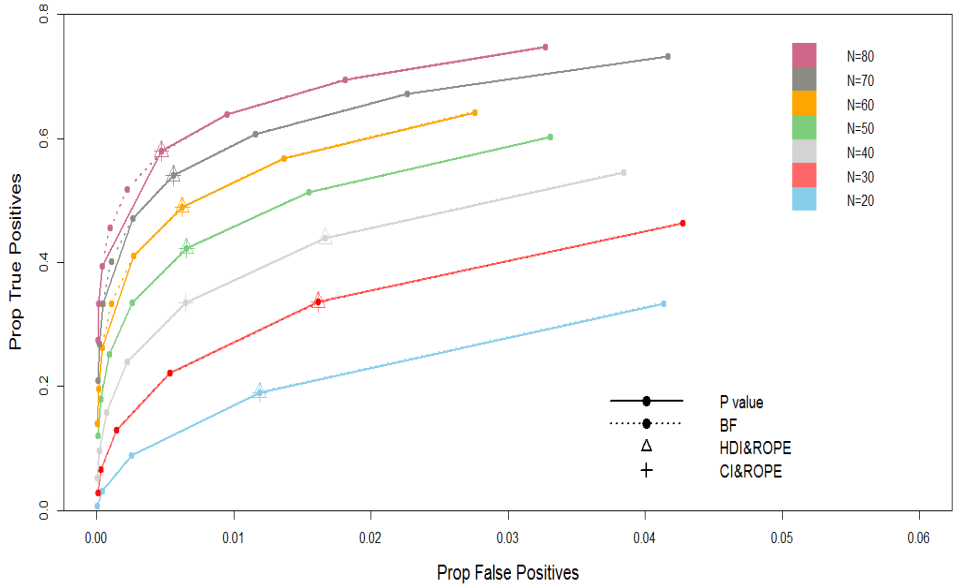
Data visualization

For each experiment and each different effect size separately, the proportion true positives against the false positives for the p-value and Bayes factor among various cut-offs and different sample sizes are plotted in a graph. Plotting all the sample sizes in one graph becomes very crowded and the information provided by the sample sizes higher than 80 did not add new information. Therefore, only the sample sizes 20 up to and including 80 are included in the following plots. The trade-off between the obtained false positives and true positives of the CI+ROPE and HDI+ROPE criteria are visualized as symbols in the graph. The different plots for the different effect sizes yielded the same information in terms of relationship between the four methods. Therefore, proportions false positives and proportions true positives are averaged and

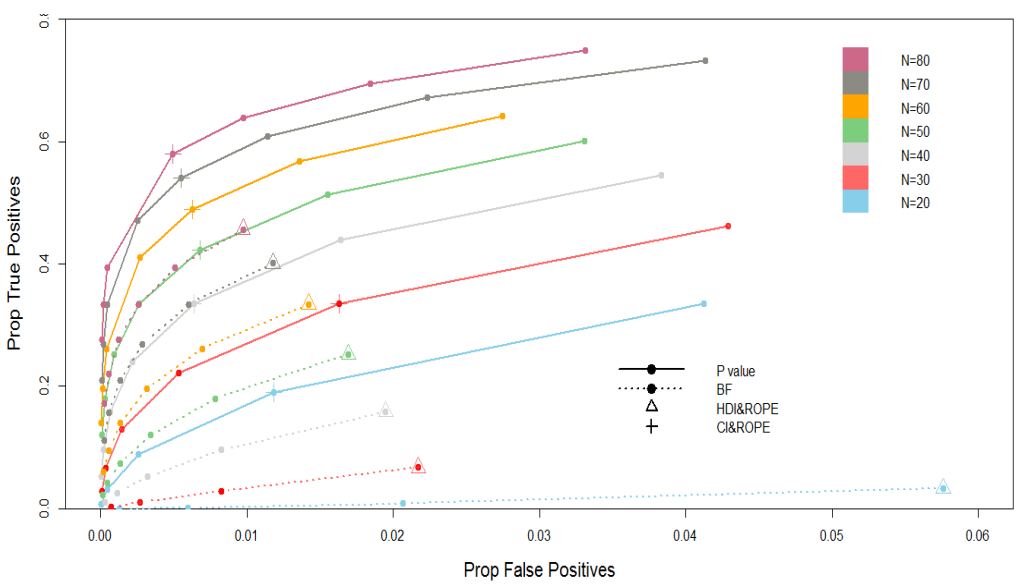
plotted in one graph for a better overview of the results. This results in four graphs; one for every experiment. They are reported under results. The plots that display the effect sizes separately are included in Appendix 1.

Results

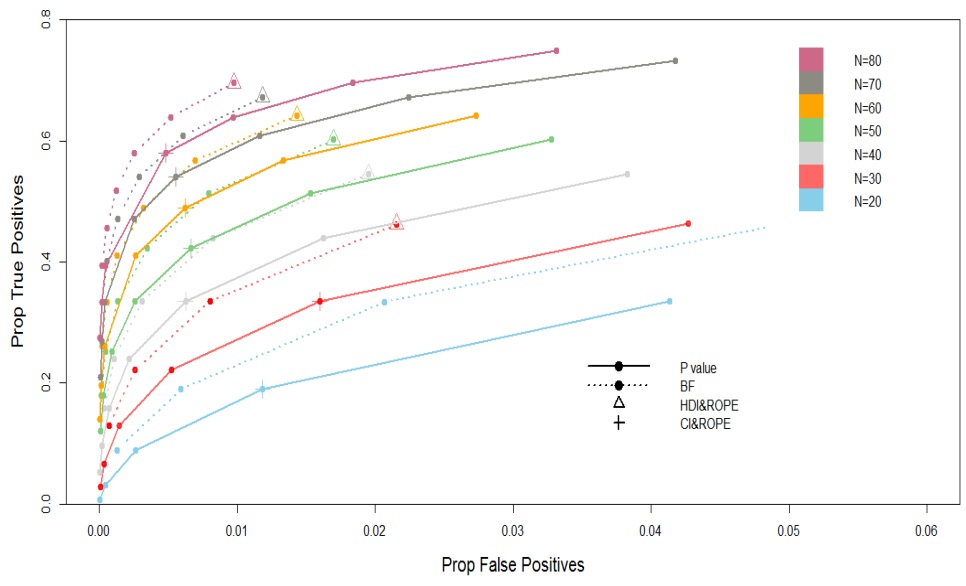
Experiment 1: *beta*(1,1)



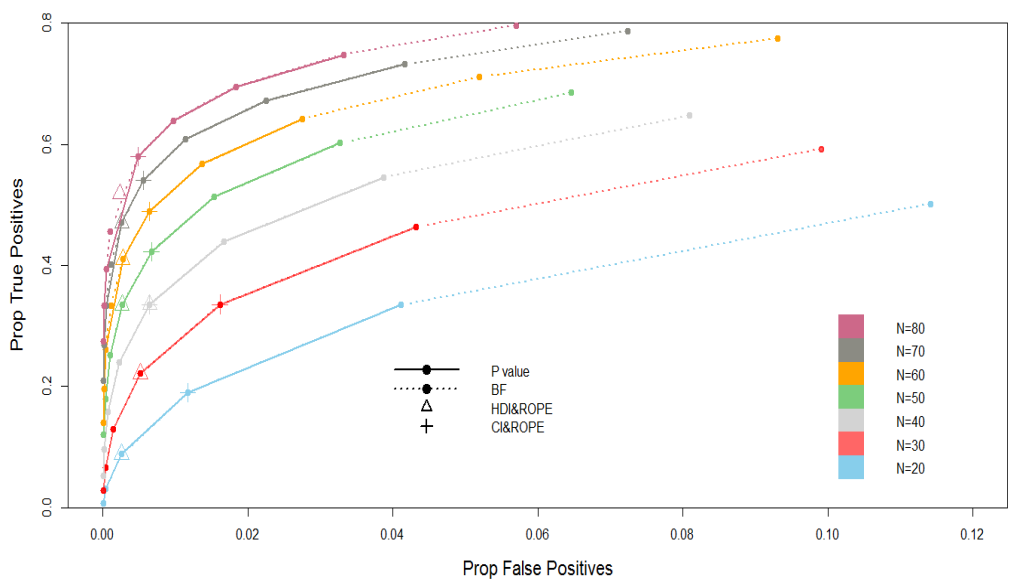
Experiment 2: *beta*(1,5)



Experiment 3: *beta*(5,1)



Experiment 4: *beta*(5,5)



General result:

In every experiment, at one point on the ROC curve of the Bayes factor, the decision threshold for the Bayes factor results in equal proportion false positives and true positives as the (95%) HDI and ROPE criterium (the ▲ in the plots). Likewise, in every experiment, at one point on the ROC curve of the p-value, the decision threshold for the p-value (α) results in equal proportion false positives and true positives as the (95%) CI and ROPE criterium.

Experiment 1: *Beta*(1,1)

In this setting, the p-value and Bayes factor yield equal ROC curves and therefore have equal overall accuracy.

The CI+ROPE and HDI+ROPE criteria have equal proportion false positives and proportion true positives in all sample sizes except for N=40. This holds for every effect size (as can be inspected from the original plots in Appendix 1). Out of curiosity, a follow-up study is conducted with the same settings as the original experiment, except that the sample size N is set at N=20 up until N=60 with steps of 1. Appendix 2 shows the resulting plot for the ROPE approaches. Of the 41 different sample sizes, the CI+ROPE and HDI+ROPE have unequal LR+'s 9 times, without any clear pattern dictating which sample sizes do or do not overlap. Remarkably, when the LR+'s are not equal, the HDI+ROPE approaches are less conservative, yielding higher false positive and true positive rates. This indicates that the 95% HDI must be smaller than the 95% CI (as the same ROPEs were used).

Experiment 2: *Beta*(1,5)

The ROC curves of the p-value are closer to the upper left-hand corner in comparison with the ROC curves of the Bayes factor. Therefore, the results yield that under these

circumstances the p-value is overall more accurate; it has a better trade-off between the proportion true positives and false positives.

Experiment 3: *Beta*(5,1)

The ROC curves of the Bayes factor are closer to the upper left-hand corner in comparison with the ROC curves of the p-value. Therefore, the results yield that in these circumstances, the Bayes factor is overall more accurate; it has a better trade-off between the proportion true positives and false positives.

Experiment 4: *Beta*(5,5)

In this setting, the p-value and Bayes factor yield equal ROC curves and therefore have an equal overall accuracy again. Although in contrast with experiment 1, the x-axis is much wider as the curves for the Bayes factor now reach higher false positive rates and higher true positive rates. In contrast with experiment 1, the CI+ROPE and HDI+ROPE do not have equal false positive rates and true positive rates except for N=40, where they do have the same false positive and true positive rate. The latter is exactly the opposite from what was observed in experiment 1.

Theoretical explanation for the observed results

Introducing $\mathbb{R}_y$, RL_y and RU_y

The results can be explained by how every method is obtained, as described in the previous chapter. However, in order to explain the results from the four experiments, I first introduce an ordering of the sampling space S (which contains all the possible number of heads given N) that can be applied to all four tests to establish notation; seeking a tool to compare the four methods in equal terms and to discover similarities and differences between the methods that explain the obtained results.

The first step is to realize that, for all the four methods, the obtained number of heads y completely determines the inference that will be made for every method. Before flipping the coin, when we know how many times we are going to flip the coin (N) and we have specified the decision thresholds by specifying α, BF_{crit}, $CI_{(1-2\alpha)*100}$, $HDI_{x\%}$ and ROPE, we can assess beforehand whether we will reject the null hypothesis for every $y \in Y$. Therefore, before flipping the coin, we can already specify a rejection region $\mathbb{R}$, the acceptance region $\mathcal{A}$ and a no decision region $\mathbb{N}$ (which is empty for the p-value) in terms of y for every method. Such that $\mathcal{A}_y \cup \mathbb{N}_y \cup \mathbb{R}_y = Y$ and $\mathcal{A}_y \cap \mathbb{N}_y \cap \mathbb{R}_y = \emptyset$**.** For a two-sided test, the rejection region consists of both low and high y's ($y > \frac{N}{2}$) and low y's ($y < \frac{N}{2}$). For a two-sided test we can split $\mathbb{R}_y$ into two subsets: $RL_y \cup RU_y = \mathbb{R}_y$, $RL_y \cap RU_y = \emptyset$ such that

$$RL_y \ni \{ y < \frac{N}{2}, \ y \in \mathbb{R}_y \}$$

$$RU_y \ni \{ y > \frac{N}{2}, \ y \in \mathbb{R}_y \}$$

The relationship between $\mathbb{R}_y$, RL_y and RU_y and false positives and true positives

By definition, the proportion false positives is $P(y \in \mathbb{R}_y | H_0)$ and the proportion true positives $P(y \in \mathbb{R}_y | H_1)$.

It is common knowledge that in frequentist NSHT a one-sided test can offer a large gain in power over the corresponding two-sided test (Neuhävser, 2004). For the total proportion false positives (Ry) it makes no difference whether RL_y is bigger than RU_y, or vice versa. However, when H_1 is true with $\theta > 0.5$, the chances of obtaining a low number of heads such that $y \in RL_y$ are much smaller than obtaining a high number of heads such that $y \in RU_y$. For this situation, given that the size of $\mathbb{R}_y$ is set, the test would have a maximum power when $RU_y = \mathbb{R}_y$ (a one-sided test in the right direction) and zero power when $RL_y = \mathbb{R}_y$ (a one-sided test in the wrong direction). The opposite is true when H_1 is true with $\theta < 0.5$. For this situation, given that the size of $\mathbb{R}_y$ is set, the test would have a maximum power when $RL_y = \mathbb{R}_y$ (a one-sided test in the right direction) and zero power when $RU_y = \mathbb{R}_y$ (a one-sided test in the wrong direction).

