NONPARAMETRIC PERMUTATION TESTING

CHAPTER 33

TONY YE

WHAT IS PERMUTATION TESTING?

Framework for assessing the statistical significance of EEG results.

Advantages:

- Does not rely on distribution assumptions
- Corrections for multiple comparisons are easy to incorporate
- Highly appropriate for correcting multiple comparisons in EEG data

WHAT IS PARAMETRIC STATISTICAL TESTING?

The test statistic is compared against a theoretical distribution of test statistics expected under the H_0.

- t-value
- χ^2-value
- Correlation coefficient

The probability (*p*-value) of obtaining a statistic under the H_0 is at least as large as the observed statistic. [INSERT 33.1A]

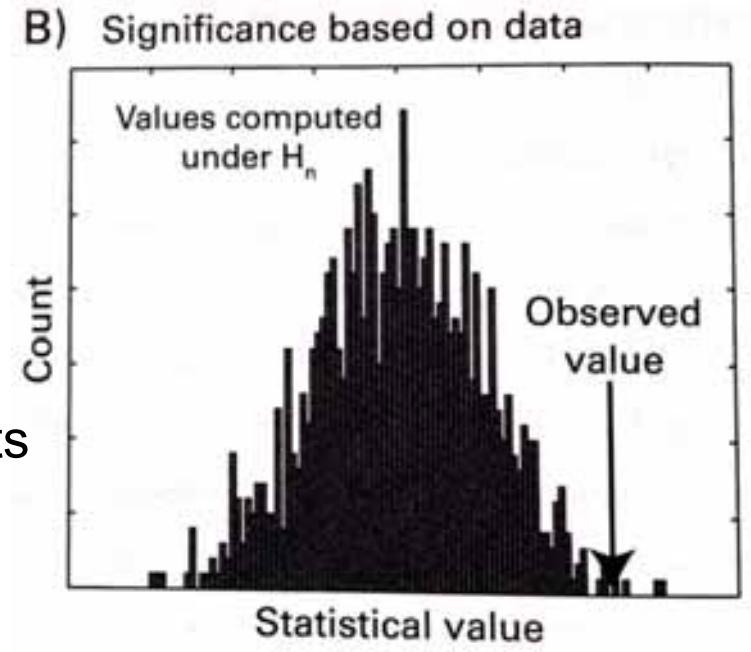
NONPARAMETRIC PERMUTATION TESTING

No assumptions are made about the theoretical underlying distribution of test statistics under the H_0.

- Instead, the distribution is created from the data that you have!

How is this done?

- Shuffling condition labels over trials
  - Within-subject analyses
- Shuffling condition labels over subjects
  - Group-level analyses
- Recomputing the test statistic



NULL-HYPOTHESIS DISTRIBUTION

Evaluating your hypothesis using a t-test of alpha power between two conditions.

Two types of tests:

- **Discrete tests**
  - Compare conditions
- **Continuous tests**
  - Correlating two continuous variables

