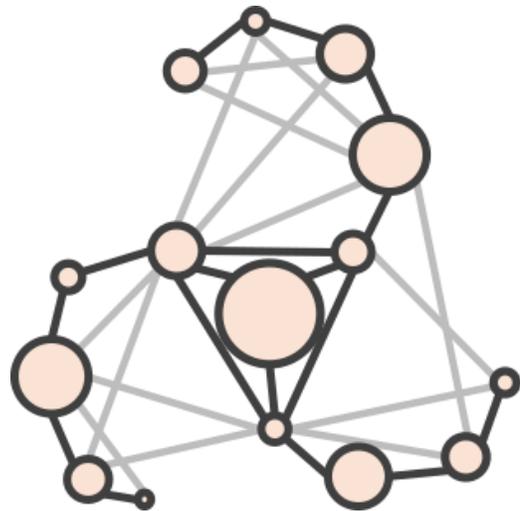


Experimental Reproducibility in Networking Research



Romain Jacob
ETH Zurich

GT Reproducibilité
GDR Réseaux et Systèmes Distribués
May 10, 2022

 @RJacobPartner

2015—2019

Doctorate from ETH Zurich

with Prof. Lothar Thiele

in real-time communication protocols
for low-power embedded systems

Since then

PostDoc in Computer Networks

with Prof. Laurent Vanbever

focus on protocol design
for “greening” the Internet

Key questions

1. How to design experiments?
2. How to analyse data?

Focus

Networking

Field

Performance evaluations

Exp. type

Goal

Foster replicability

|
To be clarified

This is an interactive session

Questions are welcome!

- Write any question in the chat;
- There will be several time slots for questions

Don't be shy

Direct question by voice
are welcome during the Q&A

Please

Stay muted during
the rest of the presentation

45' Lecture

10' Hands-on

10' Break

20' Lecture

Wrap-up & Discussions

45' Lecture

10' Hands-on

10' Break

20' Lecture

Wrap-up & Discussions

Why replicability matters
Case by example

Understanding variability
The three timescales

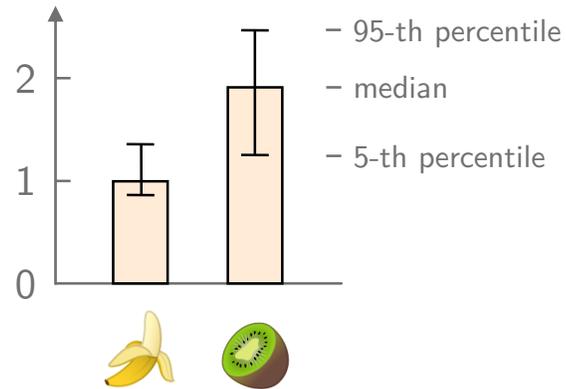
Know your data
Use the right statistics

Why replicability matters
Case by example

Understanding variability
The three timescales

Know your data
Use the right statistics

Energy consumption (normalized)



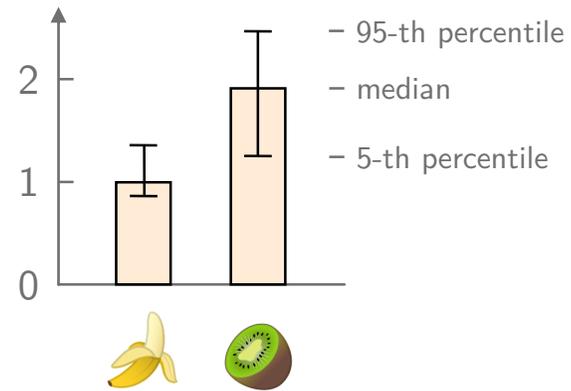
A team designed Banana, a new (and amazing!) ultra-low-power wireless communication protocol.

They set up an experiment to validate their claims.

- They deploy Banana on a **real-world testbed**;
- They run **one benchmark problem** for data collection from the IoT Bench;
- They **compare** Banana's performance against the state-of-the-art Kiwi protocol, which is re-run as part of the experiment.
- Each protocol is **tested 10 times**.



Energy consumption
(normalized)



Claim

“🍌 achieves a 2x improvement
over 🥝.”

You are
reviewing
the paper

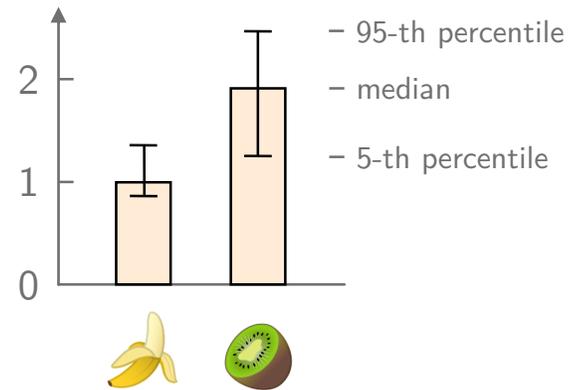
Are ten runs enough
to support this claim?

slido

Are ten runs enough?

 Start presenting to display the poll results on this slide.

Energy consumption
(normalized)



Claim

“🍌 achieves a 2x improvement
over 🥝.”

You are
reviewing
the paper

Are ten runs enough
to support this claim?

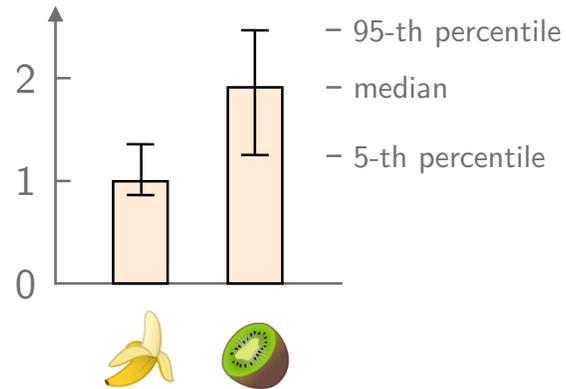
**How many runs do
you think are required?**

slido

How many do you think are required?

 Start presenting to display the poll results on this slide.

Energy consumption
(normalized)



Claim

“🍌 achieves a 2x improvement
over 🥝.”

You are
reviewing
the paper

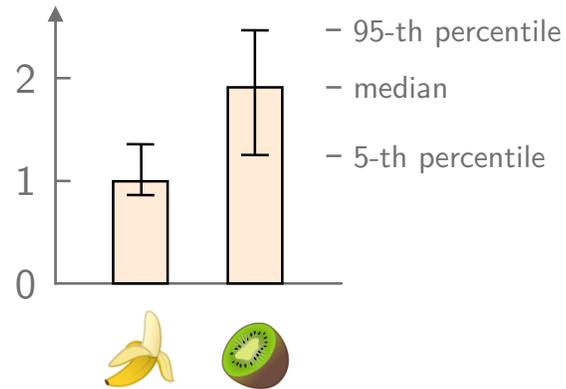
Are ten runs enough
to support this claim?

How many runs do
you think are required?

Cannot say.

▶ Which “performance”
are we talking about?

Energy consumption
(normalized)



Claim

“🍌 achieves a 2x improvement over 🥝, **in the median case.**”

You are reviewing the paper

Are ten runs enough to support this claim?

How many runs do you think **are required?**

Cannot say.

▶ Which “performance” are we talking about?

If you would repeat the experiment, do you think **you would obtain the same result?**

Hard to say.

▶ What does “same result” mean, really?

These are
hard questions!

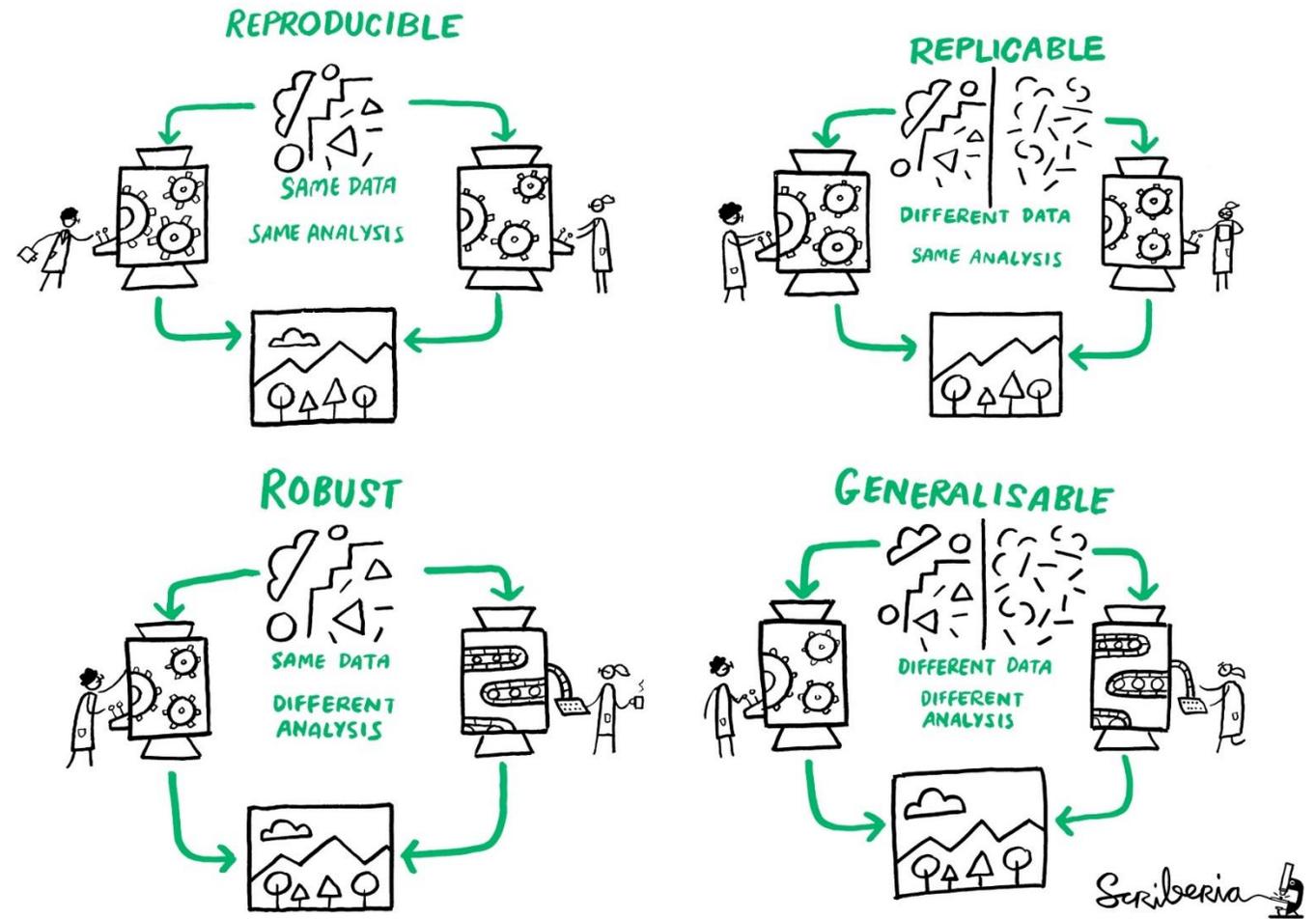
How many runs
are required?

Would you obtain
the same results?

This tutorial presents a **rational methodology**
to address these questions (and others)



What is replicability?



The Turing Way project illustration by Scriberia.
Zenodo. <http://doi.org/10.5281/zenodo.3332807>

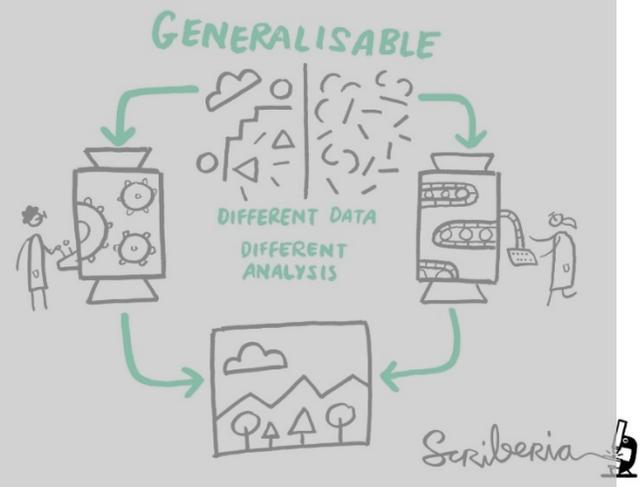
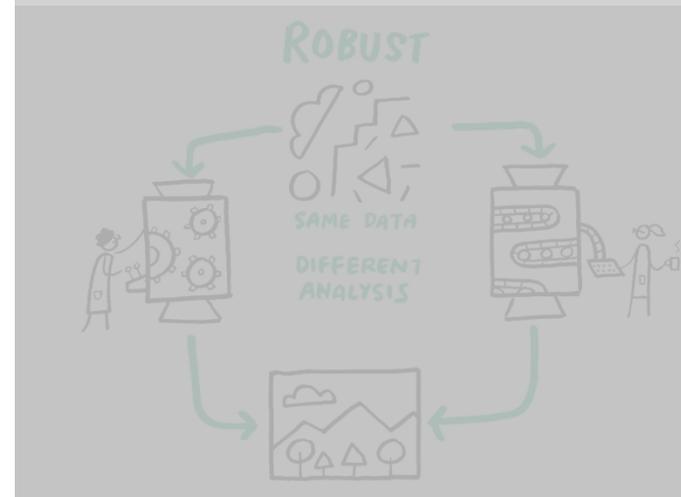
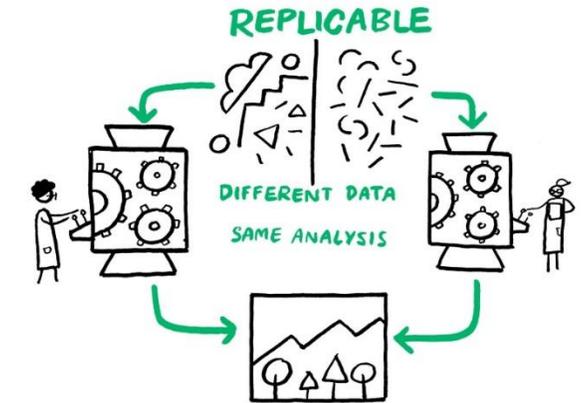
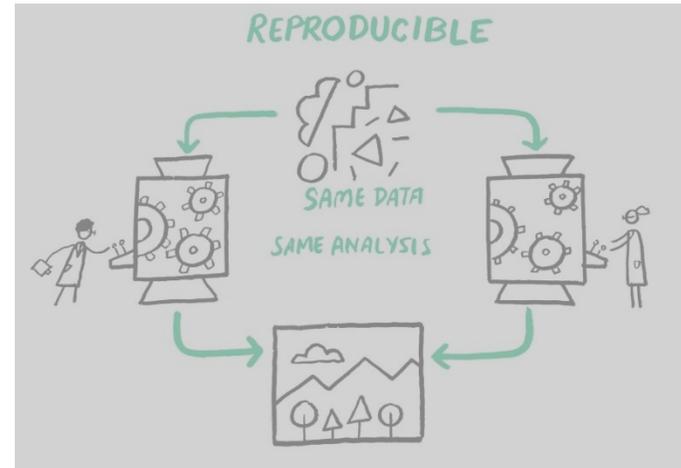
What is replicability?

SAME ANALYSIS

DIFFERENT ANALYSIS

SAME DATA

DIFFERENT DATA



The Turing Way project illustration by Scriberia.
Zenodo. <http://doi.org/10.5281/zenodo.3332807>

What is replicability? Why does it matter?

Because

No result is “science” if it cannot be independently replicated by others.

In picture



www.zbw-mediataalk.eu

“Is there a reproducibility crisis?”

Poor/no documentation
Artifacts not available
Unstable environment
Analytical bias
Falsification
etc.

90%

of surveyed scientists stated that there is a **reproducibility crisis** in their research field.

52%

Yes, a significant crisis.

38%

Yes, a slight crisis.

7%

I don't know.

3%

No, there is no crisis.

Is There a Reproducibility Crisis?
Monya Baker. Nature News (2016)

“Is there a reproducibility crisis?”
Does it really affect CS? Networking?

Is Big Data Performance Reproducible in Modern Cloud Networks?

Spoiler alert: not so much...

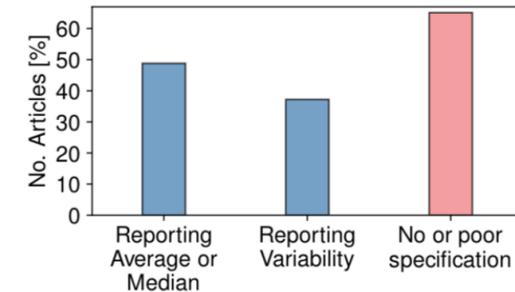
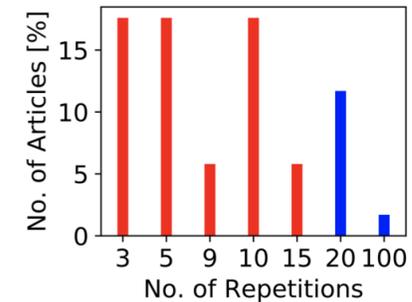
“Is there a reproducibility crisis?”

Does it really affect CS? Networking?

Variability is disconsidered in performance evaluations

Main findings:

- **Most articles report 3-10 repetitions, few report > 10**
- **> 50% of articles have no or poor experiment specification!**
- **< 50% report only average or median**
- **~ 40% report variability**
- **Cited articles > 11,000 citations**



The literature addresses replicability issues

Two examples

Mainly guidelines

The Dagstuhl Beginners Guide to Reproducibility for Experimental Networking Research

Vaibhav Bajpai
TU Munich
bajpai@in.tum.de

Anna Brunstrom
Karlstad University
anna.brunstrom@kau.se

Anja Feldmann
MPI for Informatics
anja@mpi-inf.mpg.de

Wolfgang Kellerer
TU Munich
wolfgang.kellerer@tum.de

Aiko Pras
University of Twente
a.pras@utwente.nl

Henning Schulzrinne
Columbia University
hgs@cs.columbia.edu

Georgios Smaragdakis
TU Berlin
georgios@inet.tu-berlin.de

Matthias Wählisch
Freie Universität Berlin
m.waehlich@fu-berlin.de

Klaus Wehrle
RWTH Aachen University
klaus@comsys.rwth-aachen.de

This article is an editorial note submitted to CCR. It has NOT been peer reviewed.

The authors take full responsibility for this article's technical content. Comments can be posted through CCR Online.

ABSTRACT

Reproducibility is one of the key characteristics of good science, but hard to achieve for experimental disciplines like Internet measurements and networked systems. This guide provides advice to researchers, particularly those new to the field, on designing experiments so that their work is more likely to be reproducible and to serve as a foundation for follow-on work by others.

