Analysis of pupillometry data: Preprocessing, and Statistical Analysis

Ana Vilotijević

SpringerLink

Open Access | Published: 26 August 2022

Methods in cognitive pupillometry: Design, preprocessing, and statistical analysis

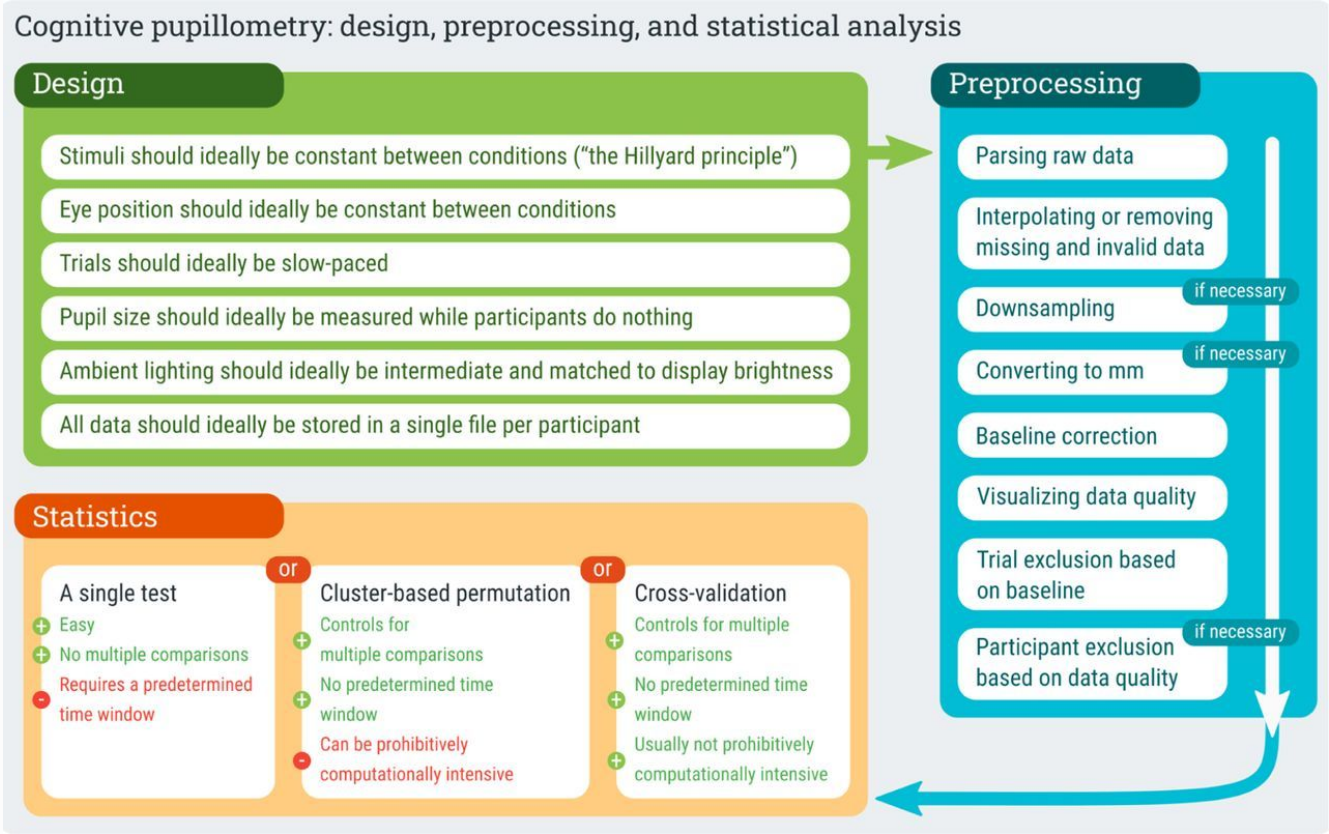
Sebastiaan Mathôt & Ana Vilotijević

Behavior Research Methods (2022) | Cite this article

795 Accesses | 15 Altmetric | Metrics

Abstract

Cognitive pupillometry is the measurement of pupil size to investigate cognitive processes such as attention, mental effort, working memory, and many others. Currently, there is no commonly agreed-upon methodology for conducting cognitive-pupillometry experiments, and approaches vary widely between research groups and even between different experiments from the same group. This lack of consensus makes it difficult to know which factors to consider when conducting a cognitive-pupillometry experiment. Here we provide a comprehensive, hands-on guide to methods in cognitive pupillometry, with a focus on trial-based experiments in which the measure of interest is the task-evoked pupil response to a stimulus. We cover all methodological aspects of cognitive pupillometry: experimental design, preprocessing of pupil-size data, and statistical techniques to deal with multiple comparisons when testing pupil-size data. In addition, we provide code and toolboxes (in Python) for



Springer Link

Open Access | Published: 26 August 2022

Methods in cognitive pupillometry: Design, preprocessing, and statistical analysis

