COMPUTER SCIENCE
VIRGINIA TECH

# Assessing Incremental Testing Practices and Their Impact on Project Outcomes

Ayaan M. Kazerouni, Clifford A. Shaffer, Stephen H. Edwards, Francisco Servant

Department of Computer Science, Virginia Tech

SIGCSE 2019, Minneapolis, MN
Thursday, February 28, 2018

VIRGINIA TECH

# Study Context

- Third year (post-CS2) Data Structures & Algorithms course
- 157 students
- 4 assignments
  - *Median 1.4 kLOC*
  - *3-4 weeks long*
- 415 implementations (unbalanced)

VIRGINIA TECH

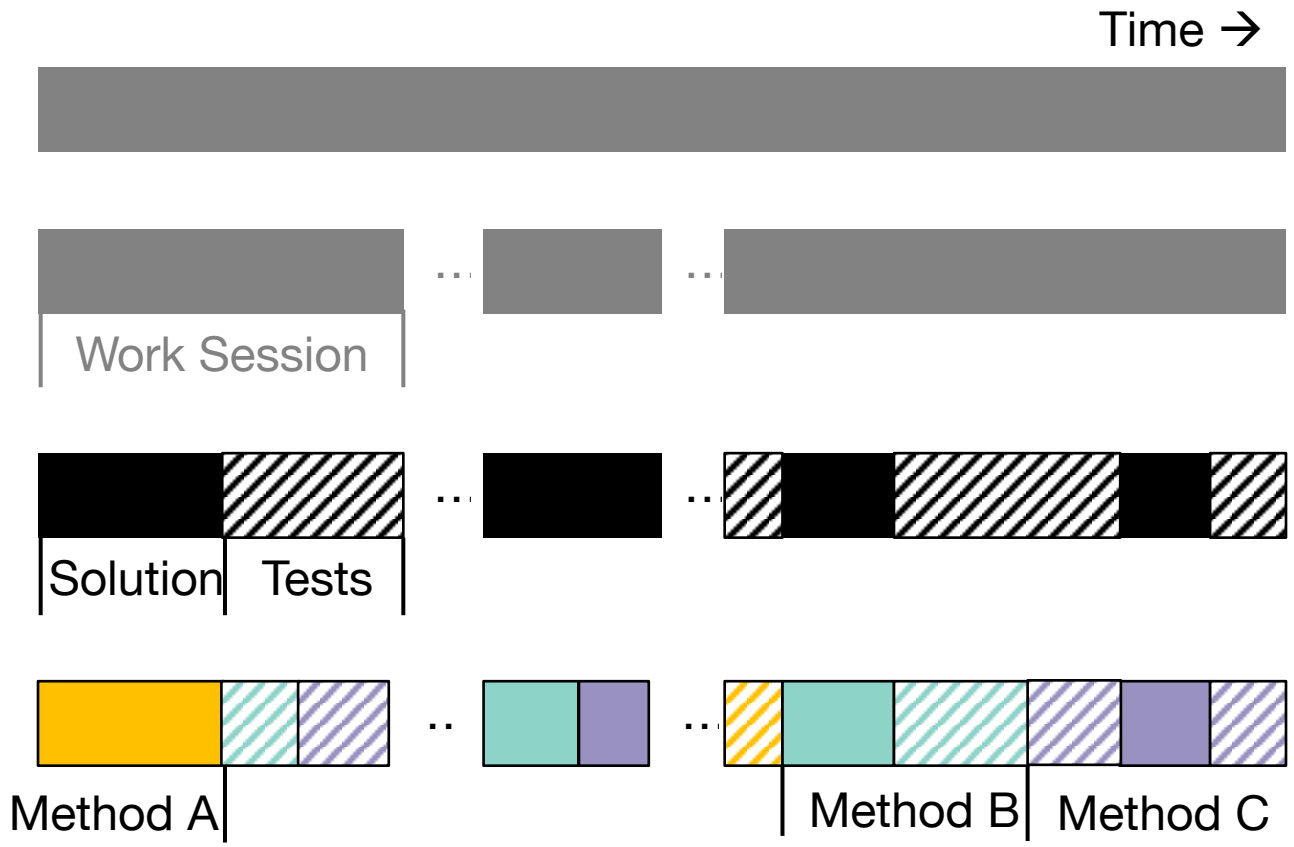
# Contributions

- Family of metrics to assess incremental testing

- Empirical study
  1. *How does the **balance** of testing effort relate to project outcomes?*
  2. *How does the **sequence** of testing effort relate to project outcomes?*

# Better Feedback on Process

## Programming effort




## Feedback

Correctness: 100%
Code coverage: 89%

Procrastination: 75% [1]

