



# Context Based A/B Test Validation

WWWDevDay2016 | 2016-04-11

Michael Nolting & Jan Eike von Seggern | SEVENVAL TECHNOLOGIES GMBH

[michael.nolting@sevenval.com](mailto:michael.nolting@sevenval.com)

[eike.seggern@sevenval.com](mailto:eike.seggern@sevenval.com)



# SEVENVAL TECHNOLOGIES

FRONTEND-EXPERTS SINCE 1999

17 YEARS OF EXPERIENCE

FRONTEND MONITORING AND OPTIMIZATION

150 EMPLOYEES

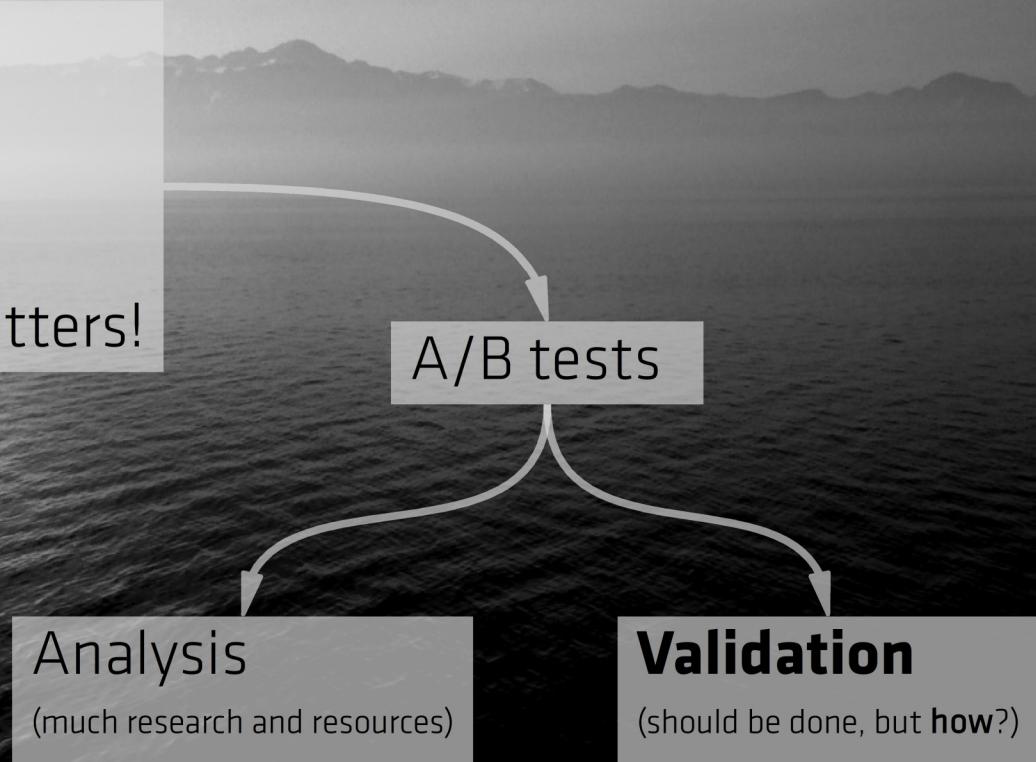
3 LOCATIONS IN EUROPE

MORE THAN 300 SATISFIED CUSTOMERS

# Optimizing Web Applications



- Data driven
- Efficient
- Reliable
- **Frontend quality matters!**





# WHY VALIDATE?

Assert that interpretation of analysis is correct.

## Example:

- Backend performance improvement
- Control (A) performs better than Treatment (B)

## Why?

- No improvement
- Assignment of backend systems uneven
- Assignment of user systems uneven



# CHALLENGES

for validating A/B tests





# CHALLENGES

for validating A/B tests

Backend systems  
(under control)



# CHALLENGES

for validating A/B tests

Frontend  
systems:

Browsers

Devices

Mobility

Backend systems  
(under control)



# CHALLENGES

for validating A/B tests

Frontend  
systems:

Browsers

Devices

Mobility

Frontend matters  
(but it's complicated)

Backend systems  
(under control)



# CHALLENGES

for validating A/B tests

Frontend  
systems:

Browsers

Devices

Mobility

Frontend matters  
(but it's complicated)

~~Backend systems  
(under control)~~



# VALIDATE THE FRONTEND

What's the frontend context of your A/B test?

Context = Everything UX depends on:

- Load time
- Event rates, e.g.
  - Page view
  - Login
  - Error
  - ...



