Justifying Your Alpha by Minimizing or Balancing Error Rates

Daniel Lakens

12-5-2019

In 1957 Neyman writes: “it appears desirable to determine the level of significance in accordance with quite a few circumstances that vary from one particular problem to the next.” Despite this good advice, social scientists developed the norm to always use an alpha level of 0.05 as a threshold when making predictions. Doing so is against the principles of Neyman-Pearson hypothesis testing. In this blog post I will explain how you can set the alpha level so that it minimizes the combined Type 1 and Type 2 error rates (thus efficiently making decisions), or balancing Type 1 and Type 2 error rates, given some smallest effect size of interest and a sample size. You can use this approach to justify your alpha level, and guide your thoughts about how to design studies efficiently.

Neyman (1933) provides an example of the reasoning process he believed researchers should go through. He explains how a researcher might have derived an important hypothesis that H0 is true (there is no effect), and will not want to ‘throw it aside too lightly’ by setting the alpha level to 0.01. In another line of research, an experimenter might be interesting in detecting factors that would lead to the modication of a standard law, where the “importance of finding some new line of development here outweighs any loss due to a certain waste of effort in starting on a false trail”, and Neyman suggests to set the alpha level to 0.1.

Miller and Ulrich (2019) explain how the quest for an optimal alpha depends on the relative seriousness of Type 1 errors (deciding there is something, when there is nothing) and Type 2 errors (deciding there is nothing, when there is something), as well as prior beliefs about the probability the null-hypothesis is true. They show how under some scenarios 0.05 is a reasonable choice for an alpha level, but under other scenarios it is not.

# Which is worse? A Type 1 Error or a Type 2 Error?

As you perform lines of research the data you collect are used as a guide to continue or abandon a hypothesis, to use one paradigm or another. The goal of well-designed experiments is to control the error rates as you make these decisions, so that you do not fool yourself too often in the long run.

Many researchers implicitly assume, without any justification, that Type 1 errors are more problematic than Type 2 errors. Cohen (1988) suggested a Type 2 error rate of 20%, and hence to aim for 80% power, but wrote “.20 is chosen with the idea that the general relative seriousness of these two kinds of errors is of the order of .20/.05, i.e., that Type I errors are of the order of four times as serious as Type II errors. This .80 desired power convention is offered with the hope that it will be ignored whenever an investigator can find a basis in his substantive concerns in his specific research investigation to choose a value ad hoc”. More recently, researchers have argued that false negative constitute a much more serious problem in science (Fiedler, Kutzner, & Krueger, 2012). I always ask my 3rd year bachelor students: What do you think?

Intermezzo: Last year I listened to a researchers at a German institute that evaluated new therapies, and made the decision of whether they would be covered by the German healthcare system. She discussed Eye Movement Desensitization and Reprocessing (EMDR) therapy. I once worked as a research assistant, analyzing data from EMDR therapy after a large firework factory exploded in Enschede, The Netherlands, in 2000. I knew the evidence that the therapy worked was very weak, and as the talk started, I hoped they had decided not to cover it under healthcare. They did, and the researchers convinced me this was a good decision. She said that, although no strong enough evidence was available that it works, the costs of the therapy (which can be done behind a computer) are very low, it was applied in settings where no really good alternatives were available (e.g., inside prisons), and risk of negative consequences was basically zero. In other words, they were aware of the fact that there was a very high probability that EMDR was a Type 1 error, but that compared to the cost of a Type 2 error, it was still better to accept the treatment.

# Balancing or minimizing error rates

Whether a Type 1 error or a Type 2 error is more costly depends, and you should make this decision for your own research line. Especially in lines of research (e.g., when you are working on a four study paper) which contain replication and extension studies, you might come to the conclusion that both Type 1 and Type 2 error matter but not in a 4:1 ratio. There might not be strong reasons to consider the one more serious than the other. Following up on a Type 1 error would eventually tell you an interesting idea is a null-effect after all. But not following up on a good idea after a Type 2 error, when that idea would have been of great interest to the scientific community, can be much more costly.

Mudge, Baker, Edge, and Houlahan (2012) explain how researchers might want to minimize the total combined error rate. This would make decision making overal most efficient. It requires researchers to choose and alpha level that, when using in the power analysis, leads to the lowest combined error rate. For example, with a 5% alpha and 80% power, the combined error rate is 25%, and if power is 99% and the alpha is 5% the combined error rate is 6%. Mudge and colleagues show that the alpha level can be chosen in ways that lower the combined error rate. This is one of the approaches we mentioned in our ‘Justify Your Alpha’ paper from 2018.