Balance of testing

Thoroughness of testing

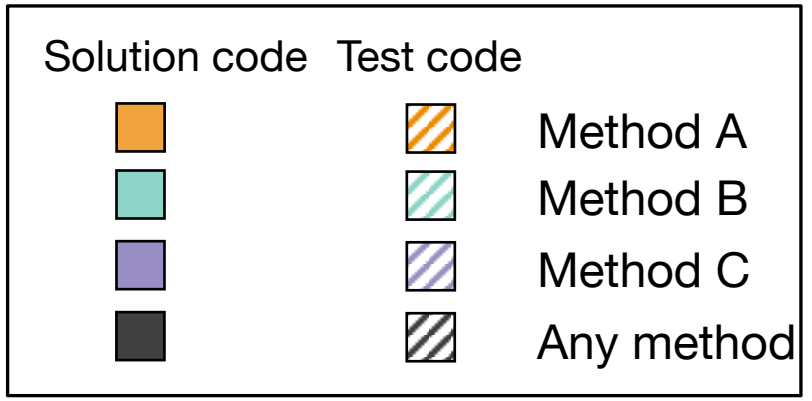
[1] *Quantifying Incremental Development Practices and Their Relationship to Procrastination*. Ayaan M. Kazerouni, Stephen H. Edwards, and Clifford A. Shaffer

VIRGINIA TECH

# Assessing Incremental Testing

# Proposed Metrics of Testing Effort

Synthetic example: sequence of developer activity



# Proposed Metrics of Testing Effort

Synthetic example: sequence of developer activity

Project-wide Overall Testing Effort

$$\frac{T}{S+T}$$

# Proposed Metrics of Testing Effort

Synthetic example: sequence of developer activity

Project-wide Overall Testing Effort

$$\frac{T}{S+T}$$

Project-wide per-Session Testing Effort

$$median\left\{\frac{T_s}{S_s+T_s}\right\}$$

# Proposed Metrics of Testing Effort

Synthetic example: sequence of developer activity

Solution code Test code

Method A

Method B

Method C

Any method

Project-wide Overall Testing Effort

$$\frac{T}{S+T}$$

Project-wide per-Session Testing Effort

$$median\left\{\frac{T_s}{S_s+T_s}\right\}$$

Method-specific Overall Testing Effort

$$median\left\{\frac{T_m}{S_m+T_m}\right\}$$

# Proposed Metrics of Testing Effort

Synthetic example: sequence of developer activity

| Solution code | Test code | |
|---|---|---|
| | | Method A |
| | | Method B |
| | | Method C |
| | | Any method |

Project-wide Overall Testing Effort

$$\frac{T}{S+T}$$

Project-wide per-Session Testing Effort

$$median\left\{\frac{T_s}{S_s+T_s}\right\}$$

Method-specific Overall Testing Effort

$$median\left\{\frac{T_m}{S_m+T_m}\right\}$$

Method-specific per-Session Testing Effort

$$median\left\{median\left\{\frac{T_{ss}}{S_{ss}+T_{ss}}\right\}_{m}\right\}$$

# Proposed Metrics of Testing Effort

Synthetic example: sequence of developer activity

Solution code Test code

Method A
Method B
Method C
Any method

Project-wide Overall Testing Effort

$$\frac{T}{S+T}$$

Project-wide per-Session Testing Effort

$$median\left\{\frac{T_s}{S_s+T_s}\right\}$$

Method-specific Overall Testing Effort

$$median\left\{\frac{T_m}{S_m+T_m}\right\}$$

Method-specific per-Session Testing Effort

$$median\left\{median\left\{\frac{T_s}{S_s+T_s}\right\}_m\right\}$$

Method-specific Overall Sequence of Testing Effort

= Method is “finalised”