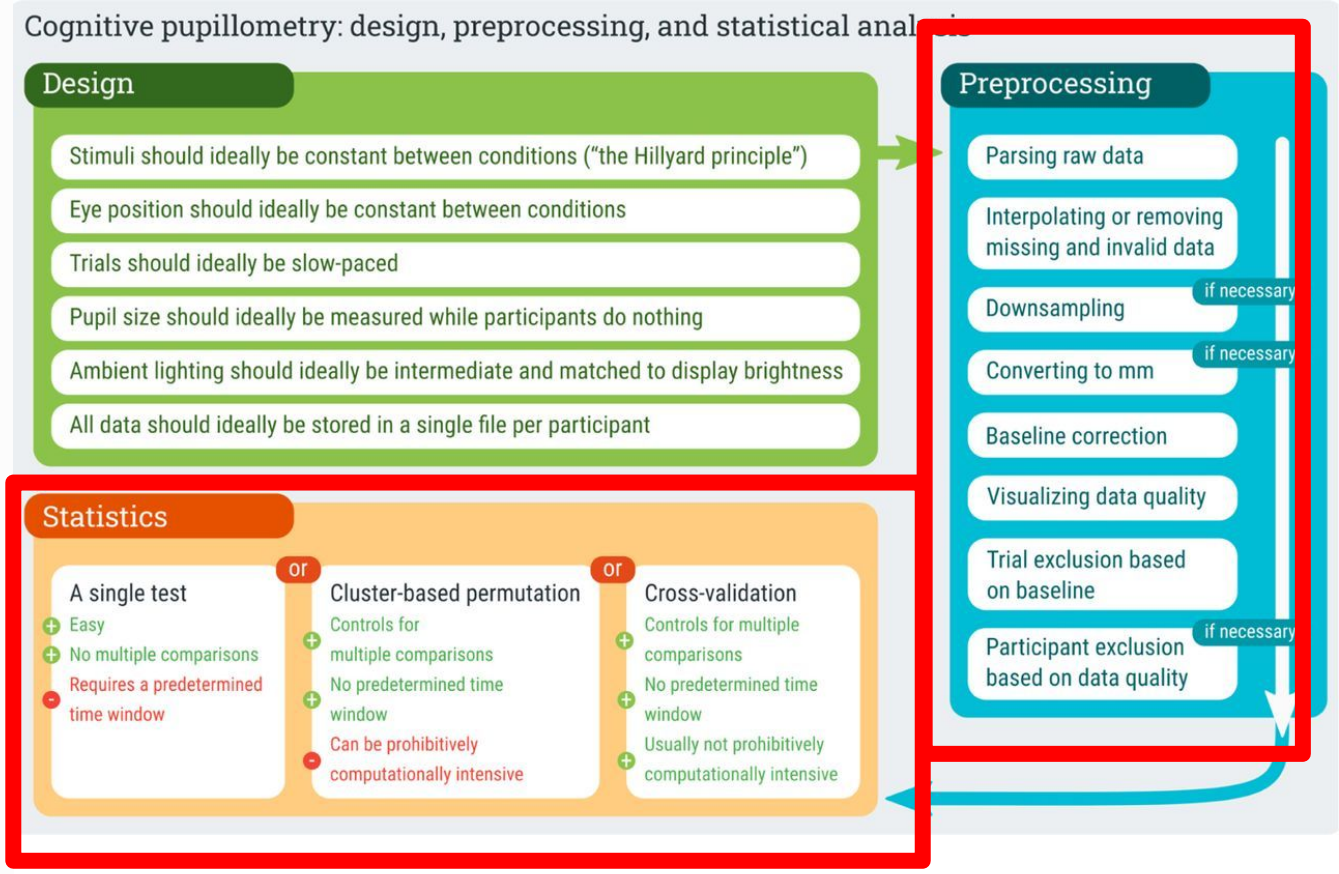
Sebastiaan Mathôt ✉ & Ana Vilotijević

Behavior Research Methods (2022) | Cite this article

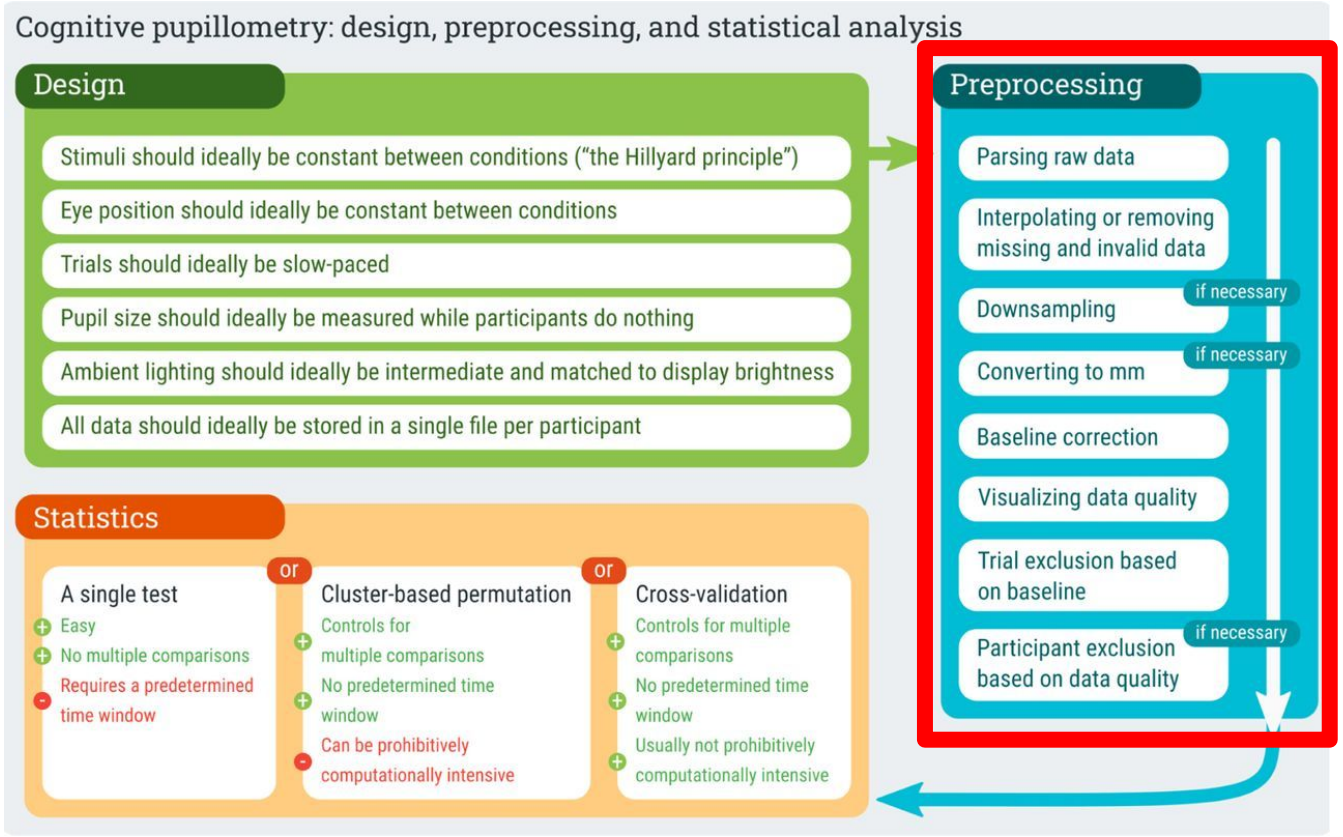
795 Accesses | **15** Altmetric | Metrics

Abstract

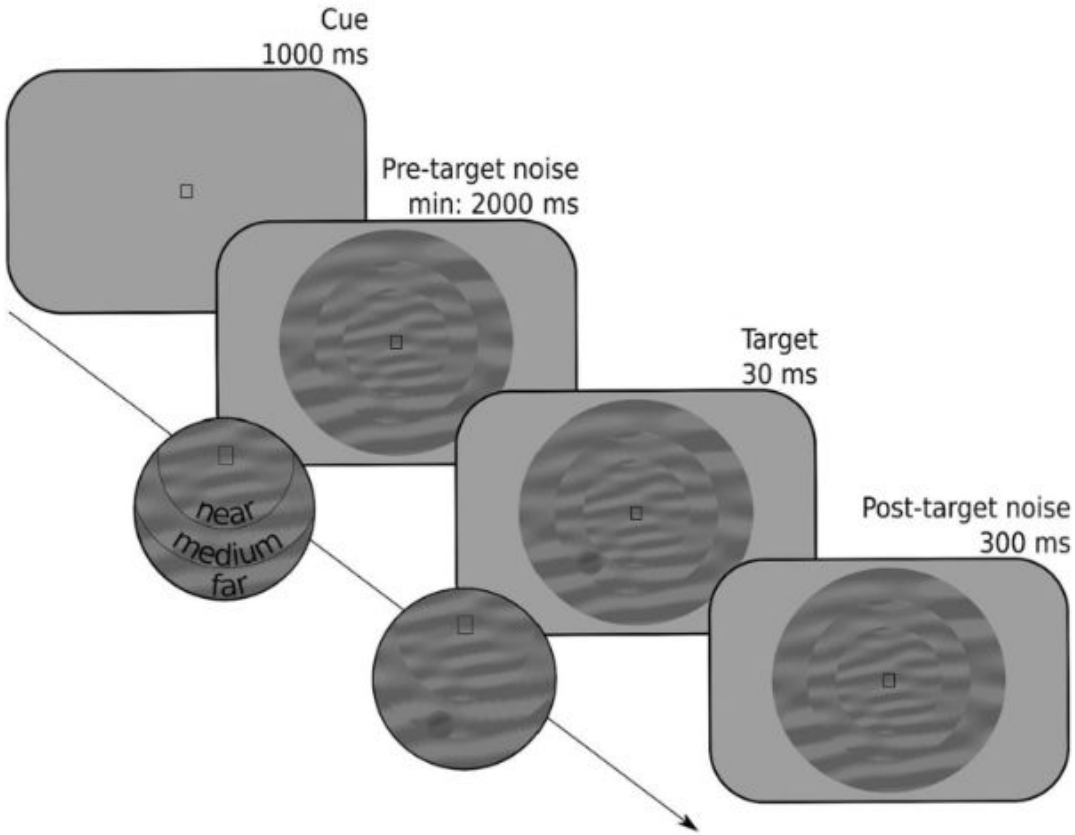
Cognitive pupillometry is the measurement of pupil size to investigate cognitive processes such as attention, mental effort, working memory, and many others. Currently, there is no commonly agreed-upon methodology for conducting cognitive-pupillometry experiments, and approaches vary widely between research groups and even between different experiments from the same group. This lack of consensus makes it difficult to know which factors to consider when conducting a cognitive-pupillometry experiment. Here we provide a comprehensive, hands-on guide to methods in cognitive pupillometry, with a focus on trial-based experiments in which the measure of interest is the task-evoked pupil response to a stimulus. We cover all methodological aspects of cognitive pupillometry: experimental design, preprocessing of pupil-size data, and statistical techniques to deal with multiple comparisons when testing pupil-size data. In addition, we provide code and toolboxes (in Python) for



