Electronic Supplementary Material

Effect size estimation from t-statistics in the presence of publication bias:
A brief review of existing approaches with some extensions

Rolf Ulrich
University of Tübingen
Jeff Miller
University of Otago
Edgar Erdfelder
University of Mannheim

Abstract

This document includes the appendices referred to in the main text.

Appendix A
Monte-Carlo simulation of the simple selection model

These simulations employed true effect sizes of $\delta = (0.0, 0.2, 0.4, 0.6)$, and $\alpha = 0.05$. The number of studies per meta-analysis was $k = (10, 20, 30)$. Therefore, these simulations evaluate the performance of the MLE procedure for relatively small values of k, which provides an especially challenging test for this procedure. Moreover, the sample size $n = n_1 = n_2$ also varied across the studies in each meta-analysis. Specifically, we used

1. $n = (20, 25, 30, \ldots, 65)$ for $k = 10$,
2. $n = (20, 20, 25, 25, \ldots, 65, 65)$ for $k = 20$, and
3. $n = (20, 20, 20, 25, 25, 25, \ldots, 65, 65, 65)$ for $k = 30$.

By comparison, an examination of the k-values of more than 400 published meta-analyses in social psychology reveals that k values of 22, 39, and 83 would have percentile ranks of 25 %, 50 %, and 75 %, respectively (Richard, Bond, & Stokes-Zoota, 2003).

Each row in Table A1 contains the simulation results for a particular combination of k and δ, summarized across the 2,000 simulated meta-analyses. The simulations at each combination were summarized by computing (a) the mean of the 2,000 estimates d, (b) the median of these estimates, (c) the standard deviation SD(d) of these estimates, (d) the average estimated standard error $\overline{\text{SE}(d)}$ across the 2,000 simulations, (e) the percentage of simulations fCI in which the estimated 95 % confidence interval actually covered the true effect size δ, and (f) the percentage of simulations in which the null hypothesis was rejected (i.e., the Type I error rate or the statistical power of the likelihood ratio test).

The results of these simulations enable the following conclusions. (a) The mean estimate d tends to be slightly negatively biased. However, this bias diminishes quickly with an increase of k or δ. (b) By contrast, the median d closely reflects the true value δ. The difference between the mean and median becomes smaller with an increase of k or δ, suggesting that the sampling distribution of d approaches a normal distribution. (c) The standard deviation SD(d) of the estimates, which is an estimate of the true standard error of estimate, also diminishes with an increase in either of these two parameters. (d) The average estimated standard error $\overline{\text{SE}(d)}$ closely matches the standard error observed across simulations, especially for larger values of k or δ. (e) Consistent with the pre-specified confidence coefficient of 95 %, the true effect size was included within the estimated confidence interval in approximately 95 % of all simulations. (f) The statistical power of the likelihood ratio test increases with δ and k, as one would expect. Moreover, for $\delta = 0$, the test produces approximately the pre-specified Type I error rate of $\alpha = 0.05$. An additional set of simulations showed that the estimation of δ also improves if the sample sizes of the single studies within a meta-analysis increase. In sum, the proposed MLE procedure for estimating δ from only significant studies has quite good statistical properties and thus appears to be a valuable tool for estimating the true effect size when publication is contingent on statistical significance in a specific positive direction.

Appendix B
Monte-Carlo simulation of the mixture model

As was done for the previous model, we conducted Monte-Carlo simulations to check the statistical properties of the estimates d and $\hat{p}_{sp}$ for the two-sample t-test with $\alpha = 0.05$. These simulations employed only effect sizes of $\delta = (0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6)$, since larger δ's yield nearly all significant results, making it difficult to identify p_{sp}. Furthermore, the values of k and p_{sp} were chosen to be $k = (20, 40, 60, 80)$ and $p_{sp} = (0.0, 0.2, 0.4, 0.6, 0.8, 1.0)$. Two thousand simulations were conducted for each combination of the parameters p_{sp}, δ, and k. The sample sizes $n = n_1 = n_2$ were again varied across the studies in each meta-analysis. In particular, these were

1. $n = (20, 24, 28, \ldots, 96)$ for $k = 20$,

2. $n = (20, 20, 24, 24, 28, \ldots, 96, 96)$ for $k = 40$,

3. $n = (20, 20, 20, 24, 24, \ldots, 96, 96, 96)$ for $k = 60$, and

4. $n = (20, 20, 20, 20, 24, \ldots, 96, 96, 96, 96)$ for $k = 80$.

Table A1

Results of Monte-Carlo simulation. Values of d as a function of number of studies k, and true effect size δ. mean d: the mean of the simulated estimates d. med d: the median of these estimates. SD(d): the standard deviation of these estimates. $\overline{SE(d)}$*: the average estimated standard error. fCI(δ): the percentage of simulations in which the estimated 95 % confidence interval covered the true effect size δ. Power: the percentage of simulations in which the null hypothesis was rejected.*

Parameters		Dependent Variables					
k	δ	mean d	med d	$SD(d)$	$\overline{SE(d)}$	fCI(δ)	Power
10	0.00	-0.05	-0.01	0.22	0.21	95.5	4.2
20	0.00	-0.03	-0.01	0.15	0.14	95.2	5.7
30	0.00	-0.01	-0.00	0.12	0.11	95.6	4.6
10	0.20	0.17	0.19	0.17	0.16	95.7	24.9
20	0.20	0.19	0.20	0.11	0.11	94.7	42.6
30	0.20	0.19	0.19	0.09	0.09	95.0	54.3
10	0.40	0.38	0.40	0.12	0.11	95.4	81.2
20	0.40	0.39	0.40	0.08	0.08	95.6	97.5
30	0.40	0.40	0.40	0.06	0.06	95.7	99.7
10	0.60	0.59	0.60	0.09	0.09	95.0	99.7
20	0.60	0.60	0.60	0.06	0.06	95.0	100.0
30	0.60	0.60	0.60	0.05	0.05	95.3	100.0

All simulations also examined the performance of both Egger's method and the rank correlation method in detecting a potential publication bias. Since the results were very similar for the two methods, we only report those for Egger's method.

The left three panels in Figure B1 show how the the model performs in detecting a potential publication bias as a function of p_{sp}, k, and δ. First, when no publication bias is present (i.e., $p_{sp} = 0$), the estimated Type I error rate is appropriately low, that is, lower than the nominal α-level of 0.05. Overall, the average proportion of inappropriately indicating a publication bias is 0.03, and this value is virtually independent of the true effect size δ and the number of studies k. This demonstrates that the model behaves somewhat conservatively, extending the observations of Rust, Lehmann, and Farley (1990), who also found Type I error rates below 5 % in simulations with different distributional assumptions and with identical rather than different sample sizes across all studies within each meta-analysis. Second, as one expects, the model's ability to detect a publication bias increases with k and with p_{sp}. For $p_{sp} \geq 0.6$, the model is quite powerful in detecting such a bias even when k is small. Finally, these results are virtually independent of δ, at least in the range

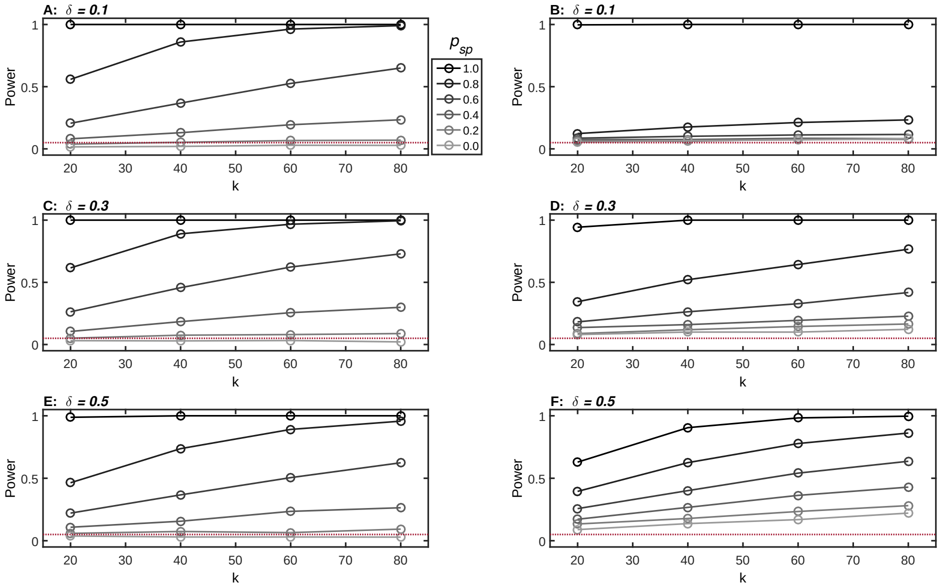

Figure B1. Statistical power of detecting a potential publication bias. The left panels A, C, and E depict the results for the mixture model and the right panels B, D, and F the results for Egger's method. Each panel shows the probability of rejecting the null hypothesis (i.e., $p_{sp} = 0$) as a function of k and p_{sp}. Note that δ increases from the top to bottom panels. The horizonal line in each panel indicates the nominal α level of 0.05.

between 0 and 0.5. For comparison, the right three panels in Figure B1 demonstrate the performance of Egger's method of testing for publication bias. First, the Type I error rate of this method clearly overshoots $\alpha = 0.05$, that is, the average Type I error rate is 0.11, and it also increases with δ. Second, this method is generally less powerful than the test provided by the mixture model. Finally, the performance of this method strongly depends on effect size δ. Except when $p_{sp} = 1$, its power is extremely low with small effects (i.e., $\delta = 0.1$), which is especially unfortunate because publication bias produces the most serious distortions of effect size estimates when the true effect is small.