$$median\left\{\frac{T_{before}}{T_{before}+T_{after}}\right\}$$

# Motivating Example from Fall 2016

Fig. 1: Good Test Writing Process

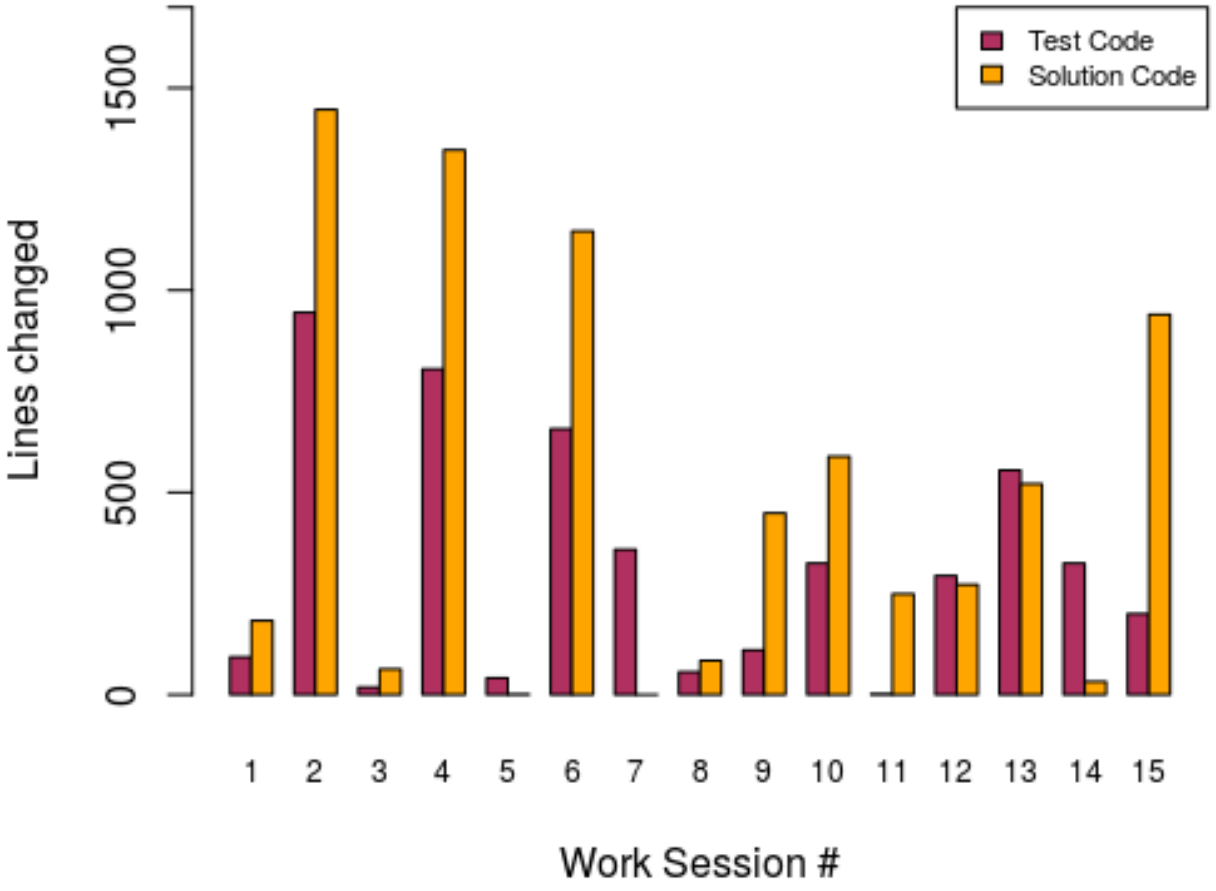
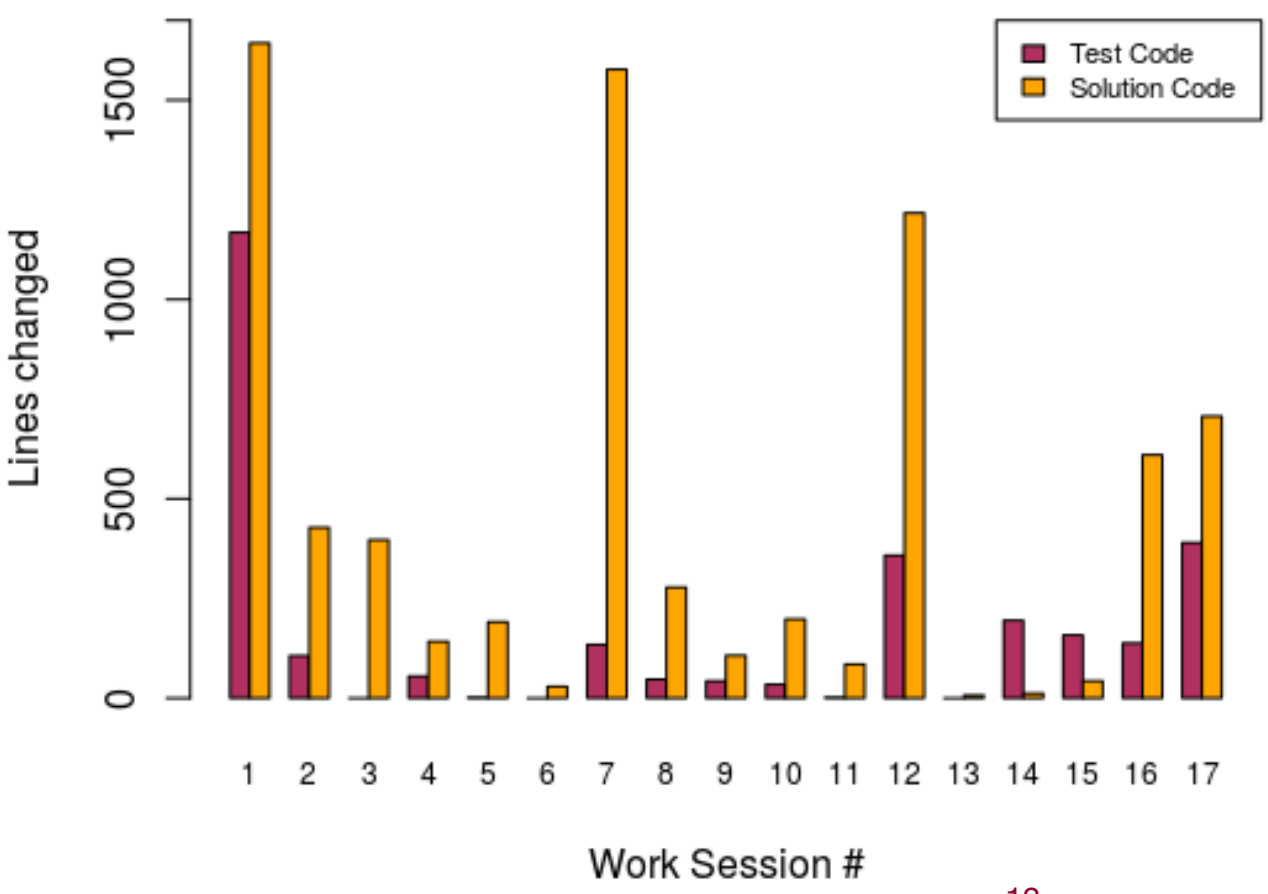