In sum, the proportion false positives does not depend on either the sizes of RL_y or RU_y separately. Instead, only the size of the two sets taken together ($\mathbb{R}_y$) determines the proportion false positives. But, when the alternative hypothesis is true, the proportion true positives is (depending on the direction of the true effect) determined by either the RL_y or RU_y.

In the next paragraphs, the "symmetry" of $\mathbb{R}_y$ (are RL_y and RU_y of equal length?) for each of the four methods will be discussed. This "symmetry" will be key in explaining the observed results of the simulation experiment.

R_y, RL_y and RU_y for the p-value

The size of R_y for the frequentist methods is almost directly specified by α, such that $P(y \in R_y \mid H_0) = \alpha$. Looking at the formula for the two-sided p-value when the null hypothesis is θ=0.5 (equation 2), we see that RL_y and RU_y are symmetrical in the sense that they contain an equal number of y's. In other words, α is equally distributed among the two tails.

R_y, RL_y and RU_y for the CI & ROPE criterium

Again, the size of R_y is almost directly specified by α, such that $P(y \in R_y \mid \theta_l \leq \theta \leq \theta_u) = \alpha$. Recalling the formula for calculating the CI (equation 3), we can see that a confidence interval is symmetrical in the sense that the interval between $\frac{y}{n}$ and the upper limit of the CI and the interval between the lower limit of the CI and $\frac{y}{n}$ are of the same length. The CI is symmetrically centered around $\frac{y}{n}$.

In the case of our coin experiment, when the ROPE is centered at θ=0.5 and the upper and lower ROPE bounds are equally far away from θ=0.5, then RL_y and RU_y are symmetrical as well. However, when the ROPE bounds are not symmetrical this does not hold as either higher y will be more likely to provide evidence to reject the null interval than lower y or vice versa; RL_y and RU_y are not equally large.

R_y, RL_y and RU_y for the Bayes Factor

Recall that the BF is the ratio of the probabilities of the observed data under the null and alternative hypothesis. R_y is set such that

$$\frac{P(y \mid y \in R_y, H_1)}{P(y \mid y \in R_y, H_0)} \geq BF_{crit}$$

From figure 3 we can see that, when the marginal likelihood of M_0 and M_1 are symmetrically centered around θ=0.5 (plot 1 and plot 3), the red line in the plot that represent the BF10 is also symmetrical and centered around θ=0.5; otherwise it is not (plot 2). When the alternative probability distribution places more credibility on 0.5<θ, then $RL_y < RU_y$ (see for example plot 2). When the prior probability distribution places more credibility on 0.5>θ, then $RL_y > RU_y$ (see for example plot 2).

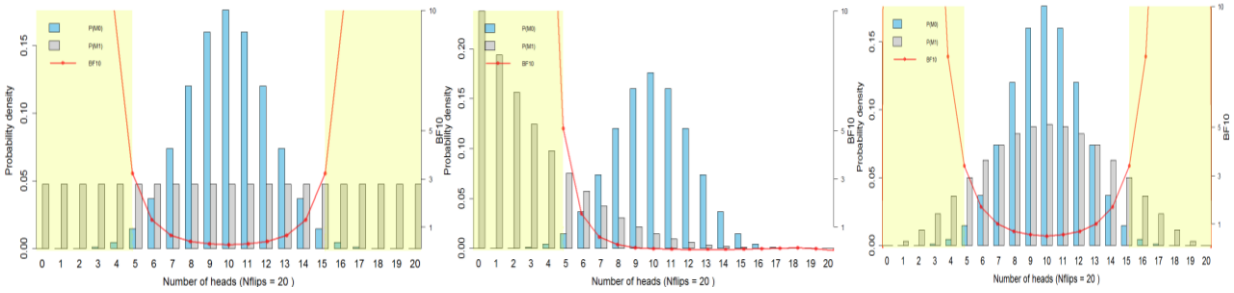


Figure 3. Three plots in which the marginal likelihoods of M_0 and M_1 are plotted and the corresponding BF's are plotted as red dots. The yellow shaded region visualizes R_y. Plot 1 displays the marginal likelihoods of M_0 and M_1 when M_1 has a uniform prior probability distribution. Plot 2 displays the marginal likelihoods of M_0 and M_1 when M_1 places more credibility on 0.5<θ. Plot 3 displays the marginal likelihoods of M_0 and M_1 when M_1 places more credibility on 0.5>θ.

R_y, RL_y and RU_y for the HDI + ROPE criterium

The 95% completely depends on the posterior distribution. Recall that when the prior distribution is *Beta*(θ|a,b), then the posterior distribution is *Beta*(θ|y+a, N-y+b). Therefore, the prior distribution determines whether the 95% HDI is symmetrical and centered at the center (θ=0.5) or not; when a = b, the posterior distribution and 95% HDI are symmetrical. As with the CI+ROPE criterium, when both the ROPE and the 95% HDI are symmetrical, the RL_y and RU_y are equally large.

Explanation result experiment I

Recall that, with a uniform prior, for all four methods, RL_y and RU_y are equally large. When we realize that R_y is fully determined by the chosen critical value for each test (and for HDI+ROPE and CI+ROPE also the ROPE interval), it becomes clear that one can adjust the critical value of a Bayesian test to match the R_y with the R_y of a frequentist test. In figure 5 this "matching" is visualized for a p-value test with $\alpha = 0.05$ and the Bayes factor with $BF_{crit} = 3$. When the R_y of a Bayesian and a frequentist test coincide, their decisions will be the same and therefore their false positives to false negatives ratio will be equal too.

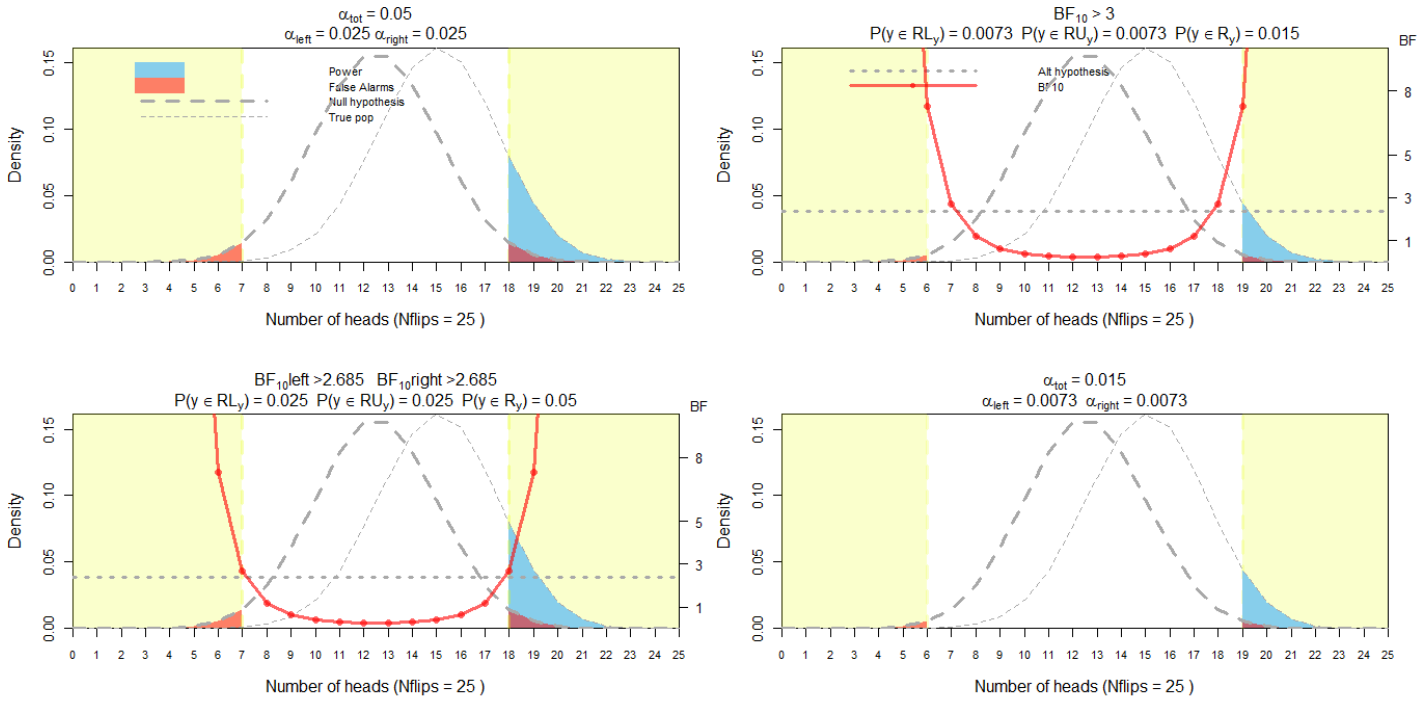


Figure 4. The first row displays the Ry (the red region) for a two-sided frequentist test with alfa = 0.05 (left) and a Bayesian null hypothesis test with BFcrit = 3 (right). The second row displays on the left the Bayesian test that would yield the same Ry as the frequentist test and on the right the alfa that would correspond with the proportion false alarms of the BF>3 test.

This explains why the ROC Curves for the BF and p-value coincide and why the observed trade-offs between true positives and false positives for the HDI+ROPE and CI+ROPE are also on the ROC curve of the Bayes factor and p-value.

As an example, table 1 displays the critical Bayes factor that would yield equal rejection regions (and therefore equal false and true positives) among different sample sizes. Here we can see the convergence between the α, that remains the same over all sample sizes, and the BFcrit, that increases with the sample size; as described by the Lindley's Paradox (Hubbard & Lindsay, 2008; Sprenger, 2013). This is because the Bayes factor is a comparative method where the sample size plays a role in interaction with the effect size, while the decision thresholds of frequentist measures are independent of the sample size.

Table 1

BF_{crit} Matched to α=0.05 for Different Sample Sizes (N)

N	10	20	30	40	50	60	70	80	90	100
Bf_{crit}	9.309	3.221	2.421	2.229	2.242	2.367	1.488	1.682	1.201	1.392

Note: The corresponding BF_{crit} that would yield equal rejection regions compared to a p-value test with α = 0.05; with M_1 specified as a *Beta*(1,1) distribution

Explanation result experiment II

In experiment 2, the alternative model M_1 that is used to calculate the Bayes factor and as a prior for the HDI places more probability on lower θ values. Therefore, $RL_y > RU_y$ for the Bayes factor and HDI+ROPE method. When the four tests have an equally large R_y, the two frequentist tests yield $RL_y = RU_y$ while the two Bayesian tests (with a prior of *Beta*(1,5)) yield $RL_y > RU_y$. From this it follows that when R_y is equally large for the four tests, the proportion false positives will be equal for the four tests, while the Bayesian tests have a lower power when 0.5<θ (which is the case in the experiment) and thus have less favorable ROC curves.

Explanation result experiment III

In experiment 3, the alternative model M_1 that is used to calculate the Bayes factor and as prior for the HDI places more probability on higher θ values. Therefore, $RL_y < RU_y$ for the Bayes factor and HDI+ROPE method. When the four tests have an equally large R_y, the two frequentist tests still yield $RL_y = RU_y$ while the two Bayesian tests (with a prior of *Beta*(5,1)) yield $RL_y < RU_y$. From this follows that when R_y is equally large for the four tests, the proportion false positives will be equal for the four tests, while the Bayesian tests have a higher power when 0.5<θ (which is the case in the experiment) and therefore more favorable ROC curves.

Explanation result experiment IV

In experiment 4, the alternative model M_1, used to calculate the Bayes factor and the HDI, places more probability on the θ values around the center (θ = 0.5). Therefore, all four tests yield $RL_y = RU_y$ again. This means that every method can again be matched to have coinciding RL_y's and RU_y's by adjusting the decision thresholds. The result is equal proportions false positives and true positives, resulting in equal ROC curves.

The only difference with the result in experiment 1 is that in this experiment the observed maximum proportion false alarms and maximum proportion true positives are higher than in experiment 1. This is the result of the alternative model M_1 placing more probability on the θ's around the center. This results in higher BF_{10}'s around the center than with a uniform prior (see the third plot in figure 3) and therefore a higher sensitivity, resulting in more false positives and more true positives. For the HDI this results in smaller HDI's and therefore a higher sensitivity and more false positives and more true positives as well.

The possibility of asymmetrical rejection regions for frequentist tests

Making the rejection region for the Bayesian and frequentist tests equal is possible when the tests have symmetrical rejection regions in the sense that $f(y \in RL_y|H_0) = f(y \in RU_y|H_0)$. Then $f(y \in R_y|H_0)$ can be set equal for every test what results in equal $f(y \in R_y|H_1)$ and therefore equal $\frac{f(y \in R_y|H_1)}{f(y \in R_y|H_0)}$. The same logic would apply to one-tailed frequentist and Bayesian tests: when $RL_y = \emptyset$ or $RU_y = \emptyset$ for both approaches, their decision thresholds can again be set such that the R_y of the four tests align perfectly, yielding equal AUC's again.