Preprocessing data



The example experiment



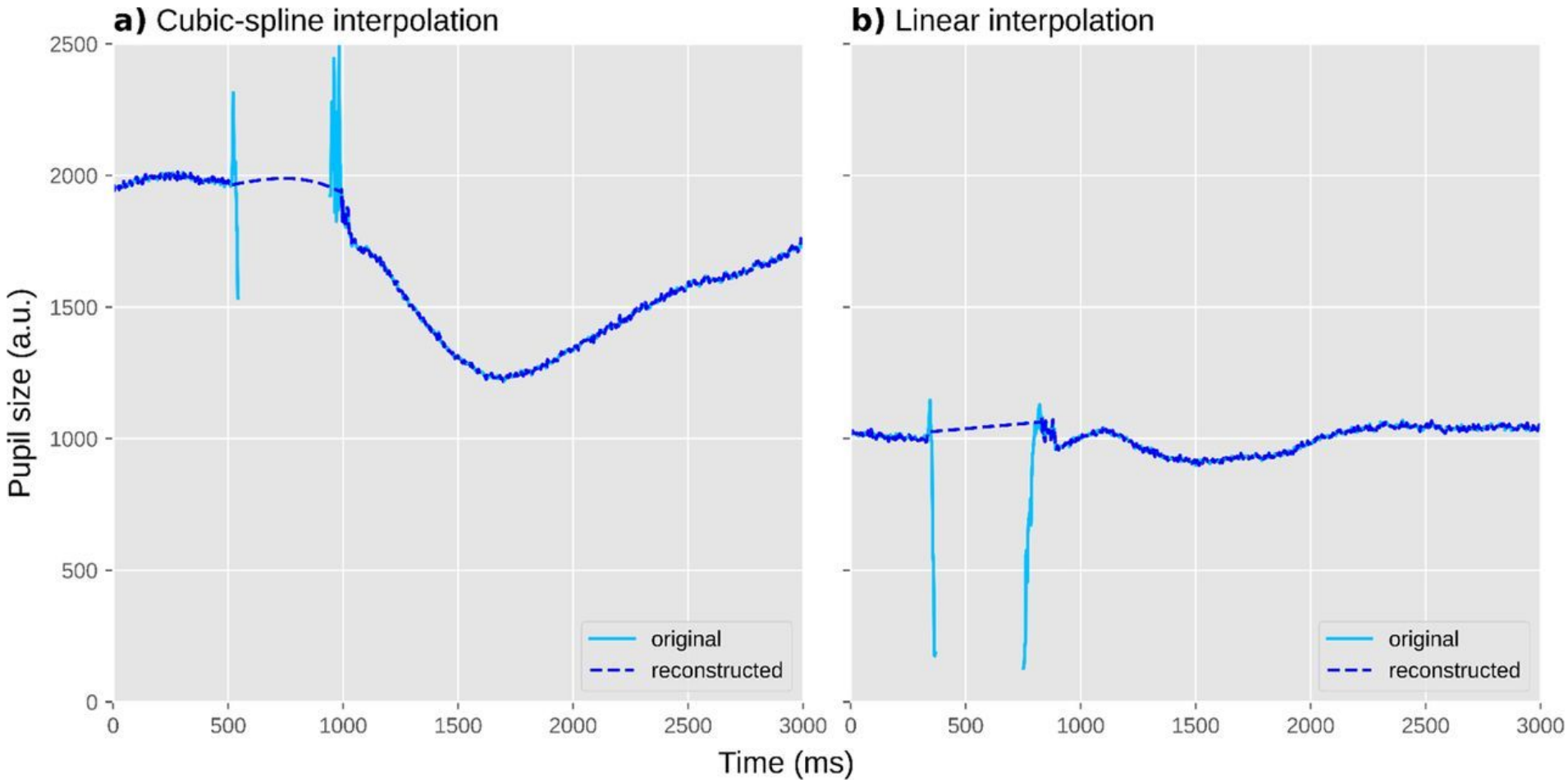
Step 1: Parsing the data

Trials ↓

response_time	cue_eccentricity	subject_nr	pupil_stream
732	medium	1	[0.00, 0.03, 0.01, … , -0.73]
1112	far	1	[0.00, 0.01, 0.00, … , 0.16]
888	near	1	[0.00, -0.03, -0.05, … , -1.11]
⋮	⋮	⋮	⋮
2604	medium	30	[0.00, 0.02, 0.02, … , 0.44]

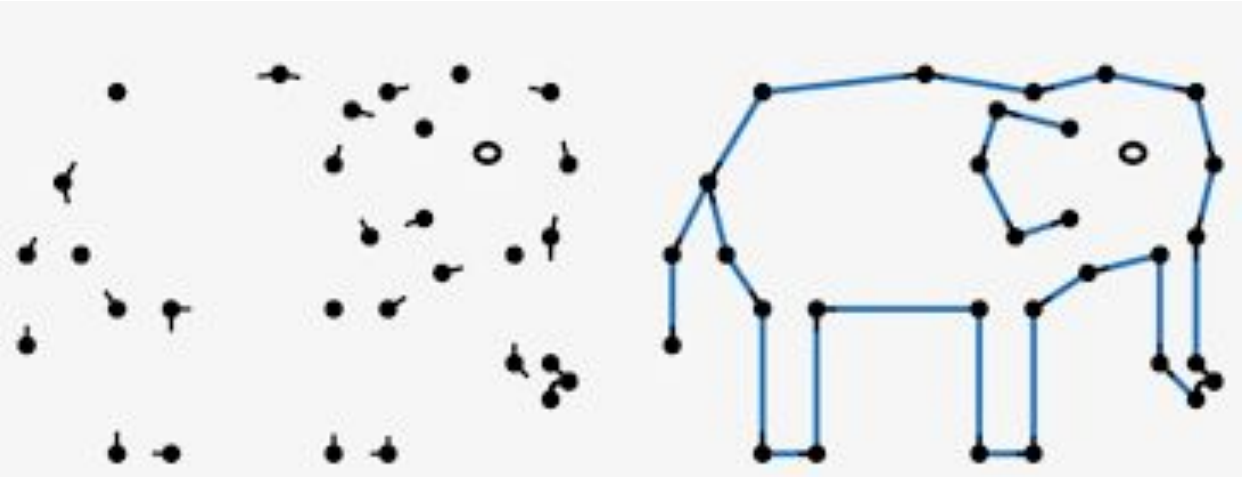
Samples →

Step 2: Interpolating blinks



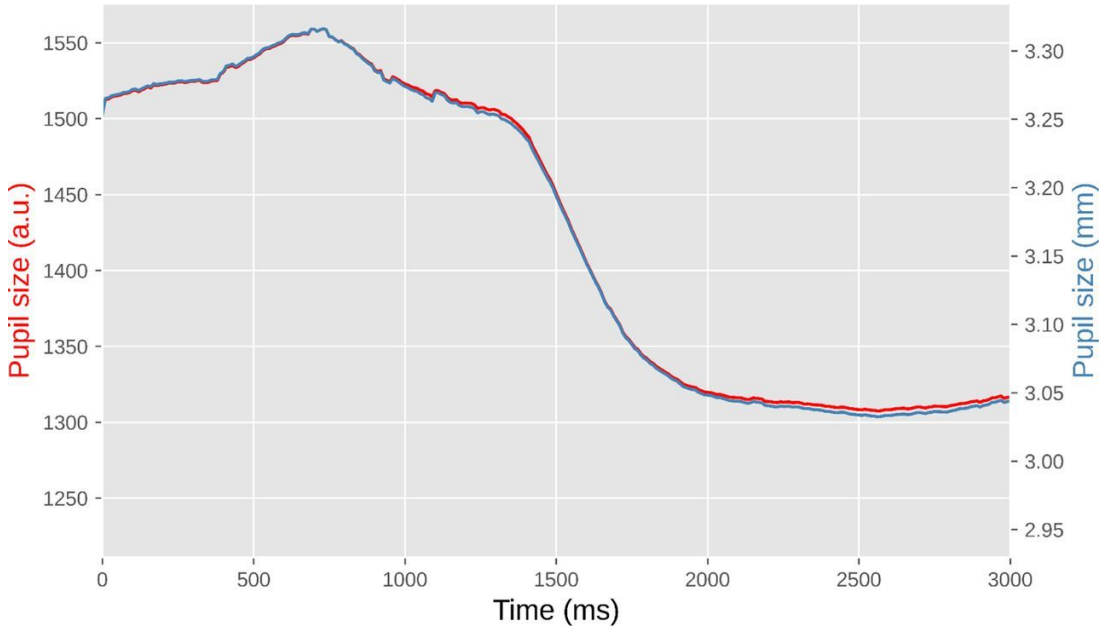
Step 3: Downsampling (if necessary)

- The latency of the pupil light response is around 200 ms, and the earliest cognitive effects on pupil size emerge around 500 ms after the triggering stimulus.
- Therefore, if you are using an eye tracker that records at a higher sampling rate (e.g. 1000Hz), it is convenient to downsample the signal to 100 Hz.