Fig. 2: Poor Test Writing Process

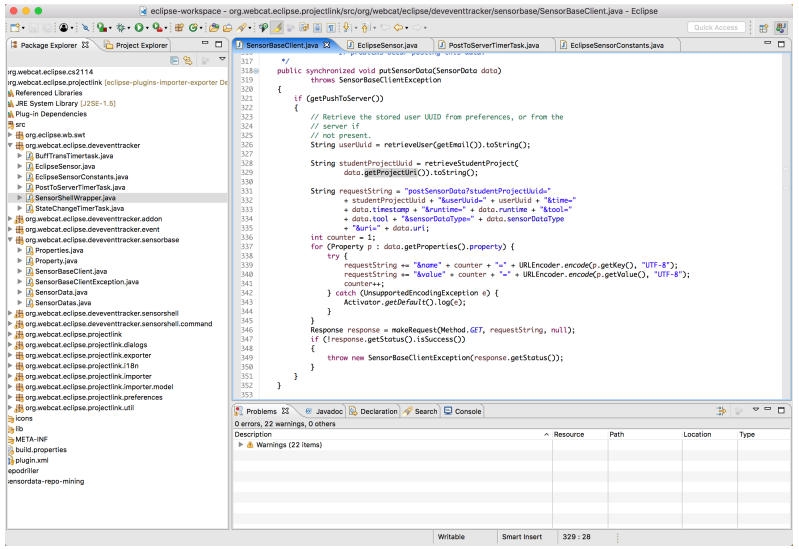


VIRGINIA TECH

# Empirical Study

# Data Collection

- 400+ project implementations



**Edit Event**

Type: Edit
Time: 1477672862
Snapshot Id: 23479b3

| Type | Size | Time |
|---|---|---|
| Change in method **`insertFront`** | +5 | 12:41:02 |
| Change in method **`getSize`** | +1 | 12:41:02 |
| Change in test for **`insertFront`** | +3 | 12:41:02 |

# Study Design

- Fixed effects: 5 measures of testing effort
- Random effects: students, assignments
- Outcome variables:
  - ***Correctness, m***easured by the percentage of reference tests passed
  - ***Code coverage*** achieved by the student's own test suite

Mixed effects model: repeated measures for each student, and for each assignment.

VIRGINIA TECH™

# Results

# Project-wide Overall Testing Effort


Expectation: Positive relationship with correctness and code coverage.

| Correctness | | Code Coverage | |
|---|---|---|---|
| Regression estimate | *p* | Regression estimate | *p* |
| 0.30 | < 0.001 * | 0.23 | < 0.001 * |

- Implementations with a higher project-wide testing effort achieved:
  - Higher semantic correctness
  - Higher code coverage

# Project-wide per-Session Testing Effort


Expectation: Positive relationship with correctness and code coverage.

| Correctness | | Code Coverage | |
|---|---|---|---|
| Regression estimate | *p* | Regression estimate | *p* |
| 0.30 | 0.005 * | 0.12 | 0.008 * |

- Implementations with higher testing effort within each work session achieved
  - Higher semantic correctness
  - Higher code coverage