CCS CONCEPTS

• General and reference → Surveys and overviews;

KEYWORDS

Experimental networking research; Internet measurements; Reproducibility; Guidance

1 INTRODUCTION

Good scientific practice makes it easy for researchers other than the authors to reproduce, evaluate and build on the work. Achieving these goals, however, is often challenging and requires planning and care. We attempt to provide guidelines for researchers early in their career and students working in the field of experimental networking research, and as a reminder for others. We begin by summarizing the terminology (§ 1.1) that will be used throughout this article. We then elaborate the goals and principles (§ 1.2), describe best practices required for reproducibility in general (§ 2) and for specific research methodologies (§ 3), provide tool recommendations (§ 4) and point to additional resources (§ 5).

Table 1: Repeatability, replicability, and reproducibility as defined by ACM [1].

Term	Level of change	
	Team	Setup
Repeatability	same	same
Replicability	different	same
Reproducibility	different	different

1.1 ACM Terminology

The terms repeatability, replicability and reproducibility are often used interchangeably and may not necessarily be used consistently within or across technical communities. Since the Association for Computing Machinery (ACM) [1] publishes a significant fraction of papers in networked systems and Internet measurements, we draw on their definitions and summarize them in Table 1.

Repeatability is achieved when a researcher can obtain the same results for her own experiment under exactly the same conditions, i.e., she can reliably repeat her own experiment ("Same team, same experimental setup").

Replicability allows a different researcher to obtain the same results for an experiment under exactly the same conditions and using exactly the same artifacts, i.e., another independent researcher can reliably repeat an experiment of someone other than herself ("Different team, same experimental setup").

Reproducibility enables researcher other than the authors to obtain the same results for an experiment under

SIGPLAN Empirical Evaluation Checklist

This checklist is meant to support informed judgement, not supplant it.

Clearly Stated Claims

- Claims not explicit:** Claims must be explicit in order for the reader to assess whether the empirical evaluation supports them. Missing claims cannot possibly be assessed. Claims should also aim to state not just what is achieved but how. *Example Violation:* [Image of a claim without methodology]
- Claims not appropriately scoped:** The truth of a claim should clearly follow from the evidence provided. Claims that are not fully supported mislead readers. 'Works for all Java' is over-broad when based on a subset of Java. Other examples are 'works on real hardware' when evaluating only with (unrealistic) simulation, and 'automatic process' when requiring human intervention. *Example Violation:* [Image of a claim with limited scope]
- Fails to acknowledge limitations:** A paper should acknowledge its limitations to place the scope of its results in context. Stating no limitations at all, or only tangential ones, while omitting the more relevant ones may mislead the reader into drawing overly strong conclusions. This could hold back efforts to publish future improvements, and may lead researchers down wrong paths. *Example Violation:* [Image of a claim with no limitations]
- Fails to compare against appropriate baseline:** Empirical evidence for a claim that a technology/system improves upon the state-of-the-art should include a comparison against an appropriate baseline. The lack of a baseline means empirical evidence lacks context. A 'straw man' baseline that is misrepresented as state-of-the-art is also problematic, as it would inflate apparent benefit. *Example Violation:* [Image of a claim with a weak baseline]
- Comparison is unfair:** Comparisons to a competing system should not unfairly disadvantage that system. Doing so would inflate the apparent advantage of the proposed system. For example, it would be unfair to compare the state-of-the-art baseline at -O0 optimization level, while using -O3 for the proposed system. *Example Violation:* [Image of a claim with unfair comparison]

Suitable Comparison

- Inappropriate suite:** Evaluations should be conducted using appropriate established benchmarks where they exist so that claimed results are more likely to generalize. Not doing so may yield results that are not sufficiently general. Established suites should be used in context: e.g. it would be wrong to use a single-threaded suite for studying parallel performance. *Example Violation:* [Image of a claim using an inappropriate benchmark suite]
- Unjustified use of non-standard suite(s):** The use of standard benchmark suites improves the comparability of results. However, sometimes a non-standard suite, such as one that is subtitled or non-isolated, is the better choice. In that case, a rationale, and possible limitations, must be provided to demonstrate why using a standard suite would have been worse. *Example Violation:* [Image of a claim using a non-standard suite]
- Kernels instead of full applications:** Kernels can be useful and appropriate in a broader evaluation. However, a claim that a system benefits applications should be tested on such applications directly, and not only on micro-kernels, which may lack important characteristics of full applications. *Example Violation:* [Image of a claim using kernels instead of full applications]

Principled Benchmark Choice

- Insufficient number of trials:** Modern systems with non-deterministic properties may require many trials (e.g., of a single time measurement) to characterize their behavior adequately. Failure to do so risks treating noise as signal. Similarly, more trials may be needed to get the system into an intended state (e.g., into a steady state that avoids warm-up effects). *Example Violation:* [Image of a claim with insufficient trials]
- Inappropriate summary statistics:** Summary statistics such as mean and median can usefully characterize many results. But they should be selected carefully, because each statistic presents an accurate view only under appropriate circumstances. An inappropriate summary may amplify noise or hide an important trend. *Example Violation:* [Image of a claim with inappropriate summary statistics]
- No data distribution reported:** A measure of variability (e.g., variance, std. deviation, quantiles) and/or confidence intervals is needed to understand the distribution of the data. Reporting just a measure of central tendency (e.g., a mean or median) can mislead the reader, especially when the distribution is bimodal or has significant variance. *Example Violation:* [Image of a claim with no data distribution reported]

Adequate Data Analysis

- Insufficiently truncated axes:** Graphs provide a visual intuition about a result. A truncated graph (with an axis not including zero) will exaggerate the importance of a difference. 'Zooming' in to the interesting range of an axis can sometimes aid exposition, but should be pointed out explicitly to avoid being misleading. *Example Violation:* [Image of a truncated graph]
- Ratios plotted incorrectly:** Incorrectly plotted ratios badly mislead visual intuition. For example, 2.0 and 0.5 are reciprocals, but their linear distance from 1.0 does not reflect that, so plotting those numbers on a linear scale significantly distorts the result. This misleading effect can be avoided either by using a log scale or by normalizing to the lowest (highest) value. *Example Violation:* [Image of a ratio plot]
- Inappropriate level of precision:** Measurements reported at a proper level of precision reveal relevant information. Under-precise reports may hide such information, and over-precise ones may overstate the accuracy of a measurement and obscure what is relevant. For example, reporting '49.9%' when the experimental error is +/- 1% overstates the level of precision of the report. *Example Violation:* [Image of a report with inappropriate precision]

Relevant Metrics

- Indirect or inappropriate proxy metric:** Proxy metrics can substitute for direct ones only when the substitution is clearly, explicitly justified. For example, it would be misleading and incorrect to report a reduction in cache misses to claim actual end-to-end performance or energy consumption improvement. *Example Violation:* [Image of a claim using an indirect metric]
- Fails to measure all important Effects:** All important effects should be measured to show the true cost of a system. For example, compiler optimizations may speed up programs at the cost of drastically increasing compile times of large systems, so the compile time should be measured as well as the program speedup. Failure to do so distorts the cost/benefit of the system. *Example Violation:* [Image of a claim missing important effects]

Insufficient information to repeat: Experiments evaluating an idea need to be described in sufficient detail to be repeatable. All parameters (including default values) should be included, as well as all version numbers of software, and full details of hardware platforms. Insufficient information impedes repeatability and comparison of future ideas and can hinder scientific progress. *Example Violation:* [Image of a claim with insufficient information]

Unreasonable platform: The evaluation should be on a platform that can reasonably be said to match the claims; otherwise, the results of the evaluation will not fully support the claims. For example, a claim that relates to performance on mobile platforms should not have an evaluation performed exclusively on servers. *Example Violation:* [Image of a claim on an unreasonable platform]

Ignores key design parameters: Parameters should be explored over a range to evaluate sensitivity to their settings. Examples include the size of the heap when evaluating garbage collection and the size of caches when evaluating a locality optimization. All expected system configurations (e.g., from warmup to steady state) should be considered. *Example Violation:* [Image of a claim ignoring design parameters]

Rated workload generator: Load generators for typical transaction-oriented systems should be 'open loop', to generate work independent of the performance of the system under test. Otherwise, results are likely to be misleading because real-world transaction servers are usually open-loop. *Example Violation:* [Image of a claim with a rated workload generator]

Tested on training set: When a system was developed with close consideration of specific examples, it is essential that the evaluation explicitly perform cross-validation, so that the system is evaluated on data distinct from the training set. For example, a static analysis should not be exclusively evaluated on programs used to inform its development. *Example Violation:* [Image of a claim tested on training set]

Misleading summary of results: The summary of the results must reflect the full range of their character to avoid misleading the reader. For example, it is not appropriate to summarize speedups of 4%, 9%, 7%, and 48% as 'up to 49%'. Instead, the full distribution of results must be reported. *Example Violation:* [Image of a misleading summary of results]

Inappropriately truncated axes: Graphs provide a visual intuition about a result. A truncated graph (with an axis not including zero) will exaggerate the importance of a difference. 'Zooming' in to the interesting range of an axis can sometimes aid exposition, but should be pointed out explicitly to avoid being misleading. *Example Violation:* [Image of a truncated graph]

Ratios plotted incorrectly: Incorrectly plotted ratios badly mislead visual intuition. For example, 2.0 and 0.5 are reciprocals, but their linear distance from 1.0 does not reflect that, so plotting those numbers on a linear scale significantly distorts the result. This misleading effect can be avoided either by using a log scale or by normalizing to the lowest (highest) value. *Example Violation:* [Image of a ratio plot]

Inappropriate level of precision: Measurements reported at a proper level of precision reveal relevant information. Under-precise reports may hide such information, and over-precise ones may overstate the accuracy of a measurement and obscure what is relevant. For example, reporting '49.9%' when the experimental error is +/- 1% overstates the level of precision of the report. *Example Violation:* [Image of a report with inappropriate precision]

The literature addresses replicability issues but it **lacks concrete answers** to practical questions

For example

- How many times should one repeat an experiment?
- Which statistical methods should one use to synthesize results?

In other words

We lack a **concrete methodology** for the design and analysis of experiments.

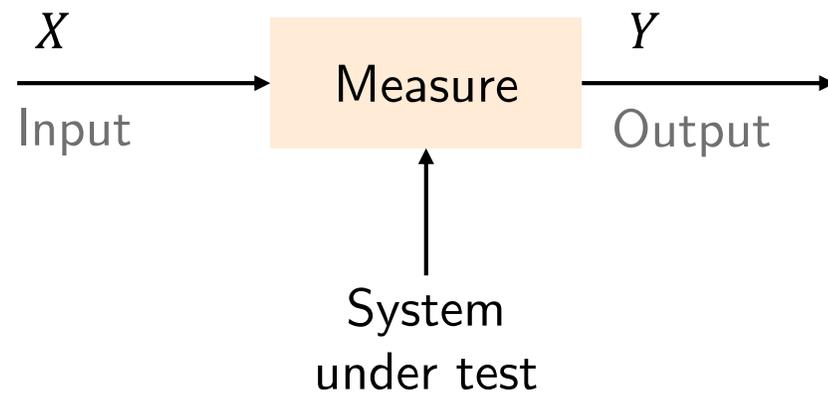
▶ That's TriScale.

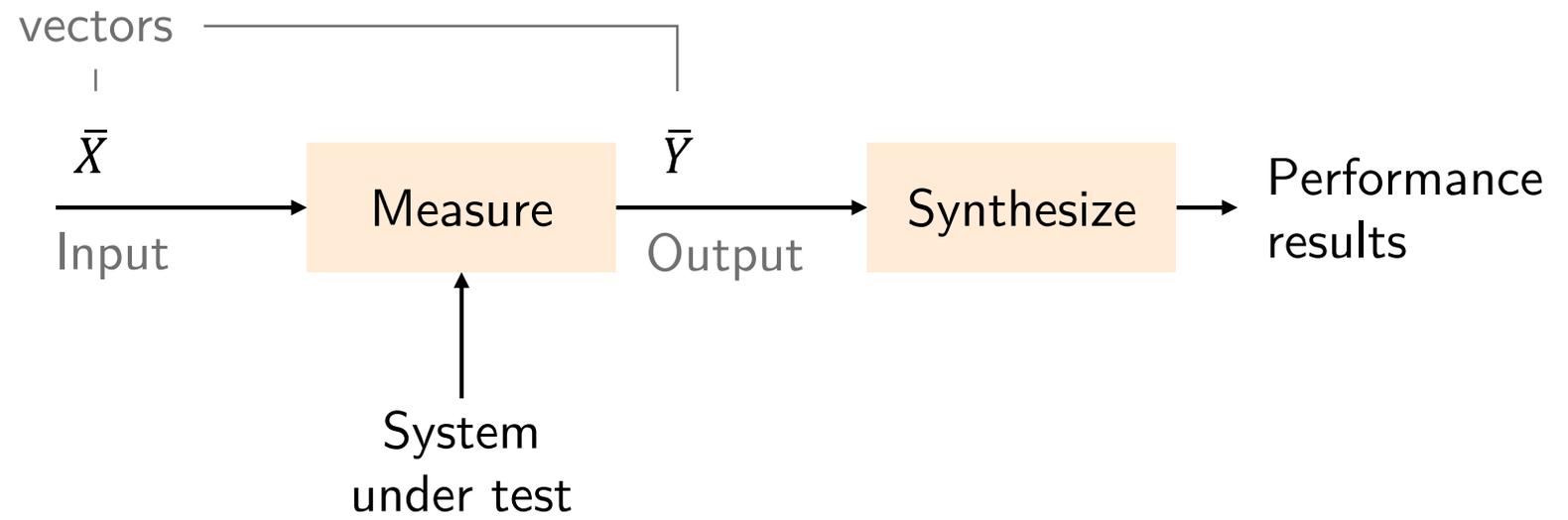
Why replicability matters
Case by example

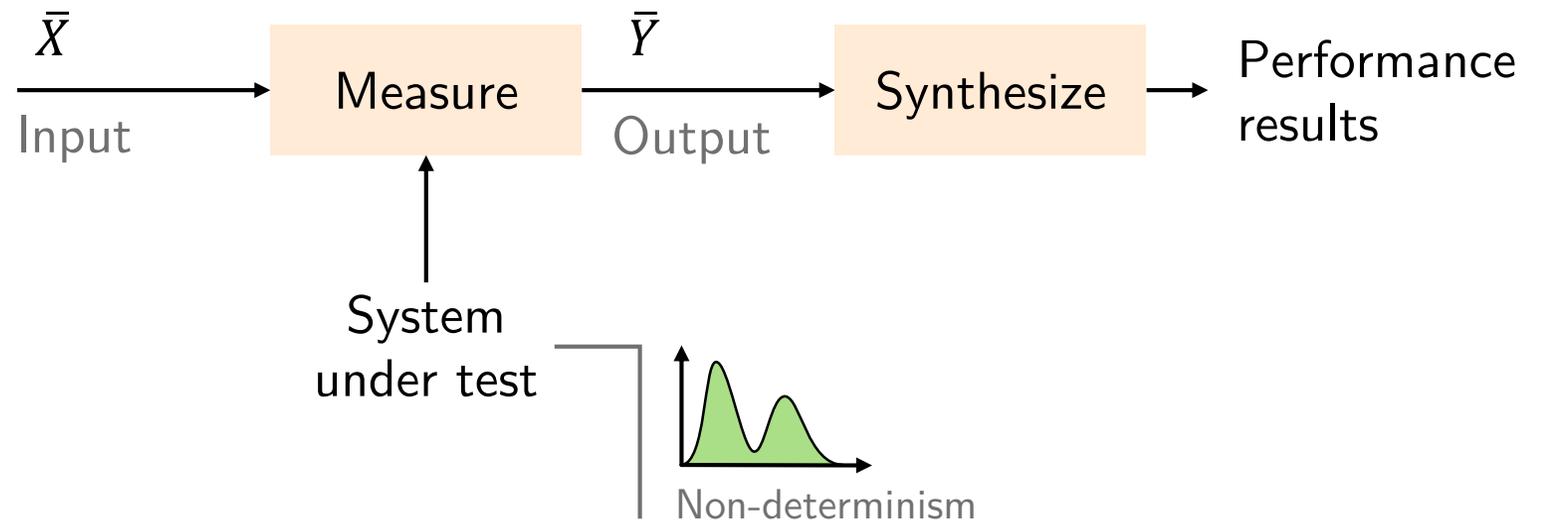
| Understanding variability
The three timescales