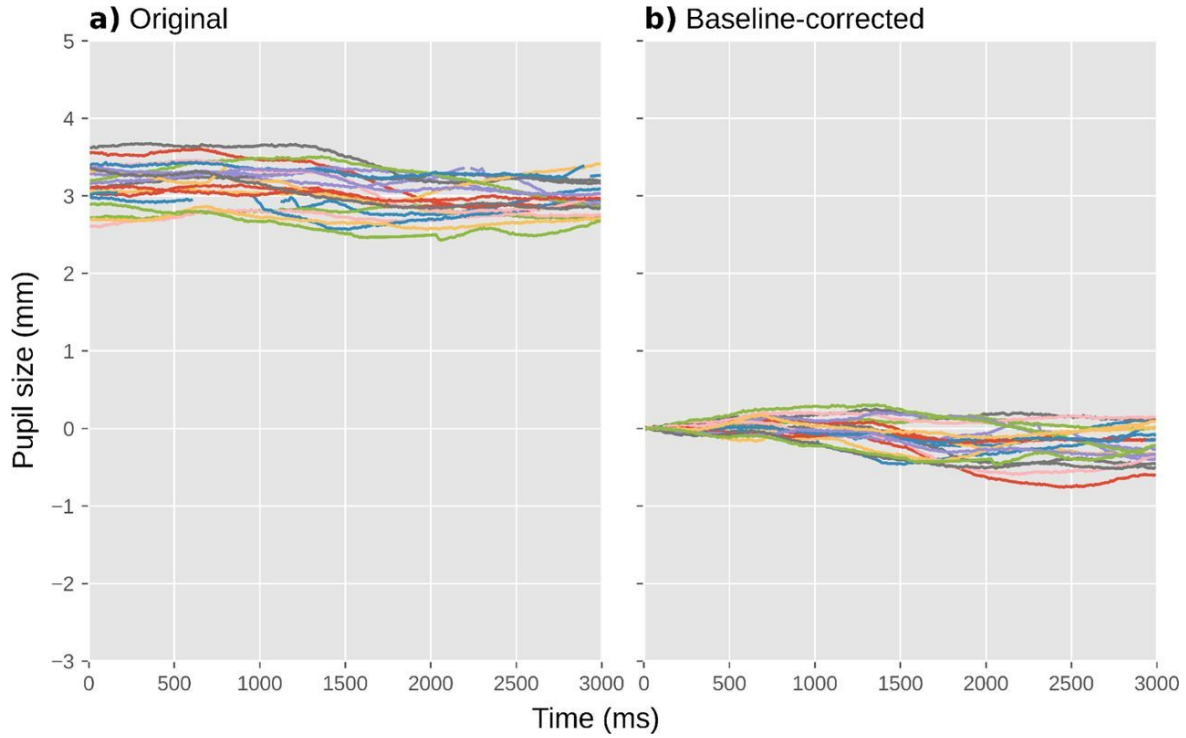
Step 4: Converting to mm (if necessary)

- Pupil size was recorded in arbitrary units and converted to millimeter (mm) of diameter
- To determine the conversion formula, we measured the size of artificial pupils (black circles printed on white paper) of known sizes.



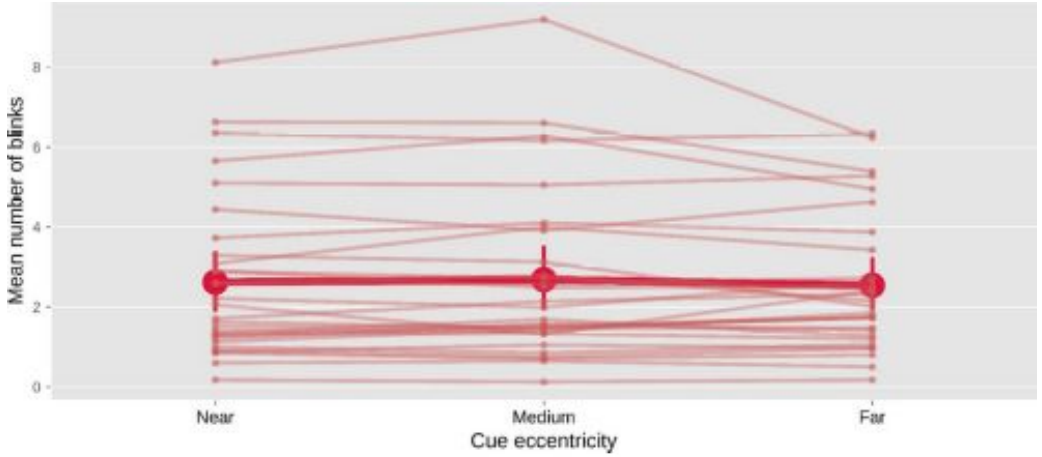
Step 5: Baseline correction

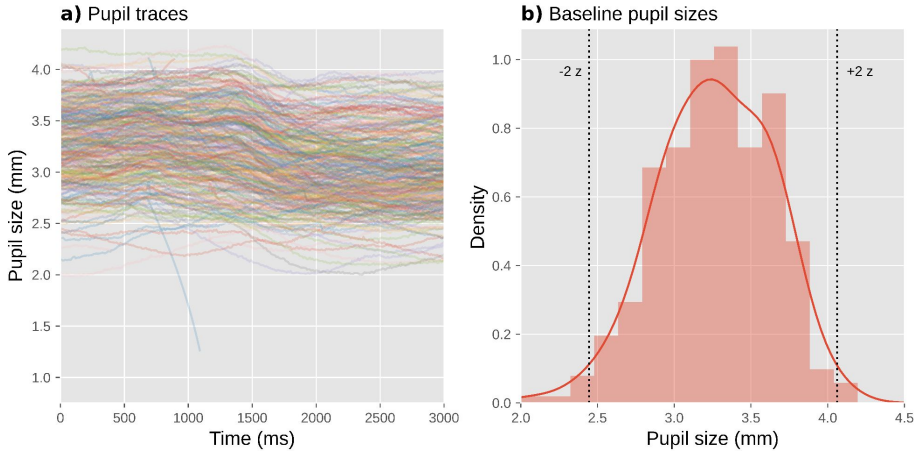
- Baseline correction is a technique to remove the impact of trial-to-trial fluctuations in pupil size
  - Done for each trial separately
  - by subtracting the mean pupil size during a baseline period from all subsequent pupil-size measurements (subtractive baseline correction)
- As a result, pupil size starts from 0 (for subtractive correction) during the baseline period on every trial, and only the change in pupil size—the task-evoked pupil response—remains.

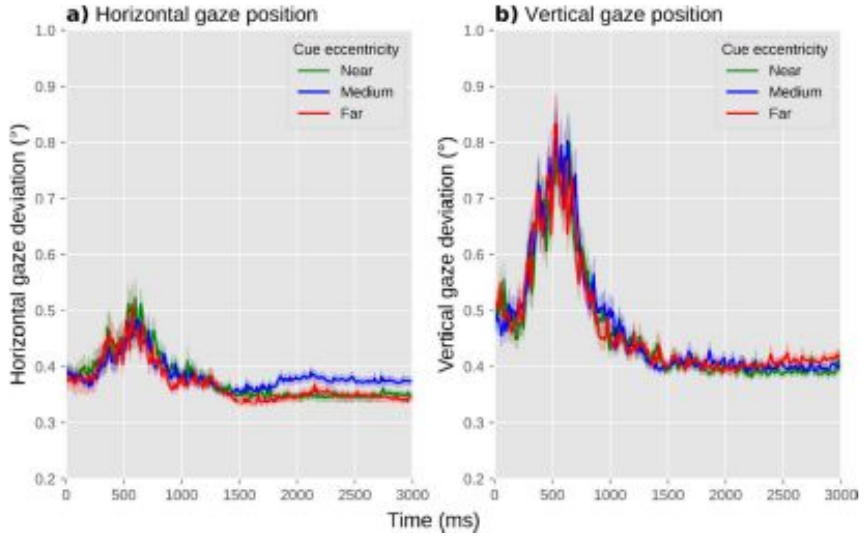


Step 6: Visualizing data quality

- Poor data quality signs:
  - spikes
  - the presence of lines that are far above (but rarely below) the others, or that start from zero but then quickly (< 200 ms) shoot upwards: these lines correspond to trials on which there were artifacts during the baseline period