In frequentist methods, traditionally, two-sided tests have symmetric rejection regions while for the BF and HDI+ROPE decision rule, in theory, rejection regions can be set however the researcher wants. For the BF this depends on how the M_1 is specified and what BF_{crit} is used. For the HDI+ROPE decision rule this depends on what prior and ROPE Is used. In experiment 2 and 3 it became clear that when an asymmetrical prior is used (and the sample size is large enough), the Bayesian methods do not have symmetric rejection regions and therefore different ROC curves than the frequentist (symmetrical) methods.

However, as Nosanchuck pointed out in 1978, nothing in theory prohibits asymmetric rejection regions for frequentist methods. He suggested a compromise between a one- and two-

tailed test in order to increase power relative to a two-tailed test, without excluding the possibility of finding unexpected effects. Rice & Gaines (1994) agreed that there is no compelling reason to consider either one- or two-tailed tests and offered guidelines to find a p-value for a "directed test (P_{dir})" by partitioning the Type-I error rate (α) into two segments α=γ+δ where $\gamma \geq \delta$ and γ is associated with the "rejection region" in the anticipated direction. The critical values for the directed frequentist test are easily found once γ/α is specified. When γ/α of a frequentist test equals $f\left(y \in RU_y \middle| H_0\right)/f(y \in \mathrm{R}_y|H_1)$ or $f\left(y \in LU_y \middle| H_0\right)/f(y \in \mathrm{R}_y|H_1)$ (depending on what the anticipated direction of the effect is) of a Bayesian test, their decision thresholds can again be set to yield equal $\frac{f(y \in \mathrm{R}_y|H_1)}{f(y \in \mathrm{R}_y|H_0)}$. *Although not tested in the simulation study, based on this rationale, one would expect that, even for asymmetrical prior probability distributions, in the example of a binomial experiment with a predetermined N, it is always possible to convert the two Bayesian tests to frequentist test that yield equal proportions false alarms and equal proportions true positives and vice versa.*

Figure 7 shows an example of the matching of a BF test where M1 is specified as a non-symmetric prior distribution *Beta*(2,1) with a p-value test with α=0.05 and vice versa. This is possible when α is not necessarily symmetrically distributed and the Bayes factor can have two different decision thresholds depending on the direction of the result.

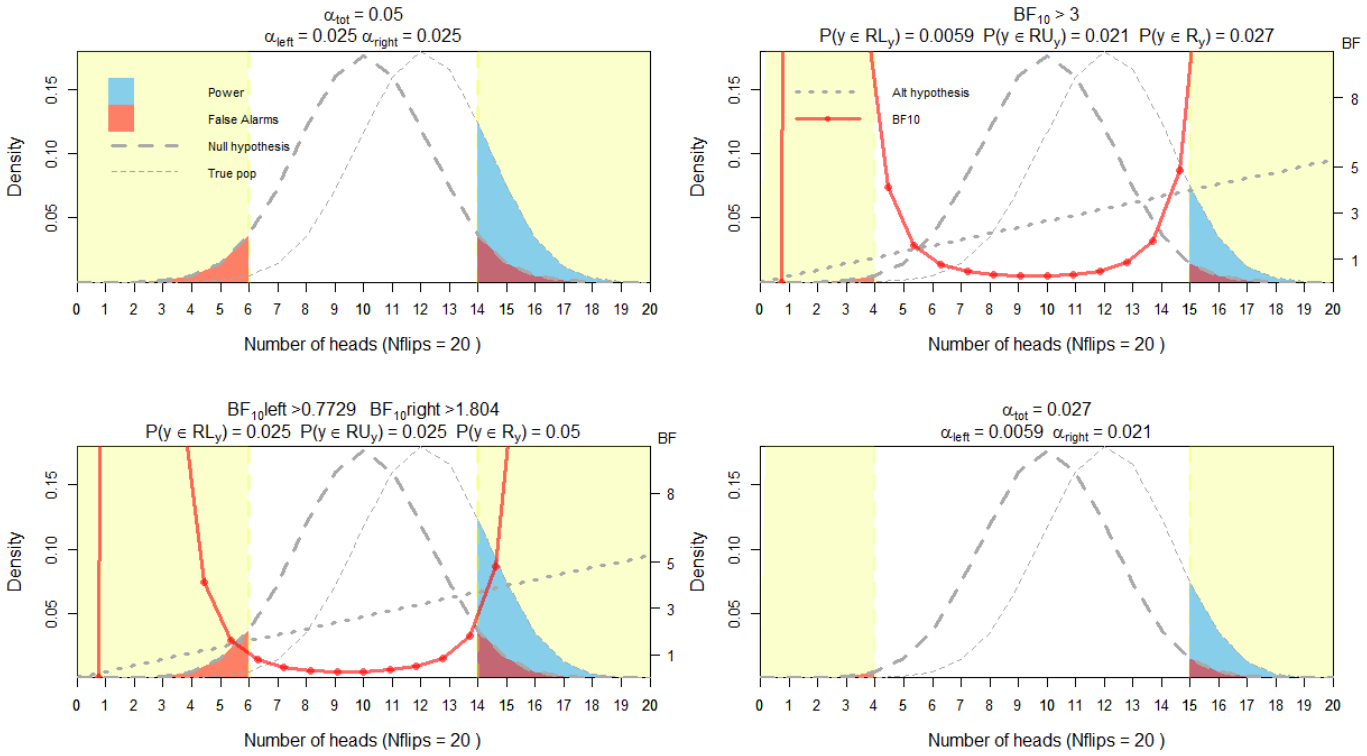


Figure 5 The Ry (the yellow shaded area) of the BF matched to the Ry of a two-sided p-value with α =0.05 (left) and the Ry of a "directed" p-value matched to the Ry of a BF>3 test with M1 (alt hypothesis) specified as a non-symmetric prior distribution beta(2,1) (right).

Discussion

Frequentist and Bayesian schools disagree with each other on how to construct statistical measures of evidence (Silva, 2018). Frequentists are concentrated on controlling pre-experimental probabilities of making wrong decisions (controlling the Type I error probability). Their decision rules are constructed based on the likelihood of obtaining the test results given that H_0 is true and how often one would obtain a false positive given that decision rule. For Bayesians, the Type I error probability does not play a central role in the construction of a decision rule. Instead, they are interested in which hypothesis or which θ values are more probable after observing the data. Their goal is to construct decision rules that rely on the post-experimental plausibility of both H_0 and H_1. The ROPE approaches can be good alternatives when the researcher can specify what effect size would be meaningful and what effect sizes would not be of interest.

The results of the simulation study show that the ROC curves of the p-value and Bayes factor coincide when M_1 is specified as a symmetric probability distribution. When this is not the case, the decision qualities of the Bayes factor are as good as the model M_1 that is used to calculate the Bayes factor. Better decision qualities are obtained when M_1 places more credibility on the θ in the same direction as the true θ; worse decision qualities are obtained when M_1 places more credibility on the θ in the opposite direction as the true θ.

The results show that the trade-off between the false positive rate and true positive rate of the CI+ROPE approach is always on the ROC curve of the p-value, and the trade-off between the false positive rate and true positive rate of the HDI+ROPE approach is always on the ROC curve of the BF. This suggests that the same applies for the ROPE approaches: better decision qualities are obtained for the HDI+ROPE approach when the prior distribution places more credibility on

the θ in the same direction as the true θ and worse decision qualities are obtained when the prior distribution places more credibility on the θ in the opposite direction as the true θ.

Despite that the Bayesian and frequentist schools are based on different rationales, the results of the simulation study suggest that aligning the rejection region for Bayesian and frequentist tests is possible when the tests have symmetrical rejection regions in the sense that $f(y \in RL_y|H_0) = f(y \in RU_y|H_0)$. Then $f(y \in R_y|H_0)$ can be set equal for the two tests, which results in equal $f(y \in R_y|H_1)$ and therefore equal ROC curves and equal decision qualities.

The ideas of Rice & Gaines (1994), who proposed asymmetrical rejection regions for frequentist tests in order to yield optimal power while still being able to find unexpected results, as discussed "*The possibility of asymmetrical rejection ratio's for frequentist tests*", provides a bridge between Bayesian tests with asymmetrical rejection regions because of asymmetric prior distributions and frequentist test. This fits with the theory of Bayarri et al. (2016), who argues that a prior distribution can simply be interpreted by a frequentist as a "weight function" for power computation. For M_0, this weight function is chosen to be a point mass at θ=0.5. For M_1, the prior distribution for the effect size under H_1 can be interpreted as a device to optimally power the procedure in the desired effect sizes.

It has been demonstrated in literature that frequentist and Bayesian schools are not in logical conflict. Instead, what separates Bayesian and frequentist approaches is how the rejection region R_y is chosen (Pericchi & Pereira, 2016; Silva, 2018). Bayarri et al. (2016) concludes that the use of the Bayes factor is actually a fully frequentist procedure. Silva (2018) generalizes the equivalence between Bayes factors and frequentist tests to all Bayesian tests. In his paper he demonstrates, through analytical arguments, the existence of a perfect equivalence between Bayesian and frequentist methods for testing when based on the same joint probability

distribution. According to Silva (2018), for each Bayesian test, one can always design an equivalent frequentist test, and conversely, for each frequentist test, one can always design an equivalent Bayesian test. This is perfectly in line with the proposed explanation of the results in this study.

The simulation study in this research has its limitations; for the HDI+ROPE and CI+ROPE approaches, the decision thresholds were not varied; hence their ROC curves were not obtained. We can only suspect that their ROC curves equal their point hypothesis tests counterpart because their false positives and true positive trade-off is always on the ROC curves corresponding to their corresponding point hypothesis test. Secondly, only symmetric α's were used to calculate the frequentist tests, therefore the simulation study does not provide evidence for the proposed theory that Bayesian tests with asymmetrical priors can always be matched to frequentist tests with asymmetrical α's. Thirdly, obviously, the simulation study was limited to binomial experiments with a predetermined N and four different statistical methods. Therefore, based on the results of the simulation study alone no inferences can be made about data from different distributions and different statistical methods. However, the results combined with the proposed framework of ordering the sample space, the possibility of using asymmetrical significance tails in the frequentists methods and the discussed literature about the equivalence of Bayesian and frequentist methods provide support for the findings of Silva (2018); it is always possible to convert a Bayesian test to a frequentist test that yields equal proportions false positives and equal proportions true positives and vice versa. When the rejection regions of the four tests coincide by adjusting the decision thresholds, they will always make the same decisions and therefore yield equal decision qualities.

Despite the implicit correspondence between the frequentist and Bayesian approaches, they do not provide the same information. The frequentist methods are based on the probability of obtaining a false positive (making conclusions based on the probability of obtaining the observed or more extreme data under the null hypothesis), while a hypothesis test based on the Bayes factor supports the hypothesis (i.e. model) under which the observed data are most likely and the HDI+ROPE test whether the 95% most credible values (with the prior taken into account) lie outside the specified ROPE.

In opposition to the common belief that a statistical test criterion should be based on just one of the two fashions, the results of this thesis suggest that hypothesis testing has not to be based on just one of these concerns but can use both criteria by calibrating the specific frequentist test into a Bayesian test (or vice versa) and report both. According to Silva (2018), control of Type I error probabilities and usage of posterior distributions can be utilized as complementary statistical devices. But because Bayesians and frequentist usually neglect the goals of each other, they can yield discordant results.

As suggested by Silva (2018), it would be better practice to combine both procedures by calibrating Bayesian decision rules into frequentist decision rules (or vice versa) can ensure effective control of Type I error probabilities while simultaneously take advantage of the Bayesian benefits, being: evaluating the strength of the evidence in the obtained data rather than also taking into account the more extreme (but not observed) values; being able to find support for the null hypothesis and providing the researcher with a tool to directly take previous findings into account (when desired) to provide cumulative evidence (Bayarri et al., 2016; Ortega & Navarrete, 2017). Both points of view are important and they can simultaneously be used in research. Therefore, in general, there is no point in arguing about what tests or rationale is

superior as both points of view provide useful additional information. In other words: they are complementary.

Future research should confirm whether it is indeed possible to construct coinciding rejection regions and therefore equal decision qualities for all frequentist and Bayesian methods and different kinds of data. Furthermore, it would be beneficial if future research would focus more on the correspondence of the Bayesian and frequentist methods, especially focusing on how the different approaches can strengthen each other, instead of getting lost in the philosophical differences. Despite some attempts in research, there is no simple (perfect) general calibration rule yet to convert Bayes factors from p-values and vice versa (Jeon & De Boeck, 2017). This is complicated because as we have seen, Bayes factor is a comparative method where the sample size plays a role in interaction with the effect size, while the proportion false positives of frequentist measures are independent of the sample size.

I contend that researchers would be less reluctant to perform and report both methods if the calibration between specific frequentist and Bayesian tests would be made easier. I aim to contribute to this process with a function in R called “alfa_to_bfcrit” (see Appendix 7) that calculates which critical Bayes factor would yield equal false alarm rates and equal power as a frequentist test that uses the specified alfa.

References

Altman, D. G. (1995). Statistics notes: Absence of evidence is not evidence of absence. *BMJ: British Medical Journal*, *311*(7003), 485.