When we wrote ‘Justify Your Alpha’ we knew it would be a lot of work to actually develop methods that people can use. For months, I would occasionally revisit the code by Mudge and colleagues, which is an adaptation of the pwr library in R, but the code was too complex and I could not get to the bottom of how it worked. After leaving this aside for some months and improving my R skills, yesterday I took a long shower and suddenly realized I did not need to understand the code by Mudge and colleagues. Instead of getting their code to work, I could just write my own code from scratch (such realizations are my justification for taking showers that are longer than is environmentally friendly).

If you want to balance or minimize error rates, the tricky thing is that the alpha level you set determined the Type 1 error rate, but through it’s influence on the statistical power, als influenced the Type 2 error rate. So I wrote a function that examines the range of possible alpha levels (from 0 to 1) and minimizes either the total error (Type 1 + Type 2) or minimizes the difference between the Type 1 and Type 2 error rates, balncing the error rates. It then returns the alpha (Type 1 error rate) and the beta (Type 2 error). You can enter any analytic power function that normally works in R and would output the calculated power.

# Minimizing Error Rates

Below is the version of the optimal\_alpha function used in this blog. Yes, I am defining a function inside another function and this could all look a lot prettier - but I don’t know how so send me a push request if you do.

optimal\_alpha <- function(power\_function, costT1T2 = 1, prior\_H1H0 = 1, error = "minimal") {  
 #Define the function to be minimized  
 f = function(x, power\_function, costT1T2 = 1, prior\_H1H0 = 1, error = "minimal") {  
 y <- 1 - eval(parse(text=paste(power\_function)))  
 # print(c(x, y, x+y)) #optional: print alpha, beta, and objective  
 if(error == "balance"){  
 max((costT1T2\*x - prior\_H1H0\*y)/(prior\_H1H0+1), (prior\_H1H0\*y - costT1T2\*x)/(prior\_H1H0+1))  
 } else if (error == "minimal"){  
 2\*(costT1T2\*x + prior\_H1H0\*y)/(prior\_H1H0+1)  
 }  
 }  
 #Run optimize to find the minimum  
 res <- optimize(f,   
 c(0, 1),   
 tol = 0.00001,  
 power\_function = power\_function,  
 costT1T2 = costT1T2,   
 prior\_H1H0 = prior\_H1H0,   
 error = error)  
 if(error == "balance"){  
 beta <- res$minimum - res$objective  
 } else if (error == "minimal"){  
 beta <- res$objective - res$minimum  
 }  
 #Store results  
 invisible(list(alpha = res$minimum,  
 beta = beta,  
 tot = res$objective  
 )  
 )  
}

The code requires requires you to specify the power function (in a way that the code *returns* the power, hence the $power at the end) for your test, where the significance level is a variable ‘x’. In this power function you specify the effect size (such as the smallest effect size you are interested in) and the sample size. In my experience, sometimes the sample size is determined by factors outside the control of the researcher. For example, you are working with a existing data, or you are studying a sample size that is limited (e.g., all students in a school). Other times, people have a maximum sample size they can feasibly collect, and accept the error rates that follow from this feasibility limitation. If your sample size is not limited, you can increase the sample size until you are happy with the error rates.

The code calculates the Type 2 error (1-power) across a range of alpha values. For example, we want to calculate the optimal alpha level for a independent *t*-test. Assume our smallest effect size of interest is d = 0.5, and we are planning to collect 100 participants in each group. We would normally calculate power as follows:

pwr.t.test(d=0.5, n=100, sig.level = 0.05, type='two.sample', alternative='two.sided')$power

This analysis tells us that we have 94% power with a 5% alpha level for our smallest effect size of interest, d = 0.5, when we collect 100 participants in each condition.

If we want to minimize our total error rates, we would enter this function in our optimal\_alpha function (while replacing the sig.level argument with ‘x’ instead of 0.05, because we are varying the value to determine the lowest combined error rate).

res <- optimal\_alpha(power\_function = "pwr.t.test(d=0.5, n=100, sig.level = x, type='two.sample', alternative='two.sided')$power")  
  
res$alpha

## [1] 0.05101728

res$beta

## [1] 0.05853977

We see that an alpha level of 0.051 slightly improved the overal error rate, since it will lead to a Type 2 error rate of 0.059 for a smallest effect size of interest of d = 0.5.