Step 7: Trials exclusion

- Blinks and recording artifacts during the baseline period may result in very small baseline pupil sizes; in turn, this results in very large baseline-corrected pupil sizes, which add variability to the data and may substantially reduce statistical power

- Therefore, trials with extreme baseline pupil sizes should be excluded from analysis
  - convert baseline pupil sizes to z-scores
  - separately for each participant
  - z-scored baseline pupil size larger than 2 or smaller than −2 are excluded

Step 8: Participants exclusion

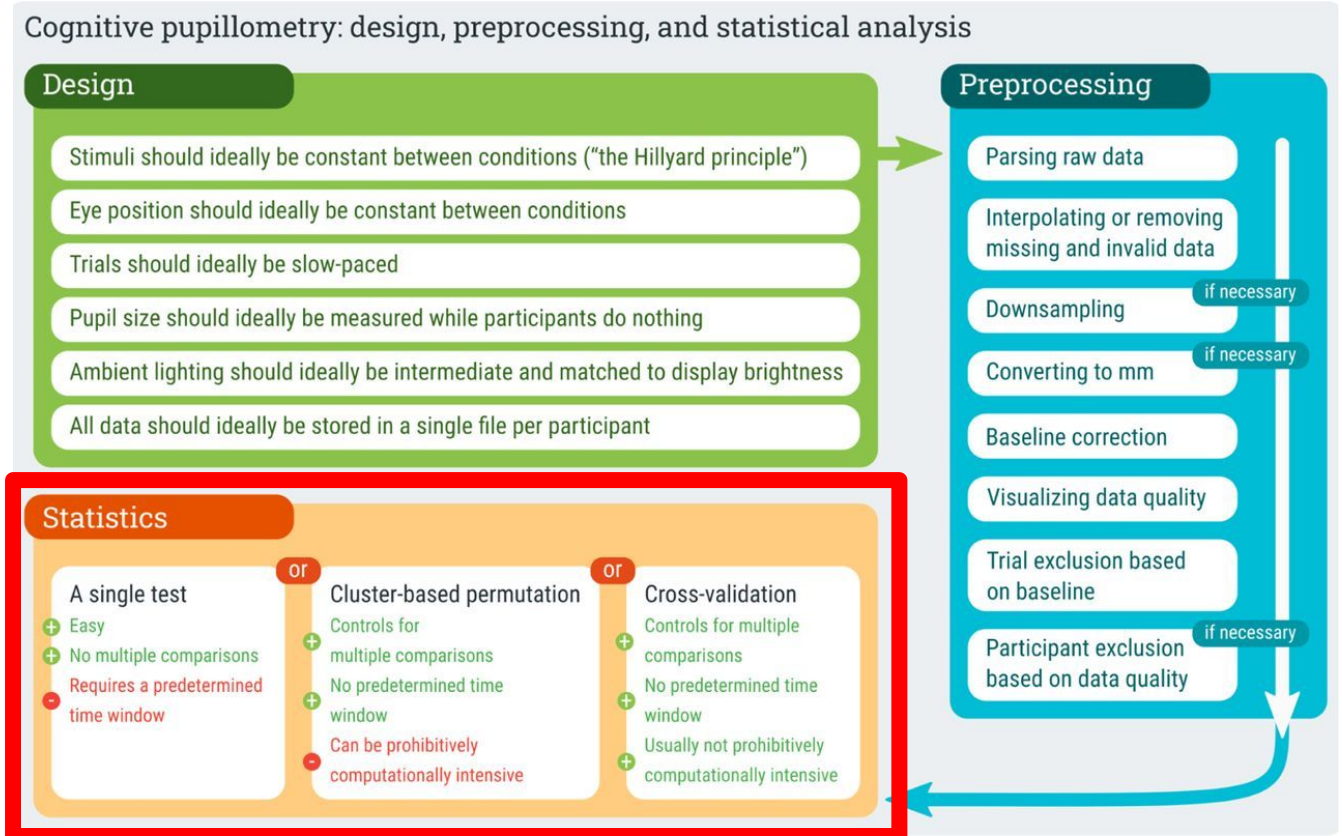
- Data quality can differ substantially between participants, for example because of
  - contact lenses/ glasses
  - eye make-up
  - or other factors that reduce the ability of the eye tracker to record the pupil
- In rare cases, this may be a reason to exclude a participant's data from analysis altogether.
- Ideally, exclusion criteria should be specified in advance

Let’s see the code!

Analysis of pupillometry data: Preprocessing, and Statistical Analysis

Ana Vilotijević

Analyzing data



Dependent variable(s)

- **Cognitive psychology experiments**
  - Does my IVs affect my DV?
- Example:
  - IV: cue eccentricity
  - DV: response time

- **Cognitive pupillometry experiments**
  - Does my IV affect my DV?
- Example:
  - IV: cue eccentricity
  - DV: pupil size (time series data)

Trials ↓

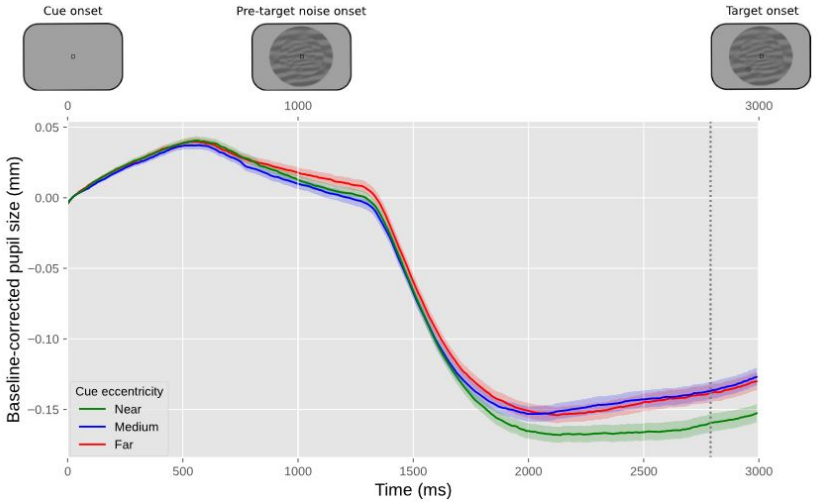
response_time	cue_eccentricity	subject_nr	pupil_stream
732	medium	1	[0.00, 0.03, 0.01, … , -0.73]
1112	far	1	[0.00, 0.01, 0.00, … , 0.16]
888	near	1	[0.00, -0.03, -0.05, … , -1.11]
⋮	⋮	⋮	⋮
2604	medium	30	[0.00, 0.02, 0.02, … , 0.44]