# Motivating Example (Reprise)

Fig. 1: Good Test Writing Process

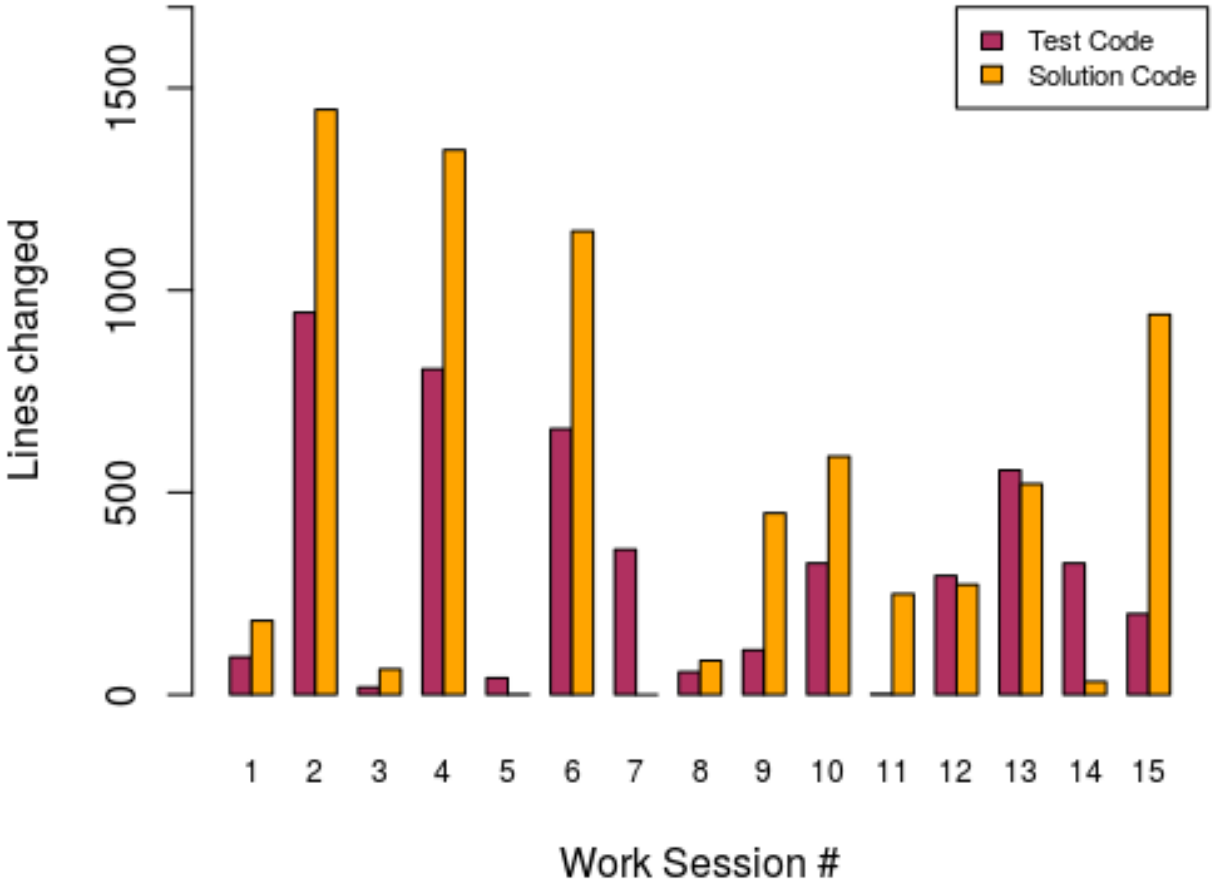
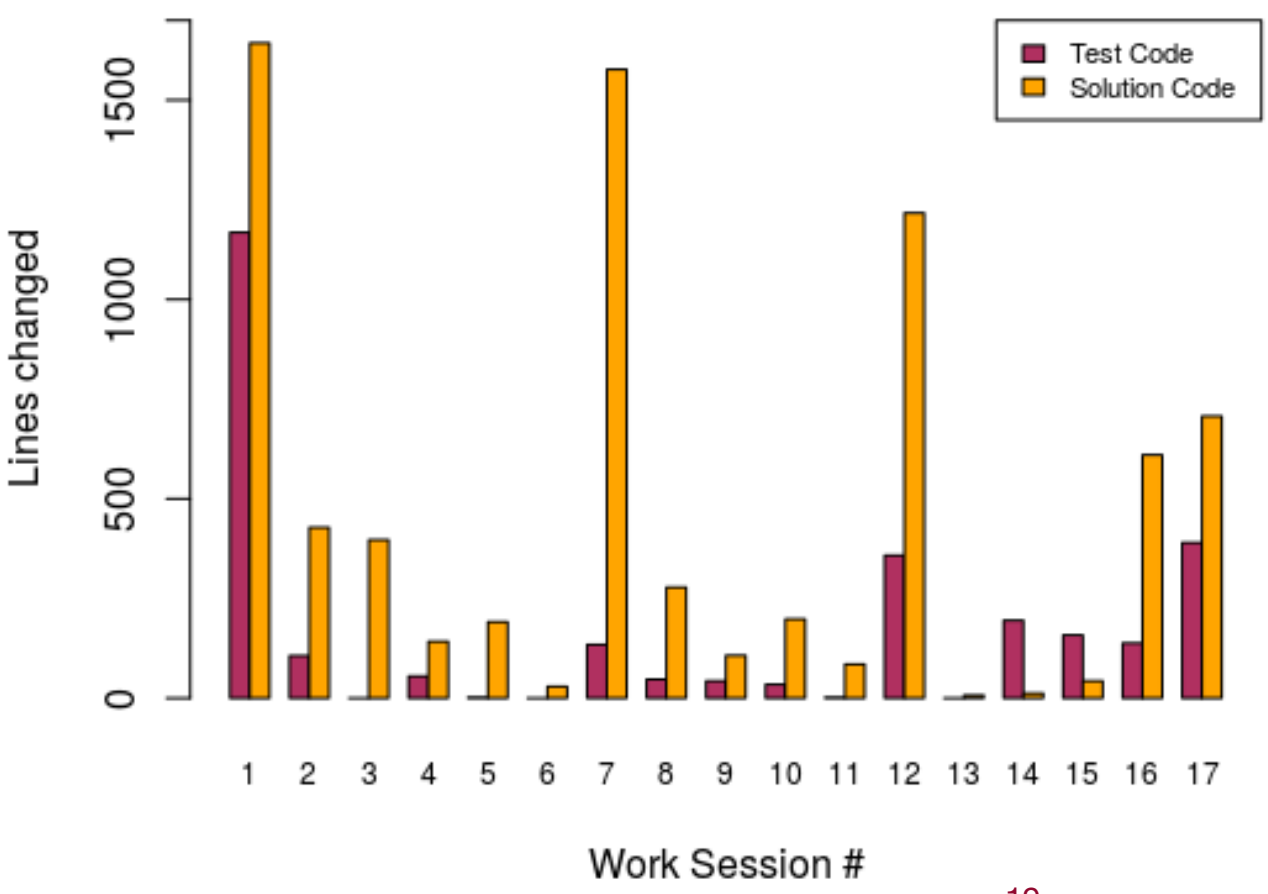



