

# Are Newbies RUSTing away?

## A Quantitative Case Study of Contributors in the Rust Compiler

Vala Zeinali,<sup>1\*</sup> Christopher Bogart,<sup>2</sup> Daniel Klug,<sup>2</sup> Jim Herbsleb<sup>2</sup>  
<sup>1</sup>Kent State University, Kent, OH  
<sup>2</sup>(SCS, ISR), Carnegie Mellon University, Pittsburgh, PA

### What's Rust?

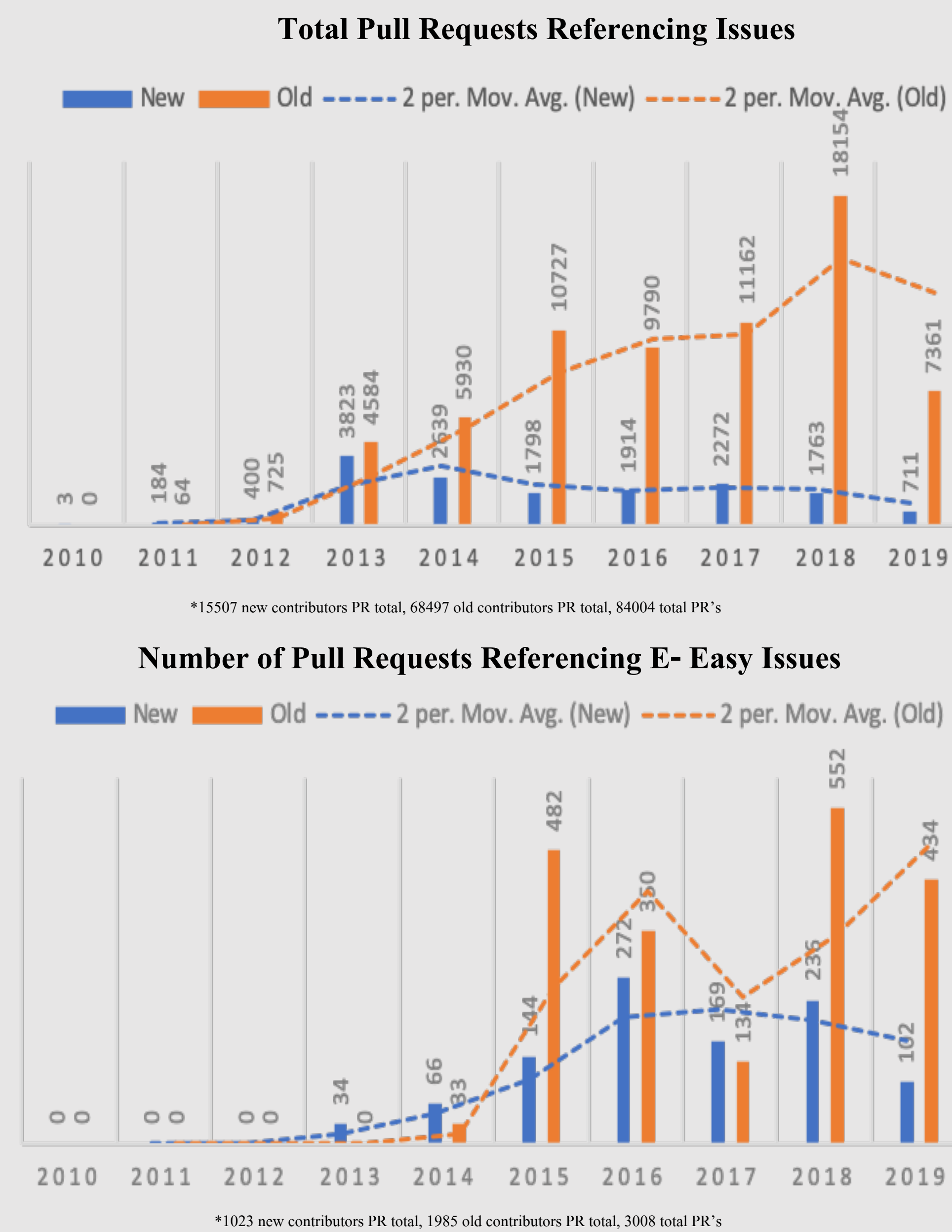
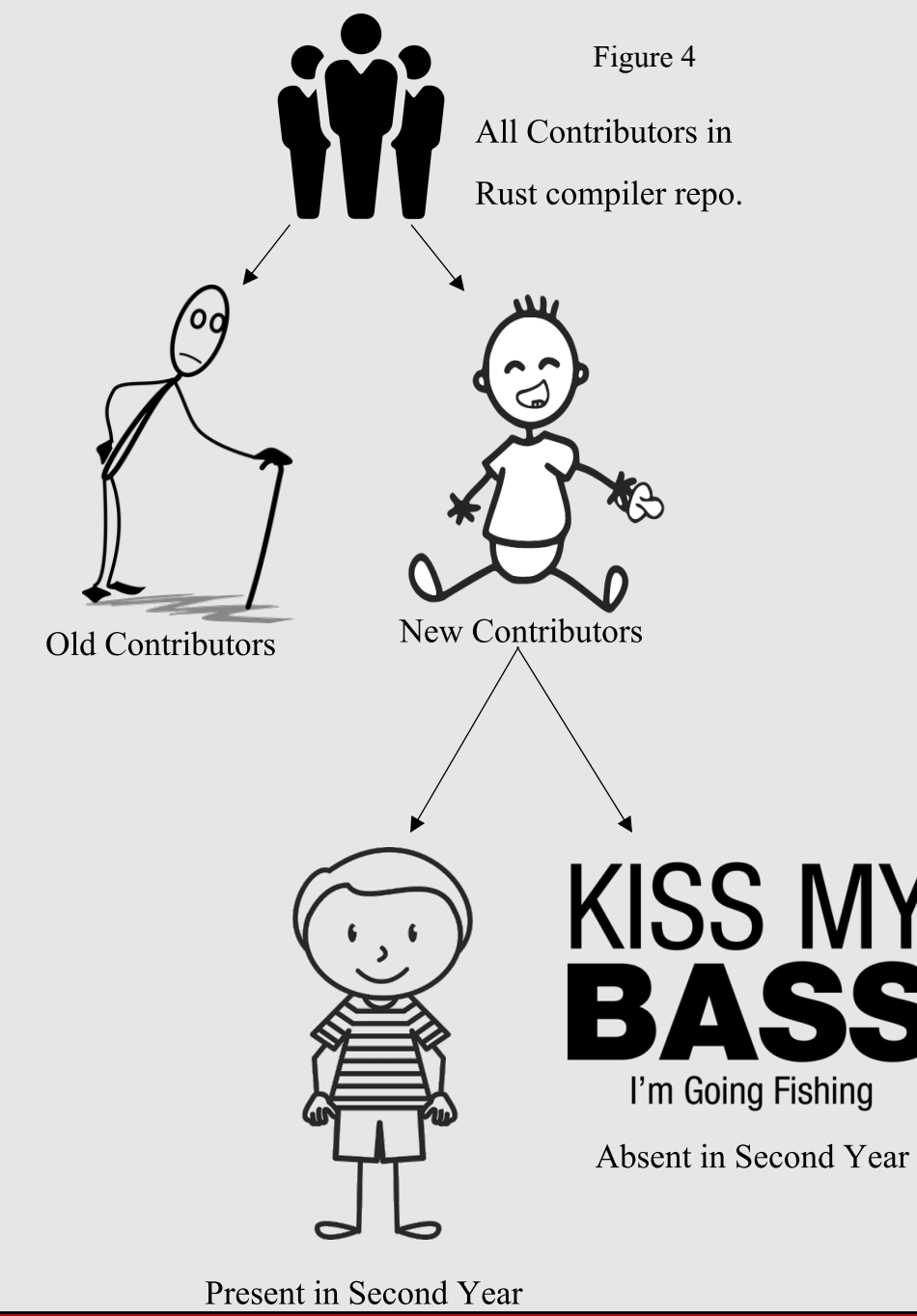
- Rust is a multi-paradigm system programming language [1] focused on safety, especially safe concurrency.[2][3] Rust is syntactically similar to C++,[4] but is designed to provide better memory safety while maintaining high performance. Rust is an open source software project on Github, a code sharing platform.