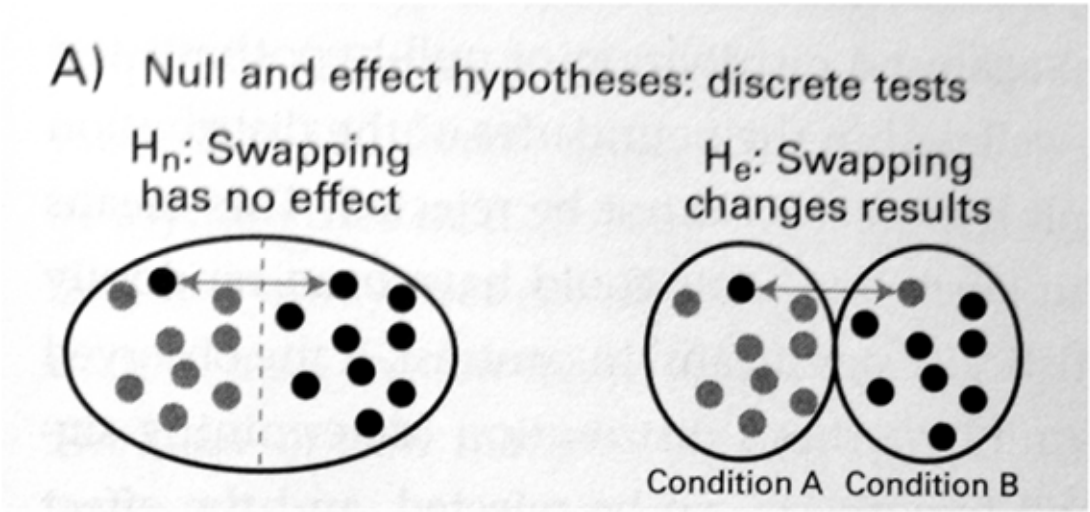
DISCRETE TESTS

Compare EEG activity between Condition A & B

- H_0 = No difference between conditions
- Random relabeling of conditions
  - Test Statistic (TS) = as large as the TS BEFORE the random relabeling.

Steps

1. Randomly swap condition labels from many trials
2. Compute *t*-test across conditions
3. If TS ≠ 0, there is sampling error or outliers



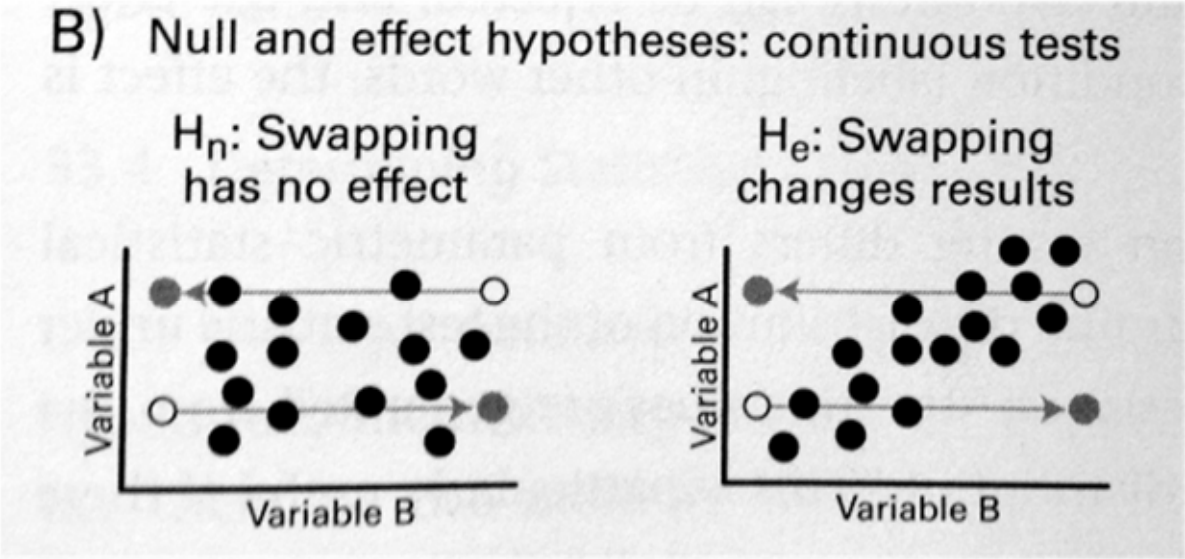
CONTINOUS TESTS

The idea:

- Testing statistical significance of a correlation coefficient

What's the difference between this and discrete tests?

- TS is created by swapping data points instead of labels



SIMILARITIES

The data are not altered

- The “mapping” of data are shuffled around.

Analysis steps:

1. Creates a H_0 TS value
2. Repeats the process many MANY times
3. The MANY iterations of H_0 values creates a distribution of TS values

SIMILARITIES

Statistical evaluation entails:

- Comparing the original TS value with the new distribution of TS

Effect is not statistically significant if…

- The original TS does not exceed the boundaries of the new distribution of TS

Effect IS statistically significant if…

- The original TS is “far away” from the new TS distribution

ITERATIONS

How many do you need?

- ~1000 for high signal-to-noise-ratio distribution
- ~250 – 500 for permutation testing at each trial, time point, and frequency.