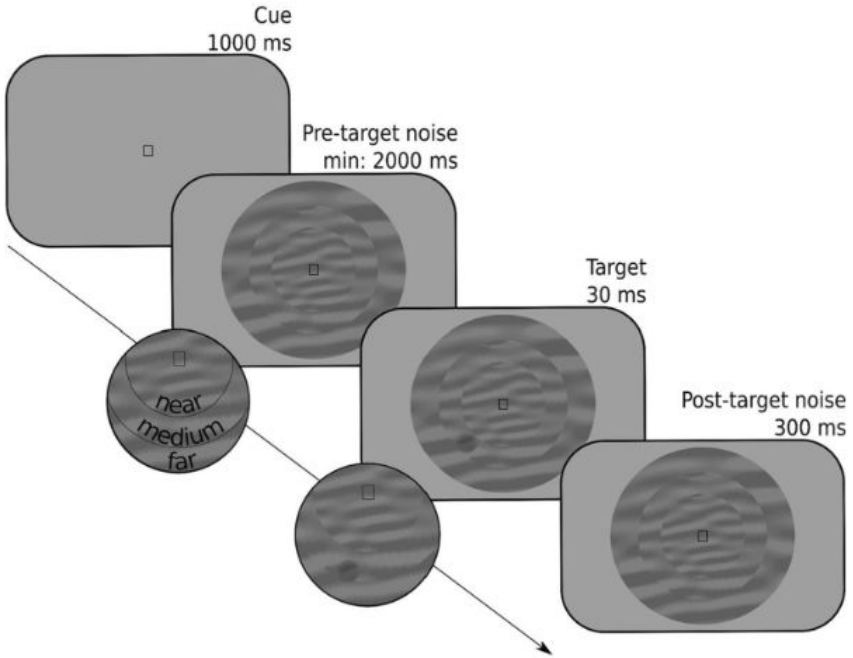
Samples →

Cognitive-pupillometry experiment example

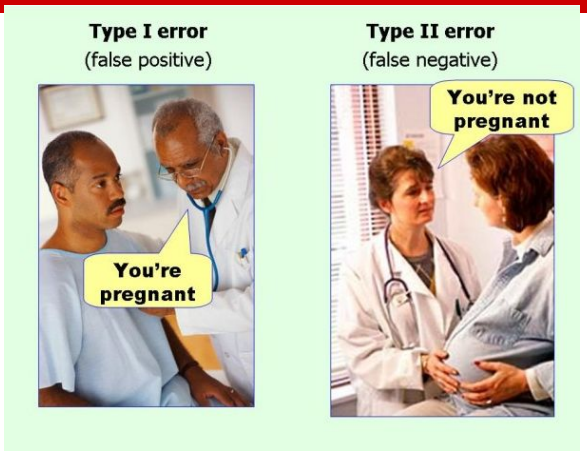
- Research goal
  - We study the relation between pupil dilatation and attentional breadth

- Hypothesis
  - Broader the breadth of attention is, the pupil dilates more



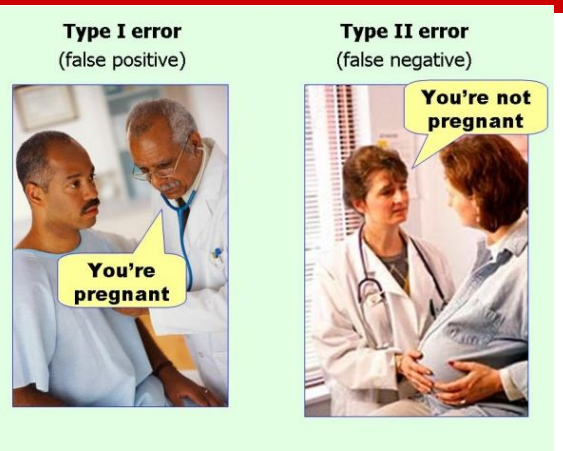


How to analyze it?



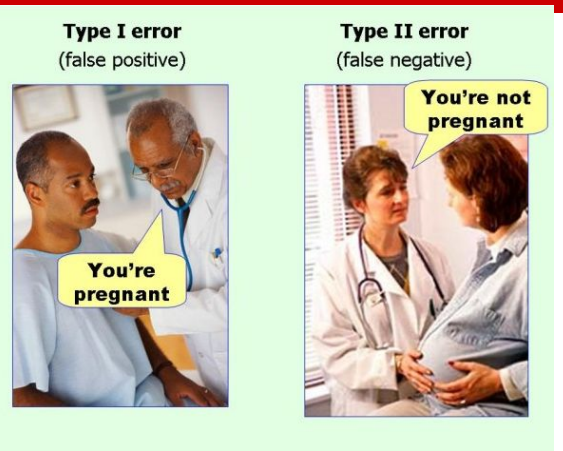
- T-test for each sample
  - multiple comparison problem
  - when conducting so many statistical tests, there is a high chance of type I errors

How to analyze it?



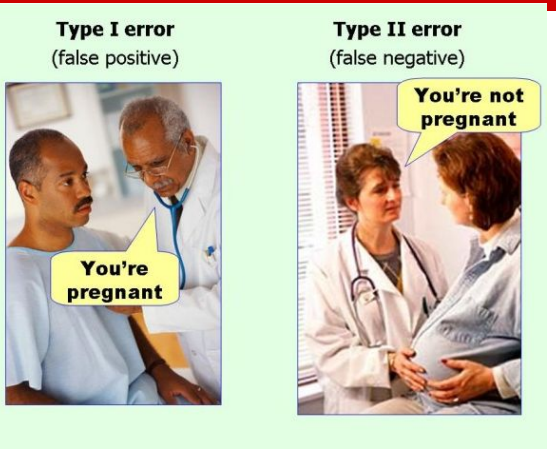
- T-test for each sample
  - multiple comparison problem
  - when conducting so many statistical tests, there is a high chance of type I errors
  - pupil-size data is "autocorrelated": it changes very little from one 10 ms sample to the next (so, Bonferroni can't help)

How to analyze it?



- T-test for each sample
  - multiple comparison problem
  - when conducting so many statistical tests, there is a high chance of type I errors
  - pupil-size data is "autocorrelated": it changes very little from one 10 ms sample to the next (so, Bonferroni can't help)
  - **so NO!**

How to analyze it?



- T-test for each sample
  - multiple comparison problem
  - when conducting so many statistical tests, there is a high chance of type I errors
  - pupil-size data is "autocorrelated": it changes very little from one 10 ms sample to the next (so, Bonferroni can't help)
  - **so NO!**

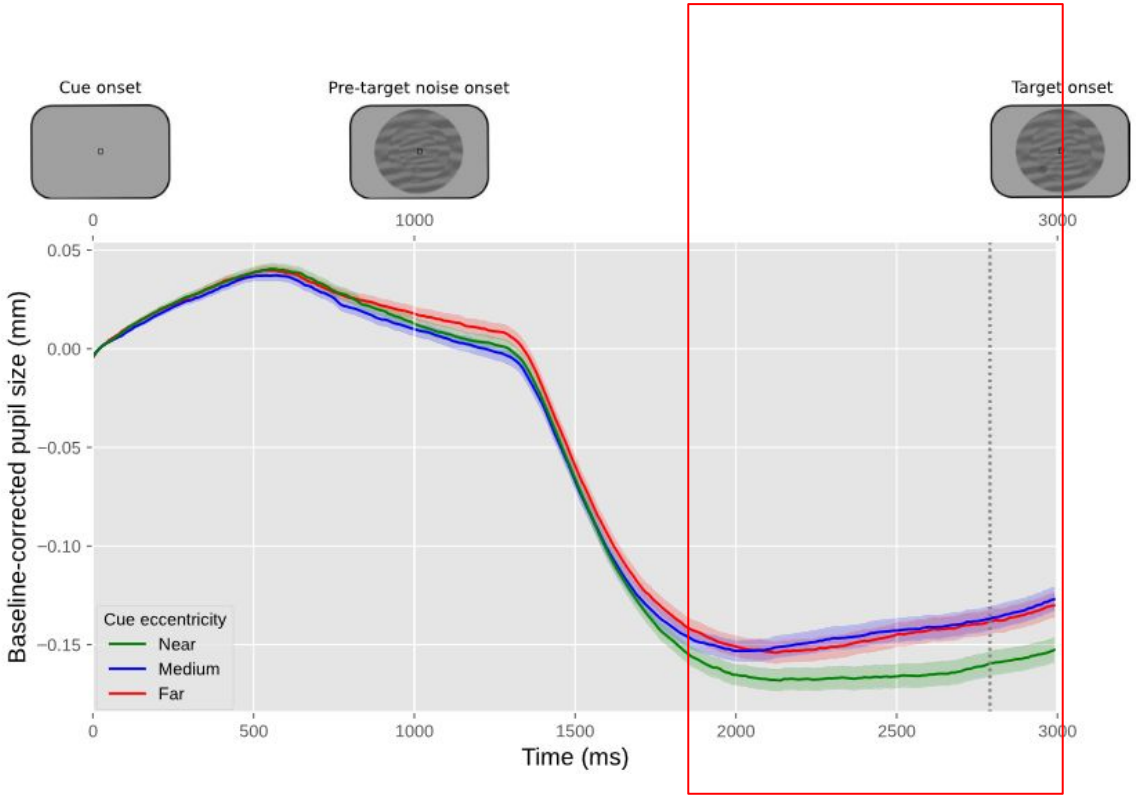
- **WHAT IF...** we say that at least 2000 contiguous milliseconds should be $p<0.05$
  - Makes sense, but it is not a formal way to correct for multiple comparisons, **so NO!** (also Hershman et al., 2022 for a similar approach using Bayes Factors). This improves the issue somewhat, but it is not a formal way to correct for multiple comparisons; therefore, no longer recommended.