Figure B2 reveals the statistical power of the model in detecting a potential effect (i.e., rejecting the null hypothesis $\delta = 0$) as a function of k, δ, and p_{sp}. Each panel shows this power for a different value of p_{sp}. First, when there is no true effect (i.e., $\delta = 0$), the estimated Type I error rate is virtually identical to the nominal α-level of 0.05. In particular, the overall average proportion of falsely indicating an effect is 0.048, and it is practically independent of p_{sp} and the number of studies k. Second, as one expects, the power increases with δ and k. As can be seen, power is high even for $k = 20$. For effect sizes of $\delta \geq 0.3$, the power is close to one. Finally,

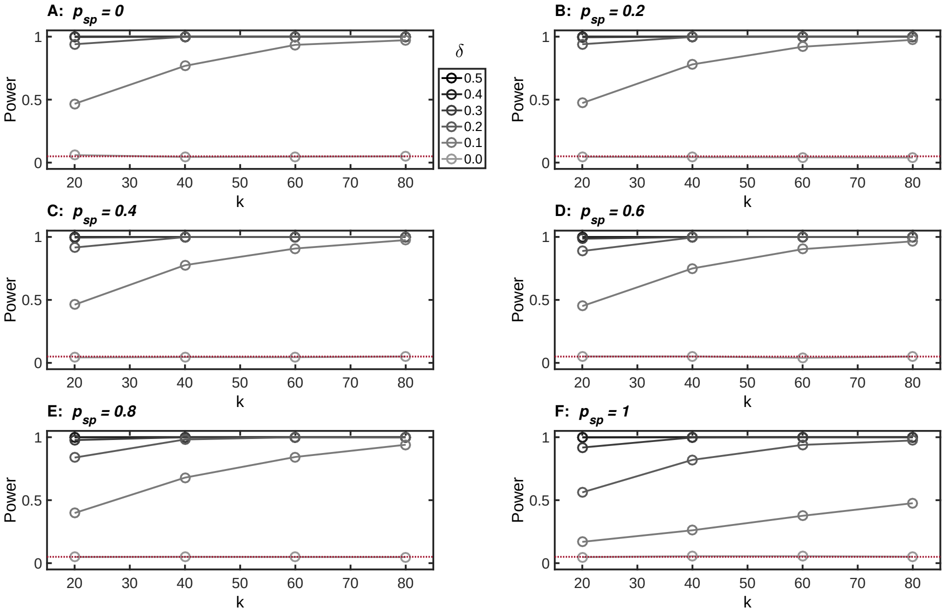

Figure B2. Statistical power of detecting a true effect. Each panel shows the probability of rejecting the null hypothesis (i.e., $\delta = 0$) as a function of k and δ. Note that p_{sp} increases from Panel A to F. The horizonal dashed line in each panel indicates the nominal α level of 0.05.

power suffers slightly for $p_{sp} = 1$, presumably because an extreme publication bias conceals information that is relevant for assessing the existence of a true effect.

Figure B3 shows the median estimate $\hat{p}_{sp}$ as a function of k, δ, and p_{sp}. Each panel plots this median against the true value p_{sp}, thus revealing whether there is a systematic bias in estimating p_{sp}. The estimates of p_{sp} are generally quite unbiased, at least for $k \geq 40$. Only for $\delta = 0$, the estimates slightly undershoot the true value of p_{sp}, that is, these estimates are somewhat conservative. However, this underestimation disappears quickly as k increases. Figure B4 shows a similar analysis for δ and clearly demonstrates that d is quite unbiased, a result that is also consistent with Rust et al. (1990).

Figure B5 shows the mean estimated coverage probability of 95 % confidence intervals for p_{sp} as a function of k and p_{sp}. Each panel plots this estimate for a different value of δ. In general, the estimated coverage probability closely resembles the target coverage probability of 0.95. However, when the true p_{sp} is either 0 or 1, the estimated 95 % CI fails to cover the true value of p_{sp}. For all other values of p_{sp}, the total average estimated coverage probability is 0.950. Therefore, when $\hat{p}_{sp}$ is close to zero or close to one, the percentile-bootstrap procedure (Hogg, McKean, & Craig, 2005) is recommended to estimate a confidence interval for p_{sp}. In that case,

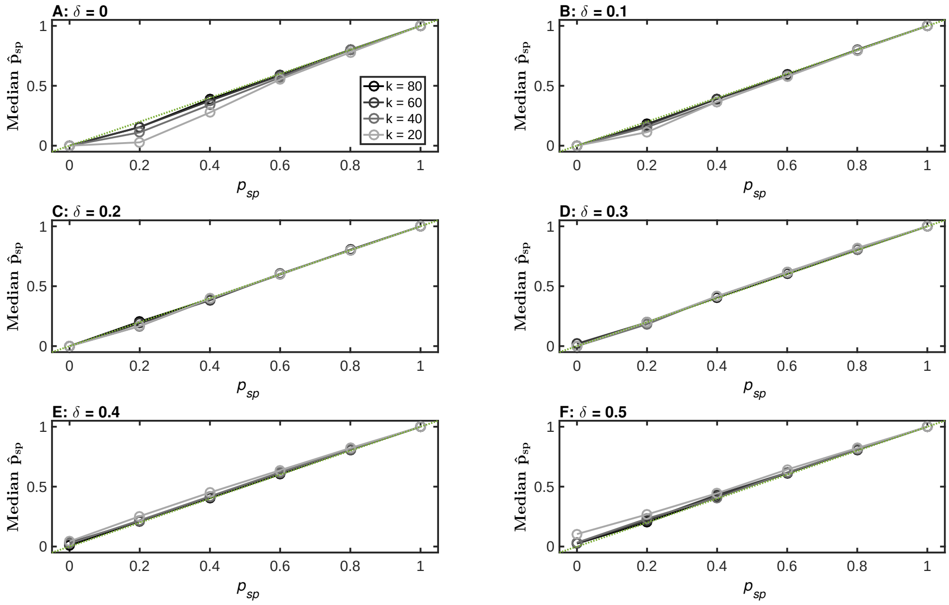

Figure B3. Median of $\hat{p}_{sp}$ as a function of k, δ, and p_{sp}. Each panel shows the median of $\hat{p}_{sp}$ plotted against the true value of p_{sp} on the x-axis. Note that δ increases from Panel A to F. Deviations from the dotted line $y = x$ indicate a systematic bias in estimating p_{sp}.

the MLE procedure needs to be repeated with at least 3,000 bootstrap samples to yield an appropriate sampling distribution for $\hat{p}_{sp}$.

Figure B6 shows the estimated coverage probability for δ. The estimated probability is close to 0.95, except when p_{sp} is one. In this case, the estimated coverage probability is too low. Therefore, if $\hat{p}_{sp}$ is very close to one, one should also use the bootstrap method to estimate the 95 % CI for δ. The total average estimated coverage probability of δ for all values $p_{sp} < 1$ is 0.956 and thus again close to the target value of 0.950.

In summary, the MLE procedure mixture model displays generally rather satisfactory properties.

1. The statistical power to detect a potential publication bias is relatively high, while the Type I error rate is kept appropriately low when there is no publication bias. This conclusion also applies to the statistical power to detect the presence of a true effect.

2. In general, the estimates of p_{sp} are satisfactory, except when p_{sp} and k are small. For $k \geq 40$, the estimates seem to be quite unbiased. By contrast, the estimate of δ seems to be unbiased in general.