Why?

- ***Less iterations*** = greater chance for unusually large/small TS by chance
- ***More iterations*** = estimates of H_0 distribution will be stronger with greater reliability in significance

WARNING!

- ***More iterations*** = longer computation times!

DETERMINING STATISTICAL SIGNIFICANCE

Method 1:

1. Count the number of H_0 values that are ***more extreme*** than the original TS value
   - **Extreme** = further to the right/left tail of the distribution

2. Divide the number of H_0 values by the total number of tests
   - $\mathbf{P_N}$ = *p*-value based on the number of suprathreshold tests

DETERMINING STATISTICAL SIGNIFICANCE

Method 2:

1. Convert original TS to the standard deviation of the H_0 distribution

2. Convert that into the *p*-value

- $\mathbf{V_e}$ = original TS
- $\mathbf{V_n}$ = vector of H_0 TS
- $\mathbf{V_n}$ bar = mean

$$Z = \frac{v_e - \overline{V_n}}{std(V_n)}$$

DETERMINING STATISTICAL SIGNIFICANCE

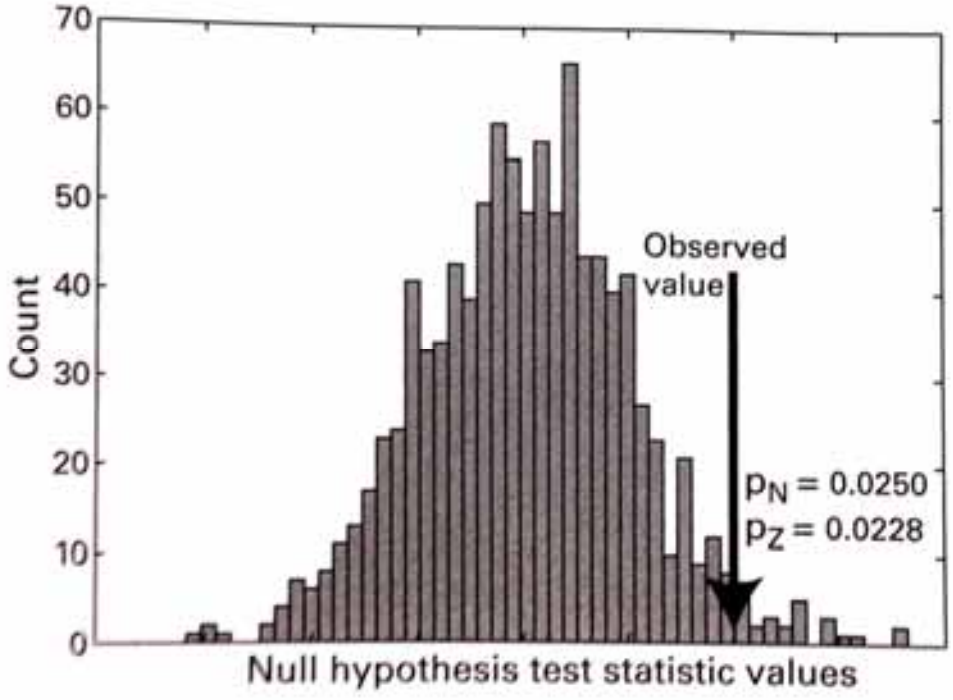
Method 2 cont'd:

- Z-value → p-value
  - Matlab function: `normcdf`

- *p*-value = $\boldsymbol{P_Z}$

Advantages:

- A *p*- and *Z*-value at each pixel can be incorporated into the H_0 distribution.