Barker, L., Rolka, H., Rolka, D., & Brown, C. (2001). Equivalence testing for binomial random variables: Which test to use? *American Statistician*, *55*(4), 279–287. https://doi.org/10.1198/000313001753272213

Bayarri, M. J., Benjamin, D. J., Berger, J. O., Sellke, T. M., Berger, J. O., & Benjamin, D. J. (2016). Rejection Odds and Rejection Ratios : A Proposal for Statistical Practice in Testing Hypotheses. *Journal of Mathematical Psychology*, *72*, 90–103.

Bolstad, B. (2016). Bayesian Inference For Binomial Propo. In *Introduction to Bayesian* (pp. 149–168).

Bolstad, W. M., & Curran, J. M. (2016). Comparing Bayesian and frequentist inference for normal mean. In *Introduction to Bayesian Statistics* (3rd ed., pp. 169–192). John Wiley & Sons, Inc.

Cohen, J. (1988). *Statistical Power Analysis for the Behavioral Sciences* (2nd ed.). New York: Lawrence Erlbaum Associates.

Cumming, G. (2014). The New Statistics : Why and How. https://doi.org/10.1177/0956797613504966

Dienes, Z. (2016). How Bayes factors change scientific practice. *Journal of Mathematical Psychology*, *72*, 78–89. https://doi.org/10.1016/j.jmp.2015.10.003

Florkowski, C. M. (2008). Sensitivity , Specificity , Receiver-Operating Characteristic (ROC) Curves and Likelihood Ratios : Communicating the Performance of Diagnostic Tests, *29*(August), 83–87.

Garc, G., & Chen, M. (2005). CALIBRATING BAYES FACTOR UNDER PRIOR PREDICTIVE DISTRIBUTIONS. *Statistica Sinica*, *15*, 359–380.

García, A. M. R., & Puga, J. L. (2018). Deciding on Null Hypotheses using P-values or Bayesian alternatives : A simulation study, *30*(1), 110–115. https://doi.org/10.7334/psicothema2017.308

Gelman, A. (2008). Objections to Bayesian statistics. *Bayesian Analysis*, *3*(3), 445–450. https://doi.org/10.1214/08-BA318

Glas, A. S., Lijmer, J. G., Prins, M. H., Bonsel, G. J., & Bossuyt, P. M. M. (2003). The diagnostic odds ratio: a single indicator of test performance. *Journal of Clinical Epidemiology*, *56*, 1129–1135. https://doi.org/10.1016/S0895-4356(03)00177-X

Hajian-Tilaki, K. (2013). Receiver Operating Characteristic (ROC) Curve Analysis for Medical Diagnostic Test Evaluation. *Caspian Journal of Internal Medicine*, *4*(2), 627.

Halsey, L. G., Curran-everett, D., Vowler, S. L., & Drummond, G. B. (2015). The fickle P value generates irreproducible results. *Nature America*, *12*(3), 179–185.

Hubbard, R., & Lindsay, R. M. (2008). Why P Values Are Not a Useful Measure of Evidence in Statistical Significance Testing, *18*(1), 69–88. https://doi.org/10.1177/0959354307086923

Jeon, M., & De Boeck, P. (2017). Decision qualities of Bayes factor and p value-based hypothesis testing. *Psychological Methods*, *22*(2), 340–360. https://doi.org/10.1037/met0000140

Johnson, V. E. (2013). Uniformly most powerful bayesian tests. *The Annals of Statistics*, *41*(4), 1716–1741. https://doi.org/10.1214/13-AOS1123

Jones, B., Jarvis, P., Lewis, J. A., & Ebbutt, A. F. (1996). Trials To Assess Equivalence : The Importance Of Rigorous Methods Published by : BMJ Stable URL :

http://www.jstor.org/stable/29732191 REFERENCES Linked references are available on JSTOR for this article : You may need to log in to JSTOR to access the lin. *BMJ: British Medical Journal*, *313*(7048), 36–39. Retrieved from http://www.jstor.org/stable/29732191

Kruschke, J. K. (2010). *Doing Bayesian Data Analysis: A Tutorial with R and BUGS*. https://doi.org/10.5964/ejop.v7i4.163

Kruschke, J. K. (2011). Bayesian assessment of null values via parameter estimation and model comparison. *Perspectives on Psychological Science*, *6*(3), 299–312. https://doi.org/10.1177/1745691611406925

Kruschke, J. K. (2015). *Doing Bayesian data analysis, Second Edition: A tutorial with R, JAGS, and Stan.* (Second Edi). Burlington, MA: Academic Press / Elsevier.

Kruschke, J. K. (2018a). Rejecting or accepting parameter values in Bayesian estimation.

Kruschke, J. K. (2018b). Supplement to Rejecting or accepting parameter values in Bayesian estimation.

Kruschke, J. K., & Liddell, T. M. (2017a). Bayesian data analysis for newcomers. *Psychonomic Bulletin and Review*, 1–23. https://doi.org/10.3758/s13423-017-1272-1

Kruschke, J. K., & Liddell, T. M. (2017b). The Bayesian New Statistics: Hypothesis testing, estimation, meta-analysis, and power analysis from a Bayesian perspective. *Psychonomic Bulletin and Review*, 1–29. https://doi.org/10.3758/s13423-016-1221-4

Lakens, D. (2017). Equivalence Tests: A Practical Primer for t Tests, Correlations, and Meta-Analyses. *Social Psychological and Personality Science*, *8*(4), 355–362. https://doi.org/10.1177/1948550617697177

Morey, R. D., Romeijn, J. W., & Rouder, J. N. (2016). The philosophy of Bayes factors and the quantification of statistical evidence. *Journal of Mathematical Psychology*, *72*, 6–18.

https://doi.org/10.1016/j.jmp.2015.11.001

Mulder, J., & Wagenmakers, E. (2016). Editors ' introduction to the special issue '" Bayes factors for testing hypotheses in psychological research : Practical relevance and new developments "'. *Journal of Mathematical Psychology*, *72*, 1–5. https://doi.org/10.1016/j.jmp.2016.01.002

Neuhävser, M. (2004). The Choice of a for One-sided Tests. *Drug Information Journal*, *38*(1), 57–60.

Ortega, A., & Navarrete, G. (2017). Bayesian Hypothesis Testing : An Alternative to Null Bayesian Hypothesis Testing : An (NHST) Alternative to Null Hypothesis Significance Testing in Psychology Hypothesis Significance Testing (NHST) in Psychology and Social Sciences and Social Sciences. In *Bayesian Inference*. https://doi.org/10.5772/intechopen.70230

Pericchi, L., & Pereira, C. (2016). Adaptative significance levels using optimal decision rules: Balancing by weighting the error probabilities. *Brazilian Journal of Probability and Statistics*, *30*(1), 70–90. https://doi.org/10.1214/14-BJPS257

Rouder, J. N., Morey, R. D., Speckman, P. L., & Province, J. M. (2012). Default Bayes factors for ANOVA designs. *Journal of Mathematical Psychology*, *56*(5), 356–374. https://doi.org/10.1016/j.jmp.2012.08.001

See, E. G., & Cohen, M. D. (2007). A practical solution to the pervasive problems of p values. *Psychonomic Bulletin & Review*, *14*(5), 779–804. https://doi.org/10.3758/BF03194105

Silva, I. R. (2018). Communications in Statistics - Theory and Methods On the correspondence between frequentist and Bayesian tests. *Communications in Statistics---Theory and Methods*, *47*(14), 3477–3487. https://doi.org/10.1080/03610926.2017.1359296

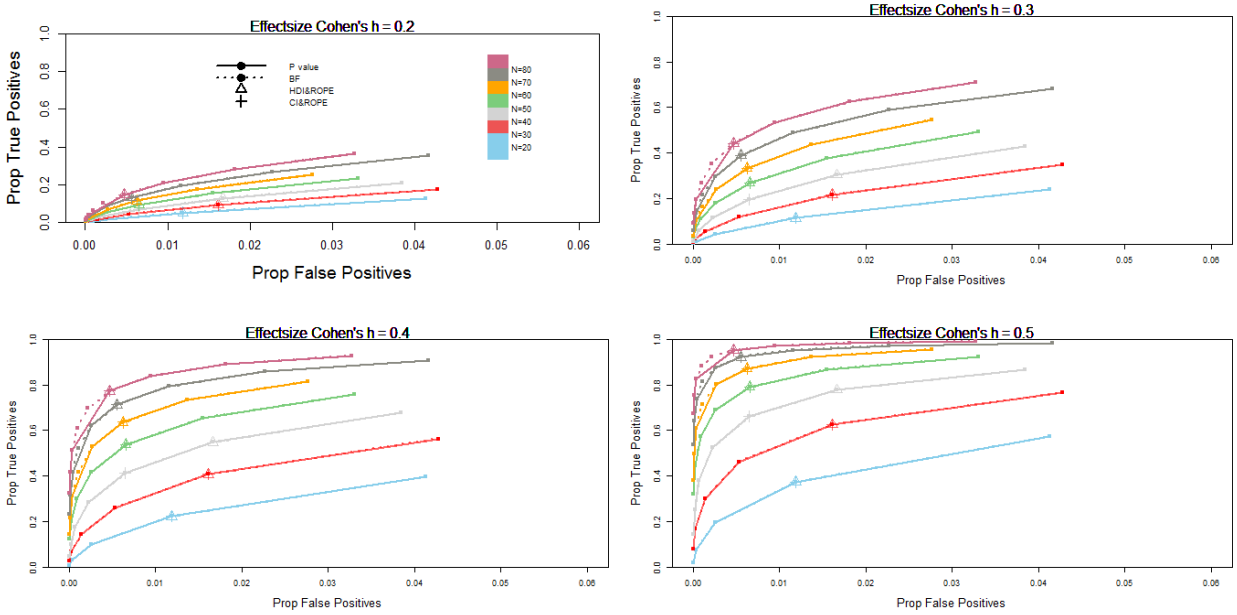
Sprenger, J. (2013). Testing a Precise Null Hypothesis : The Case of Lindley ' s Paradox, *80*(December), 733–744.

Wagenmakers, E. J., Lodewyckx, T., Kuriyal, H., & Grasman, R. (2010). Bayesian hypothesis testing for psychologists: A tutorial on the Savage-Dickey method. *Cognitive Psychology*, *60*(3), 158–189. https://doi.org/10.1016/j.cogpsych.2009.12.001

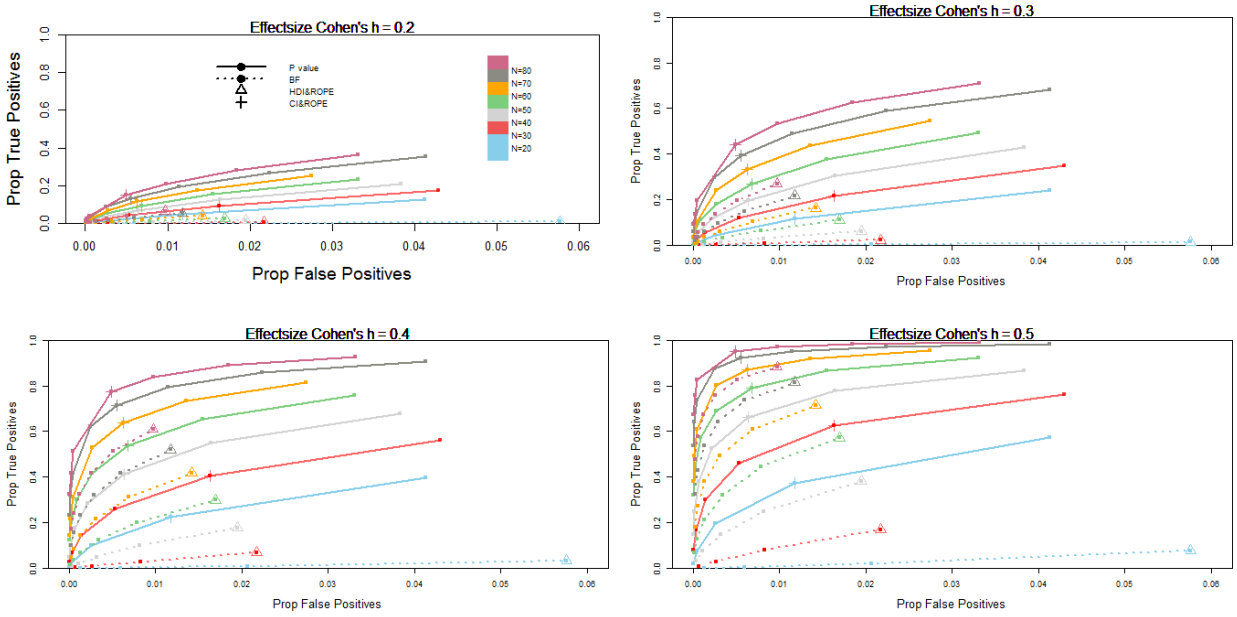
Wasserstein, R. L., Lazar, N. A., Wasserstein, R. L., Lazar, N. A., & Asa, T. (2016). The ASA ' s Statement on p-Values : Context , Process , and Purpose. *The American Statistician*, *70*(2), 129–133. https://doi.org/10.1080/00031305.2016.1154108