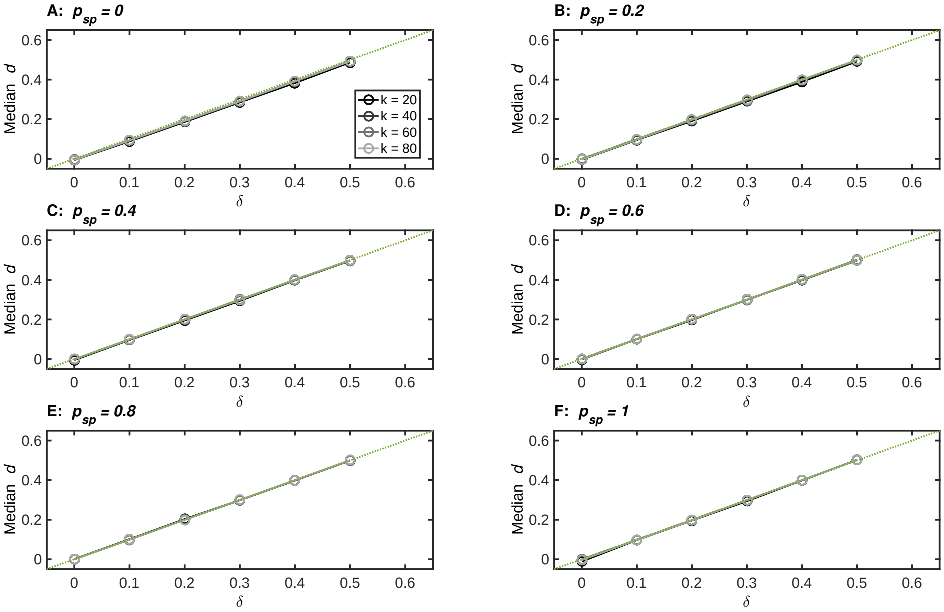

Figure B4. Median of d as a function of k, δ, and p_{sp}. Each panel shows the median of d plotted against the true value of δ on the x-axis. Note that p_{sp} increases from Panel A to F. Deviations from the dotted line $y = x$ indicate a systematic bias in estimating δ.

3. The coverage probability of the confidence interval for p_{sp} is generally satisfactory, unless p_{sp} is either close to zero or close to one. Thus when $\hat{p}_{sp}$ is close to zero or to one, bootstrapping should be used to obtain confidence intervals. The coverage probability of the confidence interval for δ is generally excellent unless p_{sp} is close to one.

Appendix C

Monte-Carlo simulation: Violation of the fixed-effect assumption

In order to assess the robustness of the present MLE procedures, we conducted two additional Monte-Carlo simulations in which the true effect size for each single study was randomly drawn from a normal distribution with mean μ_δ and variance τ^2. The first simulation explored the robustness of the simple model. Specifically, the between-study variance, τ^2, was either 0.023, 0.060, or 0.170, where $\tau^2 = 0$ represents the case of a fixed-effect model. Similar values have been used by others in previous Monte-Carlo simulations of random-effects models (see Huedo-Medina, Sánchez-Meca, Marín-Martínez, & Botella, 2006).

These selected values of τ^2 can also be linked to the I^2 index that quantifies the between-study variability relative to the total variability in meta-analyses (Higgins,

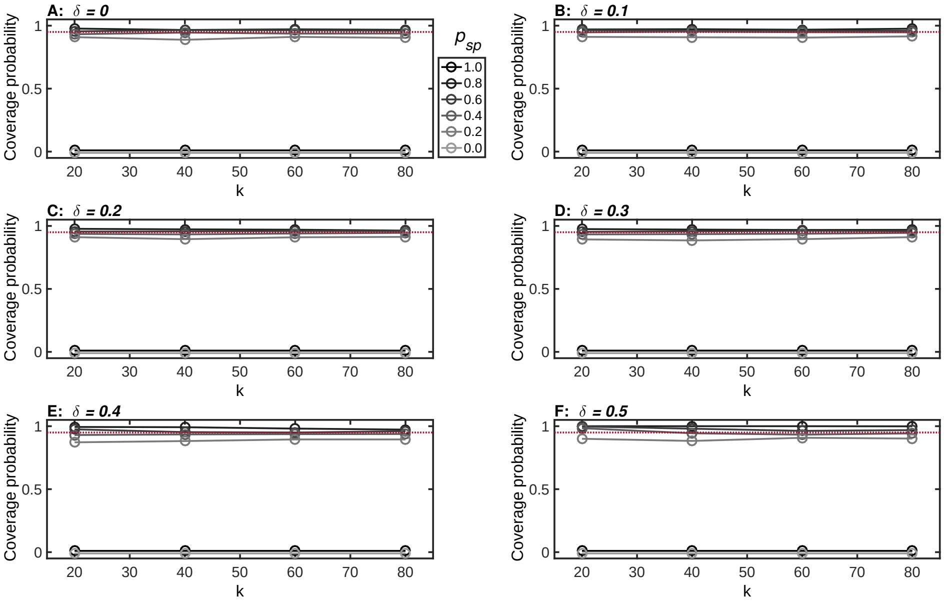

Figure B5. Estimated coverage probability of a 95 % confidence interval for p_{sp} as a function of k, δ, and p_{sp}. Each point shows the proportion of confidence intervals that covered the true p_{sp} in a simulation of 2,000 meta-analyses. Note that δ increases from Panel A to F. The dashed line marks the correct coverage probability of 0.95.

Thompson, Deeks, & Altman, 2003). Under the assumption of no publication bias, the aforementioned values of $\tau^2 = (0.0, 0.023, 0.060, 0.170)$ correspond approximately to $I^2 = (0\ \%, 25\ \%, 50\ \%, 75\ \%)$ in meta-analyses containing $k = 20$ studies with samples sizes of $n = (20, 20, 25, \ldots, 65, 65)$, as we confirmed by computer simulations similar to those of Huedo-Medina et al. (2006). Note that the index I^2 increases from 0 % to 100 % as τ^2 increases. Higgins et al. (2003) suggested that the values of $I^2 = 25\ \%, 50\ \%$, and 75 % indicate low, moderate, and high contributions of τ^2 to the total variance, respectively. In their review of 509 meta-analyses, I^2 was zero (or even negative) for almost half of these meta-analyses and thus consistent with the assumption $\tau^2 = 0$, supporting the idea that the fixed-effect assumption holds at least approximately in many cases.[1]

[1] The standard procedures (compare Borenstein, Hedges, Higgins, & Rothstein, 2009) for estimating τ and I^2 are straightforward for meta-analyses when no publication bias is present. Unfortunately, these procedures are strongly sensitive to publication bias. For example, when only significant studies are published, our simulations showed that these procedures strongly underestimate the true value of τ^2 and consequently also I^2. One major reason for this huge underestimation is that the between-study variance of d becomes smaller when only significant results enter a meta-analysis due to publication bias. It would be possible to adapt this procedure by taking into account not only this reduced variance but also the fact that truncation affects the within-study variability

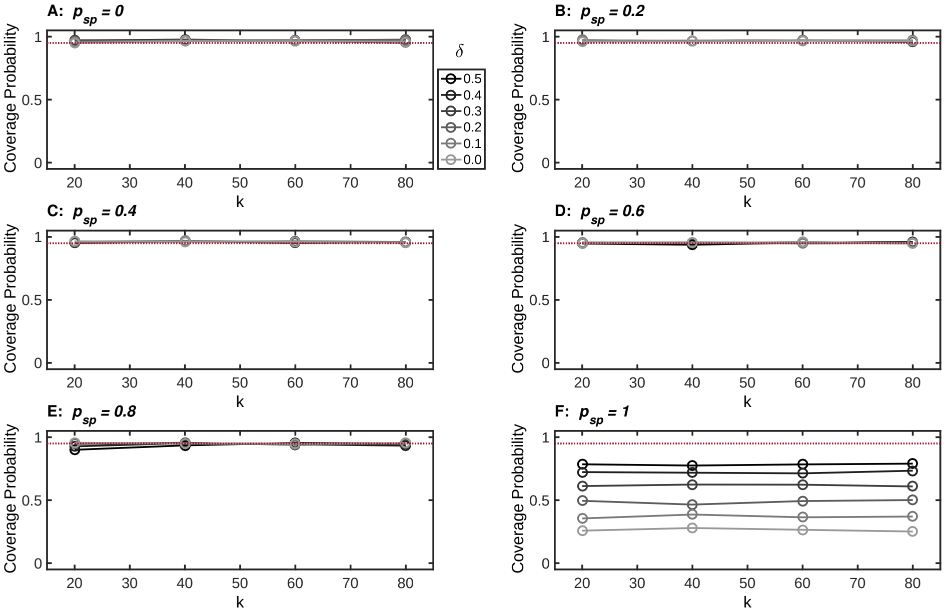

Figure B6. Estimated coverage probability of a 95 % confidence interval for δ as a function of k, δ, and p_{sp}. Each point shows the proportion of confidence intervals that covered the true δ in a simulation of 2,000 meta-analyses. Note that p_{sp} increases from Panel A to F. The dashed line marks the correct coverage probability of 0.95.