# VALIDATE THE FRONTEND

What's the frontend context of your A/B test?

Context = Everything UX depends on:

- Load time
- Event rates, e.g.
  - Page view
  - Login
  - Error
  - ...

Metrics



# VALIDATE THE FRONTEND

What's the frontend context of your A/B test?

Context = Everything UX depends on:

- Load time
- Event rates, e.g.
  - Page view
  - Login
  - Error
  - ...

**Metrics**



**Record** (via JS)

&

**Evaluate**

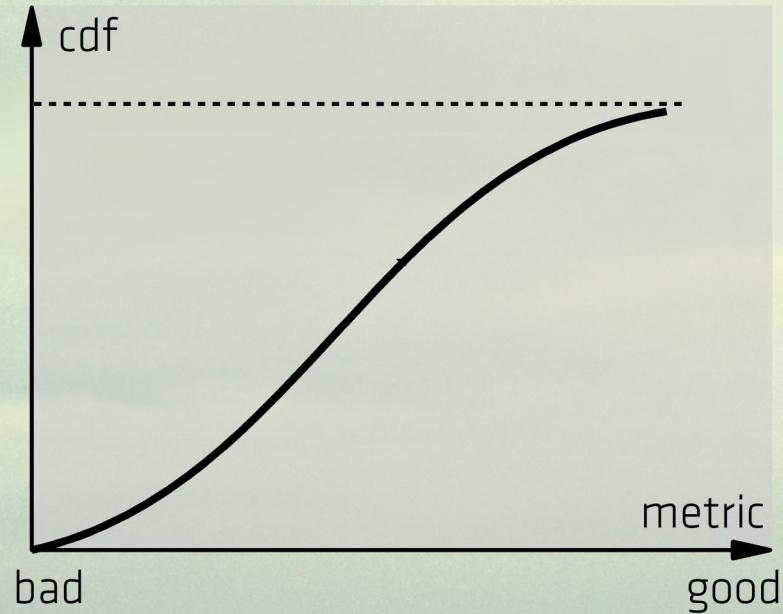
- Efficient (use pre-test sample for comparison)
- Compare A & B with pre-test sample



# EVALUATING METRICS

Is your A/B test valid?

1. Define **target space**  
bootstrap from pre-test sample



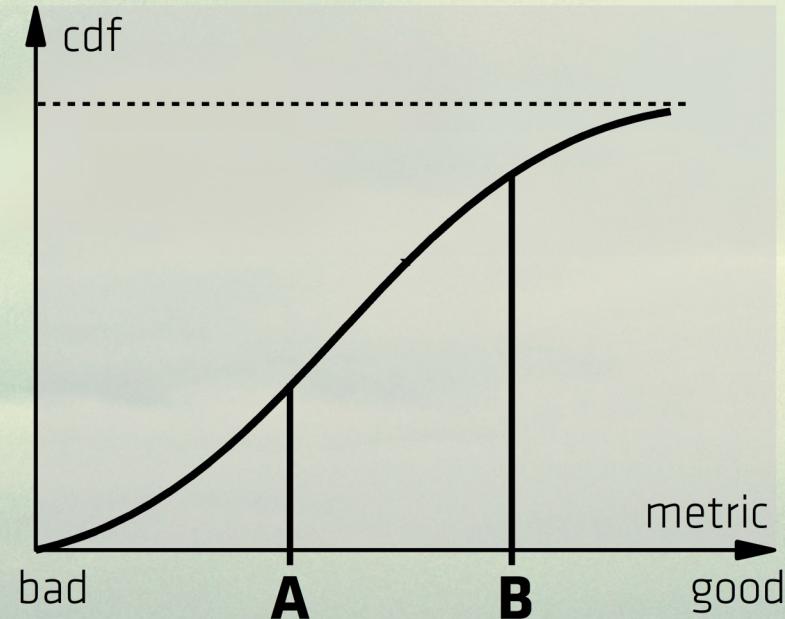


# EVALUATING METRICS

Is your A/B test valid?