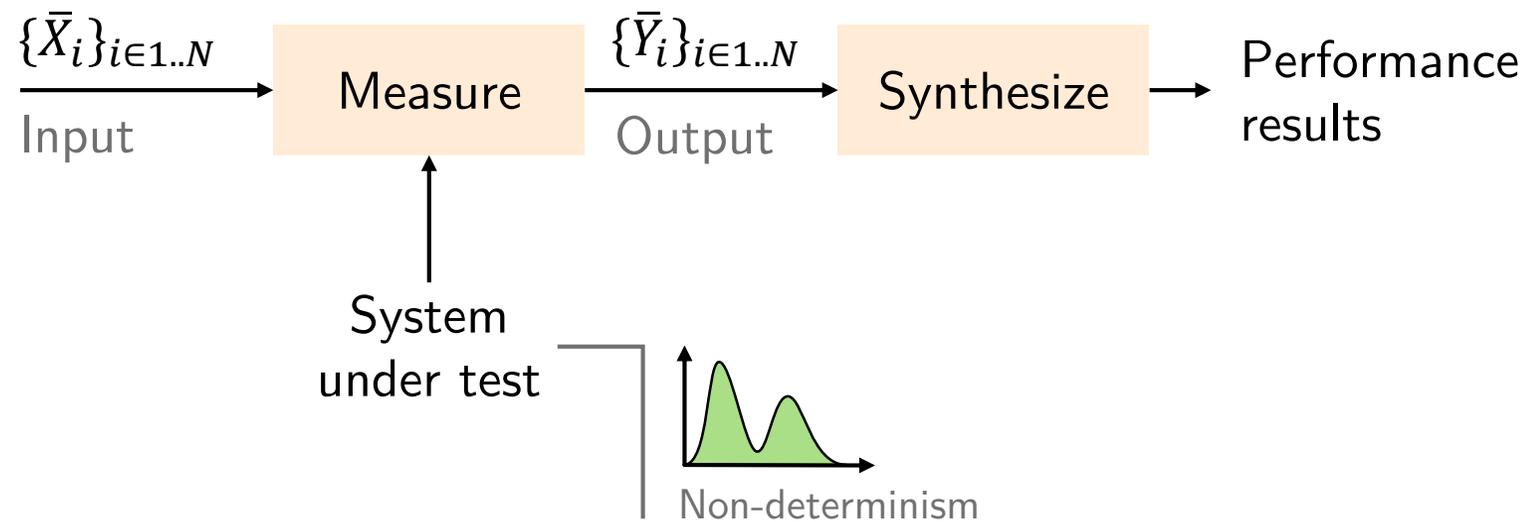
Know your data
Use the right statistics

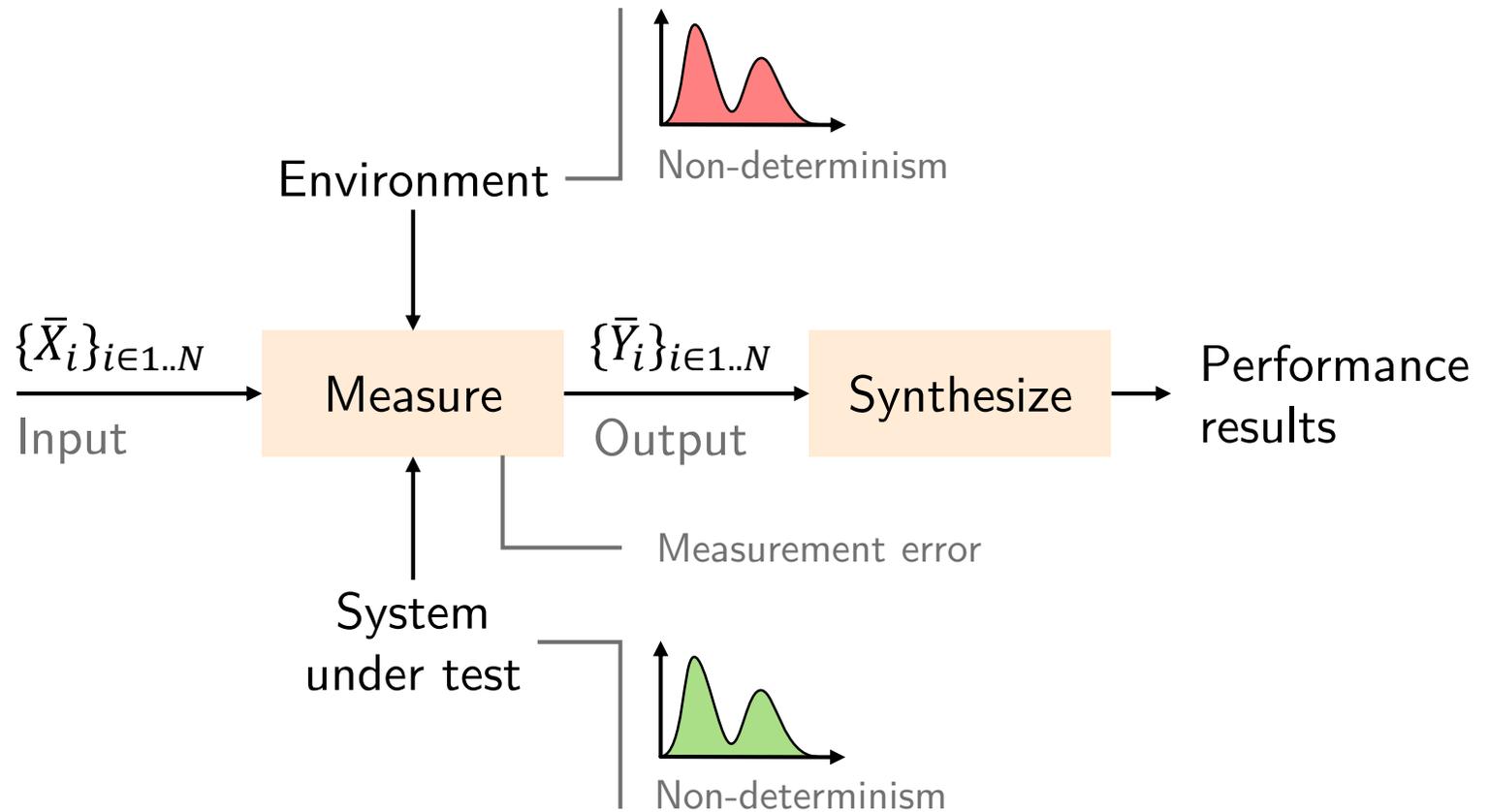
Let us have a closer look
at performance evaluations

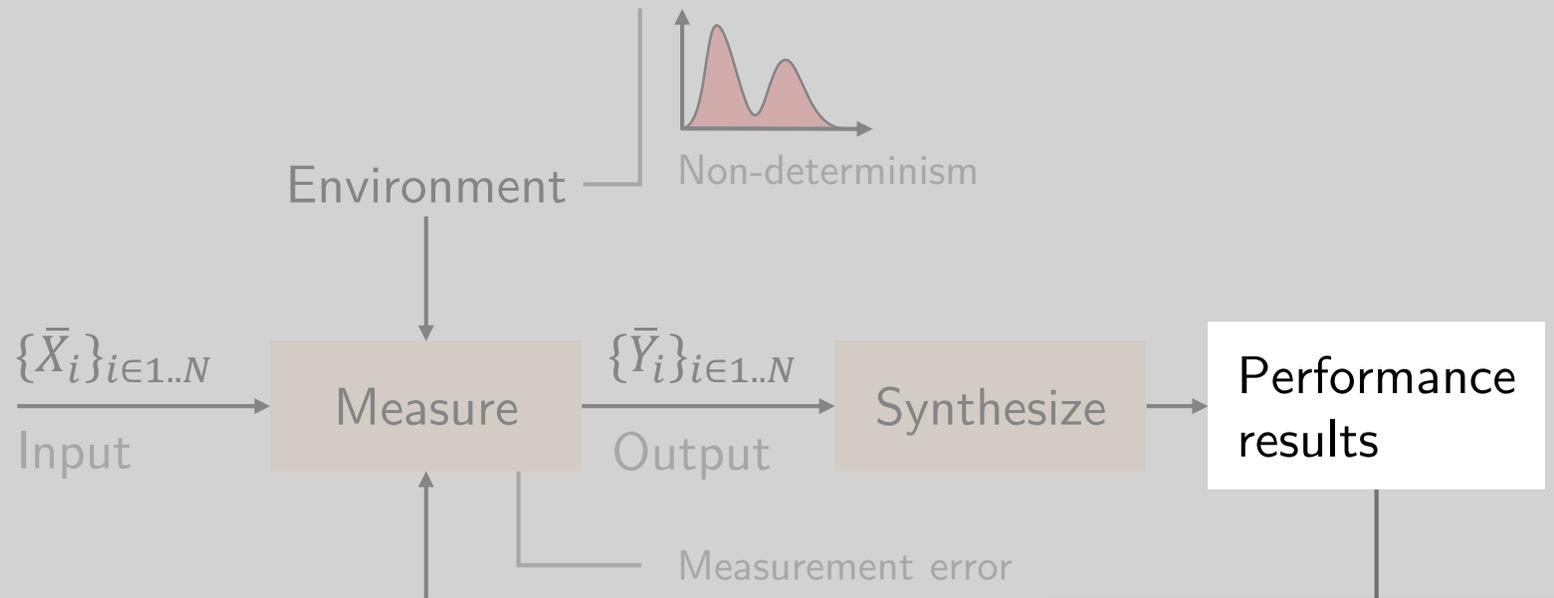






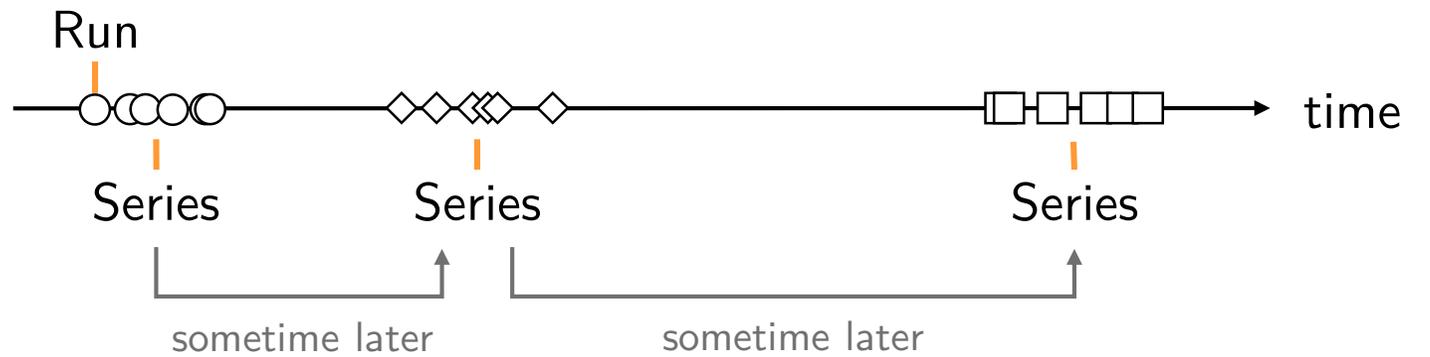
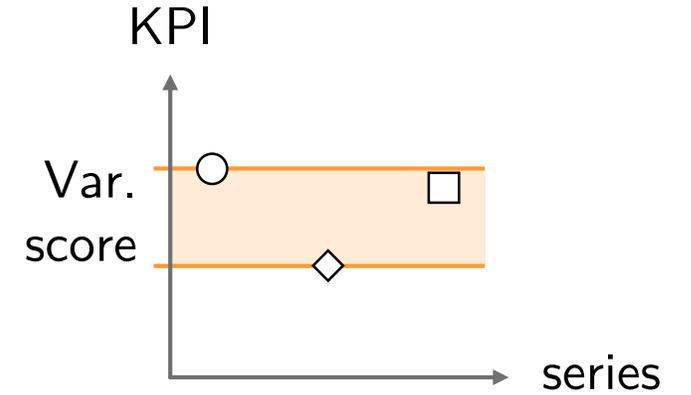
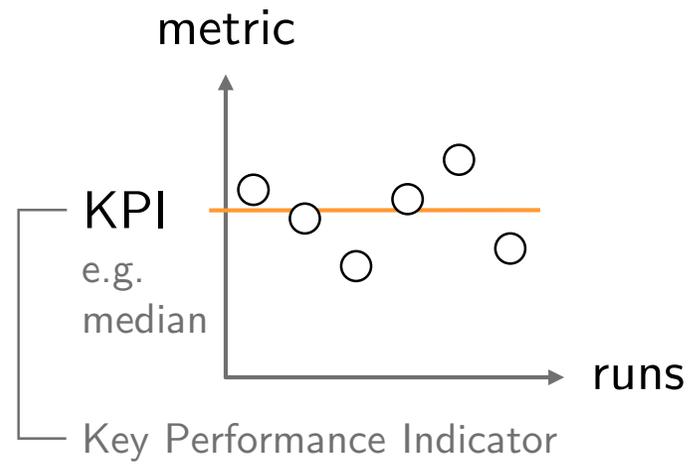
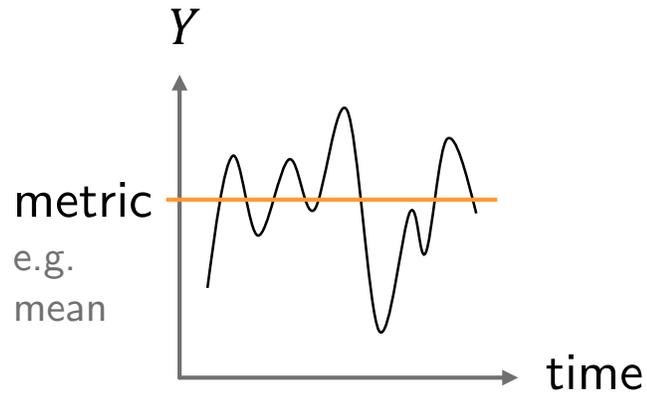




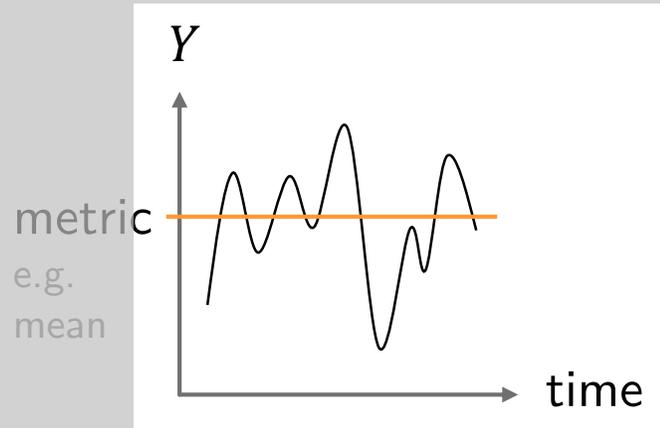


- ▶ What confidence?
- ▶ Are they replicable?

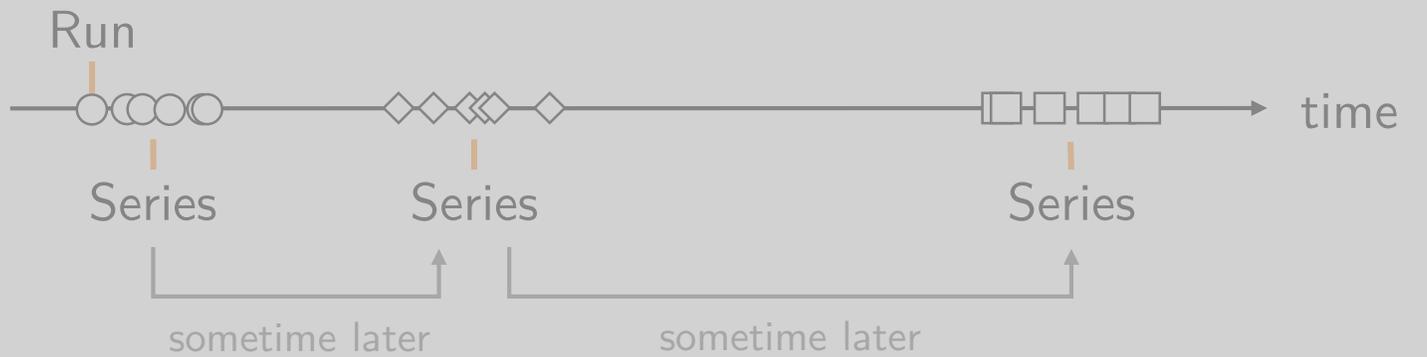
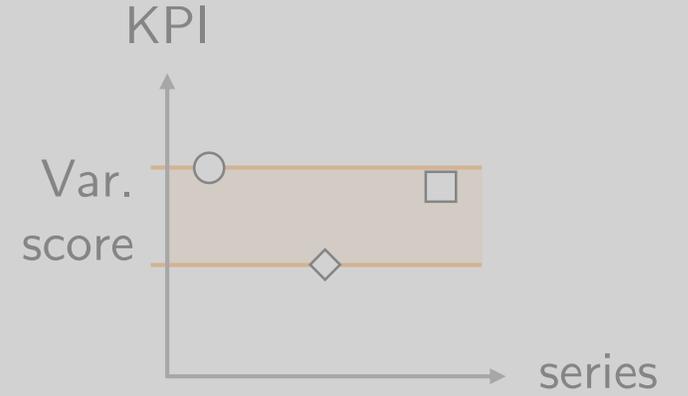
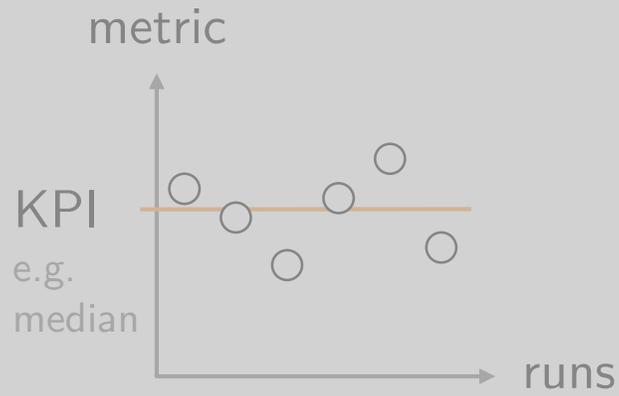
Let us set aside the causes and
consider how variability looks like



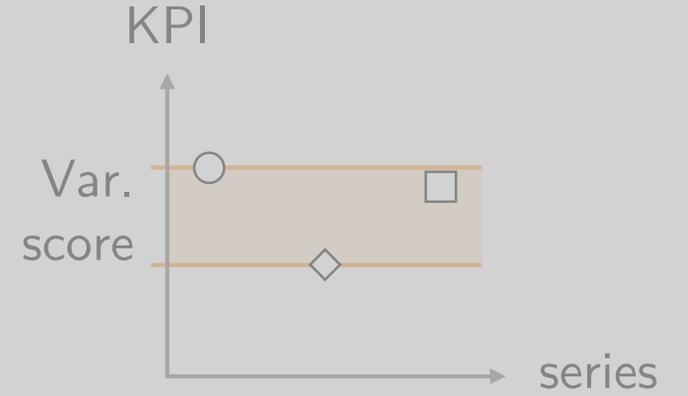
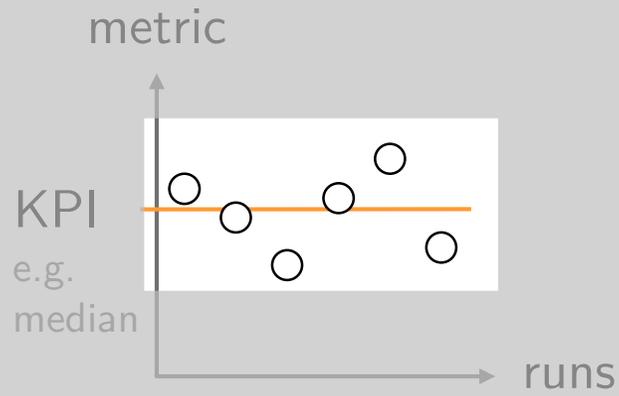
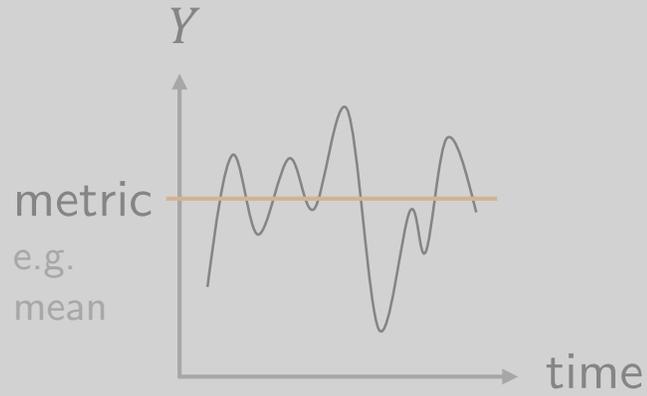
► Performance varies along three different time scales

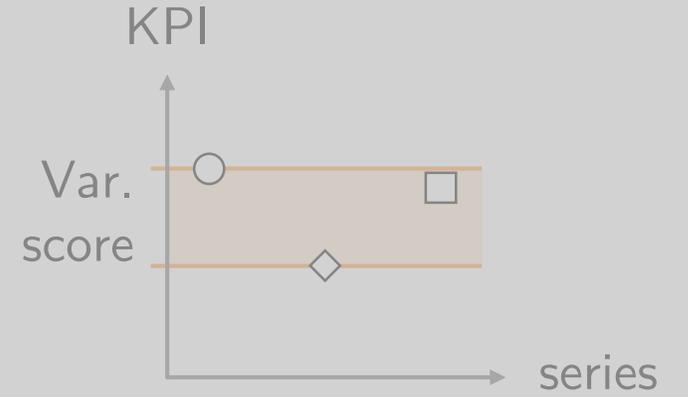
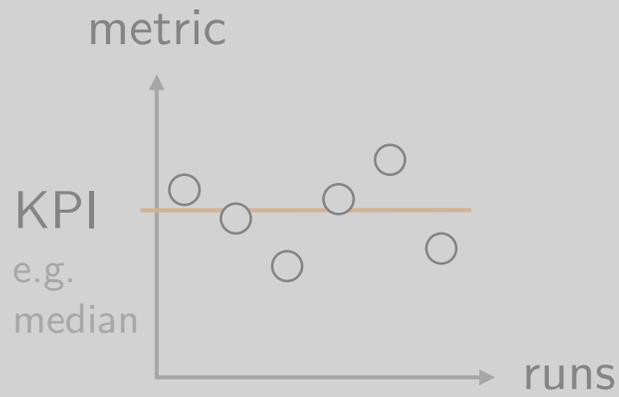
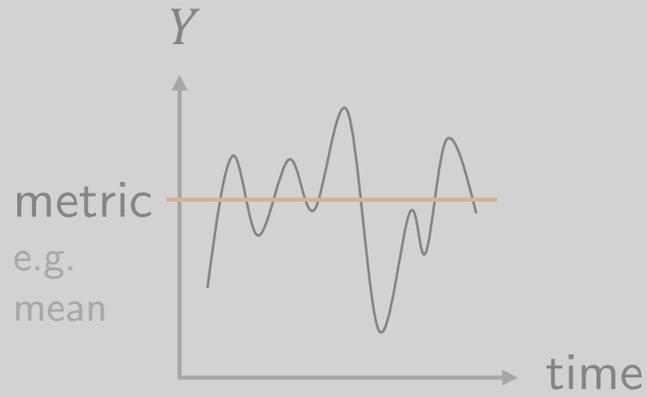


How long should a run be?