Table C1 contains the simulation results for the simple model according to which only significant results are published. The first two columns of this table depict the selected values of τ^2 and μ_δ, respectively. In many cases, the results of the model are quite robust. However, for large values of τ^2, the true effect sizes are overestimated, especially for small values of μ_δ. In this case the estimate d becomes positively biased and consequently there is a decrease in the proportion of confidence intervals containing μ_δ.

In a second simulation study, we assessed the robustness of the mixture model. The results of this study also provided reassurance that the model performs well as long as there is only mild between-study variability in δ, say $\tau < 0.15$. However, for larger values of τ the model produces more Type I errors than 5 % of the time, especially for detecting potential publication biases. In general, however, the mixture model still gives reasonable estimates of δ even when there is considerable between-study variance. The simulations also revealed that the numerical MLE procedure of this model frequently fails to converge when there is considerable between-study variability, which is usually due to numerical limitations when the t-values are very negative and thus the evaluation of the conditional t-distribution for $T > t_\alpha$ fails com-

(see Equation E.1), although such an adaptation is beyond the scope of this paper.

Table C1

Results of Monte-Carlo simulations. Across $k = 20$ simulated studies within each meta-analysis, true effect size δ was normally distributed with mean μ_δ and variance τ^2. The dependent variables are the same as in Table A1. The last column contains the estimate $\hat{\tau}^2$ obtained with the standard meta-analytic procedure for estimating τ^2.

Parameters		Dependent Variables						
τ^2	μ_δ	mean d	med d	$SD(d)$	$\overline{SE(d)}$	fCI(δ)	Power	$\hat{\tau}^2$
0.000	0.00	-0.02	-0.01	0.15	0.14	95.3	5.5	0.000
0.023	0.00	-0.00	0.01	0.15	0.14	93.5	6.3	0.000
0.060	0.00	0.06	0.08	0.14	0.13	86.3	10.9	0.000
0.170	0.00	0.21	0.22	0.10	0.10	42.9	49.8	0.001
0.000	0.20	0.18	0.20	0.11	0.11	95.1	42.8	0.000
0.023	0.20	0.21	0.23	0.11	0.10	93.0	52.9	0.000
0.060	0.20	0.27	0.28	0.09	0.09	84.0	73.5	0.000
0.170	0.20	0.38	0.38	0.07	0.08	36.9	97.9	0.006
0.000	0.40	0.39	0.40	0.08	0.08	95.2	97.4	0.000
0.023	0.40	0.42	0.42	0.08	0.08	93.0	99.2	0.000
0.060	0.40	0.46	0.47	0.07	0.07	83.7	99.9	0.002
0.170	0.40	0.53	0.53	0.06	0.07	47.9	100.0	0.025
0.000	0.60	0.60	0.60	0.06	0.06	95.6	100.0	0.001
0.023	0.60	0.61	0.61	0.06	0.06	95.2	100.0	0.003
0.060	0.60	0.63	0.63	0.05	0.06	93.3	100.0	0.012
0.170	0.60	0.67	0.67	0.05	0.06	77.0	100.0	0.057

putationally. In sum, care must be taken in interpreting d when there is considerable heterogeneity across the studies in a meta-analysis.

Appendix D
Monte-Carlo simulation: Gradual publication bias

These simulations proceeded from the assumption that the publication probability for a study depends on its significance level as specified in Equation 35. Specifically, 2,000 meta-analyses were simulated with the parameters $\delta = (0.0, 0.1, 0.2, 0.3, 0.4)$ and $k = (40, 80)$. As before, the sample sizes of each group within a meta-analysis were $n = (20, 20, 24, \ldots 96, 96)$ for $k = 40$, and $n = (20, 20, 20, 20, 24, \ldots 96, 96, 96, 96)$ for $k = 80$. A two-sample t-test was simulated for each study and the corresponding p-value calculated. This p-value determined according to Equation 35 whether or not this study was included in the meta-analysis. It is important to recognize that this selection scenario implies that the number of non-reported studies increases as sample size n decreases especially for small effect sizes. This scenario appears realistic when sample sizes vary over research projects due to different research strategies for optimizing research outcomes (Miller & Ulrich, 2016). Using the simulated data of studies selected in this fashion, the SP cutoff α within the mixture model was set to 0.05 in order to obtain the estimates $\hat{p}_{sp}$ and d.

The major results of this simulation are summarized in Table D1. The results demonstrate that the model is quite robust when its assumption of a step-like weight function (i.e., Equation 36) is violated. First, the estimates of d are only slightly too large in this case, and the power of detecting a positive effect is again high, although the Type I error rate is also slightly inflated. Second, $\hat{p}_{sp}$ tends to be larger than zero indicating that the basic mixture model is also able to detect a publication bias under this scenario despite the fact that the assumption of a true step-like weight function no longer holds. Moreover, but in agreement with the earlier simulations, the power to detect a publication bias diminishes as δ increases, basically because in that case most results are significant and so only a few outcomes would be put in the file drawer. In further simulations, we also verified that the probability of detecting a publication bias strongly increases when $P(\text{publish}|p)$ is further reduced for p-values larger than 0.05 in Equation 35.

The basic version of the mixture model also assumes that a nonsignificant result is published with probability $1 - p_{sp}$ irrespective of a study's total sample size. It seems more realistic, however, to assume that nonsignificant results are more likely to be published for studies with larger rather than smaller samples, because d can be more precisely estimated with larger samples. Therefore, we also examined the robustness of the mixture model when publication probability depends on sample size. To do that, we basically reran the previous simulations but used $p_{sp} = 0.5$ for sample sizes $n \geq 60$ per group and $p_{sp} = 0.9$ for $n < 60$. The results of these simulations are shown in Table D2. First, d recovers fairly well the true value of δ, although these estimates tend to slightly underestimate the true values. Second, and as one might expect, mean $\hat{p}_{sp}$ lies between the two values 0.5 and 0.9 of p_{sp} used in the simulations and is unaffected by the size of δ. The power to detect publication bias is almost always larger than 90 %. These results support the view that the basic

Table D1

Results of Monte-Carlo simulations in which the probability of publication increases gradually with the significance level of an outcome (see Equation 35). The first two columns show the parameters of the various simulations. The number of simulated studies was either $k = 40$ or $k = 80$. The true effect size δ was varied from 0.0 to 0.4. Each simulated study used a two-sample one-sided t-test with sample size n per group ranging from 20 to 96. In estimating the parameters within the basic mixture model, the SP cutoff α level was set to 0.05. Two thousand simulations were conducted for each combination of k and δ. The dependent variables are the mean estimates of δ and p_{sp}, the standard errors of these estimates, and the power (%) of rejecting the null hypothesis $\delta = 0$ and $p_{sp} = 0$, respectively.

Parameters		Results for δ			Results for p_{sp}		
k	δ	mean d	$SD(d)$	Power	mean $\hat{p}_{sp}$	$SD(\hat{p}_{sp})$	Power
40	0.00	0.01	0.04	9.7	0.53	0.25	30.6
40	0.10	0.14	0.04	94.0	0.39	0.26	18.6
40	0.20	0.27	0.04	100.0	0.26	0.25	8.0
40	0.30	0.35	0.04	100.0	0.24	0.24	7.4
40	0.40	0.41	0.04	100.0	0.24	0.24	6.0
80	0.00	0.01	0.03	10.2	0.56	0.20	54.4
80	0.10	0.14	0.03	99.8	0.41	0.21	33.3
80	0.20	0.26	0.03	100.0	0.25	0.21	12.5
80	0.30	0.34	0.03	100.0	0.21	0.20	8.6
80	0.40	0.42	0.03	100.0	0.20	0.19	6.3

mixture model is quite robust even under this scenario.

Appendix E
Derivation of the standard error of d

When only significant positive results enter a meta-analysis, not only is δ overestimated but also the sampling variance of d no longer corresponds to the one that is needed by meta-analysts to perform fixed-effect or random-effects meta-analyses (Borenstein et al., 2009). For example, the sampling variance of d for a two-sample t-test with only significant results would be computed with

$$\begin{aligned} Var(d|T > t_\alpha) &= \frac{n_1 + n_2}{n_1 \cdot n_2} \cdot Var\,(T|T > t_\alpha) && \text{(E.1)} \\ &= \frac{n_1 + n_2}{n_1 \cdot n_2} \cdot \int_{t_\alpha}^{\infty} [t - E(T|T > t_\alpha)]^2 \cdot f_T(t|T > t_\alpha, \epsilon, \nu)\, dt. && \text{(E.2)} \end{aligned}$$

Table D2

Results of Monte-Carlo simulations in which the probability p_{sp} depends on sample size. The first two columns show the parameters of the various simulations. The number of simulated studies was either $k = 40$ or $k = 80$. The true effect size δ was varied from 0.0 to 0.4. Each simulated study used a two-sample one-sided t-test with sample size n per group ranging from 20 to 96 and a significance level of $\alpha = 0.05$. Two thousand simulations were conducted for each combination of k and δ. The dependent variables are the mean estimates of δ and p_{sp}, the the standard error of these estimates, and the power (%) of rejecting the null hypothesis $\delta = 0$ and $p_{sp} = 0$, respectively.