Three ways to circumvent MC problem

- **using predetermined window of interest** (a-priori knowledge)
- **cluster-based permutation testing** (reduce dataspace to a single cluster metric)
- **cross-validation** (training/testing sets)

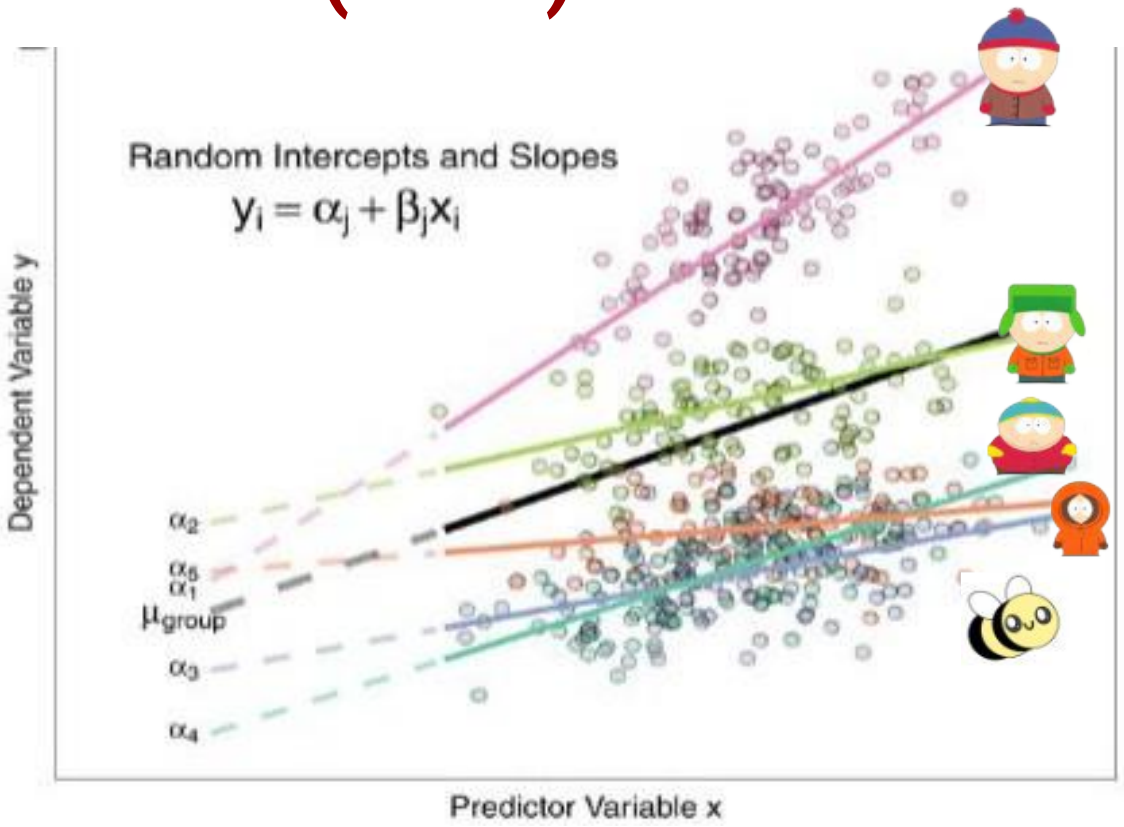
Using a predetermined window + LME

- Choosing a time window
  - When do we expect the effect?
- Take in account the sluggish eye response
  - A big enough time window
- Be aware…
  - The window is a subjective choice;
  - it is good to make this choice **before** running any statistical analyses!

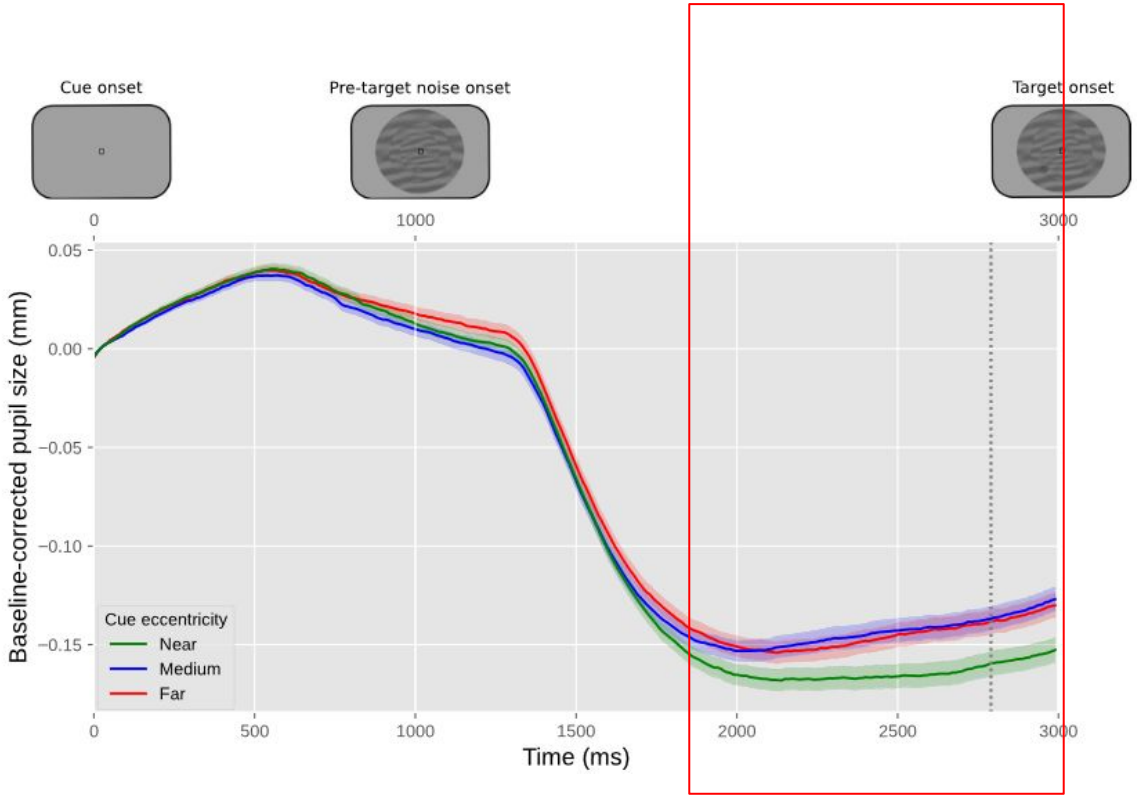


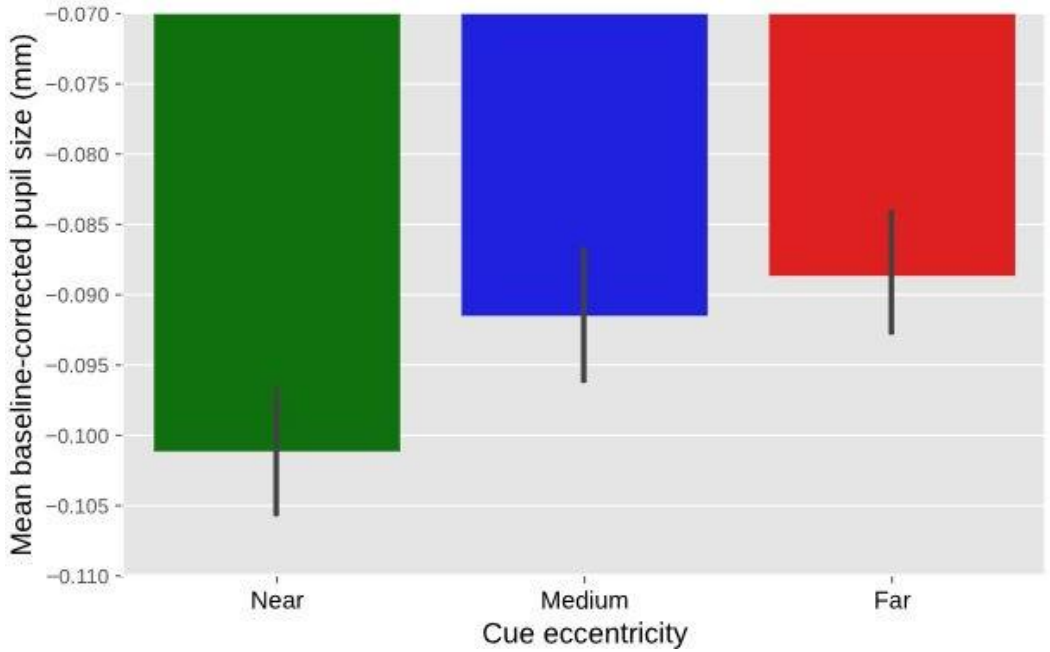
Linear Mixed Effects (LME)

subject nr	RT
	250
	1160
	840
	1970
	730
	950
	860
	940
	1170
	830
	950
	860
	840
	1070
	810
	250
	360
	340
	170