Fig. 2: Poor Test Writing Process



# Method-specific Sequence of Testing Effort

Expectation: *Positive* or *no relationship* with project outcomes.

| Correctness | | Code Coverage | |
|---|---|---|---|
| Regression estimate | p | Regression estimate | p |
| -- | 0.10 | –0.06 | < 0.001 * |

- Implementations where a higher proportion of testing for a method was done *before the method was finalised*, achieved:
  - No significant change in correctness
  - Lower code coverage

# Putting It All Together

1

- Correctness
- Code Coverage

2

- Correctness
- Code Coverage

3

- ? Correctness
- Code Coverage

VIRGINIA TECH

# Closing Remarks

VIRGINIA TECH

# Summary

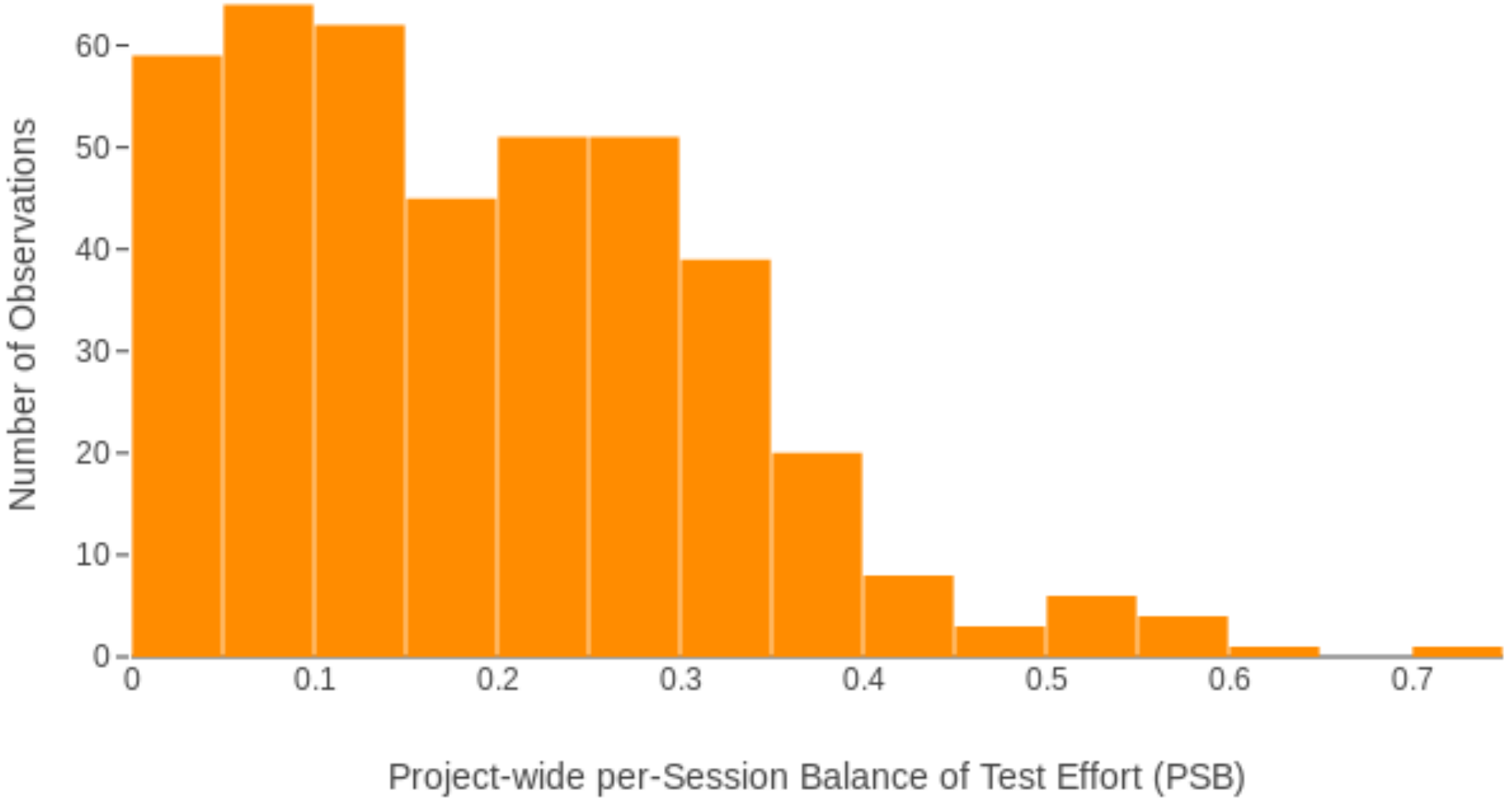
- Quantified test-writing practices
- Empirical study
  - *Higher testing effort is good (whole project and per-method)*
  - *Higher testing effort per work session is good*
  - *No such relationship on a per-method basis*
  - *Higher testing effort before finalizing relevant solution code*
    - Does not lead to improved correctness
    - Negative relationship with code coverage
- Next step: Design and deploy automated interventions for continuous feedback