Appendix 1: Effect sizes plotted separately

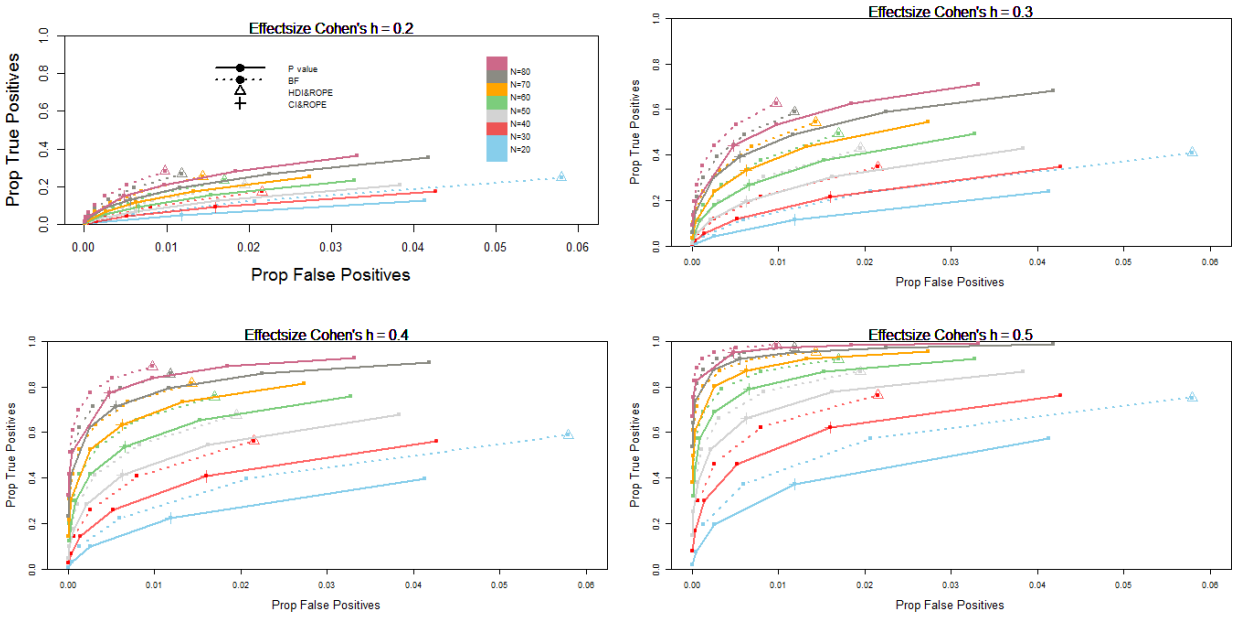
Experiment 1: *Beta*(1,1)



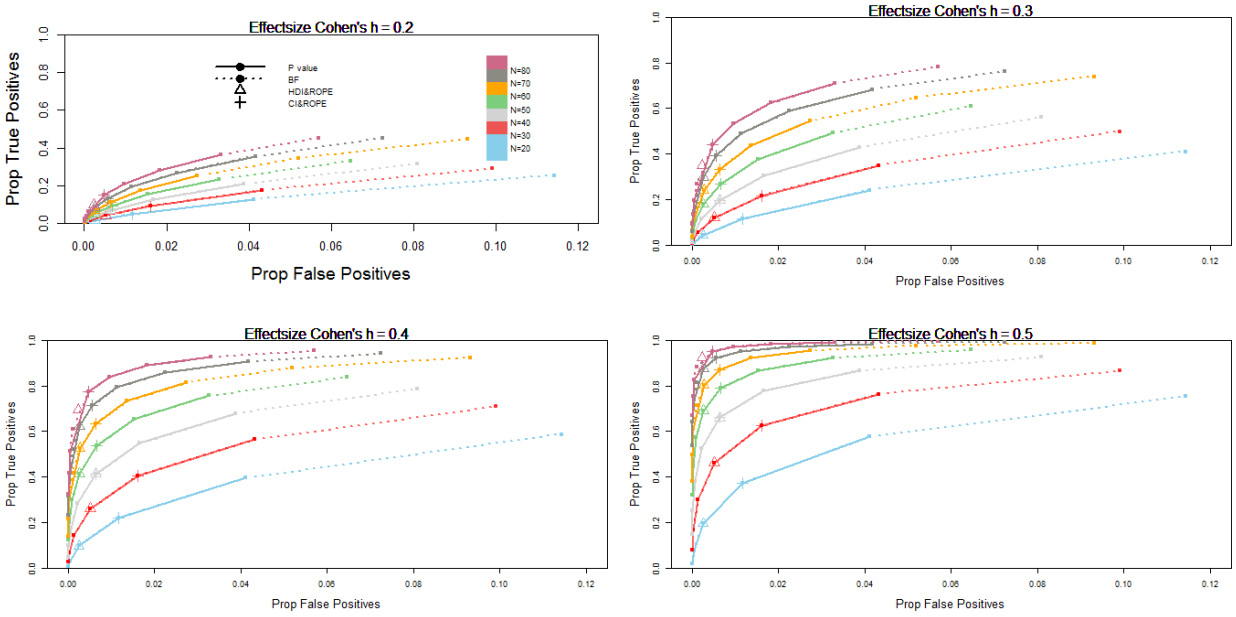
Experiment 2: *Beta*(1,5)



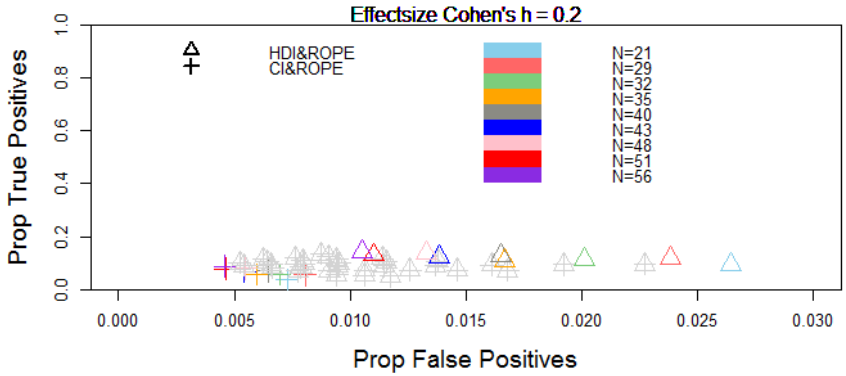
Experiment 3: *Beta*(5,1)

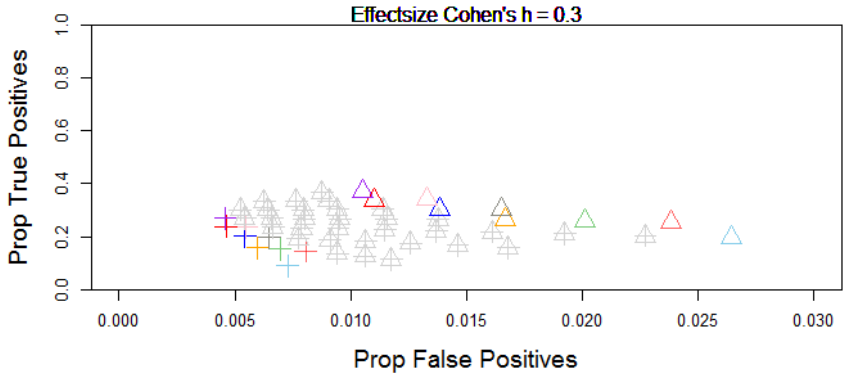


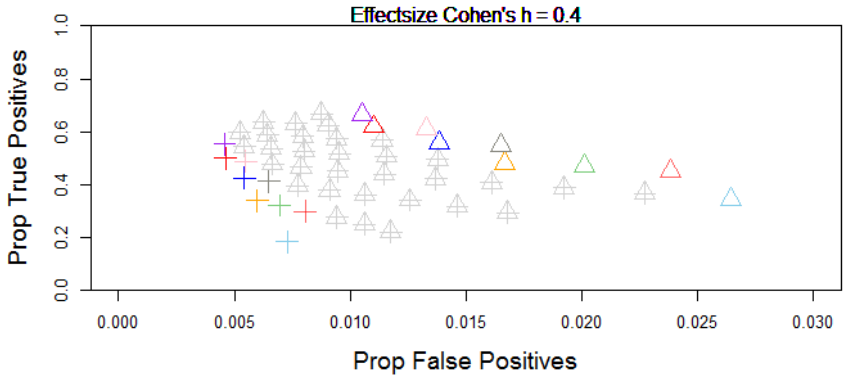
Experiment 4: *Beta*(5,5)

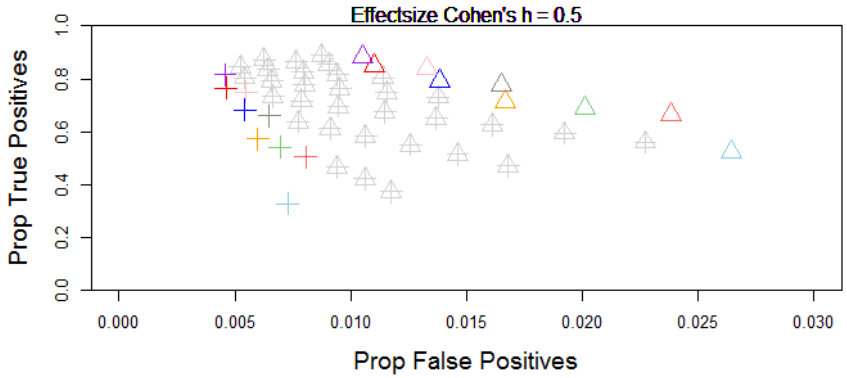


Appendix 2: Results follow-up study









Appendix 3: Deriving the Bayes rule

Deriving the Bayes rule

The formula of conditional probability is:

1. $\boldsymbol{p(y|x) = \frac{p(y,x)}{p(x)}}$

(Kruschke, 2015). In words, the definition simply says that the probability of y given x is the probability that they happen together relative to the probability that x happens at all. By multiplying both sides of the formula by p(x) we get:

2. $\boldsymbol{p(y|x)p(x) = p(y,x)}$

We can do the analogous manipulation starting with $p(x|y) = \frac{p(y,x)}{p(y)}$. We multiply both sides of the formula by p(y) to get:

3. $\boldsymbol{p(x|y)p(y) = p(y,x)}$

Formula 2 and formula 3 have both $p(y,x)$ on one side of the equation, this proves that $p(x|y)p(y) = p(y|x)p(x)$. Divide both sides of the formula by $p(x)$ to get:

4. $\boldsymbol{p(y|x) = \frac{p(x|y)p(y)}{p(x)}}$

We can re-write the denominator in terms of $p(x|y)$ by making use of the next formula:

$$p(x) = \sum_y p(x,y)$$

In words, it says that the probability of getting x is the sum of all x, y arching over all possible y's. We also know that $p(x,y) = p(x|y)p(y)$ because $p(x|y) = = \frac{p(x,y)}{p(y)}$. Combining those equitation's yields $p(x) = \sum_y p(x,y) = \sum_y p(x|y)p(y)$. Substituting this into the denominator of formula 4 we get:

5. $\boldsymbol{p(y|x) = \frac{p(x|y)p(y)}{\sum_y p(x|y)p(y)}}$

Both the equations 4 and 5 are called 'The Bayes Rule'. The input in the equations are the factors likelihood, prior and evidence:

$p(y|x) = p(x|y)\frac{p(y)}{p(x)}$ is equal to $posterior = \frac{likelihood * prior}{evidence}$.

Appendix 4: Multiplying a *beta* distribution with the likelihood function

Formally, a beta distribution has two parameters, called a and b, and the density itself is defined as:

$$p(\theta|a,b) = Beta(\theta;a,b) = \theta^{(a-1)}(1-\theta)^{b-1}/B(a,b) \quad (8)$$

Where B(a,b) is a normalizing constant that ensures that the area under the beta integrates to 1.0 (Kruschke, 2010). Substituting the likelihood function (formula 7) and the beta prior (formula 8) distribution into the Bayes' rule (formula 4) yields

$$p(\theta|y,N) = \frac{p(y|N,\theta)p(\theta)}{p(y,N)} \quad (9)$$

$$p(\theta|y,N) = \frac{\theta^{y}(1-\theta)^{n-y}\,\theta^{(a-1)}(1-\theta)^{b-1}}{[B(a,b)p(y,N)]}$$

$$p(\theta|y,N) = \frac{\theta^{((y+a)-1)}(1-\theta)^{((n-y+b)-1)}}{B(y+a,n-y+b)}$$

(Kruschke, 2010). Where the denominator is again just the normalizing factor for the corresponding beta distribution. In words, formula 9 says that, when the prior distribution is *Beta*(θ, a, b), and the data have y heads in N flips, then the posterior distribution is *Beta*(θ; y + a, n-y + b). This makes using *Beta*(a,b) prior when we have binomial observations particularly easy because we do not have to do any integration to find the posterior.

Appendix 5: Cohens' h

Intuitively, one might think that simply taking the difference between proportions would be an appropriate measure of effect size. However, according to Cohen (1977), equal differences in proportions along the distribution from 0 to 1 are not necessarily equally detectable. But subjecting the two proportions to an arcsine transformation before taking the difference solves this problem; the equal differences between arcsine transformations are equally detectable (Cohen, 1988). This effect size measure is called Cohen's h:

$$Cohen's\ h\ =\ 2\arcsin\sqrt{P_1} -\ 2\arcsin\sqrt{P_2}$$

According to Cohen, a Cohen's h = .20 corresponds to a small effect, Cohen's h = .50 to a medium effect and Cohen's h = .80 to a large effect.