1. Define **target space**  
bootstrap from pre-test sample

2. Calculate metrics of A & B  
Locate in target space  $\Rightarrow$  p-values  
(prob. to find worse data)

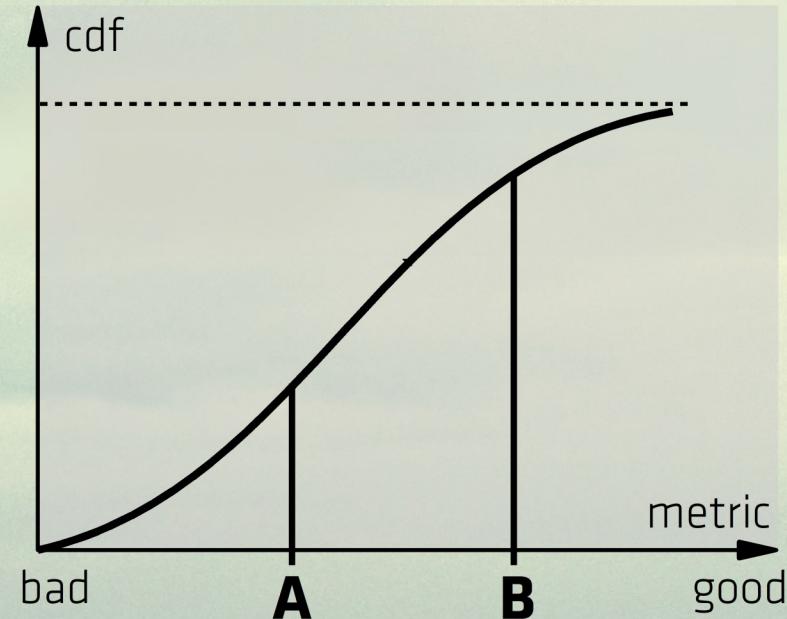




# EVALUATING METRICS

Is your A/B test valid?

1. Define **target space**  
bootstrap from pre-test sample
2. Calculate metrics of A & B  
Locate in target space  $\Rightarrow$  p-values  
(prob. to find worse data)
3. Check that A & B do not deviate significantly from each other and pre-test sample  
(use min p-values from A and B)





# EXAMPLE

Convert 160 pi/h web app to SPA

- Intention:

Decrease load time

- Conditions:

**A**: normal

**B**: SPA

- Metrics:

- Load time
- Page views
- JS Errors



# EXAMPLE

Convert 160 pi/h web app to SPA

- Intention:  
Decrease load time
- Conditions:  
**A**: normal  
**B**: SPA
- Metrics:
  - Load time
  - Page views
  - JS Errors

## Result:

**B** worse than **A**!

# EXAMPLE

Convert 160 pi/h web app to SPA

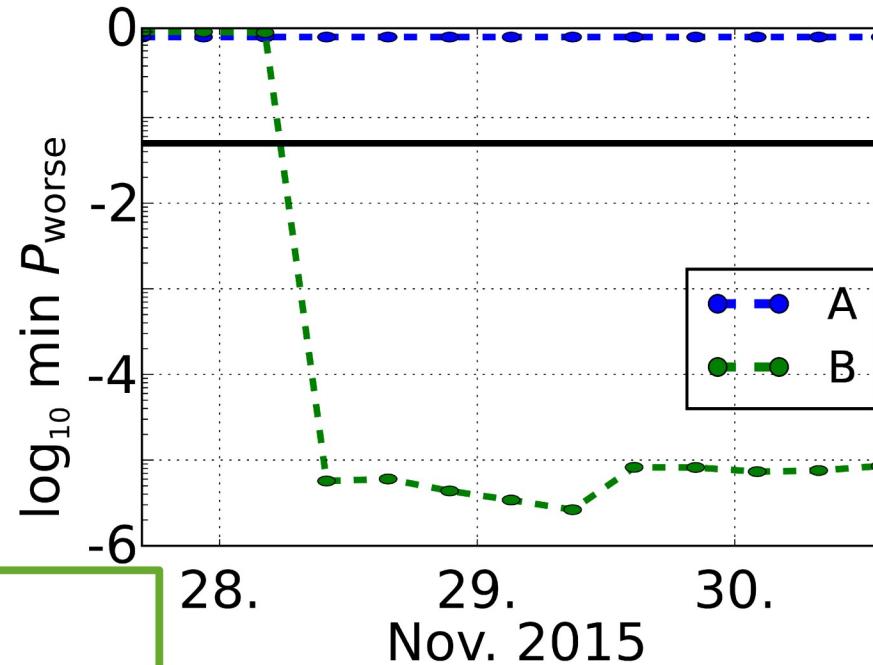
- Intention:  
Decrease load time
- Conditions:  
**A**: normal  
**B**: SPA
- Metrics:
  - Load time
  - Page views
  - JS Errors

## Result:

**B** worse than **A**!

## Why?

New JS error introduced.





# CONCLUSIONS

- A/B tests for data driven optimization  
but:
- **Validation** is important
- **Frontend** context matters