ayaan@vt.edu https://github.com/ayaankazerouni/incremental-testing

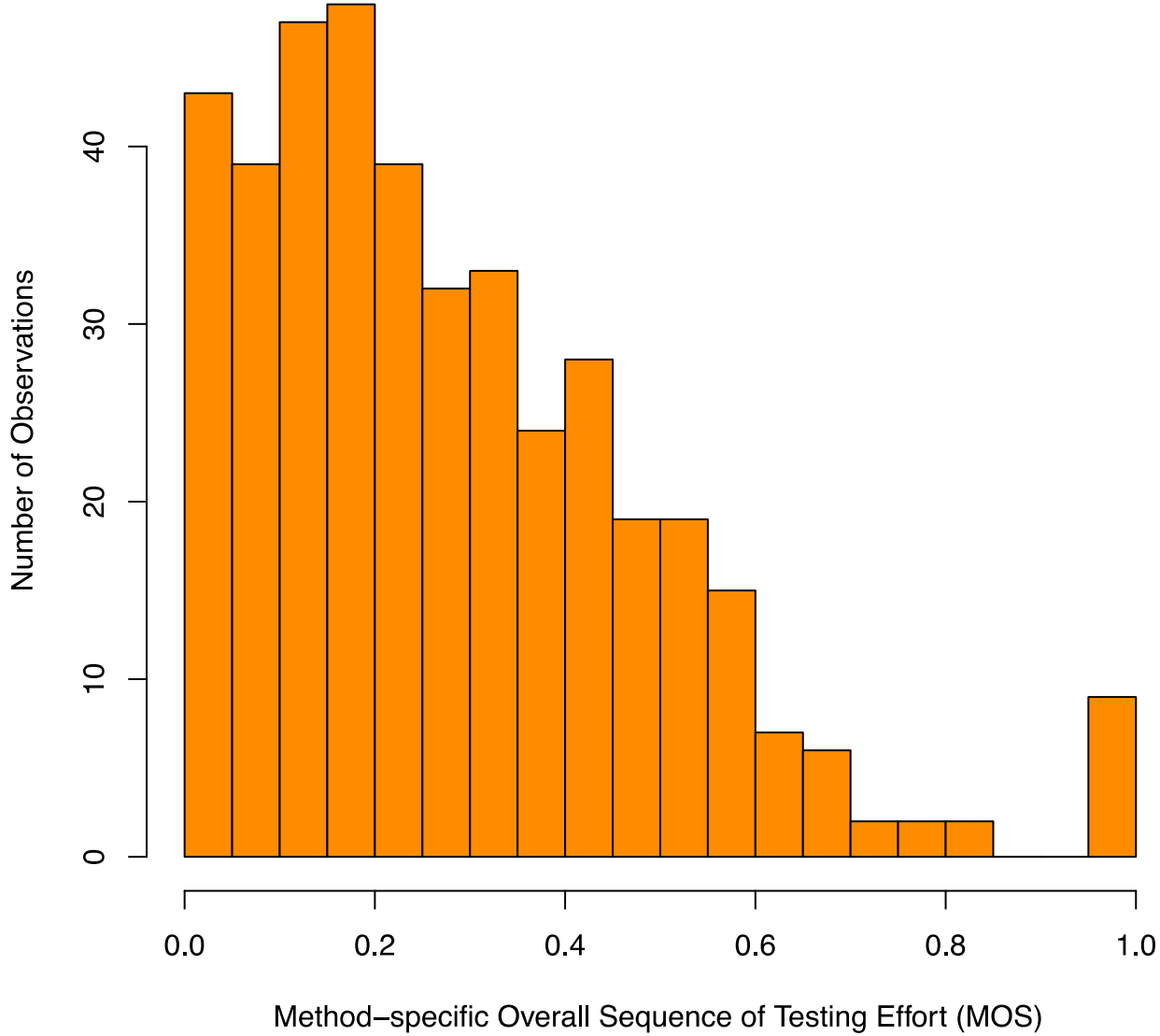
VIRGINIA TECH™

# Bonus Material

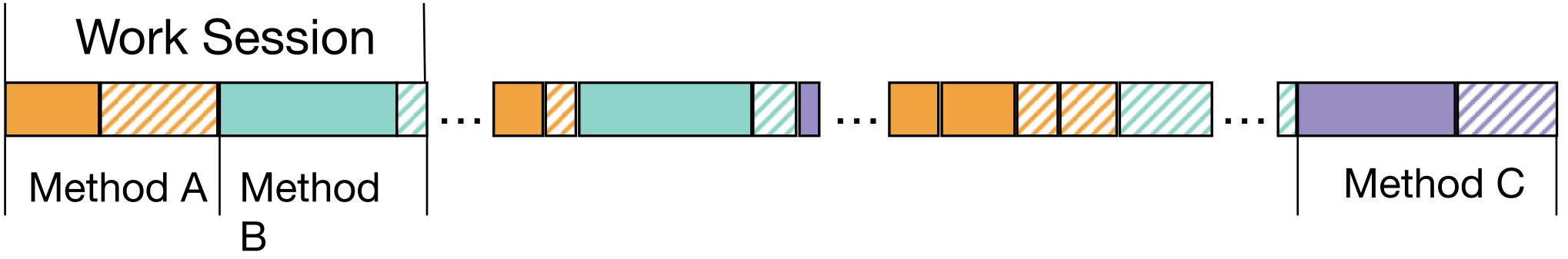
# Per-session Testing Effort: Distribution



# Method-specific Sequence of Testing Effort



# Method-specific per-Session Testing Effort




Expectation: Positive relationship with both project outcomes.

| Correctness | | Condition Coverage | |
|---|---|---|---|
| Regression estimate | *p* | Regression estimate | *p* |
| -- | 0.10 | 0.23 | < 0.001 * |

- Higher testing effort per-method, per-session achieved:
  - Higher condition coverage
  - No significant change in

# Mixed effects model (Process)

| | Correctness | | Code Coverage | |
|---|---|---|---|---|
| **Metric** | Regression estimate | p | Regression estimate | p |
| Testing per-Session | 0.30 | 0.005 * | 0.12 | 0.008 * |
| Testing per-Session per-Method | -- | 0.10 | 0.09 | 0.002 * |
| Sequence of testing | -- | 0.62 | -0.06 | 0.02 * |

# Mixed effects model (Overall)

| | Correctness | | Code Coverage | |
|---|---|---|---|---|
| **Metric** | Regression estimate | p | Regression estimate | P |
| Testing | 0.30 | < 0.001 * | 0.23 | < 0.001 * |
| Testing per-Method | -- | 0.12 | -- | 0.41 |
| Testing per-Session | -- | 0.83 | -- | 0.97 * |
| Testing per-Session, per-Method | -- | 0.97 | 0.08 | 0.01 * |
| Sequence of testing | -- | 0.74 | -0.06 | 0.03 * |

Fixed effects $R^2$ = 5%

Fixed effects $R^2$ = 10%