How many runs in a series?

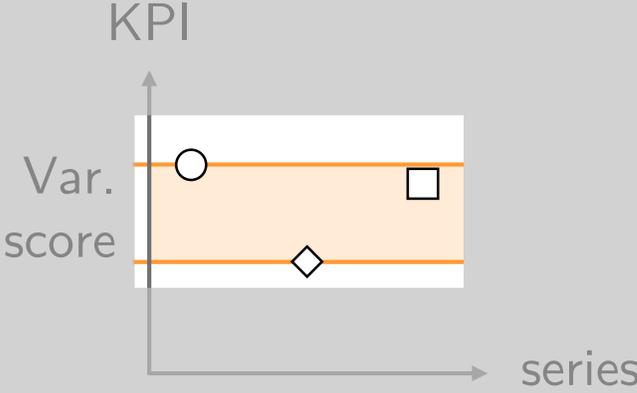
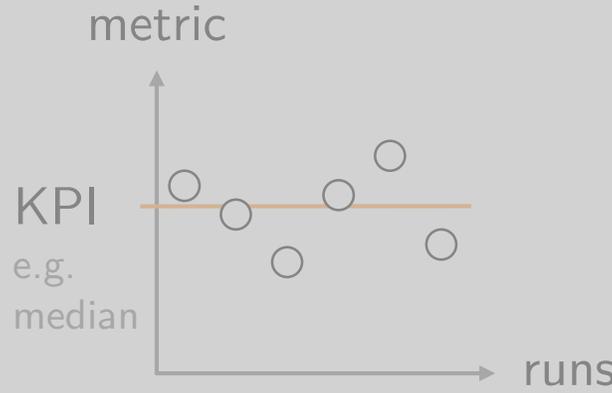
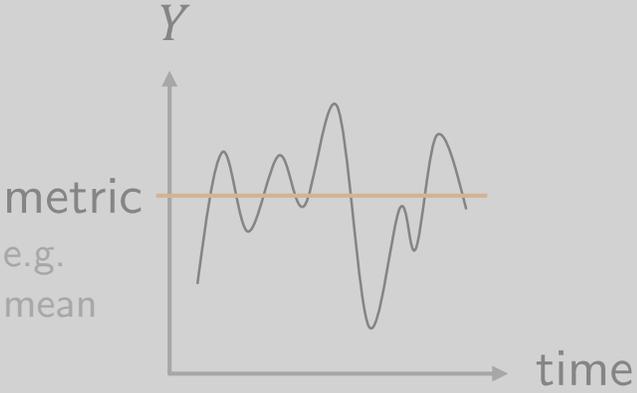




What **time span** for a series?



How many series?



The four questions of experiment design

How long should a run be?

How many runs in a series?

What time span for a series?

How many series?

The four questions of experiment design

How long should a run be?

How many runs in a series?

What time span for a series?

How many series?

Objective

Find **rational answers** to these questions

Making statistical sense



Quantify the trade-off between

- experiment effort
- confidence in the results

Why replicability matters
Case by example

Understanding variability
The three timescales

 Know your data
Use the right statistics

Let us review a few statistics basics

Statistic

def. numerical value computed
from a set of values

Let us review a few statistics basics

Descriptive
statistics

≠

Predictive
statistics

What the
collected data
is like

What the collected data
allows to **infer** about
future/other/unknown data

Let us review a few statistics basics

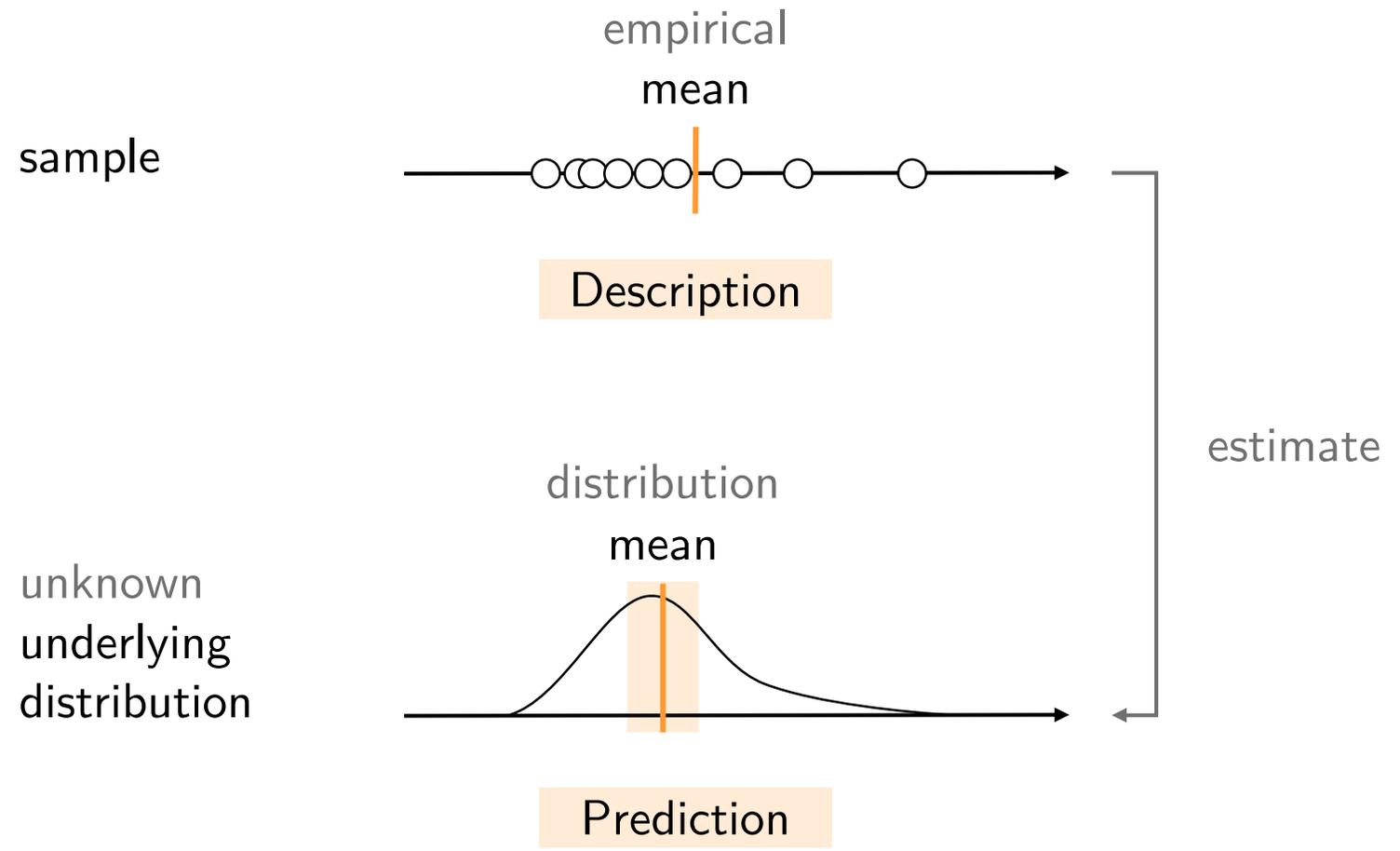
Descriptive statistics

What the collected data is like

≠

Predictive statistics

What the collected data allows to **infer** about future/other/unknown data



Descriptive
statistics

\neq

Predictive
statistics

Sample mean is X

If one draws a new sample,
the sample mean is
"likely" to be "close to" X

More formally?

Much stronger statement

▶ Replicability!

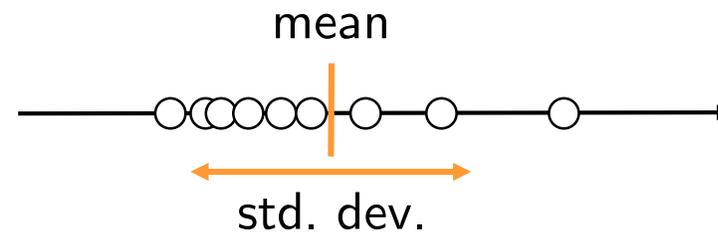
Do these statistics say anything about the expected performance? **No.**

Tendency

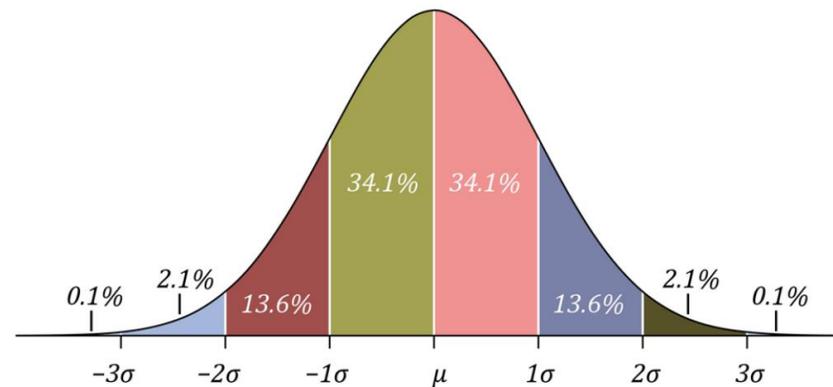
mean

Variability

standard deviation



Prediction?



Do these statistics say anything about the expected performance? **No.**

If thinking so, one makes two mistakes

#1 The mean of the **sample** is not the mean of the **underlying distribution**.

Let us review a few statistics basics

Descriptive statistics

What the collected data is like

≠

Predictive statistics

What the collected data allows to **infer** about future/other/unknown data

Do these statistics say anything about the expected performance? **No.**

If thinking so, one makes two mistakes

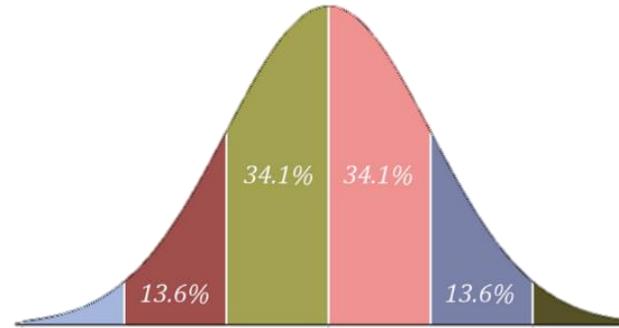
#1

The mean of the **sample** is not the mean of the **underlying distribution**.

#2

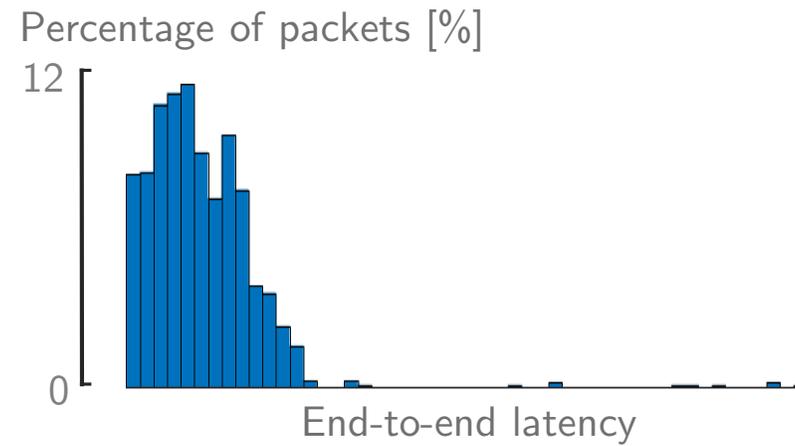
The underlying distribution is **not normal** (almost always).

Normal
Very rare



▶ Cannot be assumed unless you know **for sure**

Not normal
Ubiquitous



Do these statistics say anything about the expected performance? **No.**

If thinking so, one makes two mistakes

#1

The mean of the **sample** is not the mean of the **underlying distribution**.

▶ Use confidence intervals

#2

The underlying distribution is **not normal** (almost always).

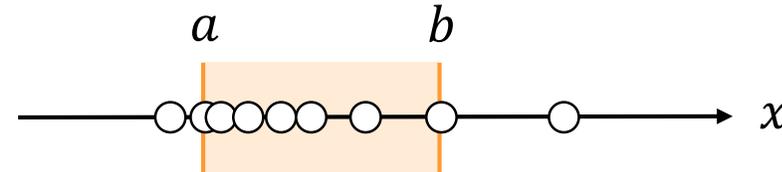
▶ Use non-parametric statistics

Confidence interval (CI)



Numerical **interval** in which lies the (unknown) **true value** of some parameter with a certain probability, called the **confidence level**

Example



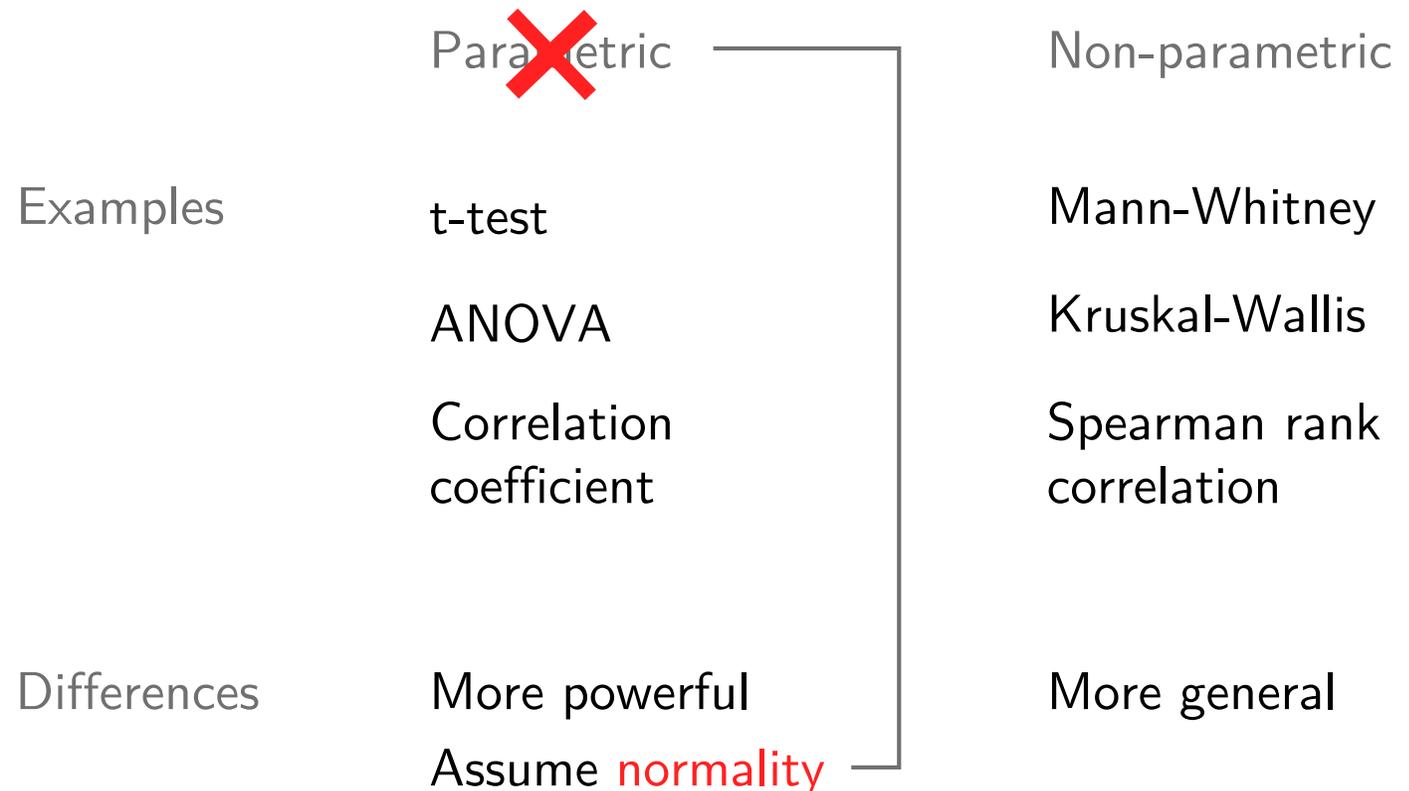
$[a, b]$ is a 95% CI for the median of x

which means that

The probability that the true median of x is within $[a, b]$ is larger or equal to 95%.

Non-parametric statistical methods

(Predictive) statistics making no assumptions on the nature of the underlying distribution



Non-parametric statistical methods

(Predictive) statistics making no assumptions on the nature of the underlying distribution

Examples

~~Parametric~~

t-test

ANOVA

Correlation coefficient

Differences

More powerful

Assume **normality**

Non-parametric

Mann-Whitney

Kruskal-Wallis

Spearman rank correlation

More general

Statistics take-away for replicability in networking

1. **Replicability** requires **predictive** statistics
2. Predictions require **confidence intervals**
3. **Non-parametric** statistics should be used;
do not assume normality!

Any questions?