Appendix 6: R code Experiments

```
       # Clear all
       graphics.off()
       rm(list=ls(all=TRUE))
       setwd("~/R/win-library/3.4")

       #------------------------------------------------------------------------------
-------------------------
       #                       Functions needed:
       #R editor RSTUDIO
       #DBDA2Eprograms.zip
(https://sites.google.com/site/doingbayesiandataanalysis/software-installation)
       #functie 'HDIofICDF.R
(BRON:http://www.indiana.edu/~kruschke/DoingBayesianDataAnalysis/Programs/)
       source("./HDIofICDF.R") #benodigde function 'HDIofICDF.R' oproepen
       #install.packages("gtools", dependencies = TRUE)
       library(gtools)
       #install.packages("stargazer", dependencies = TRUE)
       library(stargazer)
       #install.packages("stats", dependencies = TRUE)
       library(stats)
       #install.packages("Hmisc", dependencies = TRUE) #CI uitrekenen
       library(Hmisc)
       # -----------------------------------------------------------------------------
-------------------------
       #                       References
       #I copy-pasted the R-code to calculate HDI-interval, p-value and Bayes Factor
from:
         #Solutions to Exercises in Doing Bayesian Data Analysis 2nd Ed. by Kruschke ©
2015.
           #Retrieved from:
https://sites.google.com/site/doingbayesiandataanalysis/exercises  blz. 102 - 103

       #Hlavac, Marek (2018). stargazer: Well-Formatted Regression and Summary
Statistics Tables. R package version 5.2.1. https://CRAN.R-
project.org/package=stargazer
       # -----------------------------------------------------------------------------
-------------------------

       #*****************************************************************************
*************************
       #                       Set 'simple' parameters
       nullTheta = 0.5 # null hypothesis
       Nsamples = 1000 #total samples for every condition (as much as possible)
       aPrior=1 #
       bPrior=1 #

       #                       Set changing parameters
       vecalfaPvalue = as.numeric(c(0.05, 0.025, 0.02, 0.015, 0.01, 0.005, 0.001,
0.0005, 0.0002, 0.0001))#critical p-value/alfa (when do you decide there's an effect?)
       vecSamplesize= as.numeric(c(20, 30, 40, 50, 60, 70, 80, 90, 100, 150, 200, 250,
300, 400))#total flips in every sample / samplesize
       vecCohensH = as.numeric(c(0, 0.2, 0.3, 0.4, 0.5)) #effectsizes
       vecBFkrit = as.numeric(c(1.5, 2, 3, 4, 5, 6, 7, 8, 10, 15, 30, 50, 80,
100))#critical Bayes Factor (when do you decide there's an effect?)

       #                       Do you want plots & documents and/or save your
workspace?
       Plots = "off" #set "on" or "off"
       Documents = "off" #set "on" or "off"
       saveworkspace = "on" #save the workspace?
```

```
      name = "ExpX_1000000_aPrirXbpriorX" #what should be the name of the
savedworkspace (in case you want to save it)

      #*************************************************************************
************************

      ##########################start
experiment#########################################################################
      ##########################################################################################
############################
      # ------------------------------------------------------------------------
-------------------------
      #                       Calculate ROPE & HDIwidth
      # ------------------------------------------------------------------------
-------------------------

      # Calculate end-points of ROPE
      hmaxROPE= 0.1 # we accept a deviation from the nullTheta with a maximum of
Cohens h = 0.1 (0.5* a small effect) as still equivalent to our nulltheta
      ropemin = (-sin(0.5*hmaxROPE-asin(sqrt(nullTheta))))^2 #what is the lower
threshold?
      ropemax = (-sin(0.5*hmaxROPE+asin(sqrt(nullTheta))))^2 #what is the upper
threshold?

      # ------------------------------------------------------------------------
-------------------------
      #                       Make (lists with) matrices to save the data from the
experiment
      # ------------------------------------------------------------------------
-------------------------
      ##########################################################################
##########################
      #Make vectors with
names#####################################################################################

      #The amount of different testing methods depends on how many different
'critical BFs' there are
      #This for-loop creates appropiate names for the matrices we'll use
      NamevecBF = matrix(NA, ncol=length(vecBFkrit)) #make a vector with as many
places as there are inputs in vecBFkrit
      for (i in 1:length(vecBFkrit))
      {  NumberBF = as.character(vecBFkrit[i]) #put the critical BF in the name (bv
"3")
      NamevecBF[i]= paste0("BF",NumberBF)} #combine "BF" with the number & save this
in NamevecBF


      #The amount of different testing methods depends on how many different 'alfa's
there are
      #This for-loop creates appropiate names for the matrices we'll use
      NamevecPvalue = matrix(NA, ncol=length(vecalfaPvalue)) #make a vector with as
many places as there are inputs in vecBFkrit
      for (i in 1:length(vecalfaPvalue))
      {  NumberBF = as.character(vecalfaPvalue[i]) #put the critical BF in the name
(bv "3")
      NamevecPvalue[i]= paste("p-value",NumberBF)} #combine "BF" with the number &
save this in NamevecBF

      #Make also a vector with names for the methods that depend on equivalence
      NamevecEq=c("HDI&ROPE", "EqCI") #

      # make vector with "Samplesize = x"
      Flipnames = matrix(NA, nrow=1, ncol=length(vecSamplesize))
```

```
       for (f in 1:length(vecSamplesize))
       {number = as.character(vecSamplesize[f])
       Flipnames[f] = paste0("Samplesize =", number)}

       # make vector with "CohensH = x"
       CohensHnames = matrix(NA, nrow=1, ncol=length(vecCohensH))
       for (f in 1:length(vecCohensH))
       {number = as.character(vecCohensH[f])
       CohensHnames[f] = paste0("CohensH =", number)}

       ##########################################################################
#####################################
       # Make a list with matrices for the different critical Bayes Factor to save
their proportion correct decision (Power)

       BFmatrix = matrix(NA,nrow=length(vecSamplesize),ncol=(length(vecCohensH)-
1))#design of every matrix
       rep=length(vecBFkrit) #the amount of matrices needed
       BFmatricesPow = list() #make a list called 'BFmatrices' to store the different
matrices
       for(i in 1:rep){
         newmatrix = replicate(1,BFmatrix)#copy the BF matrix
         rownames(newmatrix) = paste0(("Samplesize="),c(vecSamplesize))
         colnames(newmatrix) = paste0("CohensH=",
c(vecCohensH[2:(length(vecCohensH))]))
         BFmatricesPow[[(NamevecBF[i])]] = newmatrix} #give the BF matrix a name,
pulled from the vector with names

       ##########################################################################
#####################################
       # Make a list with matrices for the different critical p-values (alfa's) to
save their proportion correct decision (Power)

       Pvaluematrix = matrix(NA,nrow=length(vecSamplesize),ncol=(length(vecCohensH)-
1))#design of every matrix
       rep=length(vecalfaPvalue) #the amount of matrices needed
       PvaluematricesPow = list() #make a list called 'PvaluematricesPow' to store the
different matrices
       for(i in 1:rep){
         newmatrix = replicate(1,Pvaluematrix)#copy the Pvaluematrix matrix
         rownames(newmatrix) = paste0(("Samplesize="),c(vecSamplesize))
         colnames(newmatrix) = paste0("CohensH=",
c(vecCohensH[2:(length(vecCohensH))]))
         PvaluematricesPow[[(NamevecPvalue[i])]] = newmatrix} #give the BF matrix a
name, pulled from the vector with names

       ##########################################################################
#####################################
       #Make a list with matrices to save the proportion correct decision for the
other methods##############################

       EqtestmatrixPow =
matrix(NA,nrow=length(vecSamplesize) ,ncol=length(vecCohensH)-1) #design of every
matrix
       EqtestmatricesPow = list() #make a list is called 'EqtestmatricesPow' to store
the different matrices
       for(i in 1:length(NamevecEq)){
         newmatrix = replicate(1,EqtestmatrixPow) #replicate RestmatriceFA to get more
matrices with the same size
         rownames(newmatrix) = c(Flipnames)
         colnames(newmatrix) = c(CohensHnames[2:length(CohensHnames)])
         EqtestmatricesPow[[NamevecEq[i]]] = newmatrix} #give the new matrix an
appropiate name
```

```
      ##############################################################################
#########################################
      #Make matrix to store the False alarm rates for the different methods
################################################

      FAmatrix =
matrix(NA,nrow=(length(vecalfaPvalue)+length(vecBFkrit)+length(NamevecEq)),ncol=(lengt
h(vecSamplesize))) #design matrix
      rownames(FAmatrix) = c(NamevecPvalue,NamevecBF, NamevecEq)
      colnames(FAmatrix) = c(Flipnames)

      ##############################################################################
#########################################
      # Make matrices to save data temporarily (they will be overwritten in every
Samplesize & BF loop) ####################

      matValues = matrix(NA,nrow=Nsamples,ncol=7) #Matrix to save the calculated
values
      colnames(matValues) <- c('prop. Head', 'p-value', 'BF', 'HDIleft', 'HDIright',
'CIleft', 'CIright') #column names

      decisionmatrix = matrix(NA,nrow=Nsamples,ncol=(1+
(length(vecalfaPvalue))+(length(vecBFkrit))+(length(NamevecEq)))) #Matrix to save the
decision for every method, based on their values and their 'critical value'
      colnames(decisionmatrix) <- c('prop. Head', c(NamevecPvalue), c(NamevecBF),
c(NamevecEq)) #column names

      # ------------------------------------------------------------------------
-------------------------
      #                       Start loops
      # ------------------------------------------------------------------------
-------------------------

      #Start big loop 1: different samplesizes (number of flips per sample)
      for (i in 1:length(vecSamplesize)) {
        Samplesize = vecSamplesize[i]

        #Make matrix to store the sampling data (0's and 1's)
        matFlips=matrix(NA,nrow=Nsamples ,ncol=Samplesize)#matrix to store the data
from sampling (0 of 1)
        colnames(matFlips) <- c(1:Samplesize)

        #Start big loop 2: different effectsizes (CohensH)
        for (j in 1:length(vecCohensH)) {
          CohensH = vecCohensH[j]

          # Calculate theta, which depends on CohensH
          if (CohensH == 0) {theta = 0.5}
          else {theta = (-sin(0.5*CohensH+asin(sqrt(nullTheta))))^2 }#Calculate Theta
depending on CohensH

#&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&
&&&&&&&&&&

          #                        Start samplingloop
          #Generate random sequence of flips and save this in the samplingmatrix:
          for ( l in 1:Nsamples)
          {matFlips[l,]= sample( c(0,1) , size=Samplesize , replace=TRUE , prob=c(1-
theta,theta) )

          #------------------------------------------------------------------------
---------------------------
```

```
          #             Calculate values & save in matValues
          # ------------------------------------------------------------------------
------------------------------

          #Col1 matValues: proportion 'heads'
          z = sum(matFlips[l,]) #sum 1's in every sample
          prop= z / Samplesize #calculate the proportion of heads
          matValues[l,1]=prop #save in matValues

          #Col2 matValues: calculate p-value
          plow = pbinom( q=z , size=Samplesize , prob=nullTheta )
          phi = 1.0-pbinom( q=z-1 , size=Samplesize , prob=nullTheta )
          pvalue = 2*min(c(plow,phi))

          matValues[l,2]=pvalue #save in matValues

          #COl3 matValues: calculate the Bayes Factor
          bf = ( exp( lbeta(aPrior+z,bPrior+Samplesize-z) - lbeta(aPrior,bPrior) ) /
( nullTheta^z * (1.0-nullTheta)^(Samplesize-z) ) ) #Calculate BayesFactor (I copy
pasted this part), eq.12.3 blz.
          matValues[l,3]=bf #save in matrix

          #Col4 & 5 matValues: calculate the endpoints of the HDI (I copy pasted this
part)
          hdi = matrix(NA,nrow=1,ncol=2) #make a vector with 2 empty places
          hdi= HDIofICDF( qbeta , shape1=aPrior+z , shape2=bPrior+Samplesize-z )#
calculate endpoints (left & right)
          matValues[l,4]= hdi [1] #save left endpoint
          matValues[l,5] = hdi [2] #save right endpoint

          #Col6 & 7 matValues: EqCI confidence interval
          CI = binconf(z, Samplesize, alpha=0.05,
                  method="wilson", #Following Agresti and Coull, the Wilson interval
is to be preferred and so is the default.
                  include.x=FALSE, include.n=FALSE, return.df=FALSE)

          matValues[l,6] = CI[2] #lower bound CI
          matValues[l,7] = CI[3] #upper bound CI

          # ------------------------------------------------------------------------
------------------------------
          #             Make a decision based on the values in matValues & save in
decisionmatrix
          # ------------------------------------------------------------------------
------------------------------

          #Col1 decisionmatrix: proportion heads
          decisionmatrix[l,1]=prop

          #Col 2:(length:vecalfaPvalue) make decisions based on different critical
Bayes Factors
          for (k in 1:length(vecalfaPvalue)) { #Loop for different critical values as
inputed in vecalfaPvalue
            alfa = vecalfaPvalue[k]

            if ( pvalue < alfa ) {decisionmatrix[l,(1+k)] = 'signif'} #  reject
0hypothesis: significant result
            else if (pvalue > alfa) {decisionmatrix[l,(1+k)] = 'nietsig'}} # don't
reject 0hypothesis: nonsignificant result

          #Next columns: make decisions based on different critical Bayes Factors
          for (k in 1:length(vecBFkrit)) { #Loop for different critical values as
inputed in vecBFkrit
```

```
            criticalBF = vecBFkrit[k]

            #Col 3 decisionmatrix: make a decision based on the BF
          if (bf < 1/criticalBF ) {decisionmatrix[l,(1+(length(vecalfaPvalue))+k)] =
'nul'} #  accept nullhypothesis
            else if (bf > criticalBF)
{decisionmatrix[l,(1+(length(vecalfaPvalue))+k)] = 'alt'} # accept
alternativehypothesis
            else if (bf > 1/criticalBF & bf < criticalBF)
{decisionmatrix[l,(1+(length(vecalfaPvalue))+k)] = 'geen'}} #no decision

          #Next column in decisionmatrix: take a decision based on HDI& ROPE
          if ((hdi[1] >= ropemin) & (hdi[2] <= ropemax))
{decisionmatrix[l,(1+(length(vecalfaPvalue))+(length(vecBFkrit))+1)] = 'nul'} #HDI is
completely inside the rope, accept nullhypothesis
          else if ((hdi[2] < ropemin | hdi[1] > ropemax))
{decisionmatrix[l,(1+(length(vecalfaPvalue))+(length(vecBFkrit))+1)] = 'alt'}#HDI is
completely outside the rope, accept alternative hypothesis
          else {decisionmatrix[l,(1+(length(vecalfaPvalue))+(length(vecBFkrit))+1)] =
'geen'} #HDI is partly inside, and partly outside, we don't make a decision yet

          #Next column decisionmatrix: decision EqCI
          if ((CI[2] >= ropemin) & (CI[3] <= ropemax)) {EqCI = "null"} # accept
nulhypothesis
            else if ((CI[3] < ropemin | CI[2] > ropemax)) {EqCI = "alt"} # accept
alternative hyphothesis
              else {EqCI = "no decision"} #we can not (yet) conclude that the true
theta differs significantly from the values in the "ROPE'

          decisionmatrix [l,(1+(length(vecalfaPvalue))+(length(vecBFkrit))+2)] = EqCI

          } #end sampling loop
          #-------------------------------------------------------------------------
-----------------------------------------
          #                     Start data Analysis
          #-------------------------------------------------------------------------
-----------------------------------------

##########################################################################################
#################################
          ###################### Make False Alarm matrix
#########################################################################

          if (theta == nullTheta) { #When the 0hypothesis is true

            for(z in 1:(length(vecalfaPvalue))){ # how many different alfa's are
tested depends on the input, therefore we need a for-loop again
              SumPvalue = sum(decisionmatrix[,(1+z)] == 'signif')
              propFAPvalue = (SumPvalue/Nsamples)
              FAmatrix [z,i]=propFAPvalue}

            for(z in 1:(length(vecBFkrit))){ # how many different critical BF are
tested depends on the input, therefore we need a for-loop again
              SumBF = sum(decisionmatrix[,(1+(length(vecalfaPvalue))+z)] == 'alt')
              propFABF = (SumBF/Nsamples)
              FAmatrix [((length(vecalfaPvalue))+z),i]=propFABF}

            HDIROPEFA =
sum(decisionmatrix[,(1+(length(vecalfaPvalue))+(length(vecBFkrit))+1)] == 'alt');#
Count how many times HDI&ROPE indicated proof for the alternative hypothesis when
there was no effect
            propFAHDIROPE = (HDIROPEFA / Nsamples);# calculate proportion false
alarms
```

```