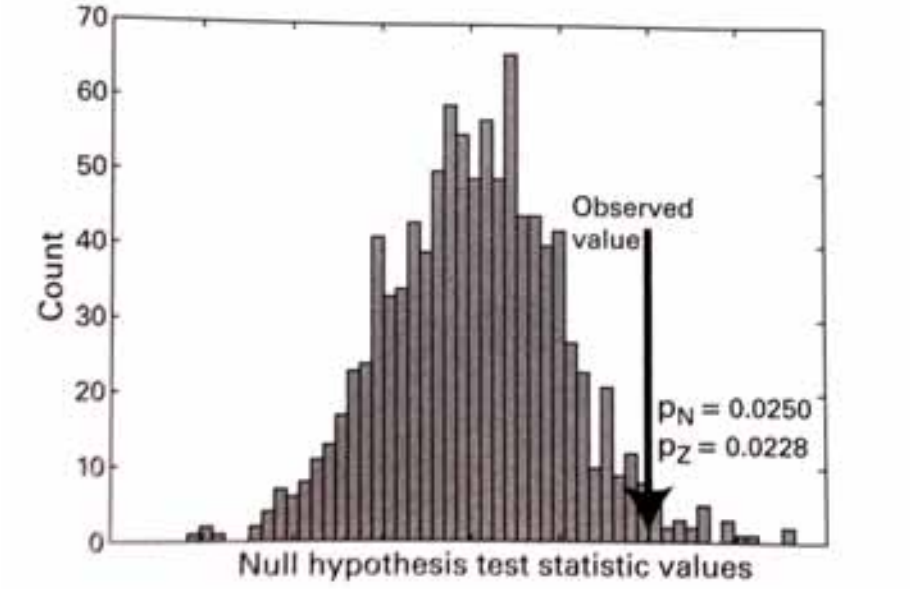
THE EVER-CHANGING P-VALUE

Warning!

- P-values can change each time you recompute the H_0 distribution

Rest-assured if you have sufficient iterations!

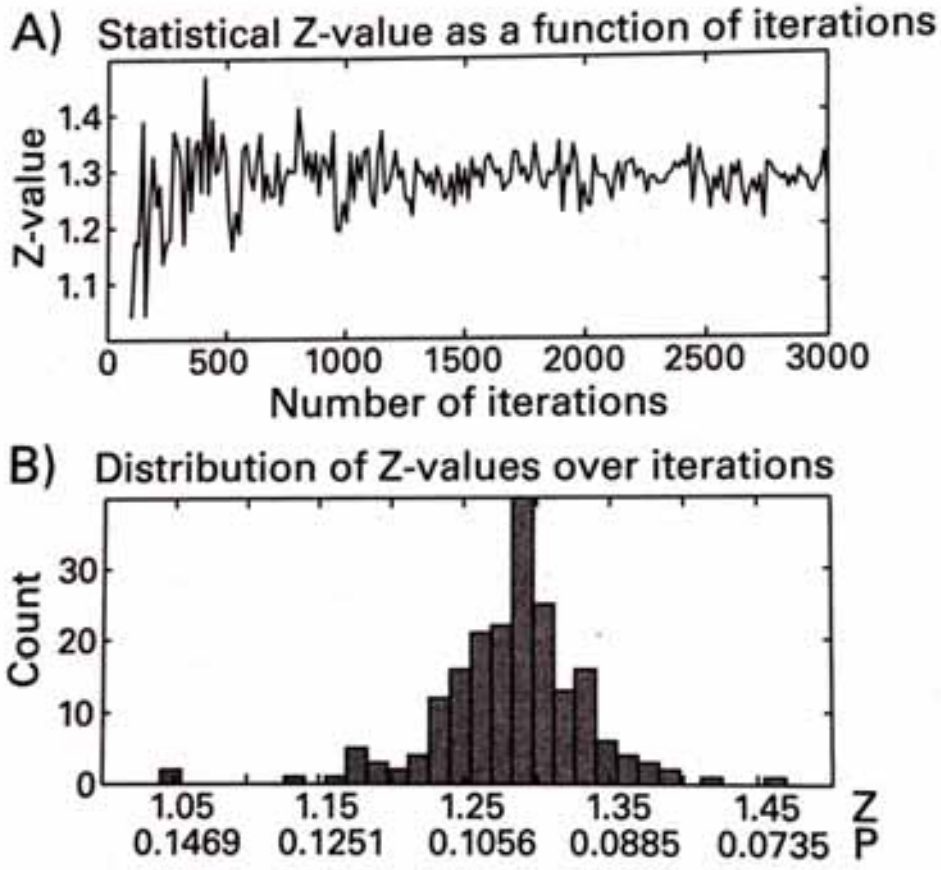
- The change in p-values should be tiny



THE EVER-CHANGING P-VALUE

What if you don't have sufficient iterations?