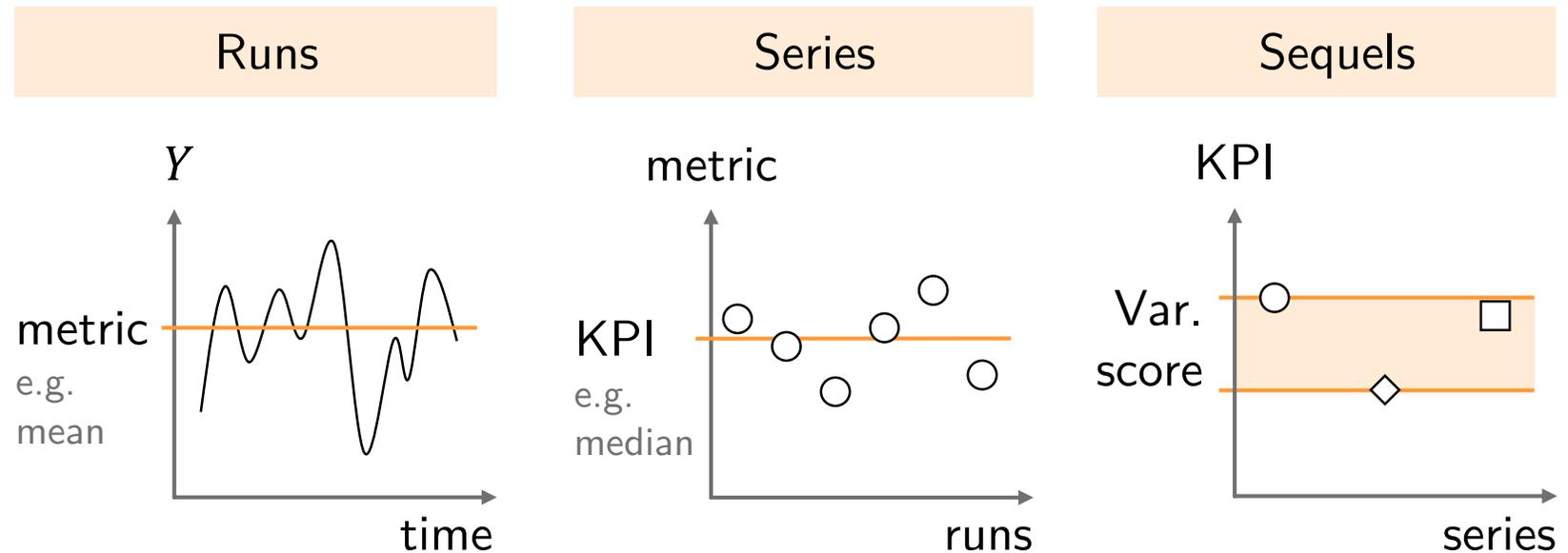
Up next



Getting started on
TriScale per se

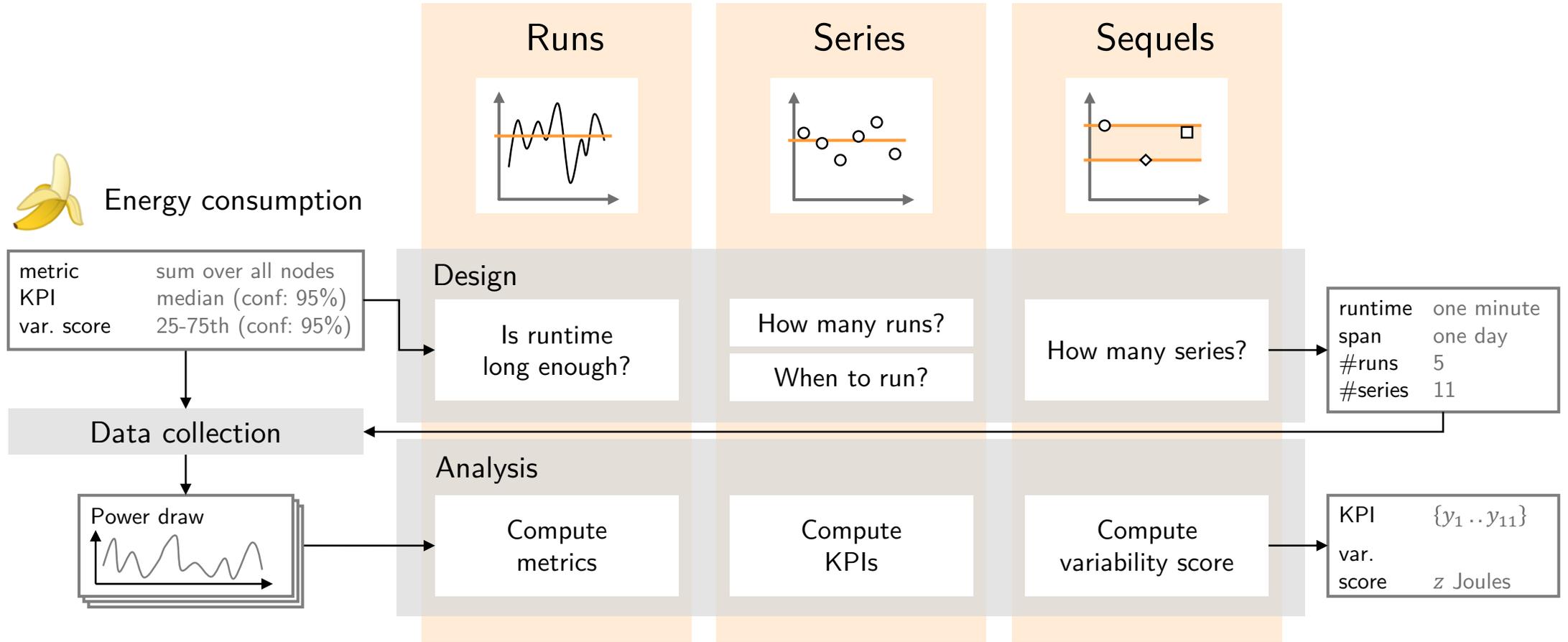
 **TriScale** is a framework helping to design and analyze networking experiments

► Divides the experiment design and data analysis into **three time scales**



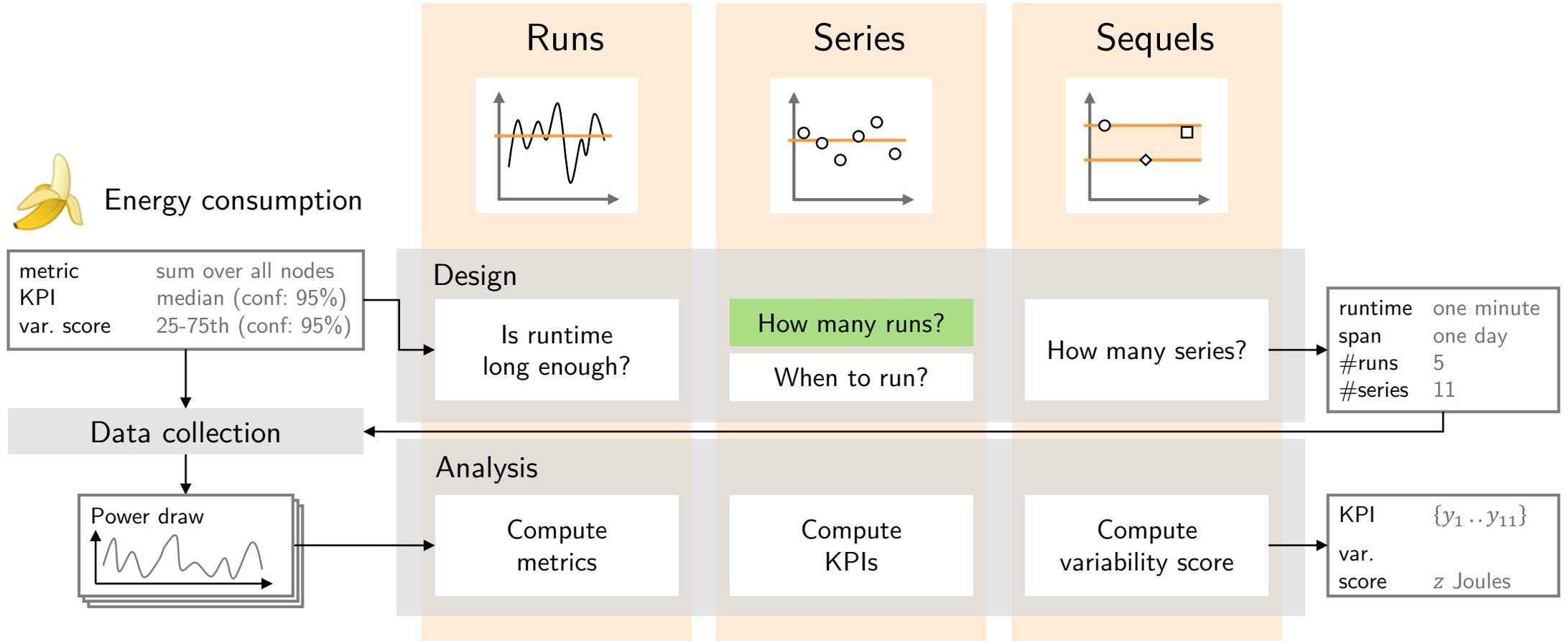


Energy consumption

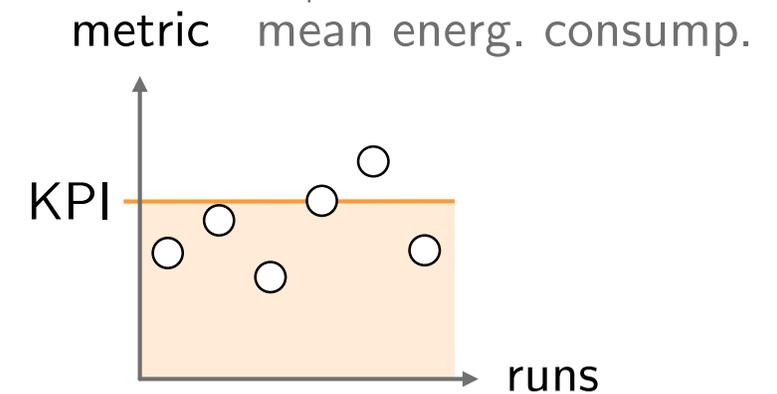




Energy consumption



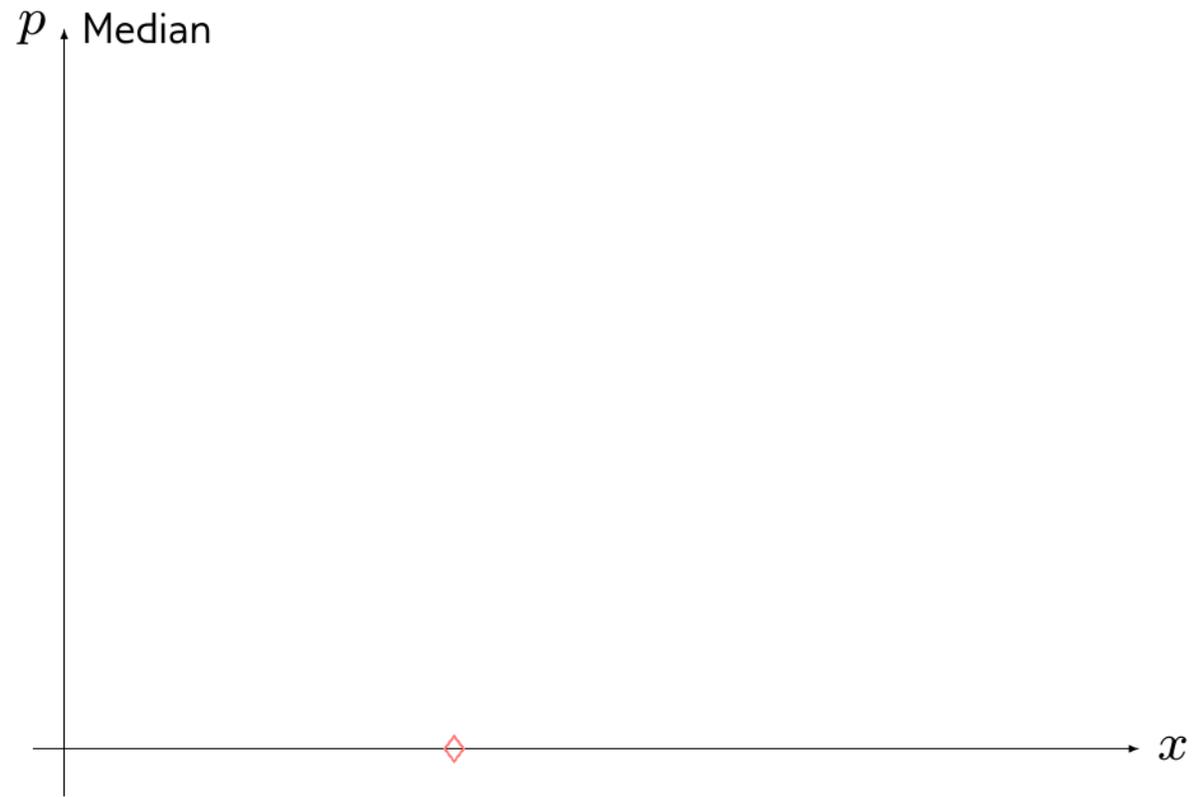
The Thompson's method provides non-parametric CI for distribution percentiles



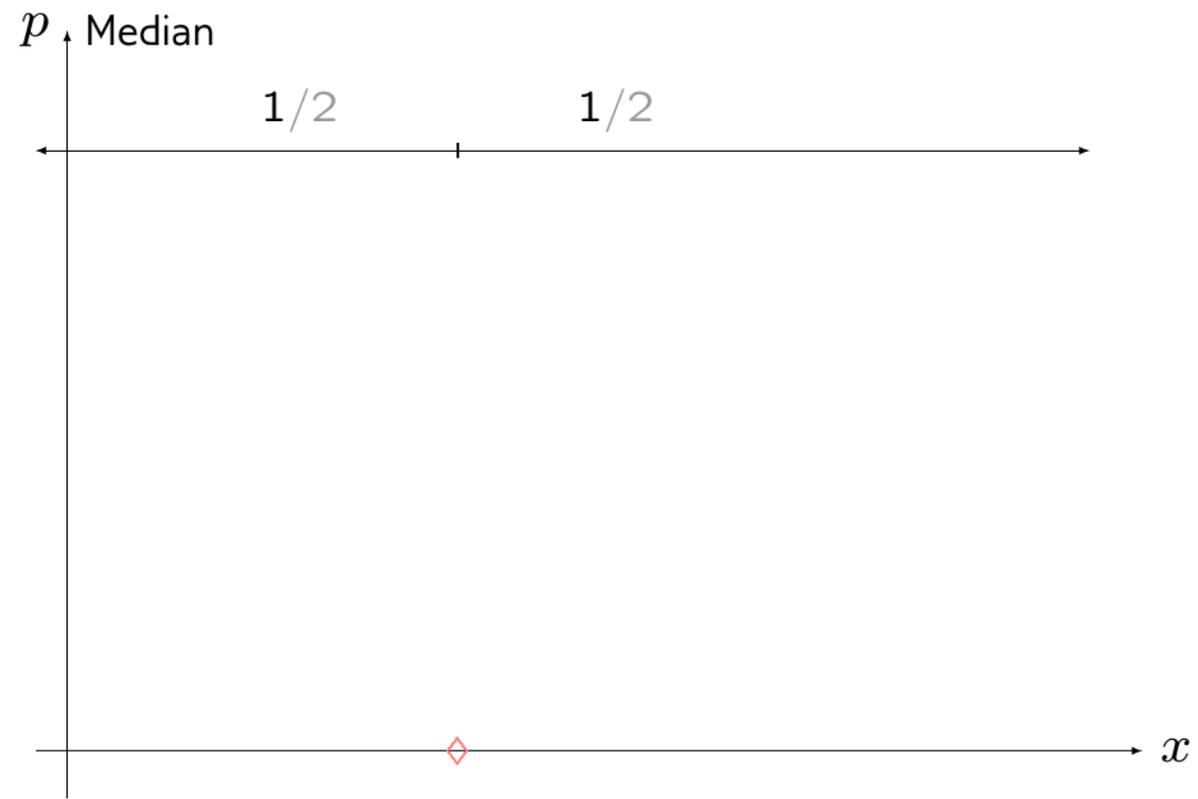
Key Performance Indicators (KPIs) are **percentiles** of the **distribution of metric** values

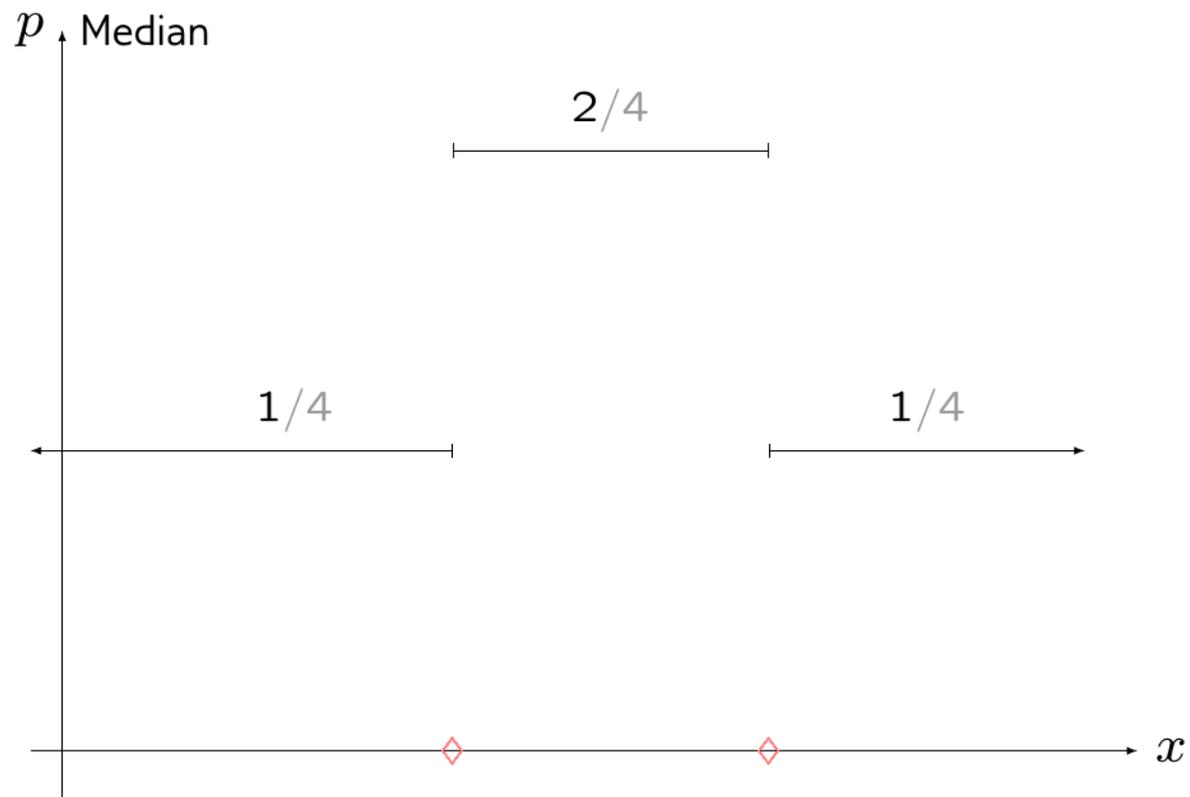
- Compute upper and lower bounds on the true percentile values for a certain confidence level
- The KPIs are defined as one such bounds