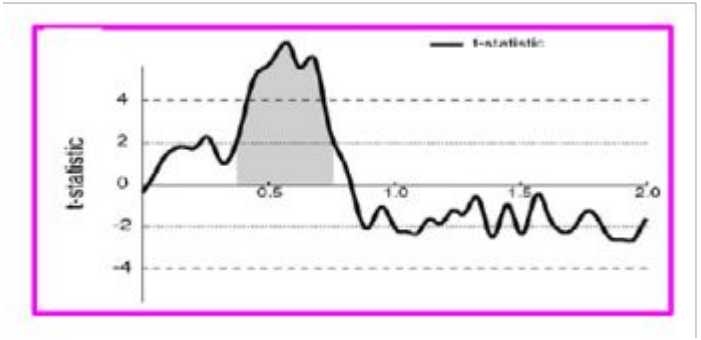
Using a predetermined window + LME





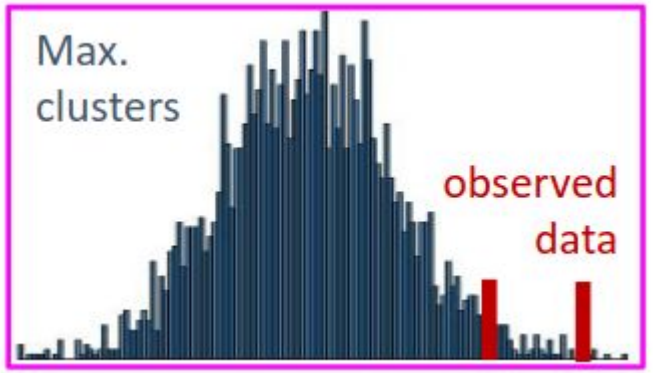
Cluster-based permutation testing

- Let's go back to 200 consecutive tests
- **given that we are likely to find some samples for which p < .05, how likely are we to find a cluster of the size that we observed or larger, assuming that the null hypothesis is true?**
  - Find a largest cluster of neighbouring data points that each exceed some threshold
  - Randomly permute conditions = create empirical null-distribution (permutation distribution) (e.g. 10.000 permutations)
  - Evaluate cluster(s) in original data under permutation distribution (p = proportion of permutations yielding a larger cluster)



pp	Orig.	Perm 1	Perm 2
1	A-B	A-B	B-A
2	A-B	B-A	B-A
3	A-B	B-A	A-B
4	A-B	A-B	B-A

And so on...



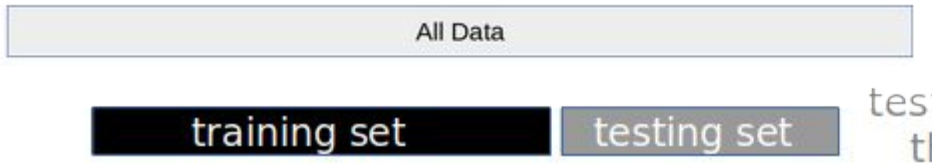
Cluster-based permutation testing

- Identify clusters in data where the conditions significantly differ
- randomly shuffling the condition labels on a trial basis
- This procedure of shuffling and analyzing is repeated a large number of times (e.g. 1000 times) which results in a distribution of false-alarm clusters.
- The p value for the actual cluster (i.e., the cluster size based on the unshuffled data) is then the proportion of false-alarm cluster sizes that are larger than or equal to the actual cluster size.

Cross-validation testing

Cross-validation testing

to localize an effect



test the effect at the location that was identified in the training set

Cross-validation testing

to localize an effect

All Data

training set | testing set

test the effect at the location that was identified in the training set

How are we going to split it?

50-50?
75-25?
90-10?

Cross-validation testing

to localize an effect

test the effect at the location that was identified in the training set

Equal parts!

"N-fold"

[4-fold is very common]

Cross-validation testing

to localize an effect

test the effect at the location that was identified in the training set

Equal parts!

"N-fold"

[4-fold is very common]

Do we just "cut" the data?

Do we split by conditions?

Cross-validation testing

to localize an effect

test the effect at the location that was identified in the training set

Cross-validation testing

to localize an effect

test the effect at the location that was identified in the training set

Equal parts!

"N-fold"

[4-fold is very common]

Split type:
interleaved/random

Cross-validation testing

to localize an effect

test the effect at the location that was identified in the training set

Equal parts!
"N-fold"
[4-fold is very common]

LME is then conducted for each sample of the training set.

LME is then conducted for each sample of the training set.

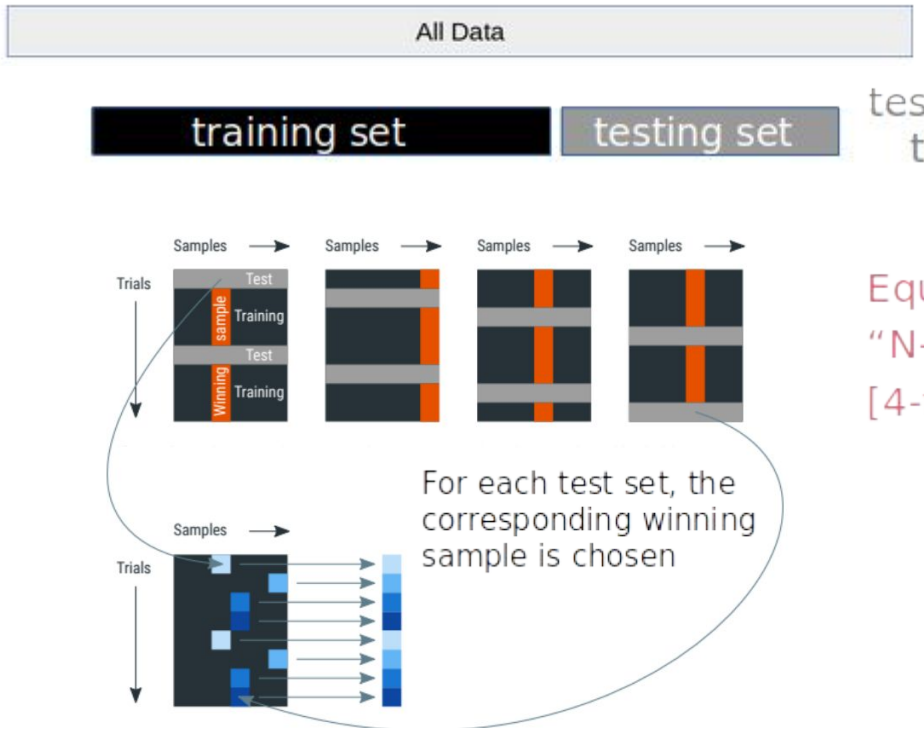
LME is then conducted for each sample of the training set.

LME is then conducted for each sample of the training set.

Split type: interleaved/random

Cross-validation testing

to localize an effect



test the effect at the location that was identified in the training set

Equal parts!
"N-fold"
[4-fold is very common]

Split type:
interleaved/random

Cross-validation testing

to localize an effect

test the effect at the location that was identified in the training set

Equal parts!
"N-fold"
[4-fold is very common]

Split type:
interleaved/random

Cross-validation testing

- 75% of the data is used for the training set
- 25% of the data is used for the test set
- the data is (by default) split in an interleaved fashion
- A linear mixed effects model is then conducted for each sample of the training set. The sample that yields the highest z value in the training set is used as the sample-to-be-tested for the test set. This procedure is repeated four times, using a different training set each time, until all samples have been part of a test set, and a sample-to-be tested has therefore been determined for the entire dataset.
- Finally, a single linear mixed effects model is conducted using the sample-to-be tested for each trial as a dependent measure. This means that the dependent variable consists of a column of (baseline-corrected) pupil-size values that correspond to different samples for different trials.