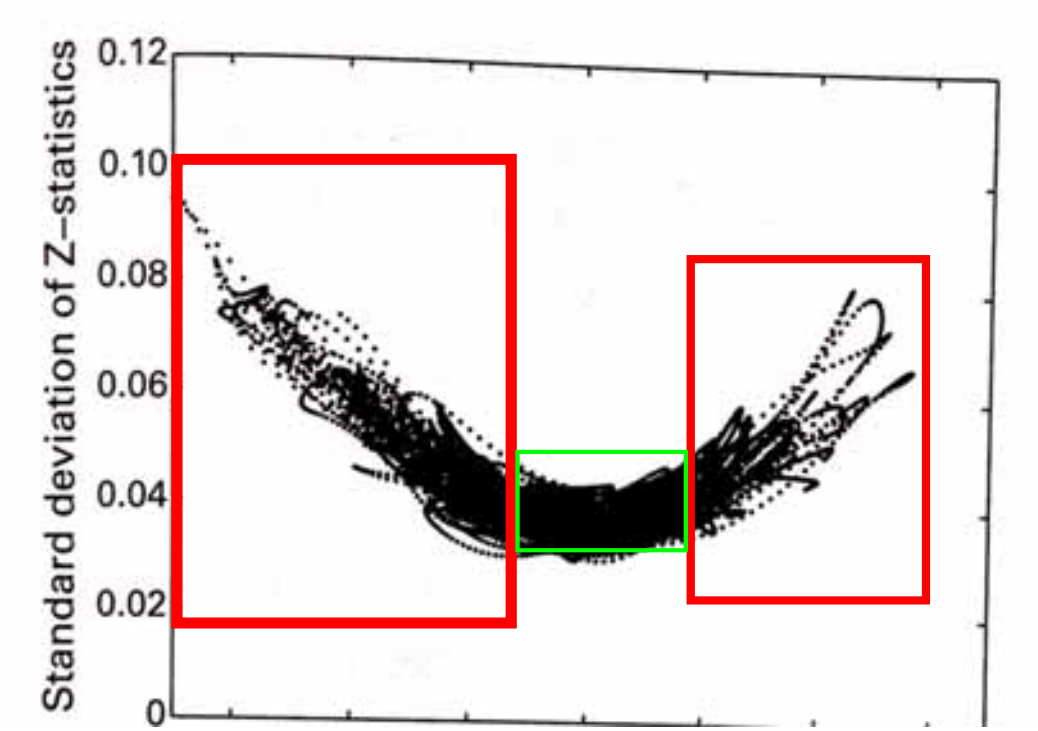
- You may get p-values that **CAN** affect your interpretation!



THE EVER-CHANGING P-VALUE

Variability depends on:

- Clean data
- Result's significance



MULTIPLE COMPARISONS

The Bonferroni correction is available to use!

When to use Bonferroni:

- Hypothesis-driven analyses
  - Testing effect for 3 different electrodes in ***ONE*** time-frequency window
- If there are not too many tests
- If you expect robust effects

MULTIPLE COMPARISONS

When *NOT* to use Bonferroni correction:

- Exploratory data-driven analyses
- Many tests over:
  - Time points
  - Frequency bands
  - Electrodes

Why not?

1. Bonferroni correction assumes that the tests are *independent* – which many EEG results are NOT
2. The p-value will drop and hide actual effects
3. Bonferroni correction is based only on the number of tests, instead of the information that can be found in the tests.

PERMUTATION TESTING TO THE RESCUE!

This framework already incorporates multiple comparison corrections!

Unlike Bonferroni, permutation testing:

1. Corrects for ***information*** in the tests, instead of number of tests.
2. Provides stable p-values that can detect effects regardless of correlated data.