            FAmatrix [((length(vecalfaPvalue))+(length(vecBFkrit))+1),i] =
propFAHDIROPE

            EqCIFA =
sum(decisionmatrix[,(1+(length(vecalfaPvalue))+(length(vecBFkrit))+2)] == 'alt');
            propFAEqCI = (EqCIFA / Nsamples);
            FAmatrix [((length(vecalfaPvalue))+(length(vecBFkrit))+2),i] = propFAEqCI

          }else{


################################################################################
################################
          ###################### Make Power matrices
################################################################################

            for(z in 1:(length(vecalfaPvalue))){ # how many different alfa's are
tested depends on the input, therefore we need a for-loop again
              SumPvalue = sum(decisionmatrix[,(1+z)] == 'signif')
              proportionFAPvalue = (SumPvalue/Nsamples)
              PvaluematricesPow[[z]][i,(j-1),1]=proportionFAPvalue}

            # i= place in vecSamplesize, for every Samplesize we test, the proportion
FA is stored in the i'th column
            # j= place in vecCohensH
            # j-1 = CohensH > 0  (there's an effect). In the 'Power matrices' we only
want the samples where there was actually an effect
            # therefore, every CohensH (except CohensH = 0) is stored in the j-1'th
column
            # Therefore we store the data in the [i,(j-1),1]'th place

            for (o in 1:length(vecBFkrit)) { # how many different critical BF are
tested depends on the input, therefore we need a for-loop again
              CorrectRejectionsBF = sum(decisionmatrix[,(1+length(vecalfaPvalue)+o)]
== 'alt');
              PowerBF = (CorrectRejectionsBF / Nsamples);
              BFmatricesPow[[o]][i,(j-1),1]=PowerBF}

            CorrectRejectionsHDIROPE =
sum(decisionmatrix[,(1+(length(vecalfaPvalue))+(length(vecBFkrit))+1)] == 'alt');
            PowerHDIROPE = (CorrectRejectionsHDIROPE / Nsamples);
            EqtestmatricesPow$`HDI&ROPE`[i,(j-1),1]=PowerHDIROPE

            CorrectRejectionsEqCI =
sum(decisionmatrix[,(1+(length(vecalfaPvalue))+(length(vecBFkrit))+2)] == 'alt');
            PowerEqCI = (CorrectRejectionsEqCI / Nsamples);
            EqtestmatricesPow$EqCI[i,(j-1),1]=PowerEqCI

          } #end IF-statement
        } #End loop CohensH (j)
      } #End loop Samplesize (i)

      #-------------------------------------------------------------------------------
-----------------------------------
      #                     Make plots & docs
      #-------------------------------------------------------------------------------
-----------------------------------
      if (Plots == "on") {

      vecSamplesize <- as.numeric(vecSamplesize) # convert factor to numeric
      NlinesPvalueBF <- (length(vecalfaPvalue)+length(vecBFkrit)) # count how many
lines are needed for the interactionplot (depends on input vecBFkrit)
```

```
      ##############################################################################
#####################################
      ###################### Plot 1: Proportion FA p-value vs
BF#############################################################################
      png(paste0("False Alarms_p-value.v.s.BF_Nsamples=",Nsamples,".png"), 1500,
1000) #save plot as png image

      # get the range for the x and y axis
      xrange <- range(vecSamplesize)
      yrange <- c(0, 0.06)

      # set up the plot
      plot(xrange, yrange, type="n", xlab="Samplesize",
           ylab="Proportion FA", cex.lab=2)
      colours <- rainbow(NlinesPvalueBF) #choose as many colours as lines in plot
      plotchar <- seq(18,18+NlinesPvalueBF,1)

      # add lines
      for (j in 1:NlinesPvalueBF)
      {lines(vecSamplesize, (as.vector(FAmatrix[(j),])), type = "o", lwd=2,
             lty=1 , col=colours[j], pch=plotchar[1])}

      # add a title
      title("Proportion FA p-value & Bayes Factor", cex.main = 2.5)

      # add a legend
      legend(xrange[1], yrange[2], c(NamevecPvalue,NamevecBF), cex=1.5, col=colours,
             pch=plotchar[1], lty=1)

      # Save the file.
      dev.off()

      ##############################################################################
#####################################
      ###################### Plot 2: Proportion FA Eq.
tests#########################################################################
      png(paste0("False Alarms_Eq.Test_Nsamples=",Nsamples,".png"), 1500, 1000) #save
plot as png image
      NlinesEq= length(NamevecEq)

      # get the range for the x and y axis
      xrange <- range(vecSamplesize)
      yrange <- c(0, 0.03)

      # set up the plot
      plot(xrange, yrange, type="n", xlab="Samplesize",
           ylab="Proportion FA", cex.lab=2)
      colours <- rainbow(NlinesEq) #choose as many colours as lines in plot
      plotchar <- seq(18,18+NlinesEq,1)

      # add lines
      for (j in 1:NlinesEq)
      {lines(vecSamplesize,
(as.vector(FAmatrix[((length(vecalfaPvalue)+length(vecBFkrit))+j),])), type = "o",
lwd=2,
             lty=1 , col=colours[j], pch=plotchar[1])}

      # add a title
      title("Proportion FA p-value & Bayes Factor", cex.main = 2.5)

      # add a legend
      legend(xrange[1], yrange[2], c(NamevecEq), cex=1.5, col=colours,
             pch=plotchar[1], lty=1)
```

```
       # Save the file.
       dev.off()

       ##############################################################################
#####################################
       ###################### Plot 3: Power p-value vs BF
##############################################################################
       # The amount of plots we want depends on how many different CohensH-values are
imputed
       # Therefore we have to calculate the number of rows and collumns we put into
'par(mfrow = c(xrow, ycol))
       # We can calculate this with "(0.5*(length(vecCohensH)-1))". We use vecCohensH-
1 because vecCohensH[1]=0, we only want CohensH>0 in this plot
       ycol = as.integer(0.5*(length(vecCohensH)))
       xrow = as.integer(0.5*(length(vecCohensH))+0.5) #control for odd number if
necessary

       # Why does +0.5 control for a odd number?
       # If we add +0.5, the rownumber doesn't change when "0.5*(length(vecCohensH))"
is an even number (as.integer makes from (2+0.5) -> 2),
       # but when it's an odd number it does change (as.integer makes from (2.5+0.5) -
> 3).

       png(paste0("Power_BFvsPvalue_Nsamples=",Nsamples,".png"), (xrow*1500),
(ycol*1500)) #save plot as png image

       ##  Set up plotting in xrows and ycolumns, plotting along rows first.
       par( mfrow = c(xrow,ycol))

       # get the range for the x and y axis
       xrange <- range(vecSamplesize)
       yrange <- c (0, 1)
       for (x in 1:(length(vecCohensH)-1)) # for-loop for making all the Power plots

         # Set up plot
       { plot(xrange, yrange, type="n", xlab="Samplesize",
            ylab="Proportion correct", main = paste("CohensH =", vecCohensH[(1+x)]),
cex.lab=2.5, cex.main=3)

       colours <- rainbow(NlinesPvalueBF)
       plotchar <- seq(18,18+NlinesPvalueBF,1)

       # add lines
       for (j in 1:(length(vecalfaPvalue))) #the other lines depend on the input in
'vecalfaPvalue'
       {lines(vecSamplesize, (as.vector(PvaluematricesPow[[j]][,x,1])), type = "o",
lwd=3,
              lty=1, col=colours[j], pch=plotchar[1])}

       for (k in 1:(length(vecBFkrit))) #the other lines depend on the input in
'vecBFkrit'
       {lines(vecSamplesize, (as.vector(BFmatricesPow[[k]][,x,1])), type = "o", lwd=3,
              lty=1, col=colours[j+k], pch=plotchar[1])}

       # add a legend, only in the first plot
       if (x == 1)
       {legend(xrange[1], yrange[2], c(NamevecPvalue,NamevecBF), cex=2, col=colours,
              pch=plotchar[1], lty=1)}

       } # end for-loop for making plots

       dev.off() # save the image
```

```
      ##############################################################################
#####################################
      ##################### Plot 4: Power Eq.testin
##############################################################################

      # The amount of plots we want depends on how many different CohensH-values are
imputed
      # Therefore we have to calculate the number of rows and collumns we put into
'par(mfrow = c(xrow, ycol))
      # We can calculate this with "(0.5*(length(vecCohensH)-1))". We use vecCohensH-
1 because vecCohensH[1]=0, we only want CohensH>0 in this plot
      ycol = as.integer(0.5*(length(vecCohensH)))
      xrow = as.integer(0.5*(length(vecCohensH))+0.5) #control for odd number if
necessary

      # Why does +0.5 control for a odd number?
      # If we add +0.5, the rownumber doesn't change when "0.5*(length(vecCohensH))"
is an even number (as.integer makes from (2+0.5) -> 2),
      # but when it's an odd number it does change (as.integer makes from (2.5+0.5) -
> 3).

      png(paste0("Power_Eqtest_Nsamples=",Nsamples,".png"), (xrow*1500), (ycol*1600))
#save plot as png image

      ##  Set up plotting in xrows and ycolumns, plotting along rows first.
      par( mfrow = c(xrow,ycol))

      # get the range for the x and y axis
      xrange <- range(vecSamplesize)
      yrange <- c (0, 1)
      for (x in 1:(length(vecCohensH)-1)) # for-loop for making all the Power plots

        # Set up plot
      { plot(xrange, yrange, type="n", xlab="Samplesize",
             ylab="Proportion correct", main = paste("CohensH =", vecCohensH[(1+x)]),
cex.lab=2.5, cex.main=3)

        colours <- rainbow(NlinesEq)
        plotchar <- seq(18,18+NlinesPvalueBF,1)

        # add lines
        for (j in 1:NlinesEq) #the other lines depend on the input in 'vecalfaPvalue'
        {lines(vecSamplesize, (as.vector(EqtestmatricesPow[[j]][,x,1])), type = "o",
lwd=3,
               lty=1, col=colours[j], pch=plotchar[1])}

        # add a legend, only in the first plot
        if (x == 1)
        {legend(xrange[1], yrange[2], c(NamevecEq), cex=2, col=colours,
                pch=plotchar[1], lty=1)}

      } # end for-loop for making plots

      dev.off() # save the image

      } # end if-function plot


      ######################################## final matrices
######################################################################
      #make a list with matrices for every effectsize
```

```
      matriceEffectsizes =
matrix(NA,ncol=length(vecSamplesize),nrow=(2*length(vecBFkrit)+2*length(vecalfaPvalue)
))   #design of every matrix
      matricesEffectsizes = list() #make a list called 'BFmatrices' to store the
different matrices
      for(i in 2:(length(vecCohensH))){
        newmatrix = replicate(1,matriceEffectsizes)#copy the BF matrix
        rownames(newmatrix) = c(paste0(("FA"),NamevecPvalue),
paste0(("FA"),NamevecBF), paste0(("Pow"),NamevecPvalue), paste0(("Pow"),NamevecBF))
        colnames(newmatrix) = Flipnames
        matricesEffectsizes[[as.character(vecCohensH[i])]] = newmatrix #give the BF
matrix a name, pulled from the vector with names
      }

      #fill big matrix
      for(i in 1:(length(vecCohensH)-1)){ #copy FAmatrix

matricesEffectsizes[[i]][1:(length(vecalfaPvalue)+length(vecBFkrit)),,1]=FAmatrix[1:(l
ength(vecalfaPvalue)+length(vecBFkrit)),]

        #Pow matrices
        for(j in 1:(length(vecBFkrit))){
          for(k in 1:(length(vecalfaPvalue))){
            for(l in 1:( length(vecSamplesize))){

matricesEffectsizes[[i]][(length(vecalfaPvalue)+length(vecBFkrit))+k,l,1] =
PvaluematricesPow[[k]][l,i,1]

matricesEffectsizes[[i]][(length(vecalfaPvalue)+length(vecBFkrit)+length(vecalfaPvalue
))+j,l,1] = BFmatricesPow[[j]][l,i,1]
            }}}}

      #make a matrix with the means over the four effect sizes
      meanmatriceEffectsizes =
matrix(NA,ncol=length(vecSamplesize),nrow=(2*length(vecBFkrit)+2*length(vecalfaPvalue)
))   #design of every matrix
      rownames(meanmatriceEffectsizes) = c(paste0(("FA"),NamevecPvalue),
paste0(("FA"),NamevecBF), paste0(("Pow"),NamevecPvalue), paste0(("Pow"),NamevecBF))
      colnames(meanmatriceEffectsizes) = Flipnames
      for(c in 1:length(vecSamplesize)){
        for(r in 1:(2*length(vecBFkrit)+2*length(vecalfaPvalue))){
          meanmatriceEffectsizes[r,c] =
mean(c(matricesEffectsizes[[1]][r,c,1],matricesEffectsizes[[2]][r,c,1],matricesEffects
izes[[3]][r,c,1],matricesEffectsizes[[4]][r,c,1]))
        }}

      meanHDI = matrix(NA,ncol=length(vecSamplesize),nrow=1)   #design of every
matrix
      for(c in 1:length(vecSamplesize)){
        meanHDI[c] = mean(EqtestmatricesPow$`HDI&ROPE`[c,,1])}

      meanCI = matrix(NA,ncol=length(vecSamplesize),nrow=1)   #design of every matrix
      for(c in 1:length(vecSamplesize)){
        meanCI[c] = mean(EqtestmatricesPow$EqCI [c,,1])}


      # save workspace
      if (saveworkspace == 'on')
      {save.image(paste0("~/R/win-library/3.4/", name, ".RData"))} #save workspace

      ########################  Data visualization
##############################################################################
```

```
      ##############################################################################
#########################################

      #make 4 plots for 4 different effectsizes
      dev.new()
      par(mfrow =c(2,2))
      colours =c("skyblue", rgb(1,0,0,0.6), "lightgrey", "palegreen3"
                 ,"orange","ivory4","palevioletred3")#,"","n","n","n","","n","n","n"

      for(i in 1:(length(vecCohensH)-1)){
        plot(c(0,0.12),c(0,1), yaxs= "i", type = "n", col= colours,
             xlab = "Prop False Positives", ylab = "Prop True Positives" ,  cex.lab =
1.4 )
        if (i == 1) {
          legend("top",
                 c("p-value","BF", "HDI&ROPE","CI&ROPE"), cex = 1, inset = 0.06,
                 lty = c(1,3,NA,NA), y.intersp = 0.8,
                 pch = c(19,19,2,3), pt.bg = c("white","white" ,"white" ,"white" ),
                 x.intersp = 1,
                 pt.cex = c(1.5,1.5,1.5,1.5),  lwd=2,
                 box.lty = 0)

          op = par
          legend("topright", paste0("N=",
                                    as.character(vecSamplesize[7:1])), inset = 0.01,
                 x.intersp = 0.1,
                 y.intersp = 0.7, fill=colours[7:1], border = NA,
                 box.lty = 0)
        }
        for(l in 1:( length(vecSamplesize))){

          lines(matricesEffectsizes[[i]][1:length(vecalfaPvalue),l,1],

matricesEffectsizes[[i]][(length(vecalfaPvalue)+length(vecBFkrit)+1):(length(vecalfaPv
alue)+length(vecBFkrit)+length(vecalfaPvalue)),l,1],
                type="l", lty=1, col= colours[l], lwd=2)

          lines(matricesEffectsizes[[i]][1:length(vecalfaPvalue),l,1],

matricesEffectsizes[[i]][(length(vecalfaPvalue)+length(vecBFkrit)+1):(length(vecalfaPv
alue)+length(vecBFkrit)+length(vecalfaPvalue)),l,1],
                type="p", pch=19, lty=1, col= colours[l], lwd=2)


lines(matricesEffectsizes[[i]][(length(vecalfaPvalue)+1):(length(vecalfaPvalue)+length
(vecBFkrit)),l,1],

matricesEffectsizes[[i]][((2*length(vecalfaPvalue)+length(vecBFkrit)+1):(2*length(vecB
Fkrit)+2*length(vecalfaPvalue))),l,1],
                type="b", pch=19, lty=3, col= colours[l], lwd=2)
          lines(FAmatrix[length(vecalfaPvalue)+length(vecBFkrit)+1, l], type="p",
pch=2, cex=2, col= colours[l],
                EqtestmatricesPow[[1]][l,i,1])
          lines(FAmatrix[length(vecalfaPvalue)+length(vecBFkrit)+2, l], type="p",
pch=3, cex=2, col= colours[l],
                EqtestmatricesPow$EqCI[l,i,1])

              result = paste("Effectsize Cohen's h =", vecCohensH[[i+1]])
          mtext(result,3)
        }}

      #make one plot that displays the mean over the 4 effectsizes
      plot()
```

```
       dev.new()

       #set up plot
       plot(c(0,0.12),c(0,0.8), yaxs= "i", type = "n", col= colours,
            xlab = "Prop False Positives", ylab = "Prop True Positives", cex.lab =
1.4 )

       #add legend
       legend("bottom",
              c("p-value","BF", "HDI&ROPE","CI&ROPE"), cex = 1,
              lty = c(1,3,NA,NA), y.intersp = 0.8, inset = 0.1,
              pch = c(19,19,2,3), pt.bg = c("white","white" ,"white" ,"white" ),
              x.intersp = 1,
              pt.cex = c(1.5,1.5,1.5,1.5),  lwd=2,
              box.lty = 0)

       op = par(cex= 1)
       legend("bottomright", paste0("N=",
                                    as.character(vecSamplesize[7:1])), inset = 0.01,
              y.intersp = 0.8, fill=colours[7:1], border = NA,
              box.lty = 0)

       # add lines in plot
       for(l in 1:( length(vecSamplesize))){

         lines(meanmatriceEffectsizes[1:length(vecalfaPvalue),l],

meanmatriceEffectsizes[(length(vecalfaPvalue)+length(vecBFkrit)+1):(length(vecalfaPval
ue)+length(vecBFkrit)+length(vecalfaPvalue)),l],
               type="l", pch=20, lty=1, col= colours[l], lwd=2)

         lines(meanmatriceEffectsizes[1:length(vecalfaPvalue),l],

meanmatriceEffectsizes[(length(vecalfaPvalue)+length(vecBFkrit)+1):(length(vecalfaPval
ue)+length(vecBFkrit)+length(vecalfaPvalue)),l],
               type="p", pch=19, lty=1, col= colours[l], lwd=2)


lines(meanmatriceEffectsizes[(length(vecalfaPvalue)+1):(length(vecalfaPvalue)+length(v
ecBFkrit)),l],

meanmatriceEffectsizes[((2*length(vecalfaPvalue)+length(vecBFkrit)+1):(2*length(vecBFk
rit)+2*length(vecalfaPvalue))),l],
               type="b", pch=19, lty=3, col= colours[l], lwd=2)

         lines(FAmatrix[length(vecalfaPvalue)+length(vecBFkrit)+1, l], type="p",
pch=2, cex=2, col= colours[l],
               meanHDI[l])
         lines(FAmatrix[length(vecalfaPvalue)+length(vecBFkrit)+2, l], type="p",
pch=3, cex=2, col= colours[l],
               meanCI[l])

       }
```