What would happen if we had decided to collect 200 participants per group, or only 50? With 200 participants per group we would have more than 99% power for d = 0.05, and relatively speaking, a 5% Type 1 error with a 1% Type 2 error is slightly out of balance. In the age of big data, we nevertheless see this all the time. Large sample sizes, where people have huge power, but they still solemnly stick to the 5% Type 1 error rate, even though the combined error rates can be smaller, if the alpha level was lowered. If we just replace 100 by 200 in the function above, we see the combined Type 1 and Type 2 error rates are the lowest if we set the alpha level to 0.00866. I think it makes a lot of sense to lower the alpha level, when not doing so would lead to an imbalance in the ratio between Type 1 and Type 2 errors.

If our maximum sample size we were willing to collect was 50 per group, the optimal alpha level to reduce the combined Type 1 and Type 2 error rates is 0.13. This means that we would have a 13% probability of deciding there is an effect when the null hypothesis is true. This is quite high! However, is we had used an 5% Type 1 error rate, the power would have been 69.69%, with a 30.31% Type 2 error rate, while the Type 2 error rate is ‘only’ 16.56% after increasing the alpha level to 0.13. We increase the Type 1 error rate by 8%, to reduce the Type 2 error rate by 13.5%.

# Balancing Error Rates

You can choose to minimize the combined error rates, but you can also decide that it makes most sense to you to balance the error rates. For example, you think a Type 1 error is just as problematic as a Type 2 error, and therefore, you want to design a study that has balanced error rates for a smallest effect size of interest (e.g., a 5% Type 1 error rate and a 95% Type 2 error rate). Whether to minimize error rates or balance them can be specified in an additional argument in the function. The default it to minimize, but by adding error = "balance" an alpha level is given so that the Type 1 error rate equals the Type 2 error rate.

res <- optimal\_alpha(power\_function = "pwr.t.test(d=0.5, n=100, sig.level = x, type='two.sample', alternative='two.sided')$power", error = "balance")  
  
res$alpha

## [1] 0.05488516

res$beta

## [1] 0.05488402

Repeating our earlier example, the alpha level is 0.055, such that the Type 2 error rate, given the smallest effect size of interest and the and the sample size, is also 0.055. I feel that even though this does not minimize the overal error rates, it is a justification strategy for your alpha level that often makes sense. If both Type 1 and Type 2 errors are equally problematic, we design a study where they are weighed equally.

# Relative costs and prior probabilities

So far we have assumed a Type 1 error and Type 2 error are equally problematic. But you might believe Cohen (1988) was right, and Type 1 errors are exactly 4 times as bad as Type 2 errors. Or you might think they are twice as problematic, or 10 times as problematic. However you weigh them, as explained by Mudge et al., 2012, and Ulrich & Miller, 2019, you should incorporate those weights into your decisions.

The function has another optional argument, costT1T2, that allows you to specify the relative cost of Type1:Type2 errors. By default this is set to 1, but you can set it to 4 (or any other value) such that Type 1 errors are 4 times as costly as Type 2 errors. This will change the weight of Type 1 errors compared to Type 2 errors, and thus also the choice of the best alpha level.

res <- optimal\_alpha(power\_function = "pwr.t.test(d=0.5, n=100, sig.level = x, type='two.sample', alternative='two.sided')$power", error = "minimal", costT1T2 = 4)  
  
res$alpha

## [1] 0.01918735

res$beta

## [1] 0.1787393

Now, the alpha level that minimized the *weighted* Type 1 and Type 2 error rates is 0.019.

Similarly, you can take into account prior probabilities that either the null is true (and you will observe a Type 1 error), or that the alternative hypothesis is true (and you will observe a Type 2 error). By incorporating these expectations, you can minimize or balance error rates in the long run (assuming your priors are correct). Proprs can be specified using the prior\_H1H0 argument, which by default is 1 (H1 and H0 are equally likely). Setting it to 4 means you think the alternative hypothesis (and hence, Type 2 errors) are 4 times more likely than that the null hypothesis is true (and hence, Type 1 errors).

res <- optimal\_alpha(power\_function = "pwr.t.test(d=0.5, n=100, sig.level = x, type='two.sample', alternative='two.sided')$power", error = "minimal", prior\_H1H0 = 4)  
  
res$alpha

## [1] 0.1183584

res$beta

## [1] -0.03128996

If you think H1 is four times more likely to be true than H0, you need to worry less about Type 1 errors, and now the alpha that minimizes the weighted error rates is 0.118. It is always difficult to decide upon priors (unless you are Omniscient Jones) but even if you ignore them, you are making the decision that H1 and H0 are equally plausible.

# Conclusion