Parameters		Results for δ			Results for p_{sp}		
k	δ	mean d	$SD(d)$	Power	mean $\hat{p}_{sp}$	$SD(\hat{p}_{sp})$	Power
40	0.00	-0.01	0.04	9.2	0.80	0.14	84.5
40	0.10	0.08	0.04	49.1	0.80	0.12	90.3
40	0.20	0.18	0.05	95.5	0.79	0.12	90.0
40	0.30	0.28	0.05	100.0	0.80	0.11	92.7
40	0.40	0.39	0.04	100.0	0.81	0.12	90.6
80	0.00	-0.01	0.03	9.4	0.82	0.09	98.8
80	0.10	0.08	0.03	78.8	0.81	0.08	99.6
80	0.20	0.18	0.04	99.9	0.80	0.08	99.6
80	0.30	0.28	0.04	100.0	0.80	0.08	99.4
80	0.40	0.39	0.03	100.0	0.82	0.08	99.6

In order to illustrate the preceding equation assume $n_1 = 30$, $n_2 = 20$, $\alpha = 0.05$, $\delta = 0.30$, which would yield $Var(d|T > 1.68) = 0.16^2$. The sampling variance of d in meta-analysis, however, is typically computed under the assumption of no publication bias, that is, (see Hedges & Olkin, 1985, p. 80, Equation 8)

$$\begin{aligned} Var(d) &= \frac{n_1 + n_2}{n_1 \cdot n_2} \cdot Var\,(T) && \text{(E.3)} \\ &= \frac{n_1 + n_2}{n_1 \cdot n_2} \cdot \int_{-\infty}^{\infty} [t - E(T)]^2 \cdot f_T(t|\epsilon, \nu)\, dt && \text{(E.4)} \\ &\approx \frac{n_1 + n_2}{n_1 \cdot n_2} + \frac{\delta^2}{2 \cdot (n_1 + n_2 - 3.94)}. && \text{(E.5)} \end{aligned}$$

For the above numerical example this yields $Var(d) = 0.29^2$. Thus, when a meta-analysis is performed exclusively on significant results, not only is the estimated effect size too large, but also the confidence interval surrounding this estimate would be too large.

Appendix F

Computation of the predicted proportion of significant results

According to the mixture model, nonsignificant results from the SP-path will be put into the file drawer. Consequently, significant results are predicted to be over-represented in a meta-analysis. This bias towards significant results is predicted to increase with p_{sp}. Under the assumptions of this model, it is possible to compute the predicted proportion of significant results in a meta-analysis for any combination of δ and p_{sp}, and this is given by the following conditional probability

$$\begin{aligned} P(\text{Result is significant}|\text{Result is published}) &= \frac{P(\text{PE} \cap \text{s}) + P(\text{SP} \cap \text{s})}{P(\text{Result is published})} \\ &= \frac{1 - F_T(t_\alpha|\epsilon, \nu)}{1 - p_{sp} \cdot F_T(t_\alpha|\epsilon, \nu)} \end{aligned}$$

The left panel of Figure F1 depicts this probability as a function of δ and p_{sp} for a one-sample t-test with $n = 20$. The line for $p_{sp} = 0$ gives the power of this t-test, which starts at the significance level $\alpha = 0.05$ for $\delta = 0$ and then increases gradually towards one as the effect size increases. This line reflects the proportion of *all* studies predicted to give significant results, and it serves as a baseline for evaluating the biasing effects of $p_{sp} > 0$. The remaining lines demonstrate how significant results become over-represented among published results when p_{sp} increases. This happens because, as p_{sp} increases, an increasing number of nonsignificant results from the SP-path will be put into the file drawer, as is shown in the panel on the right. As one expects, the proportion of studies in the file drawer not only increases with p_{sp} but also decreases toward zero with increasing δ. Therefore, if the SP-path is the dominant publication strategy (i.e., large p_{sp}), many results will be put into the file drawer when the true effects are small. Figure F2 shows how these results would change if n is increased from 20 to 100. First, and as one would expect, the conditional probability of a significant result given publication increases with larger n. Second, the proportion of nonsignificant results that are put into the file drawer decreases as n increases, except when $\delta = 0$.

Appendix G

Extension of the mixture-model to two-sided t-tests

The preceding models presuppose that the publication of results depends on the statistical significance *and* on the direction of the results obtained. For example, when the efficacy of a new medical intervention is statistically supported, this would be regarded as a positive and thus publishable result. Likewise, an experimental psychologist testing the hypothesis that forgetting occurs because of interference will consider the results as positive if the experimental group (with an interference task) demonstrates significantly more forgetting than the control group (without interference). In both examples, results will most likely be regarded as negative both if they are nonsignificant and if they are significant opposite to the direction of the

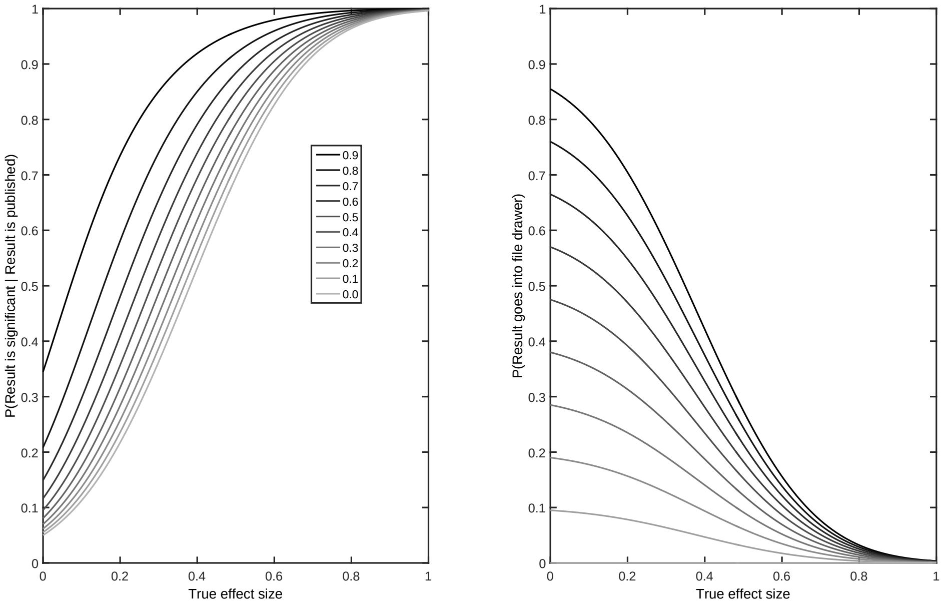

Figure F1. Left Panel: Predicted proportion of significant results in a meta-analysis (i.e., the conditional probability of a significant result given that the result is published) as a function of δ and p_{sp}. Right Panel: The compound probability $P(\mathrm{SP}\cap\mathrm{ns})$ that a result goes into the file drawer as a function of δ and p_{sp}. These results apply to a one-sided t-test with $n = 20$ and $\alpha = 0.05$.

researcher's hypothesis.

In some research scenarios, however, a researcher's hypothesis may not specify in advance the direction of results. For example, a gender psychologist may wonder whether the degree of aerophobia differs between males and females. For this research, the direction of the difference would not matter, and the results of the study would be regarded as positive and hence publishable any time that males and females differed significantly. In this case, the researcher would employ a two- rather than one-sided t-test, so the preceding mixture model would not apply. It is, however, possible to modify this model to accommodate two-sided t-tests.