Appendix 7: R code functie alfa_to_BFcrit

```
      alfa_to_BFcrit <- function(N, alfa, aprior, bprior, nullvalue, ratio=
c("BF10", "BF01"))
        # by: Joukje Willemsen
        # calculate the corresponding critical bayes factor that would yield
equal "rejection regions" as the specified alfa

      {

        # only 1 output if aprior = bprior (symmetrical rejection regions):
        if (aprior == bprior) {

        x=qbinom(0.5*alfa, size=N, prob = nullvalue) #from alfa to minimal
observed number of heads to be significant
        if (2*(pbinom(x, N, nullvalue)) > alfa) {x= qbinom(0.5*alfa, size=N,
prob = nullvalue)-1}

        if (ratio=="BF01") {BFcrit= 1/(( exp( lbeta(aprior+x,bprior+N-x) -
lbeta(aprior,bprior) ) / ( nullvalue^x * (1.0-nullvalue)^(N-x) ) ))}
        if (ratio=="BF10") {BFcrit= ( exp( lbeta(aprior+x,bprior+N-x) -
lbeta(aprior,bprior) ) / ( nullvalue^x * (1.0-nullvalue)^(N-x) ) )   }
          return(BFcrit)
        }

        # otherwise 2 outputs are necessary (nonsymmetrical rejection
regions)
        else {
            xl=qbinom(0.5*alfa, size=N, prob = nullvalue, lower.tail=TRUE)
#from alfa to maximal observed number of heads to be significant (left tail)
            if (2*(pbinom(xl, N, nullvalue)) > alfa) {xl= qbinom(0.5*alfa,
size=N, prob = nullvalue, lower.tail=TRUE)-1}

            xu=qbinom(0.5*alfa, size=N, prob = nullvalue, lower.tail=FALSE)
#from alfa to minimal observed number of heads to be significant (right tail)
            if (2*(pbinom(xu, N, nullvalue)) > alfa) {xu= qbinom(0.5*alfa,
size=N, prob = nullvalue, lower.tail=FALSE)+1}

            if (ratio=="BF10") {
              A = ( exp( lbeta(aprior+xl,bprior+N-xl) -
lbeta(aprior,bprior) ) / ( nullvalue^xl * (1.0-nullvalue)^(N-xl) ) )
              lA = ( exp( lbeta(aprior+(xl-1),bprior+N-(xl-1)) -
lbeta(aprior,bprior) ) / ( nullvalue^(xl-1) * (1.0-nullvalue)^(N-(xl-1)) ) )

              B = ( exp( lbeta(aprior+xu,bprior+N-xu) -
lbeta(aprior,bprior) ) / ( nullvalue^xu * (1.0-nullvalue)^(N-xu) ) )
              uB = ( exp( lbeta(aprior+(xu+1),bprior+N-(xu+1)) -
lbeta(aprior,bprior) ) / ( nullvalue^(xu+1) * (1.0-nullvalue)^(N-(xu+1)) ) )


              if (A>lA) {signa = ">="}
              if (A<lA) {signa = "<="}

              if (B<uB) {signb = ">="}
              if (B>uB) {signb = "<="}
```

```
              return (paste("BFcritleft", signa, A, "  BFcritright", signb,
B)) }

         if (ratio=="BF01") {
              A = (1/( exp( lbeta(aprior+xl,bprior+N-xl) -
lbeta(aprior,bprior) ) / ( nullvalue^xl * (1.0-nullvalue)^(N-xl) ) ) )
             lA = (1/( exp( lbeta(aprior+(xl-1),bprior+N-(xl-1)) -
lbeta(aprior,bprior) ) / ( nullvalue^(xl-1) * (1.0-nullvalue)^(N-(xl-
1)) ) ) )

             B = (1/( exp( lbeta(aprior+xu,bprior+N-xu) -
lbeta(aprior,bprior) ) / ( nullvalue^xu * (1.0-nullvalue)^(N-xu) ) ) )
             uB = (1/( exp( lbeta(aprior+(xu+1),bprior+N-(xu+1)) -
lbeta(aprior,bprior) ) / ( nullvalue^(xu+1) * (1.0-nullvalue)^(N-
(xu+1)) ) ) )


             if (A>lA) {signa = ">="}
             if (A<lA) {signa = "<="}

             if (B<uB) {signb = ">="}
             if (B>uB) {signb = "<="}

        return (paste("BFcritleft", signa, (A), "  BFcritright", signb,
(B))) }
        }
      }
```