NONPARAMETRIC PERMUTATION TESTING

Two methods available

1. Corrects for multiple comparisons by using the pixel to determine the threshold
2. Corrects by using the cluster to determine threshold.

Generally, how do these methods work?

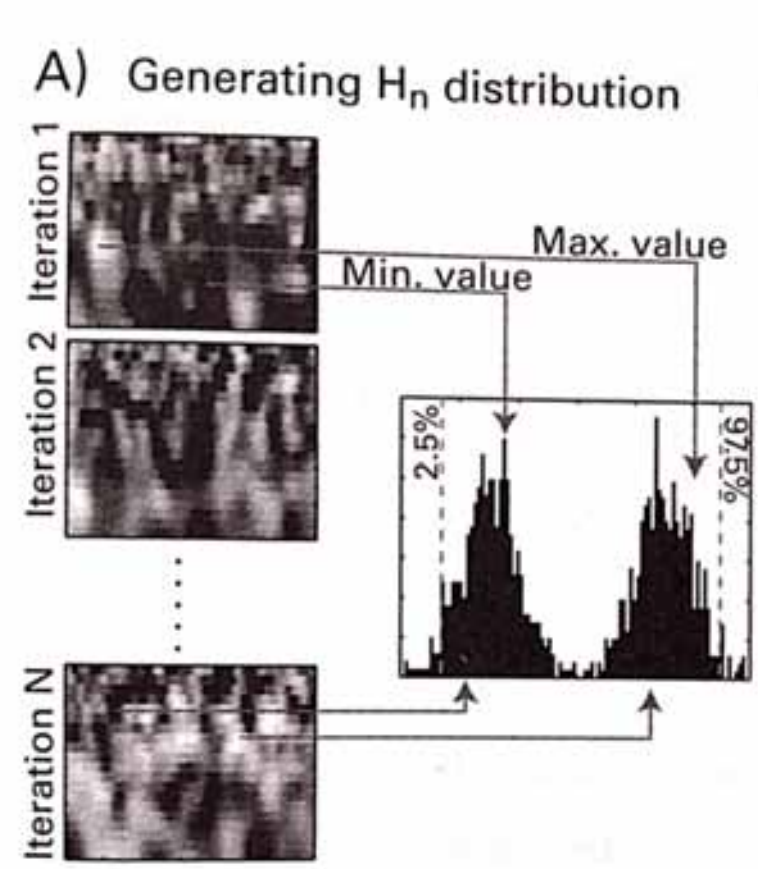
- Approaches the H_0 distribution at the time-frequency map level, instead of pixel level
- Reflecting information from the ***ENTIRE*** time-frequency-electrode space

PIXEL-BASED STATISTICS

Creating a distribution that contains the pixel from each iteration with the most extreme statistical value.

Steps:

1. Generate TS values under the H_0 (as previously outlined)
2. Store one or two pixels with the most extreme TS values in a matrix
3. For + **AND** – effects: Define the statistical threshold for 2.5 percentile and 97.5th percentile
4. For + **OR** – effects: Define threshold for 95th or 5th percentile, respectively



PIXEL-BASED STATISTICS

Things to note:

- A summary of the most extreme H_0 TS are saved across all pixels, at each iteration
  - Map-level thresholding

- Single pixels can be statistically significant
  - Even if neighboring pixels are non-significant
  - Interpretability depends on experimental design and size of time-frequency pixels

CLUSTER-BASED CORRECTION

What is a cluster?

- A group of contiguously significant points in time-frequency-electrode space
- Can be seen after applying a threshold with any pixel that has a value below it set to zero

What is cluster-based correction?

- Significance = enough neighboring pixels with suprathreshold values.
- Individual pixels that are significant aren't really significant

CLUSTER-BASED CORRECTIONS

“Big enough”?