According to this modification, a study taking the SP-path is only published if the resulting t-value is either smaller than $-t_{\alpha/2}$ or larger than $t_{\alpha/2}$. An analogous mathematical derivation as for the preceding mixture model yields the PDF of t values for this two-sided model version,

$$f_T(t) = \frac{f_T(t, |\epsilon, \nu) \cdot [1 - p_{sp} \cdot I(-t_{\alpha/2} < t < -t_{\alpha/2})]}{1 - p_{sp} \cdot [F_T(t_{\alpha/2}|\epsilon, \nu) - F_T(-t_{\alpha/2}|\epsilon, \nu)]}, \tag{G.6}$$

where $I = 1$ if $-t_{\alpha/2} < t < -t_{\alpha/2}$ and zero otherwise. This distribution is illustrated in

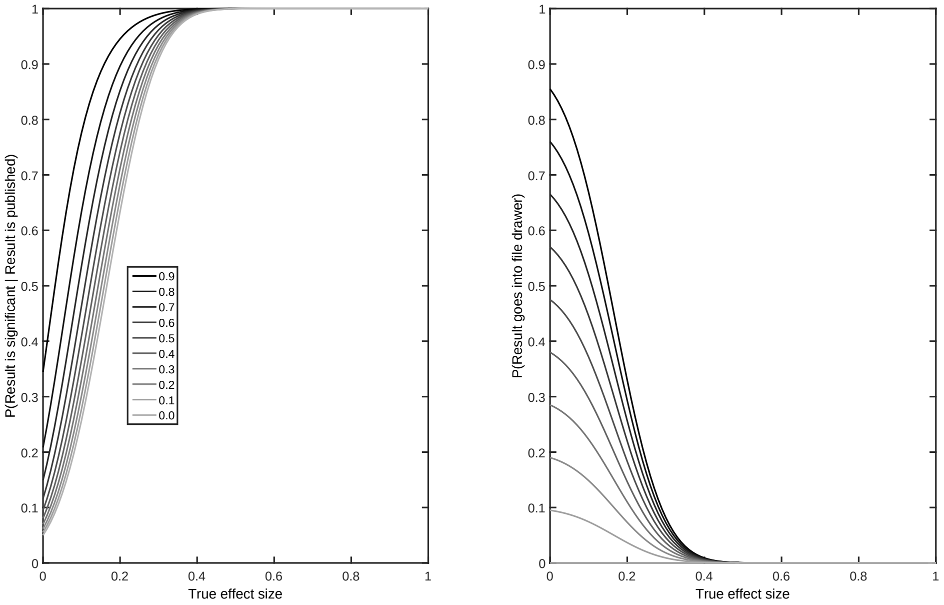

Figure F2. Left Panel: Predicted proportion of significant results in a meta-analysis (i.e., the conditional probability of a significant result given that result is published) as a function of δ and p_{sp}. Right Panel: The compound probability $P(\text{SP} \cap \text{ns})$ that a result goes into the file drawer as a function of δ and p_{sp}. These results apply to a one-sided t-test with $n = 100$ and $\alpha = 0.05$.

Figure G1, and it can be seen to resemble the model of Iyengar and Greenhouse (1988). In contrast to their formulation, however, the present version allows estimation of p_{sp}.

As one expects, this two-sided version makes different predictions for small values of δ compared to the predictions of the preceding mixture model for one-sided t-tests. However, as δ increases, the predictions of this model become virtually identical to those of the one-sided version. Moreover, for the special case of $p_{sp} = 1$, this two-sided version reduces to the model suggested by Hedges (1984), which was discussed in the Introduction of the main text. Hedges's special case assumes that only significant results, irrespective of the direction of the outcome, will be published. Therefore, the above formulation also includes Hedges's model of two-sided t-tests as a special case.

Appendix H of this supplement contains the R code for estimating δ and p_{sp} under the assumptions of this modified model. To illustrate the procedure, we used the same hypothetical example with $k = 20$ that was used to illustrate the one-sided case (Table 2). This yields $d = 0.14$, $SE = 0.05$, $CI_{95\%} = [0.04, 0.24]$, and the null hypothesis of $\delta = 0$ is rejected, $\chi^2 = 10.7$, $df = 1$, $p < .001$. Furthermore, the program outputs $\hat{p}_{sp} = 0.70$, $SE = 0.18$, $CI_{95\%} = [0.31, 0.92]$, and in this case the

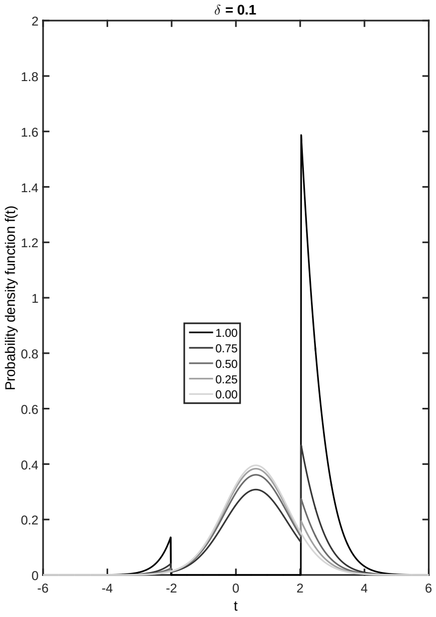

Figure G1. Probability density functions predicted by the random mixture model. Each panel depicts the PDF for various values of the mixture probability $p_{sp} = (0, 0.25, 0.50, 0.75, 1)$. Left panel: $\delta = 0.1$. Right panel: $\delta = 0.2$. The underlying t-distribution is associated with a two-sided one-sample t-test and $n = 40$.

associated likelihood ratio test of $p_{sp} = 0$ only approaches statistical significance, $\chi^2 = 3.7$, $df = 1$, $p = .057$.

Appendix H
Model 1: R code for estimating δ when all results are significant

The following R code estimates δ when only significant studies enter a meta-analysis. Note that the following code can also be downloaded: BiasR.zip and BiasMatlab.zip.

R code for two-sample *t*-test

```
EstimationTwoSampleSigOnly <- function(t,n1,n2,alpha,ConCof){

  # Significant results only, two-sample t-test

  #### Input
  #  t: vector of t-values
  #  n1,n2:  vectors of sample sizes for each study
  #  alpha: significance level (one sided test, right tail)
  #  ConCof: confidence coefficient for calculating CI

  #### Output
  #  d_est: estimate of delta
  #  SE: standard error of estimate
  #  CI: confidence interval of delta
  #  X2: square of likelihood ratio test, df=1, H0: d=0
  #  p: p-value associated with X2

  # Starting values of d for nlminb-optimization
  dset <- c(0.6, 0.3, 0.0)

  # Compute critical values for all studies
  c <-  qt(1-alpha, n1+n2-2)
  nu <- n1+n2-2
  r <- sqrt(n1*n2/(n1+n2))

  f  <-  function(d)
    -LogLike(d,t,nu,r,c)

  # Convergence check with different starting values
  j <- 0
  fval <- dmax <- rep(NA, length(dset))
  for(d0 in dset){
    j <- j+1
    tmp  <-  nlminb(d0, f)
    fval[j] <- tmp$objective
    dmax[j] <- tmp$par
  }

  y <- min(fval, na.rm = TRUE)
  i <- which.min(fval)

  if (abs(var(dmax)) > 1e-3){
    cat('Check␣starting␣value␣\n')
    print(list(t=t, t.mean=mean(t),
               d_start=dset,
               dmax=dmax))
    d_est=NA; SE=NA
    CI <- c(NA, NA)
    X2=NA; p=NA
  }

  d_est  <-  dmax[i]
  SE_CI  <-  se(d_est,t,nu,r,c,ConCof)
  X2  <-  abs(-2*(LogLike(0,t,nu,r,c)-LogLike(d_est,t,nu,r,c)))

  list( d_est=d_est,
        SE=SE_CI$SE,
        CI=SE_CI$CI,
        X2=X2,
        p=1-pchisq(X2,1)  )
}

LogLike <- Vectorize(
  function (d,t,nu,r,c){
    ncp <- d*r
    L  <-  sum(dt(t,nu,ncp, log=TRUE) -
                  pt(c,nu,ncp,lower.tail=FALSE, log.p=TRUE))
    #   For nonsignificant results only, replace the preceding lines
    #   by the following lines.
    #   L  <-  sum(dt(t,nu,ncp, log=TRUE) -
    #                  pt(c,nu,ncp, log.p=TRUE))
    L
  }, vectorize.args="d")

se <- function (d,t,nu,r,c,ConCof){
  h <- 0.0001    # differential
  g  <-  function(d) LogLike(d,t,nu,r,c)
  I <- (g(d+h) - 2*g(d) + g(d-h))/h^2 # Fisher info at x = d
  SE <- 1/sqrt(-I) # Standard error of estimate
  p1 <- (1-ConCof)/2
  p2 <- ConCof+p1
  z <- qnorm(c(p1, p2),0,1)
  CI <- c(d+z[1]*SE, d+z[2]*SE) # Confidence interval
  list(SE=SE, CI=CI)
}
```

