Clarification for Cutoff

Problem Statement

We are dealing with a scenario where we have conducted $N = 144$ independent tests. Each test has a probability $P(Y_i = 1) = p_0 = 0.1$ of being significant just by chance. We want to determine the minimum number of tests, n, that need to be significant to ensure that the observed significance is not just due to chance, under a specified error probability e (e.g., $e = 0.1$ or $e = 0.05$).

*In terms of the "consistency" you've mentioned, it should be $e = p_0 = 0.1$

Statistical Approach

The sum of significant tests, $S = \sum_{i=1}^{N} Y_i$, follows a binomial distribution $\text{Bin}(N = 144, p_0 = 0.1)$. We aim to find the cutoff n such that the probability of having more than n significant tests by chance is less than the error probability e. Mathematically, this is represented as:

$$P(S > n) = 1 - P(S \leq n) < e$$

Thus, we need to find n such that:

$$P(S \leq n) \geq 1 - e$$

In the context of a binomial distribution, this probability can be calculated as:

$$P(S \leq n) = F_S(n) = \sum_{i=1}^{n} \binom{144}{i} \times 0.1^i \times 0.9^{144-i}$$

Calculation in R

In R, we can use the `qbinom` function to find this cutoff n. The function `qbinom` is used to find the quantile function of the binomial distribution. The parameters for `qbinom` in our case are:

- `p`: The cumulative probability, which is $1 - e$.
- `size`: The number of trials, which is $N = 144$.
- `prob`: The probability of success on each trial, which is $p_0 = 0.1$.

The R code to calculate n is:

```
e <- 0.1 # or 0.05, depending on the error probability
p <- 1 - e
size <- 144
prob <- 0.1
n <- qbinom(p, size, prob)

print(n)
```

```
## [1] 19
```

"n=21" is got by setting e=0.05, while keeping prob=0.1:

```r
e <- 0.05
p <- 1 - e
size <- 144
prob <- 0.1
n <- qbinom(p, size, prob)

print(n)
```

```
## [1] 21
```

The previous "n=12" is got by setting e=0.05, prob=0.05, and size=126: (I added 1 to the outcome, since the n is the value which S should be **greater** than, see the formula $P(S > n) = 1 - P(S \leq n) < e$; so in this way, for previous 2 cases, the cutoffs should be 20 and 22)

```r
e <- 0.05
p <- 1 - e
size <- 126
prob <- 0.05
n <- qbinom(p, size, prob)

print(n)
```

```
## [1] 11
```

October memo version

N=126, i.e., size=126

```r
e <- 0.1
p <- 1 - e
size <- 126 # Adjusted number of tests
prob <- 0.1
n <- qbinom(p, size, prob)

print(n)
```

```
## [1] 17
```

Cutoff should be 17 or 17+1=18.

Another Way of Thinking

There is something interesting:

We are particularly interested in tests that include specific significant variables, i.e., `FTR_ratio_sAPPb_sAPPa` or `FCR_ratio_sAPPb_sAPPa`. It turns out that only 4 independent model configurations contain these variables:

Independent Variable Combinations (16):

1. (cct_sAPPb)
2. (pdr_sAPPb)
3. (cct_sAPPa, cct_ratio_sAPPb_sAPPa)
4. (pdr_sAPPa, pdr_ratio_sAPPb_sAPPa)
5. (cct_sAPPb, cct_ratio_sAPPb_sAPPa)
6. (pdr_sAPPb, pdr_ratio_sAPPb_sAPPa)
7. (cct_sAPPb, cct_ratio_sAPPb_totalAb)
8. (pdr_sAPPb, pdr_ratio_sAPPb_totalAb)
9. (FTR_ratio_sAPPb_sAPPa, dlt_ratio_sAPPb_sAPPa)
10. (FCR_ratio_sAPPb_sAPPa, dlt_ratio_sAPPb_sAPPa)
11. (FTR_sAPPb, FTR_ratio_sAPPb_Ab40, dlt_sAPPb, dlt_ratio_sAPPb_Ab40)
12. (FCR_sAPPb, FCR_ratio_sAPPb_Ab40, dlt_sAPPb, dlt_ratio_sAPPb_Ab40)
13. (FTR_sAPPa, FTR_ratio_sAPPb_sAPPa, dlt_sAPPa, dlt_ratio_sAPPb_sAPPa)
14. (FCR_sAPPa, FCR_ratio_sAPPb_sAPPa, dlt_sAPPa, dlt_ratio_sAPPb_sAPPa)
15. (FTR_sAPPb, FTR_ratio_sAPPb_sAPPa, dlt_sAPPb, dlt_ratio_sAPPb_sAPPa)
16. (FCR_sAPPb, FCR_ratio_sAPPb_sAPPa, dlt_sAPPb, dlt_ratio_sAPPb_sAPPa)

So the total number of relevant tests is reduced to $N = \text{control} \times \text{independent} = 9 \times 4 = 36$.

Then $S = \sum_{i=1}^{N} Y_i$ follows a binomial distribution $\text{Bin}(N = 36, p_0 = 0.1)$

The R code to calculate the new cutoff n with the adjusted total number of tests is:

```
e <- 0.05
p <- 1 - e
size <- 36 # Adjusted number of tests
prob <- 0.1
n <- qbinom(p, size, prob)

print(n)
```

```
## [1] 7
```

So the cutoff is n=6 or n'=6+1=7 now. (It is the same case for October's version, since configurations concerning FTR and FCR haven't been changed.)

```
e <- 0.05
p <- 1 - e
size <- 36 # Adjusted number of tests
prob <- 0.1
n <- qbinom(p, size, prob)

print(n)
```

```
## [1] 7
```