95%-CI for the median



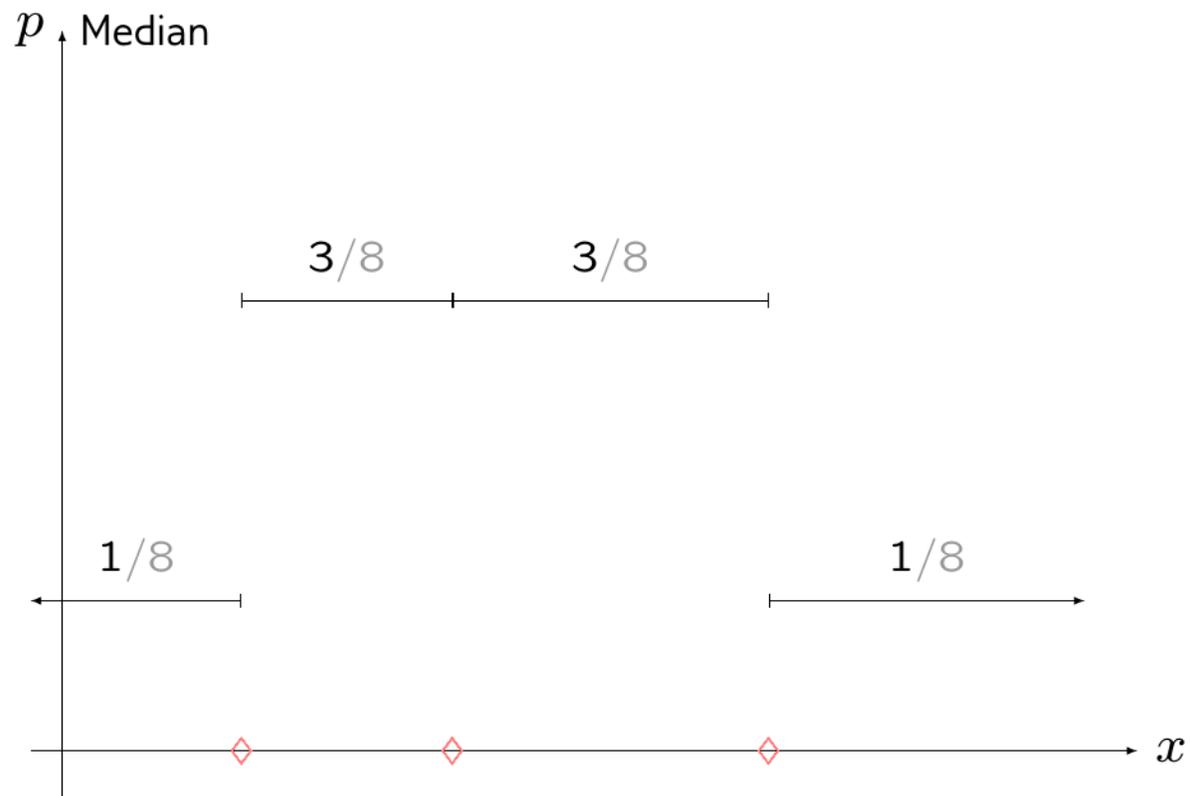
Adapted from Hanspeter Schmid and Alex Huber





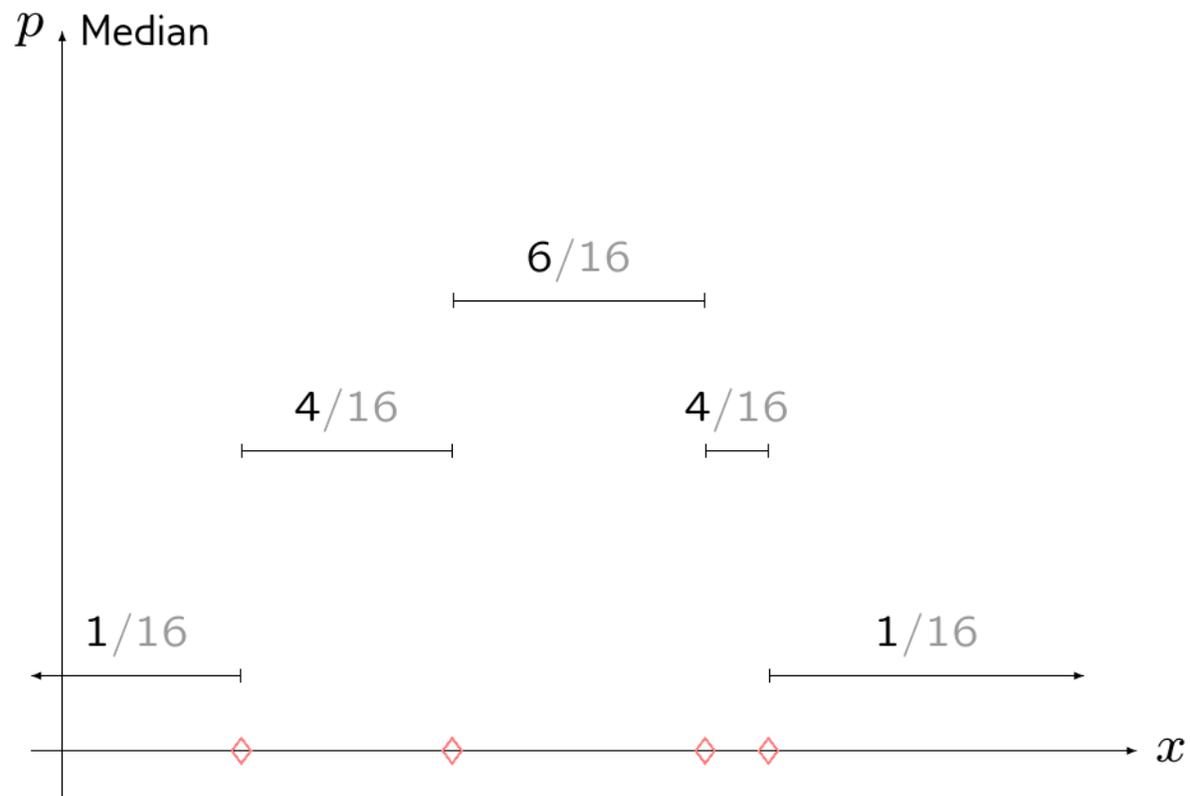
Hypothesis

Samples are **i.i.d.**



Hypothesis

Samples are **i.i.d.**



Hypothesis

Samples are **i.i.d.**

The Thompson's method provides non-parametric CI for distribution percentiles

Probability of any P_p to be between two consecutive samples

$$\mathbf{P} \{x_k \leq P_p \leq x_{k+1}\} = \binom{N}{k} p^k (1-p)^{N-k}$$

Binomial distribution

Allows to derive lower and upper bounds for any percentile

$$\mathbf{P} \{x_m \leq P_p\} = 1 - \sum_{k=0}^{m-1} \binom{N}{k} p^k (1-p)^{N-k}$$

The Thompson's method provides non-parametric CI for distribution percentiles

For any confidence c

For any percentile P_p

$$N \geq \frac{\log(1 - c)}{\log(1 - p)}$$

For any confidence c
For any percentile P_p

$$N \geq \frac{\log(1 - c)}{\log(1 - p)}$$

95% CI
 $c = 0,95$

Minimal number
of runs in a series

Median
 $p = 0,5$

6

25-th
 $p = 0,25$

11

1-th
 $p = 0,01$

299

0.001-th
 $p = 0,00001$

299572

We might want to rethink
the idea of “five-nines” claims...

95%CI on the median

Minimum 6 samples

CI starts excluding most extreme values

CI gets narrower with more samples in general

—

N = 8



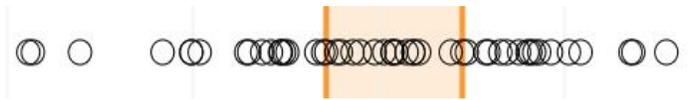
N = 9



N = 10



N = 50



N = 75



N = 100



N = 200



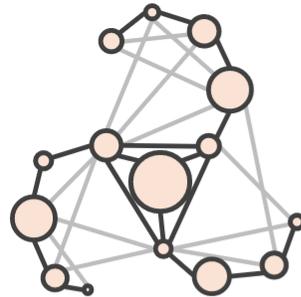
N = 1000



Let's practice!

Go to
tryscale.ethz.ch

TriScale



TriScale

A Framework Supporting Replicable
Performance Evaluations in Networking

[View the Project on GitHub](#)
romain-jacob/tryscale



**A Framework Supporting Replicable
Performance Evaluations in Networking**

[Paper](#) [Code](#) [Tutorial](#) [Discussion](#)

Following a live tutorial session? Here are the links you're looking for

Hands-on

Part 1 [launch](#) [binder](#)

Part 2 [launch](#) [binder](#)

When designing their performance evaluations, networking researchers often encounter questions such as:

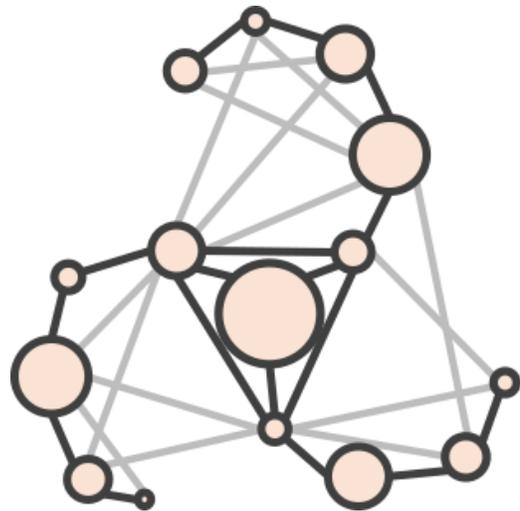
- How long should a run be?
- How many runs to perform?
- How to account for the variability across multiple runs?
- What statistical methods should be used to analyze the data?

Despite the best intentions, researchers often answer these questions differently, thus impairing the replicability of evaluations and the confidence in the results.

Improving the standards of replicability has recently gained traction overall, as well as within the networking community. As an important piece of the puzzle, we developed a systematic methodology that streamlines the design and analysis of performance evaluations, and we have implemented this methodology into a framework called *TriScale*.



Experimental Reproducibility in Networking Research



Resuming at 15:05
Strasbourg time

Enjoy your break!

45' Lecture

10' Hands-on

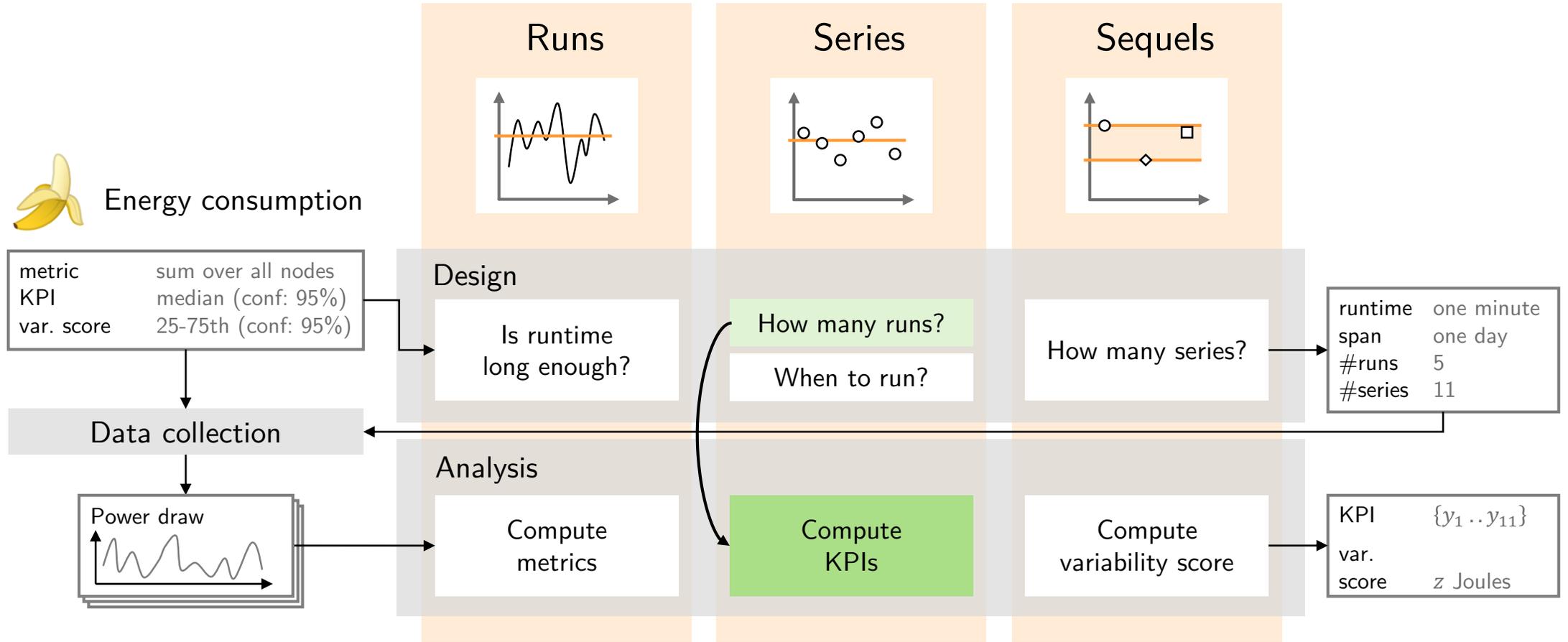
10' Break

20' Lecture

Wrap-up & Discussions



Energy consumption



Assessing replicability
How to be fair and general?

Independence assumption
The elephant in the room

Dealing with seasonality
Patterns in networks

Assessing replicability
How to be fair and general?

Independence assumption
The elephant in the room

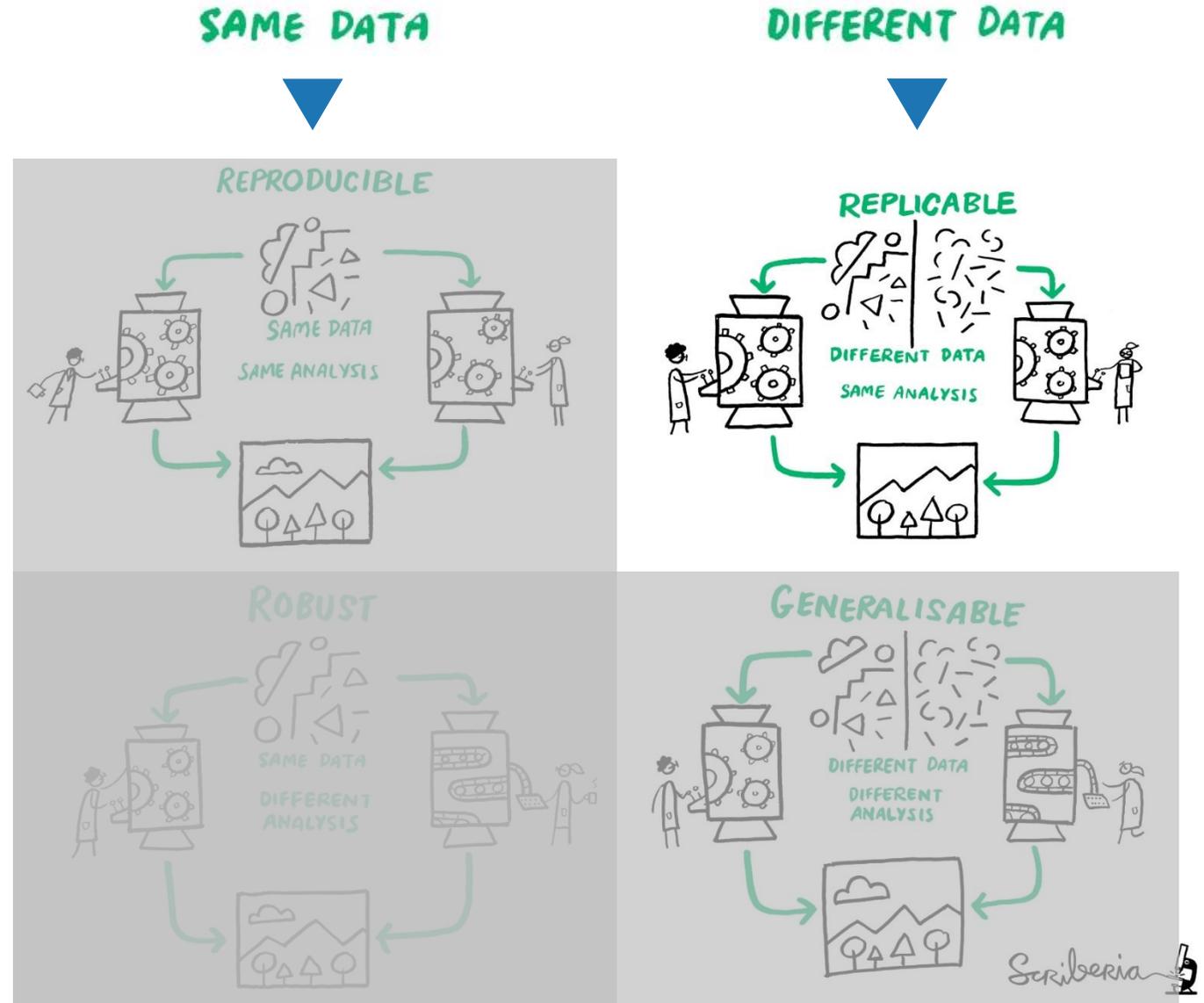
Dealing with seasonality
Patterns in networks

How to assess replicability?

What are “same” results?

SAME ANALYSIS

DIFFERENT ANALYSIS



The Turing Way project illustration by Scriberia.
Zenodo. <http://doi.org/10.5281/zenodo.3332807>

How to assess replicability?

What are
“same” results?

Problems

- Statistical tests are good at checking that things are **different**
- “Similarity” tests all boil down to testing whether differences below **some threshold** ————— Set how?