R code for one-sample *t*-test

```
EstimationOneSampleSigOnly <- function(t,n,alpha,ConCof){

  # Significant results only, one-sample t-test

  #### Input
  # t: vector of t-values
  # n: sample size of each study
  # alpha: significance level (one sided test, right tail)
  # ConCof: confidence coefficient for calculating CI

  #### Output
  # d_est: estimate of delta
  # SE: standard error of estimate
  # CI: confidence interval of delta
  # X2: square of Likelihood ratio test, df=1, H0: d=0
  # p: p-value associated with X2

  # Starting values of d for nlminb-optimization
  dset <- c(0.6, 0.3, 0.0)

  # Compute critical values for all studies
  c <- qt(1-alpha, n-1)
  nu <- n-1
  r <- sqrt(n)

  f <- function(d)
    -LogLike(d,t,nu,r,c)

  # Convergence check with different starting values
  j <- 0
  fval <- dmax <- rep(NA, length(dset))
  for(d0 in dset){
    j <- j+1
    tmp  <-  nlminb(d0, f)
    fval[j] <- tmp$objective
    dmax[j] <- tmp$par
  }

  y <- min(fval, na.rm = TRUE)
  i <- which.min(fval)

  if (abs(var(dmax)) > 1e-3){
    cat('Check starting value \n')
    print(list(t=t, t.mean=mean(t),
               d_start=dset,
               dmax=dmax))
    d_est=NA; SE=NA
    CI <- c(NA, NA)
    X2=NA; p=NA
  }

  d_est  <-  dmax[i]
  SE_CI  <-  se(d_est,t,nu,r,c,ConCof)
  X2  <-  abs(-2*(LogLike(0,t,nu,r,c)-LogLike(d_est,t,nu,r,c)))

  list( d_est=d_est,
        SE=SE_CI$SE,
        CI=SE_CI$CI,
        X2=X2,
        p=1-pchisq(X2,1)  )
}

LogLike <- Vectorize(
  function (d,t,nu,r,c){
    ncp <- d*r
    L  <-  sum(dt(t,nu,ncp, log=TRUE) -
                 pt(c,nu,ncp,lower.tail=FALSE, log.p=TRUE))
    ## For nonsignificant results only, replace the preceding
    ## lines by the following lines.
    #   L  <-  sum(dt(t,nu,ncp, log=TRUE) -
    #                pt(c,nu,ncp, log.p=TRUE))
    L
  }, vectorize.args="d")
```

```
se <- function (d,t,nu,r,c,ConCof){
  h <- 0.0001    # differential
  g  <-  function(d) LogLike(d,t,nu,r,c)
  I  <-  (g(d+h) - 2*g(d) + g(d-h))/h^2 # Fisher info at x = d
  SE  <-  1/sqrt(-I) # Standard error of estimate
  p1 <- (1-ConCof)/2
  p2 <- ConCof+p1
  z <- qnorm(c(p1, p2),0,1)
  CI  <-  c(d+z[1]*SE, d+z[2]*SE)  # Confidence interval
  list(SE=SE, CI=CI)
}
```

Appendix I
Model 2: R code for estimating δ of the mixture model

The following R code estimates δ and probability p_{sp} when selective publishing is based on a one-sided t-test.

R code for two-sample t-test

```
EstimationTwoSampleMix <- function(t,n1,n2,alpha,ConCof){

  # mixture model, two-sample t-test

  #### Input
  # t: vector of t-values
  # n1 and n2: vectors with sample sizes of each group
  # alpha: significance level (one sided test, right tail)

  #### Output
  # d_est:   estimate of delta
  # SE_d:    standard error of d_est
  # CI_d:    confidence interval for delta
  # p_est:   estimate of p
  # SE_est:  standard error of p_est
  # CI_p:    confidence interval for p
  # X2:      Chi-square of likelihood ratio test, df=1, H0: d=0
  # p:       p-value associated with X2
  # XX2:     Chi-square of likelihood ratio test, df=1, H0: p=0
  # pp:      p-value associated with XX2

  ##  Define starting values of p and d for nlminb
  pstart <-  c(.2, .5, .8)
```

```
dstart <- c(0, .2, .5)

## Compute critical values for all studies;
#  r is needed to compute delta
nu <- n1+n2-2
c <- qt(1-alpha, nu)
r <- sqrt(n1*n2/(n1+n2))

## Estimate d and p
f1 <- function(x)
  -LogLike(x[1],x[2],t,nu,r,c)

# Checks local minima for different starting values defined above
fmax <- Inf
i <- 0
fs <- rep(NA, length(dstart))
for (p0 in pstart){
  for (d0 in dstart){
    i <- i+1
    # many warnings due to precision (does not affect results)
    suppressWarnings(
      tmp <- nlminb(c(d0, log(p0/(1-p0))), f1)
    )
    fs[i] <- tmp$objective

    if (tmp$objective<fmax){
      d_est <- tmp$par[1]
      l_est <- tmp$par[2]
      fmax <- tmp$objective
    }
  }
}
if (sd(fs)>1e-3){
  cat('Check starting values\n')
  print(fs)
}

p_est <- 1/(1+exp(-l_est))
SEtmp <- se(d_est,p_est,t,nu,r,c)
SE_d <- SEtmp$SE_d
SE_l <- SEtmp$SE_l
SE_p <- SEtmp$SE_p
```

```
  ## Confidence Intervals
  p1 <- (1-ConCof)/2
  p2 <- ConCof+p1
  z <- qnorm(c(p1, p2),0,1)
  CI_d  <-  d_est+z*SE_d
  CI_l  <-  l_est+z*SE_l
  CI_p  <-  1/(1+exp(-CI_l))

  ## Compute likelihood ratio test for d
  f0 <- function(y) -LogLike(0,y,t,nu,r,c)
  y <- nlminb(log(p_est/(1-p_est)),f0)$par
  p_0 <- 1/(1+exp(-y))
  X2 <- -2*(LogLike(0,log(p_0/(1-p_0)),t,nu,r,c) -
            LogLike(d_est,log(p_est/(1-p_est)),t,nu,r,c))
  p <- 1-pchisq(abs(X2),1)

  ## Compute likelihood ratio test for p
  f0 <- function(y) -LogLikeP0(y,t,nu,r)
  y <- nlminb(d_est,f0)$par
  XX2 <- -2*(LogLikeP0(y,t,nu,r) -
             LogLike(d_est,log(p_est/(1-p_est)),t,nu,r,c))
  pp <- 1-pchisq(abs(XX2),1)

  list(d_est=d_est, SE_d=SE_d, CI_d=CI_d,
       p_est=p_est, SE_p=SE_p, CI_p=CI_p,
       X2=X2, p=p, XX2=XX2, pp=pp)
}

## Fisher Info for d and p se <- function(d,p,t,nu,r,c){
  h <- 0.0001
  if (p+h > 1)
    p <- p-h
  if (p-h < 0)
    p <- p+h

  I <- matrix(NA, 2,2)
  f <- function(d,p) LogLike(d,log(p/(1-p)),t,nu,r,c)
  I[1,1] <-  -(f(d+h,p)-2*f(d,p)+f(d-h,p))/h^2
  I[2,2] <-  -(f(d,p+h)-2*f(d,p)+f(d,p-h))/h^2
  I[1,2] <-  -(f(d+h,p+h)+f(d-h,p-h)-f(d-h,p+h)-f(d+h,p-h))/(4*h^2)
  I[2,1] <-  I[1,2]
  Cov <- solve(I)
  SE_d <- abs(sqrt(Cov[1,1]))
  SE_p <- abs(sqrt(Cov[2,2]))

  l <- log(p/(1-p))
  ff <- function(d,l) LogLike(d,l,t,nu,r,c)
  I[1,1] <-  -(ff(d+h,l)-2*ff(d,l)+ff(d-h,l))/h^2
  I[1,2] <-  -(ff(d+h,l+h)+ff(d-h,l-h)-ff(d-h,l+h)-ff(d+h,l-h))/(4*h^2)
  I[2,1] <-  I[1,2]
  I[2,2] <-  -(ff(d,l+h)-2*ff(d,l)+ff(d,l-h))/h^2
  Cov <- solve(I)
  SE_l <- abs(sqrt(Cov[2,2]))

  list(SE_d=SE_d, SE_p=SE_p, SE_l=SE_l)
}

LogLike <- Vectorize(
  function (d,l,t,nu,r,c){
    p <- 1/(1+exp(-l))
    ncp <- d*r
    ## for extreme values, precision of dt() / pt() causes warnings
    MLE <- sum(log(MixPdf(t,p,nu,ncp,c)))
    MLE
  }, vectorize.args=c("d", "l"))

MixPdf <- Vectorize(
  function (t,p,nu,ncp,c){
    beta <- pt(c,nu,ncp)
    A <- 1-p*beta
    I <- ifelse(t<=c, 1, 0)
    # For the mixture model with suppressing significant results,
    #  replace the next lines with the preceding lines.
    # A <- 1-p*(1-beta)
    # ifelse(t<=c, 0, 1)
    f <- dt(t,nu,ncp)*(1-p*I)/A
    f
  }, vectorize.args=c("t", "nu", "ncp","c"))

LogLikeP0 <- Vectorize(
  function(d,t,nu,r){
    ncp <- d*r
    L <- sum(dt(t,nu,ncp, log= TRUE))
    L
  }, vectorize.args=c("d"))
```

