

Tutorial: Addressing researcher degrees of freedom through minP adjustment

Maximilian Mandl

18-06-2024

1 General idea and intuition: minP

To gain an initial intuition for the minP procedure, we introduce the example from our paper (Mandl et al., 2024): »The minP procedure has the major advantage that it has a relatively intuitive principle, as illustrated by the following example. In a comment on a study by Mathews et al. (2008) claiming that breakfast cereal intake before pregnancy is positively associated with the probability to conceive a male fetus, Young et al. (2009) reinterpret the small p-value of 0.0034 obtained in the original article. They notice that Mathews et al. (2008) did not only analyse the association between fetal sex and the consumption of breakfast cereals, but also many other food items—a typical case of multiple testing. Based on the analysis of permuted data (i.e. data with randomly shuffled fetal sex status), Young et al. (2009) argue that “one would expect to see a p-value as small as 0.0034 approximately 28 percent of the time when nothing is going on”. Implicitly, they apply the minP procedure for adjusting the smallest raw p-value of 0.0034 to 0.28 in this context where multiple tests are performed to investigate multiple food items. Our suggestion consists of translating this approach into the context of the analytical researcher degrees of freedom towards addressing the statistical factors of the replication crisis.«´

To make use of the minP adjustment method, we basically need two main ingredients: **(1)** the original unadjusted p-values based on the different analytical specifications and **(2)** the p-values obtained from the permuted datasets. If we then want to adjust the minimal p-value obtained from the different specifications, we count how often the minimal p-value is smaller for the permuted data sets compared to the unadjusted minimal p-value. This proportion is our final result and thus our adjusted p-value.

2 Toy example

Just imagine a researcher wants to perform a two-sided hypothesis test on two variables X_1 and X_2 . Subsequently, we may choose between a two-sample t-test or a nonparametric two-sample Wilcoxon-test. Therefore, we are left with two analytical specifications in total ($m = 2$). Note that in reality the number of analytical specifications m can be much higher and that the degree of adjustment is a function of the total number of specifications and the dependency between the hypotheses.

2.1 Create data

In the following we create a dataframe with two normally distributed variables ($n = 100$) with different means and the same variance.

```
set.seed(1)
x1 = rnorm(n = 100, mean = 0, sd = 0.1)
x2 = rnorm(n = 100, mean = 0.05, sd = 0.1)
data = data.frame(x1 = x1, x2 = x2)

data = data.frame(group = rep(c("0", "1"), each = 100), variable = c(x1, x2))
```

2.2 Run initial analyses

Now, we apply our two analytical specifications ($m = 2$) resulting in two different initial (and unadjusted) p-values.

```
result1 = t.test(variable ~ group, data = data, var.equal = TRUE)

result2 = wilcox.test(variable ~ group, data = data, alternative = "two.sided",
                      conf.level = 0.95, paired = TRUE)

result1$p.value; result2$p.value

## [1] 0.007742591
## [1] 0.01189869
# save minimal p-value for comparison
minimal.pvalue = result1$p.value
```

2.3 Permutation

Subsequently, we want to apply the minP adjustment method. We run a simple for-loop with 1000 iterations ($B = 1000$) to shuffle the group assignment for every iteration i of the loop. Here, we again use the *sample* function to randomly shuffle without replacement. Then we apply the hypothesis tests to each shuffled data set. We save the results for every iteration in an empty dataframe with dimension $[m, B]$, i.e. $m = 2$ being the number of specifications and $B = 1000$ being the number of permutations. In every iteration i we get a single p-value for each specification resulting in two p-values per loop.

```
p = data.frame(matrix(NA, nrow = 2, ncol = 1000))
for (i in 1:1000){
  set.seed(i)
  df1 = data
  df1$variable = df1$variable[sample(nrow(df1))]

  result1 = t.test(variable ~ group, data = df1, var.equal = TRUE)

  result2 = wilcox.test(variable ~ group, data = df1, alternative = "two.sided",
                        conf.level = 0.95, paired = TRUE)

  p[, i] = c(result1$p.value, result2$p.value)
}
```

2.4 Extract minimal p-values

Subsequently, we extract the minimal p-values for every permutation. We do this using the *apply* function to extract the minimal p-value of every column.

```
min.permuted = apply(p, 2, min)
```

2.5 Adjust p-values

If we want to adjust the smallest p-value from our initial analyses we simply count how often the minimal p-values from the permuted data sets are smaller than the original p-value.

```
sum(min.permuted < minimal.pvalue)/1000
```

```
## [1] 0.009
```

In our simple toy example the original p-value gets adjusted to $p = 0.009$.

3 References

[*] Mandl, M.M., Backer-Pennrich, A.S., Hinske, L.C., Hoffmann, S., Boulesteix, A.-L.: Addressing researcher degrees of freedom through minP adjustment. arXiv preprint arXiv:2401.11537 (2024).

[*] Mathews, F., Johnson, P.J., Neil, A.: You are what your mother eats: evidence for maternal preconception diet influencing foetal sex in humans. *Proceedings of the Royal Society B: Biological Sciences* 275(1643), 1661–1668 (2008).

[*] Young, S.S., Bang, H., Oktay, K.: Cereal-induced gender selection? most likely a multiple testing false positive. *Proceedings of the Royal Society B: Biological Sciences* 276(1660), 1211–1212 (2009).