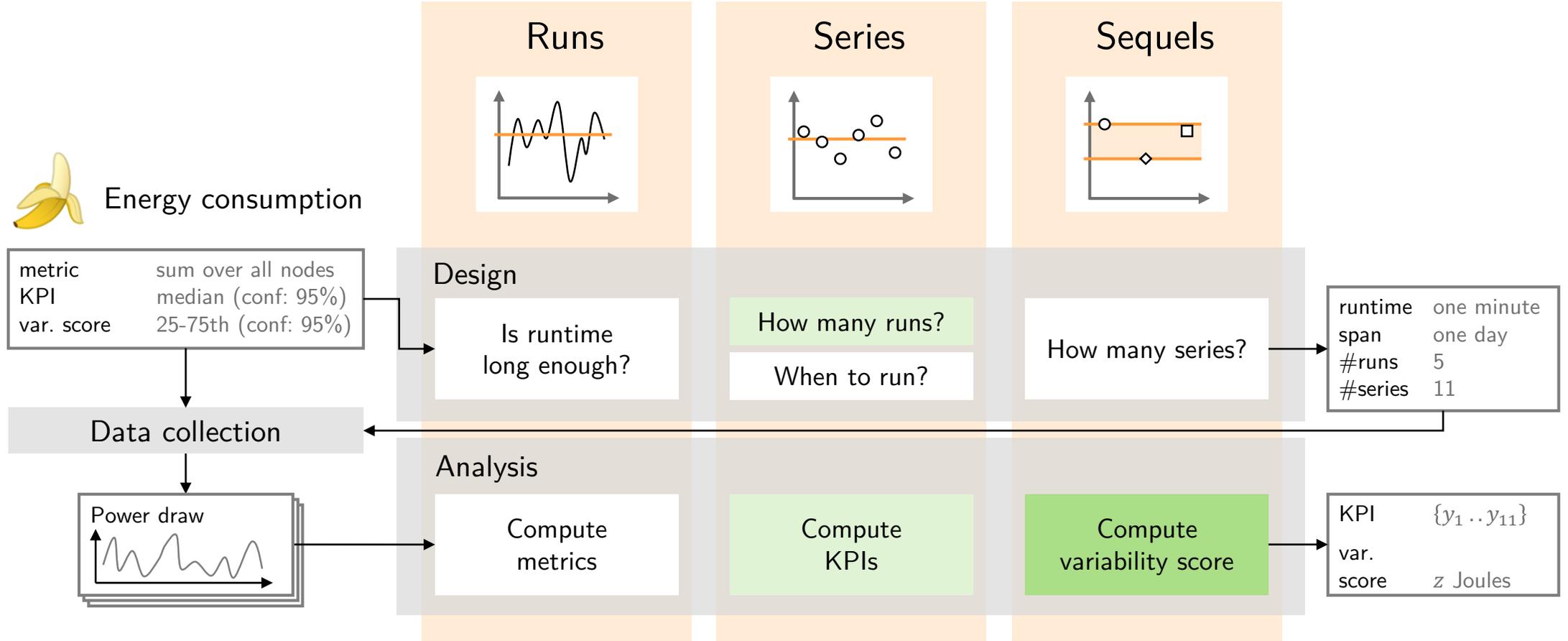
Our approach

Do not assess replicability as a binary criterion

Quantify variability



Energy consumption

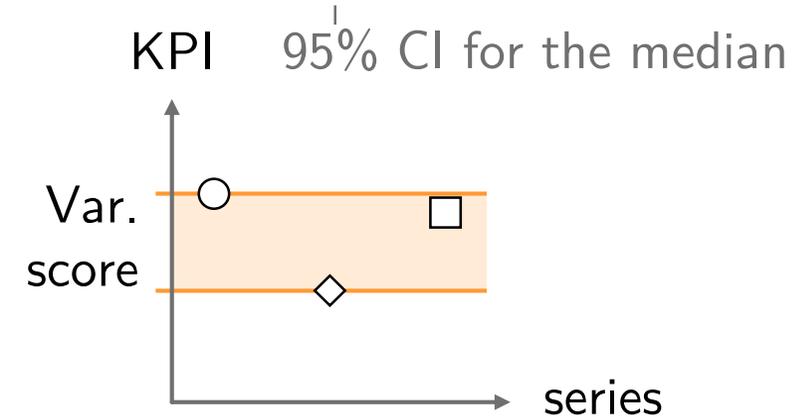


The Thompson's method provides **non-parametric CI** for distribution percentiles



Variability scores are **percentile ranges** of KPI values

- Compute upper and lower bounds on the true percentile values for a certain confidence level
- Variability scores are defined as ranges between these bounds

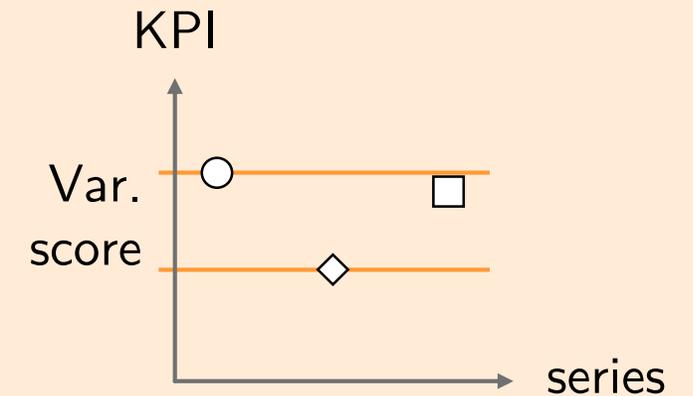


Two-sided 75%-CI for the median

The Thompson's method provides non-parametric CI for distribution percentiles

Variability scores are
percentile ranges
of KPI values

- Compute upper and lower bounds on the true percentile values for a certain confidence level
- Variability scores are defined as ranges between these bounds



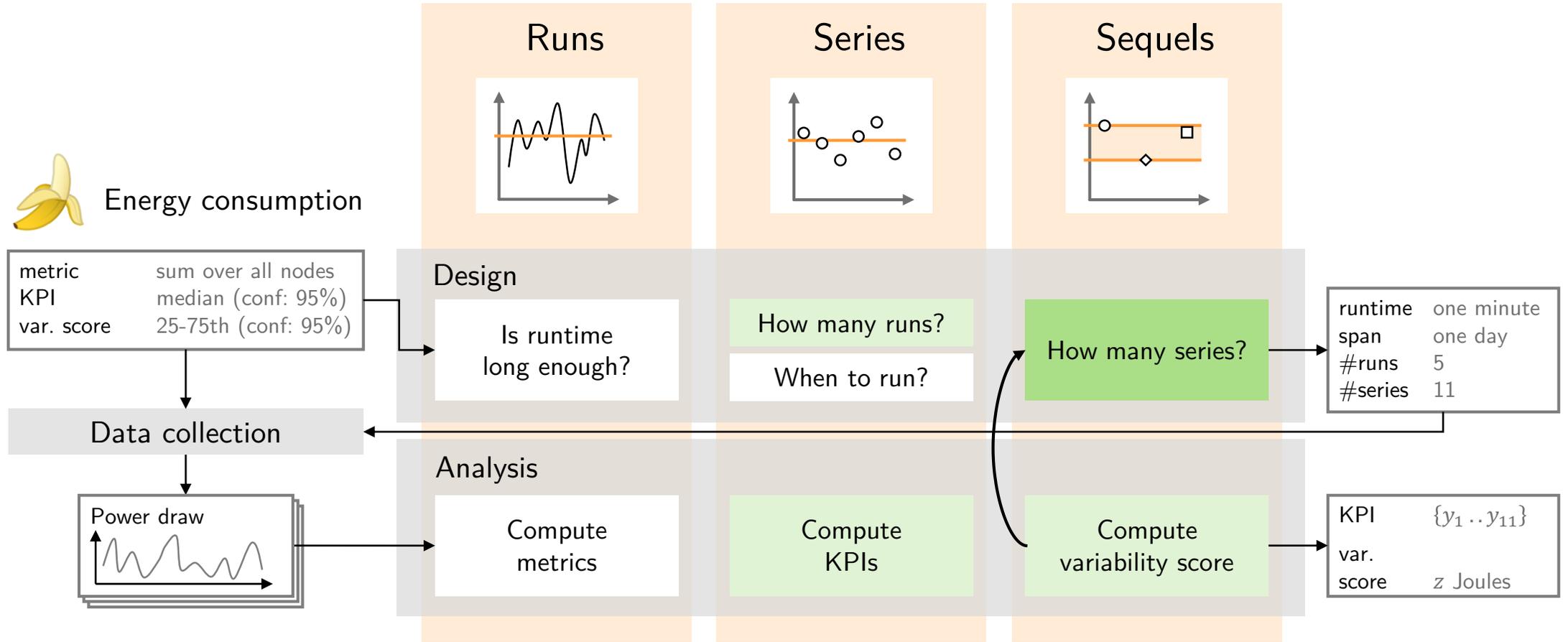
Two-sided 75%-CI
for the median



If a binary cut is desired,
base it on the score



Energy consumption



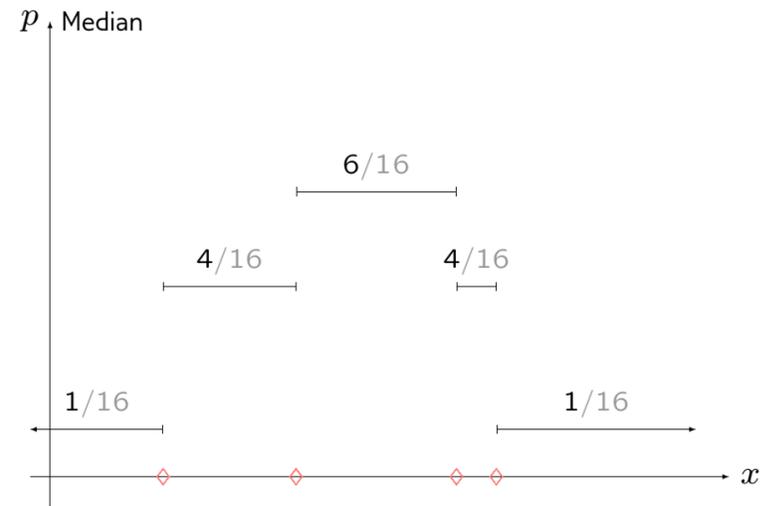
Let's talk about
independence

	Runs	Series	Sequels
Design	Is runtime long enough?	How many runs? When to run?	How many series?
Analysis	Compute metrics	Compute KPIs	Compute variability score

Assessing replicability
How to be fair and general?

| Independence assumption
The elephant in the room

Dealing with seasonality
Patterns in networks



Hypothesis

Samples are **i.i.d.**

I.I.D. is the acronym for
Independent and Identically Distributed

I.I.D. is the acronym for Independent and Identically Distributed

Memoriless

i.e.

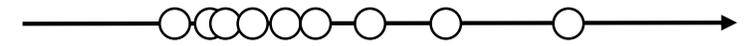
Future samples
are not correlated
to past samples

I.I.D. is the acronym for
Independent and **Identically Distributed**

All samples are drawn from
the same underlying distribution

I.I.D. is the acronym for Independent and **Identically Distributed**

Identically distributed sample



Underlying distribution



I.I.D. is the acronym for Independent and Identically Distributed

~~Identically distributed~~ ^{biased} sample

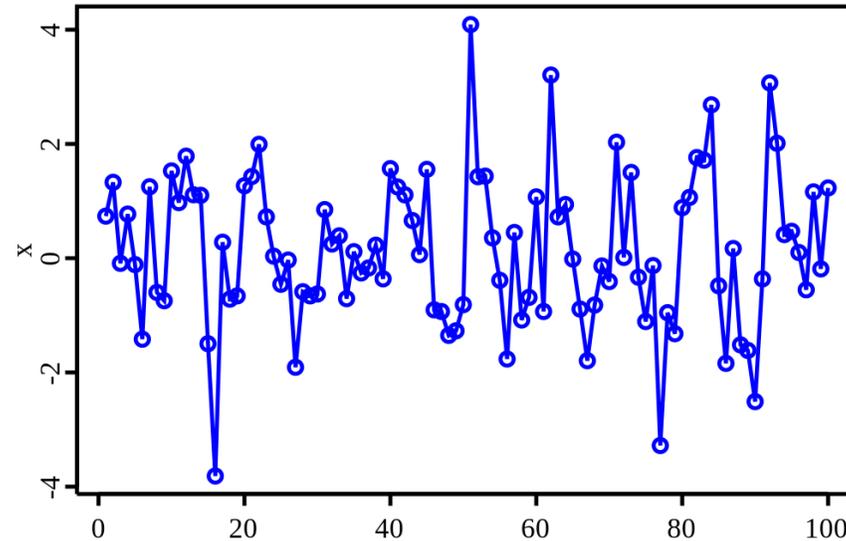


Underlying distribution
^{change}



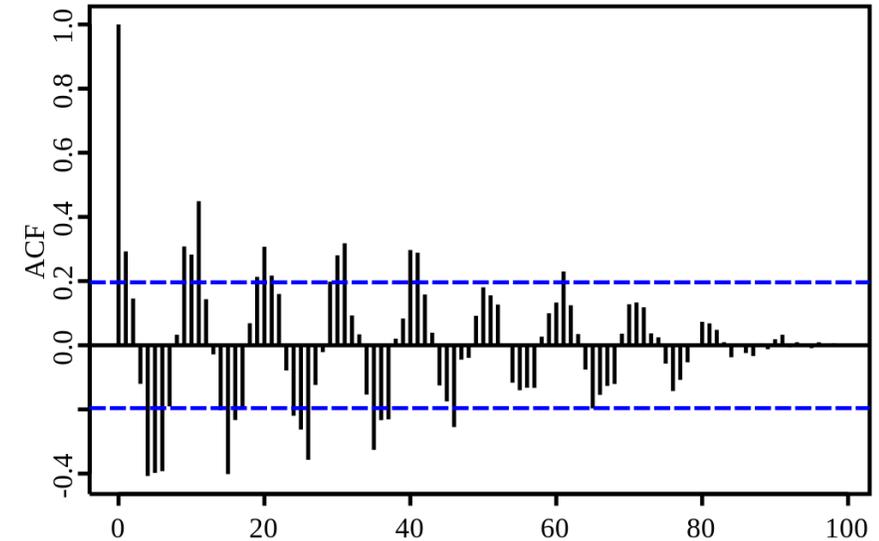
I.I.D. is the acronym for Independent and Identically Distributed

Correlation
i.e., non-independence
can be seen in an
autocorrelation plot



100 random numbers
with a "hidden" sine function

commons.wikimedia.org/wiki/File:Acf.svg



The autocorrelation plot reveals
the hidden structure in the data

In general

We often say “independence”
when we mean “i.i.d.-ness”

What if there is
no independence?

- Samples are biased
- Data do not contain
as much information
as it appears to.
- “Fake” effects

Independence is a property of the experiment design (not of the data!)

We often say
or write

“Data is i.i.d.”

Not mathematically correct statement

▶ We mean that the samples were collected from an i.i.d. experiment — ?

An experiment is i.i.d. if all its factors are selected in an i.i.d. way

Factor?

Any parameters that affect the outcome of an experiment

e.g.,
Time of the day

Factor values must be selected

- in a memoriless fashion
- using the same random procedure

Independent
Identically Distributed

An experiment is i.i.d. if all its factors are selected in an i.i.d. way, but this is often **impossible**

Uncontrollable factors

External interference may be unavoidable

Imperfect randomization

Experiments cannot overlap in time

Hidden factors

What about temperature?

Independence is often impossible to guarantee, but we can test if it **appears** to hold

Empirical i.i.d. test

- No trend
- No correlation structure

Implemented in TriScale

Two caveats

- Imprecise
- No future guarantees

Especially with few samples

Can only detect correlation that was captured in the sample

▶ Things may change...

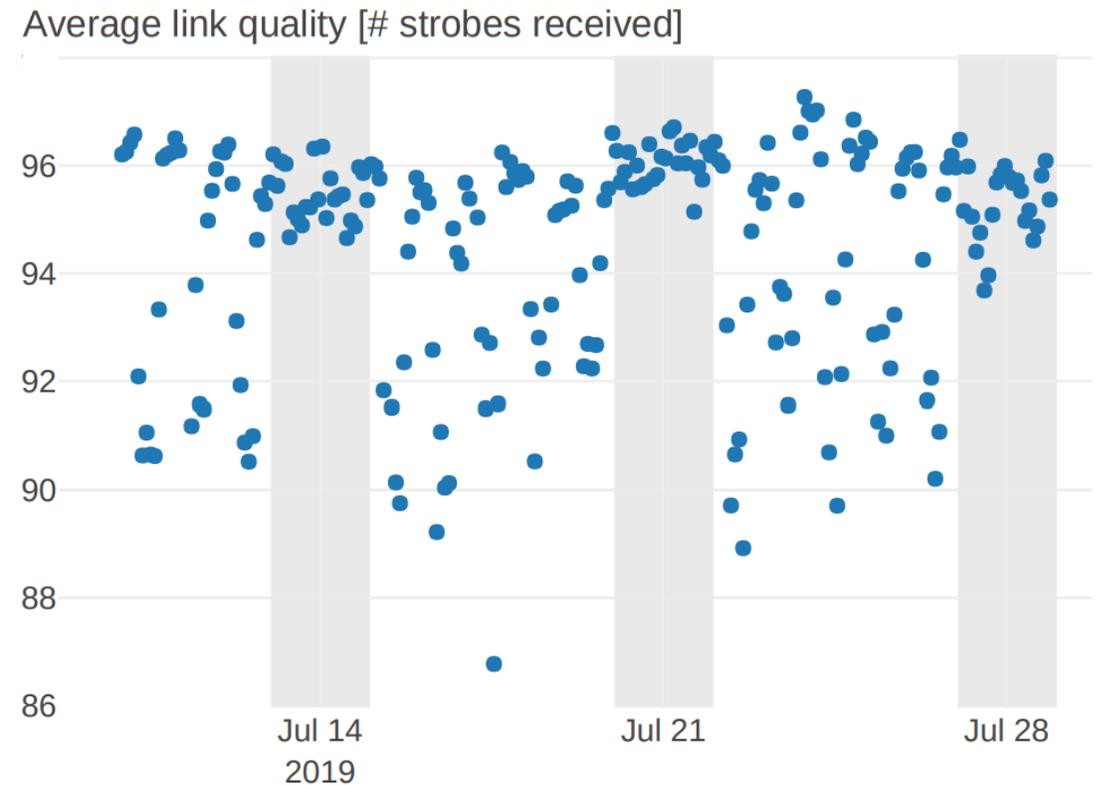
Assessing replicability
How to be fair and general?