R code for one-sample *t*-test

```
EstimationOneSampleMix <- function(t,n,alpha,ConCof){

  #Mixture model, one-sample t-test

  #### Input
  # t: vector of t-values
  # n: vector of sample sizes
  # alpha: significance level (one sided test, right tail)

  #### Output
  # d_est:   estimate of delta
  # SE_d:    standard error of d_est
  # CI_d:    confidence interval for delta
  # p_est:   estimate of p
  # SE_est:  standard error of p_est
  # CI_p:    confidence interval for p
  # X2:      Chi-Square of likelihood ratio test, df=1, H0: d=0
  # p:       p-value associated with X2
  # XX2:     Chi Square of likelihood ratio test, df=1, H0: p=0
  # pp:      p-value associated with XX2

  ##  Define starting values of p and d for nlminb
  pstart <-  c(.2, .5, .8)
  dstart <- c(0, .2, .5)

  ## Compute critical values for all studies;
  #  r is needed to compute delta
  nu <- n-1
  c <- qt(1-alpha, nu)
  r <- sqrt(n)

  ##  Estimate d and p
  f1  <-  function(x)
    -LogLike(x[1],x[2],t,nu,r,c)

  # Checks local minima for different starting values defined above
  fmax <- Inf
  i <- 0
```

```
fs <- rep(NA, length(dstart))
for (p0 in pstart){
  for (d0 in dstart){
    i <- i+1
    # many warnings due to precision (does not affect results)
    suppressWarnings(
      tmp <- nlminb(c(d0, log(p0/(1-p0))), f1)
    )
    fs[i] <- tmp$objective

    if (tmp$objective<fmax){
      d_est <- tmp$par[1]
      l_est <- tmp$par[2]
      fmax <- tmp$objective
    }
  }
}
if (sd(fs)>1e-3){
  cat('Check starting values\n')
  print(fs)
}

p_est <- 1/(1+exp(-l_est))
SEtmp <- se(d_est,p_est,t,nu,r,c)
SE_d <- SEtmp$SE_d
SE_l <- SEtmp$SE_l
SE_p <- SEtmp$SE_p

##  Confidence Intervals
p1 <- (1-ConCof)/2
p2 <- ConCof+p1
z <- qnorm(c(p1, p2),0,1)
CI_d  <-  d_est+z*SE_d
CI_l  <-  l_est+z*SE_l
CI_p  <-  1/(1+exp(-CI_l))

## Compute likelihood ratio test for d
f0 <- function(y) -LogLike(0,y,t,nu,r,c)
y <- nlminb(log(p_est/(1-p_est)),f0)$par
p_0 <- 1/(1+exp(-y))
X2 <- -2*(LogLike(0,log(p_0/(1-p_0)),t,nu,r,c) -
            LogLike(d_est,log(p_est/(1-p_est)),t,nu,r,c))
  p <- 1-pchisq(abs(X2),1)

  ## Compute likelihood ratio test for p
  f0 <- function(y) -LogLikeP0(y,t,nu,r)
  y <- nlminb(d_est, f0)$par
  XX2 <- -2*(LogLikeP0(y,t,nu,r) -
               LogLike(d_est,log(p_est/(1-p_est)),t,nu,r,c))
  pp <- 1-pchisq(abs(XX2),1)

  list(d_est=d_est, SE_d=SE_d, CI_d=CI_d,
       p_est=p_est, SE_p=SE_p, CI_p=CI_p,
       X2=X2, p=p, XX2=XX2, pp=pp)
}

## Fisher Info for d and p se <- function(d,p,t,nu,r,c){
  h <- 0.0001
  if (p+h > 1)
    p <- p-h
  if (p-h < 0)
    p <- p+h

  I <- matrix(NA, 2,2)
  f <- function(d,p) LogLike(d,log(p/(1-p)),t,nu,r,c)
  I[1,1] <- -(f(d+h,p)-2*f(d,p)+f(d-h,p))/h^2
  I[2,2] <- -(f(d,p+h)-2*f(d,p)+f(d,p-h))/h^2
  I[1,2] <- -(f(d+h,p+h)+f(d-h,p-h)-f(d-h,p+h)-f(d+h,p-h))/(4*h^2)
  I[2,1] <- I[1,2]
  Cov <- solve(I)
  SE_d <- abs(sqrt(Cov[1,1]))
  SE_p <- abs(sqrt(Cov[2,2]))

  l <- log(p/(1-p))
  ff <- function(d,l) LogLike(d,l,t,nu,r,c)
  I[1,1] <- -(ff(d+h,l)-2*ff(d,l)+ff(d-h,l))/h^2
  I[1,2] <- -(ff(d+h,l+h)+ff(d-h,l-h)-ff(d-h,l+h)-ff(d+h,l-h))/(4*h^2)
  I[2,1] <- I[1,2]
  I[2,2] <- -(ff(d,l+h)-2*ff(d,l)+ff(d,l-h))/h^2
  Cov <- solve(I)
  SE_l <- abs(sqrt(Cov[2,2]))

  list(SE_d=SE_d, SE_p=SE_p, SE_l=SE_l)
}
```

```
LogLike <- Vectorize(
  function (d,l,t,nu,r,c){
    p <- 1/(1+exp(-l))
    ncp <- d*r
    ## for extreme values, precision of dt() / pt() causes warnings
    MLE <- sum(log(MixPdf(t,p,nu,ncp,c)))
    MLE
  }, vectorize.args=c("d", "l"))

MixPdf <- Vectorize(
  function (t,p,nu,ncp,c){
    beta <- pt(c,nu,ncp)
    A <- 1-p*beta
    I <- ifelse(t<=c, 1, 0)
    # For the mixture model with suppressing significant results,
    #  replace the next lines with the preceding lines.
    # A <- 1-p*(1-beta)
    # ifelse(t<=c, 0, 1)
    f <- dt(t,nu,ncp)*(1-p*I)/A
    f
  }, vectorize.args=c("t", "nu", "ncp","c"))

LogLikeP0 <- Vectorize(
  function(d,t,nu,r){
    ncp <- d*r
    L <- sum(dt(t,nu,ncp, log= TRUE))
    L
  }, vectorize.args=c("d"))
```

Appendix J

Model 3: R code for estimating δ of the mixture model for a two-sided t-test

In order to estimate δ and p_{sp} for a two-sided t-test, the function MixPdf in the preceding code of Model 2 has to be replaced by the following function and α has to be set to 0.025 instead to 0.05.

```
  MixPdf<-Vectorize(
    function(t,p,nu,ncp,c){
      A <- dt(t,nu,ncp)
      B <- pt(c,nu,ncp)
      C <- pt(-c,nu,ncp)
      publish <- 1-p*(B-C)
      I <- ifelse(abs(t)>=c,0,1)
```

```
        f <- A/publish*(1-p*I)
        f
    }, vectorize.args=c("t","nu","ncp","c"))
```

References

Borenstein, M., Hedges, L. V., Higgins, J. P. T., & Rothstein, H. R. (2009). *Introduction to meta-analysis.* Chichester, England: Wiley & Sons. doi: 10.1002/9780470743386

Hedges, L. V. (1984). Estimation of effect size under nonrandom sampling: The effects of censoring studies yielding statistically insignificant mean differences. *Journal of Educational and Behavioral Statistics*, *9*, 61–85. doi: 10.3102/10769986009001061

Hedges, L. V., & Olkin, I. (1985). *Statistical methods for meta-analysis.* Orlando, Florida: Academic Press, Inc.

Higgins, J. P. T., Thompson, S. G., Deeks, J. J., & Altman, D. G. (2003). Measuring inconsistency in meta-analyses. *BMJ: British Medical Journal*, *327*(7414), 557–560. doi: 10.1136/bmj.327.7414.557

Hogg, R. V., McKean, J., & Craig, A. T. (2005). *Introduction to mathematical statistics* (6th ed.). New Dehli: Pearson Prentice Hall.

Huedo-Medina, T. B., Sánchez-Meca, J., Marín-Martínez, F., & Botella, J. (2006). Assessing heterogeneity in meta-analysis: Q statistic or I2 index? *Psychological Methods*, *11*, 193–206. doi: 10.1037/1082-989X.11.2.193

Iyengar, S., & Greenhouse, J. B. (1988). Selection models and the file drawer problem. *Statistical Science*, *3*, 133–135. doi: 10.1214/ss/1177013019

Miller, J., & Ulrich, R. (2016). Optimizing research payoff. *Perspectives on Psychological Science*, *11*, 664-691. doi: 10.1177/1745691616649170

Richard, F. D., Bond, C. F., & Stokes-Zoota, J. J. (2003). One hundred years of social psychology quantitatively described. *Review of General Psychology*, *7*, 331–363. doi: 10.1037/1089-2680.7.4.331

Rust, R. T., Lehmann, D. R., & Farley, J. U. (1990). Estimating publication bias in meta-analysis. *Journal of Marketing Research*, *27*, 220–226. Retrieved from `www.jstor.org/stable/3172848`