- Number of extracted frequencies
- Resolution of the results
- Temporally downsampled

Example:

- 1 time point @ 1 ms
  - Significance is false
- 1 time point @ several Hz and several hundreds of ms
  - Significance is valid

CLUSTER-BASED CORRECTIONS

Non-data-driven method:

1. Predefine a number of target time and frequency points
   - E.g., Clusters of 200 ms, 3 Hz
2. Remove clusters that are less than that number

Data-driven method:

1. Perform permutation testing (as previously outlined)
2. At each iteration, apply a threshold to the time-frequency map using an uncorrected p-value (“precluster threshold”)
3. Threshold the H_0 iteration map

THRESHOLDING STRATEGIES

Method 1:

- Use parametric statistics
  - P-value from your t-test or correlation coefficient
- Ideal for normally distributed data

Method 2:

- Loop through the iterations **TWICE**
  - Once to build the H_0 distribution at each pixel
  - Second to threshold them using nonparametric pixel-based significance thresholding.

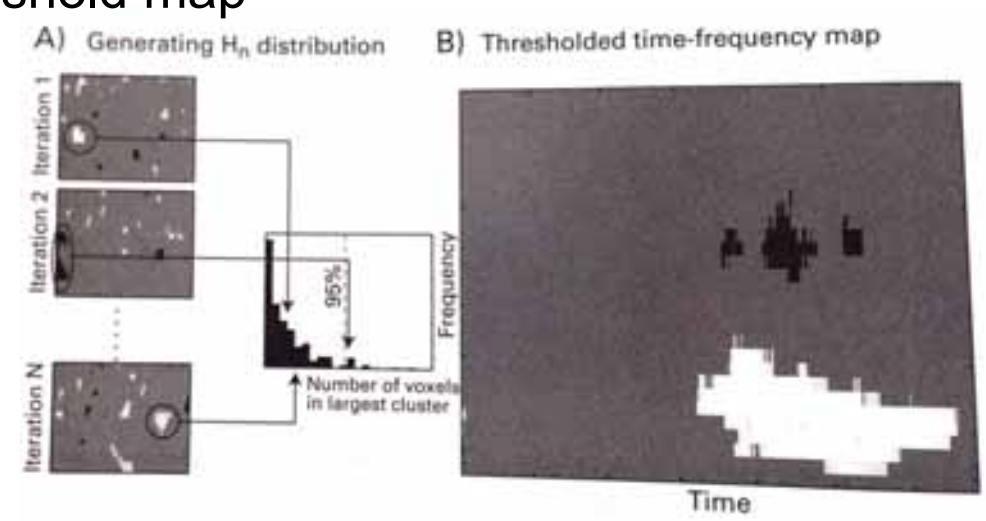
THRESHOLDING STRATEGIES

Now you have a distribution of the largest suprathreshold clusters under the H_0

- Threshold the map of observed statistical values using uncorrected p-value

Next steps:

1. Identify clusters in threshold map
2. Remove clusters that are less than 95th% of the largest cluster distribution



CLUSTER-BASED METHOD SUMMARY

Performs map-level thresholding

Corrections are based on the information within the data, instead of the number of tests performed

Precluster threshold affects the cluster correction threshold!

- Small p-value = remove many clusters
- Larger p-value = leave many clusters

Extremely sensitive to large clusters

- Localized true effects may go unnoticed!

FALSE DISCOVERY RATE (FDR) METHOD

How does it work?

- Controls for the probability of Type I errors within a distribution of p-values

```
function [pID,pN] = fdr(p,q)
```

- pID = p-value based on independence or positive dependence
- pN = nonparametric p-value
- p = p-value vector
- q = FDR level

Limitations

- Critical p-value is based on the number of tests performed AND distribution of p-values

SUMMARY

Shuffling depends on your focus of analysis and hypothesis

- Comparing two conditions
  - Shuffle condition labels
- Correlations of time-frequency and reaction times over trials
  - Shuffle mapping of reaction time to trials
- Connectivity between two electrodes
  - Shuffle ordering of time segments

If still unsure…

- What effects does your hypothesis concern?
- Sometimes, shuffling can be performed with more than one option

SUMMARY

What about complex statistical designs?

- There is a lack of support for complex analyses in cognitive electrophysiology
- Take a hypothesis-testing approach with SPSS, SAS, or R first

How can you report your analyses in a methods section?

- What variables were shuffled?
- How many iterations?
- How were p-values created?
- Which multiple comparisons correction did you use?