Independence assumption
The elephant in the room

| Dealing with seasonality
Patterns in networks

One common danger to beware of is seasonal components

Periodic patterns in the
experimental conditions

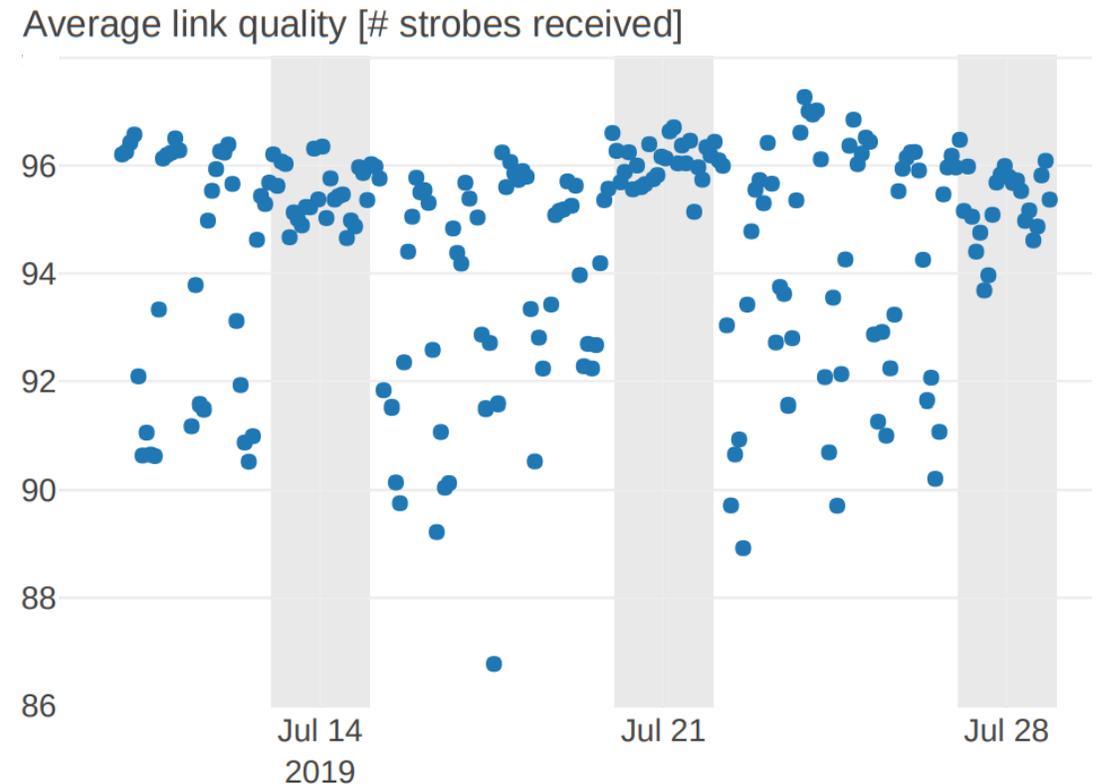


Average link quality on the Flocklab testbed
July 2019

One common danger to beware of is seasonal components

In TriScale

The **time span** of a series of runs should be a multiple of the largest seasonal component

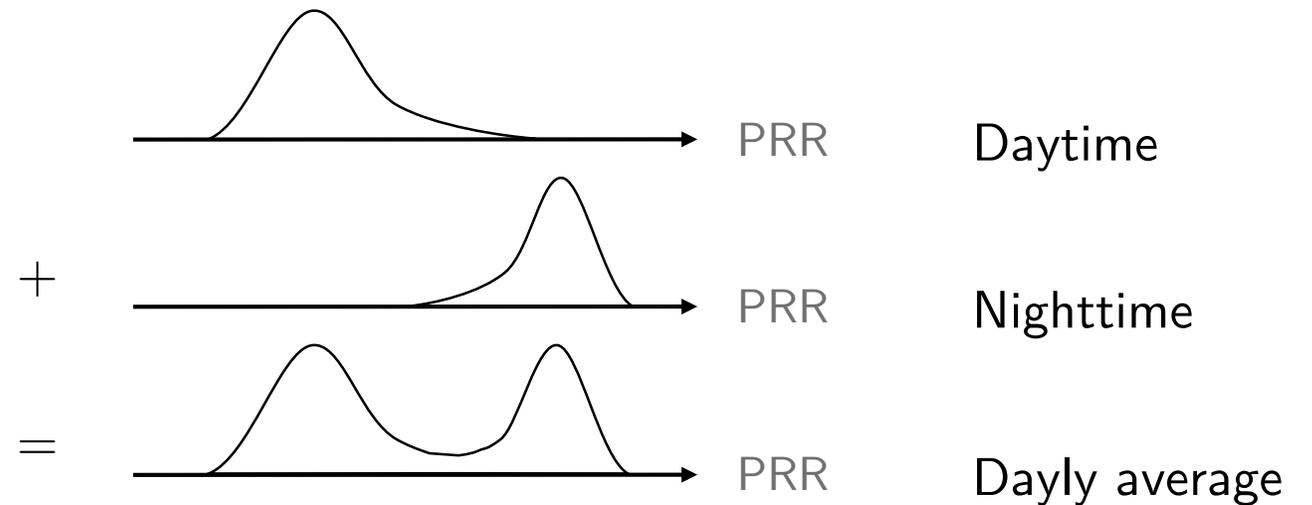


One common danger to beware of is seasonal components

In TriScale

The **time span** of a series of runs should be a multiple of the largest seasonal component

Intuition



▶ Randomly sample this joint distribution
(not truly “identically distributed” experiment)

Identifying seasonal components is a fairly difficult task

Requires

1. Long-term monitoring
of the environment
2. Definition of a metric for “link quality”
which is relevant for the system under test

▶ Hidden factors!

Hard work
but **important!**

We can see that in practice...

Any questions?

Up next



Hands-on session

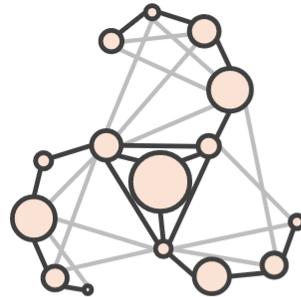
Data analysis

Seasonality

Let's practice!

Go to
triscala.ethz.ch

TriScale



TriScale

A Framework Supporting Replicable
Performance Evaluations in Networking

[View the Project on GitHub](#)
romain-jacob/triscala



**A Framework Supporting Replicable
Performance Evaluations in Networking**

[Paper](#) [Code](#) [Tutorial](#) [Discussion](#)

Following a live tutorial session? Here are the links you're looking for

Hands-on

Part 1 [launch](#) [binder](#)

Part 2 [launch](#) [binder](#)

When designing their performance evaluations, networking researchers often encounter questions such as:

- How long should a run be?
- How many runs to perform?
- How to account for the variability across multiple runs?
- What statistical methods should be used to analyze the data?

Despite the best intentions, researchers often answer these questions differently, thus impairing the replicability of evaluations and the confidence in the results.

Improving the standards of replicability has recently gained traction overall, as well as within the networking community. As an important piece of the puzzle, we developed a systematic methodology that streamlines the design and analysis of performance evaluations, and we have implemented this methodology into a framework called *TriScale*.



Why replicability matters
Case by example

Understanding variability
The three timescales

Know your data
Use the right statistics

Why replicability matters
Case by example

| Understanding variability
The three timescales

Know your data
Use the right statistics

Why replicability matters
Case by example

Understanding variability
The three timescales

 Know your data
Use the right statistics

Why replicability matters
Case by example

Understanding variability
The three timescales

Know your data
Use the right statistics

Assessing replicability
How to be fair and general?

Independence assumption
The elephant in the room

Dealing with seasonality
Patterns in networks

Why replicability matters
Case by example

Understanding variability
The three timescales

Know your data
Use the right statistics

Assessing replicability
How to be fair and general?

Independence assumption
The elephant in the room

Dealing with seasonality
Patterns in networks

Why replicability matters
Case by example

Understanding variability
The three timescales

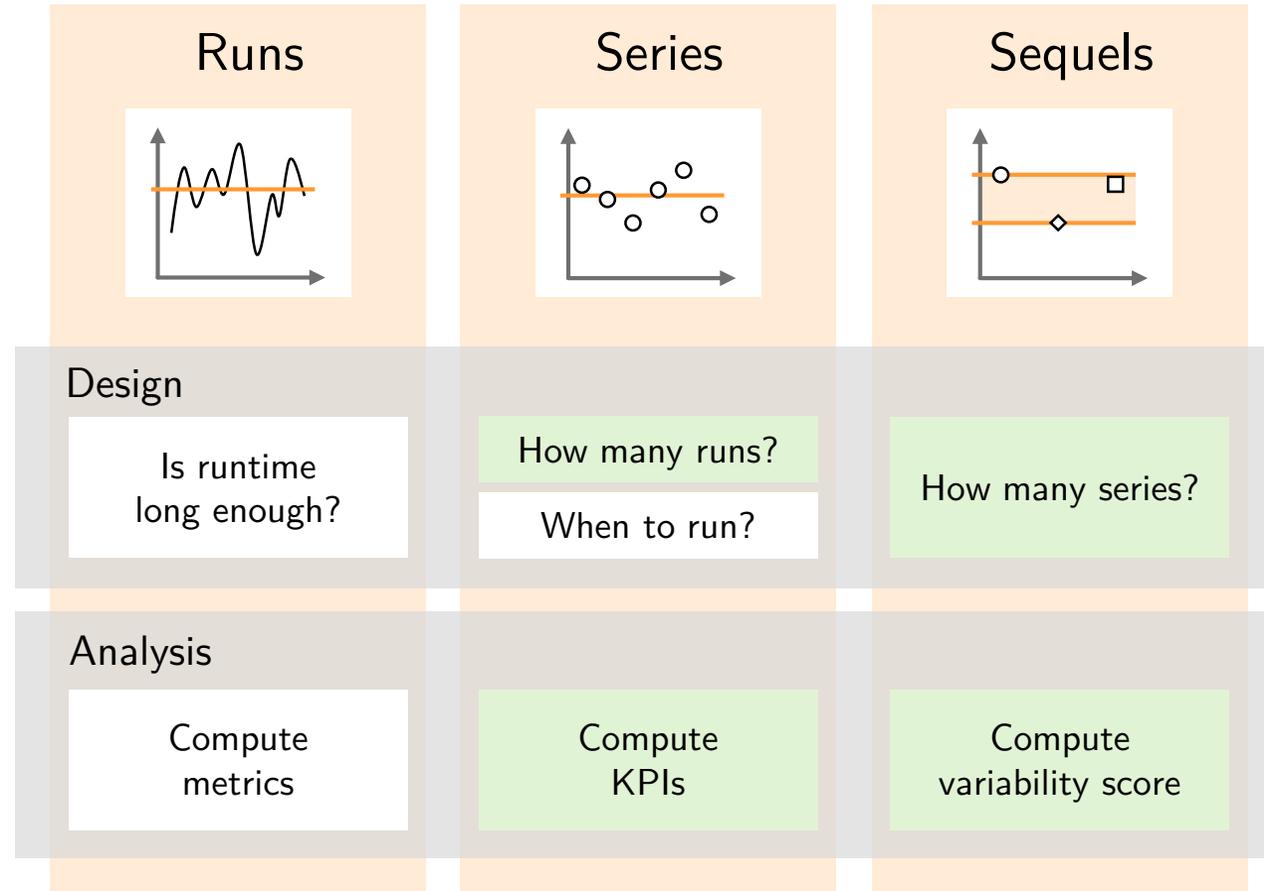
Know your data
Use the right statistics

Assessing replicability
How to be fair and general?

Independence assumption
The elephant in the room

Dealing with seasonality
Patterns in networks

Some other boxes...



In  **TriScale**

- Convergence of runs

For future work

- Comparison of confidence intervals

Interested? Find our more!

10.5281/zenodo.4596442 - 11 Mar 2021

TriScale: A Framework Supporting Replicable Performance Evaluations in Networking

Romain Jacob
ETH Zurich
jacobr@ethz.ch

Marco Zimmerling
TU Dresden
marco.zimmerling@tu-dresden.de

Carlo Alberto Boano
TU Graz
cboano@tugraz.at

Laurent Vanbever
ETH Zurich
lvanbever@ethz.ch

Lothar Thiele
ETH Zurich
thiele@ethz.ch

ABSTRACT

When designing their performance evaluations, networking researchers often encounter questions such as: How long should a run be? How many runs to perform? How to account for the variability across multiple runs? What statistical methods should be used to analyze the data? Despite their best intentions, researchers often answer these questions differently, thus impairing the replicability of their evaluations and the confidence in their results.

To support networking researchers, we propose a *systematic methodology* that streamlines the design and analysis of performance evaluations. Our approach hierarchically partitions the performance evaluation in a sequence of stages building on top of each other, following the principle of separation of concerns. The idea is to first understand, for each stage, the temporal characteristics of variability sources, and then to apply, for each source, rigorous statistical methods to derive performance results with *quantifiable confidence* in spite of the inherent variability. We implement an instance of that methodology in a software framework called *TriScale*. For each performance metric, *TriScale* computes a variability score that estimates, with a given confidence, how similar the results would be if the evaluations were replicated, in other words, *TriScale* quantifies the replicability of evaluations. We apply *TriScale* to four different use cases (congestion control, wireless embedded systems, failure detection, video streaming), demonstrating that *TriScale* helps to generalize and strengthen previously published results.

Improving the standards of replicability in networking is a crucial and complex challenge; with *TriScale*, we make an important contribution to this endeavor by providing for the first time a rationale and statistically sound experimental methodology.

1 INTRODUCTION

The ability to replicate an experimental result is essential for making a scientifically sound claim. In networking research, replicability¹ is a well-recognized problem due to the *inherent variability of the experimental conditions*: the uncontrollable dynamics of real networks [17, 51] and the time-varying performance of hardware and software components [11, 49, 73] cause major changes in the experimental conditions, making it difficult to replicate results and quantitatively compare different solutions [4]. In addition, *differences in*

¹Different terminology is used to refer to different aspects of replicability research [8, 9]. In this paper, we refer to replicability as the ability of different researchers to follow the steps described in published work, collect new data using the same tools, and eventually obtain the same results, within the margins of experimental error. This is usually called replicability [1] but sometimes referred to as reproducibility.

Submitted to ACM SIGCOMM Computer Communication Review

the methodology used to design an experiment, process the measurements, and reason about the outcomes impair the ability to replicate results and assess the validity of claims reported by other researchers. Without replicability, any performance evaluation is questionable, at best.

To be replicable, performance evaluations must account for the inherent variability of networking experiments on different time scales. Therefore, experiments are typically repeated to increase the confidence in the conclusions. To facilitate this, the networking community has put great efforts into developing testbeds [35] and data collection frameworks [40]. However, we lack a *systematic methodology* that specifies how to design and analyze performance evaluations. The literature is currently limited to generic guidelines [5, 52, 63] and recommendations [8, 43, 97] which leave open critical questions *before* an experiment (How many runs? How long should a run be?) and *after* (How to process the data and analyze the results?). Without a systematic methodology, networking researchers often design and analyze similar experiments in different ways, making them hardly comparable [12]. Yet, strong claims are being made (“our system improves latency by 35%” while confidence is often discussed only in qualitative ways (“with high confidence”), if at all [73, 82]. Furthermore, it is currently unclear how to assess whether an experiment is indeed replicable. We argue that a systematic methodology is needed to help resolve this situation.

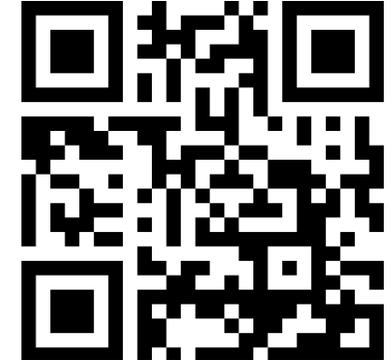
We identify four key challenges that must be addressed in the design of such a methodology.

Rationality The methodology must rationalize the experiment design by linking the design questions (e.g. How many runs?) with the desired confidence in the results.

Robustness The methodology must be robust against the variability of the experimental conditions. The data analysis must use statistics that are compatible with the nature of networking data and be able to quantify the expected performance variation shall the evaluation be replicated.

Generality The methodology must be applicable to a wide range of performance metrics, evaluation scenarios (emulator, testbed, in the wild), and network types (wired, wireless).

Conciseness The methodology must describe the experimental design and the data analysis in a concise and unambiguous way to foster replicability while minimizing the use of highly treasured space in scientific papers.



tiny.cc/triscale



triscale.ethz.ch

zenodo

DOI 10.5281/zenodo.3464273

Getting the TriScale work published has been... complicated.

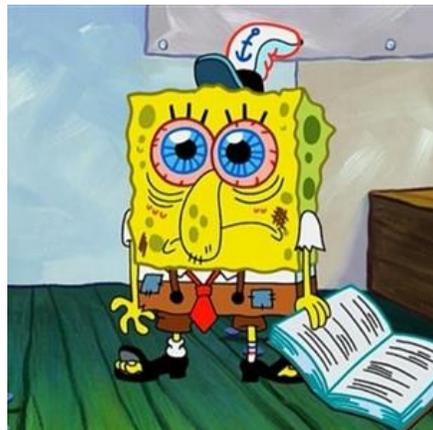
Rejected at

NSDI'20

SIGCOMM'20

SIGMETRICS'21

CCR'21



while receiving comments like

- Solid work with great tooling.
- Our community clearly has a problem with reproducibility and this paper presents very promising solutions.
- Every PhD student should read this paper.

... wait what?

Getting the TriScale work published has been... complicated.

Rejected at

NSDI'20

SIGCOMM'20

SIGMETRICS'21

CCR'21

Accepted at

JSys'21

while receiving comments like

- Solid work with great tooling.
- Our community clearly has a problem with reproducibility and this paper presents very promising solutions.
- Every PhD student should read this paper.



Journal of Systems Research

Diamond Open Access

Free to read. Free to publish.

No page limits

No paper caps

Different paper types

- Solution Usual ones
- Problem Position/white papers
- SoK Survey++
- Tool Typically hard to publish

Student board

Transparent reviewing

- Review summary
- Reviewers named
- Public reviews (anonym)

Artifact Evaluation

Independent **but compulsory**



Journal of Systems Research

13 different areas

- Networking
- Configuration Management for Systems
- Computer Architecture
- Real-time and Cyber-physical Systems
- Streaming Systems
- Systems for ML and ML for systems
- Distributed Consensus
- Data Science and Reproducibility
- Serverless Systems
- System Security
- Active Storage
- Wireless Embedded Systems
- Decentralized Systems



Journal of Systems Research

 jsys.org

13 different areas

- Networking

Area Chairs

Francis Yan

Sangeetha Abdu Jyothi

Area Board

Amreesh Phokeer

Ang Chen

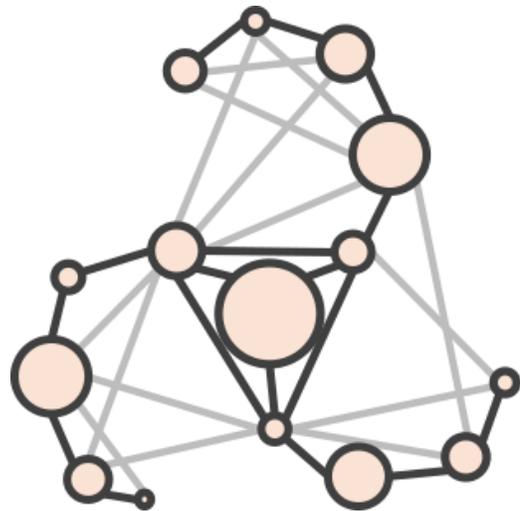
Arpit Gupta

Colin Perkins

Daehyeok Kim

Srinivas Narayana

Experimental Reproducibility in Networking Research



Please fill out
our short survey!



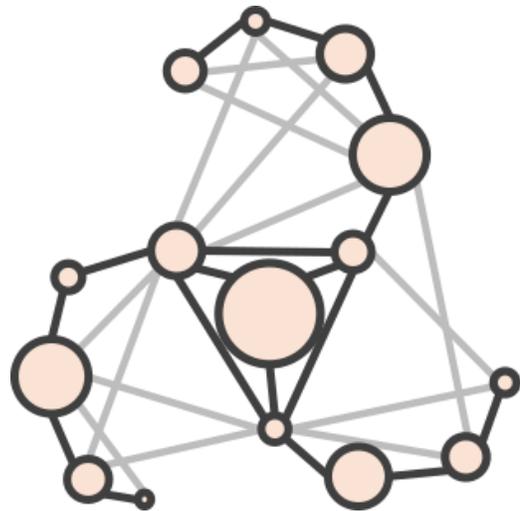
triscala.ethz.ch

feedback tutorial

Experimental Reproducibility in Networking Research

 triscala.ethz.ch

[feedback](#) [tutorial](#)



Romain Jacob
ETH Zurich

Collaboration with
Marco Zimmerling Laurent Vanbever
Carlo Alberto Boano Lothar Thiele



@RJacobPartner