### Their Claim

- The Rust community claims they are a community of people who come together to make working a rewarding experience for all [1]. Rust desires to attract and retain new contributors to their environment. One initiative to get new users involved is through labeling issues that are deemed as easy as “E-Easy” as a signal.

### Our Question

- Our research digs deeper to see how new and old contributors in the Rust compiler team evolve. We then raise the question: what helps new users stick around for their second year? Will completing issues with “E-Easy” labels help new users stay contributing after their matriculation in the Rust compiler repository?



### Methodology

- Mined nine years of Github repository data on the Rust compiler team.
- Exploration of overall effort put in by all levels of contributors over time and within their first two years in the environment.  $effort = additions\ in\ lines\ of\ code\ (LOC) + deletions\ in\ LOC$
- Next, we classify all contributors in the repository as “new” or “old” contributors for every year and graph the shift in moving average contributor effort (figure 4).

```
if (PrProposedYear != AuthorsFirstActionYear) THEN
  Status ← OLD
ELSE
  Status ← NEW
```

- We explore if issues labeled E-Easy help new contributors stick around and put effort in their second year. We label the contributors as “present” or “absent” based on if they had effort in their second year (figure 4).

```
if (SecondYearEffort > 0) THEN
  Status ← PRESENT
ELSE
  Status ← ABSENT
```

- We run a multi-linear regression model to predict second year effort based first year effort and number of pushed pull requests referencing issues that containing the label “E-Easy”.

$$LOG(effort_{y2}) \sim LOG(effort_{y1})_{LOG(x1)} + LOG(num\_easy\_y1)_{LOG(x2)} - C$$

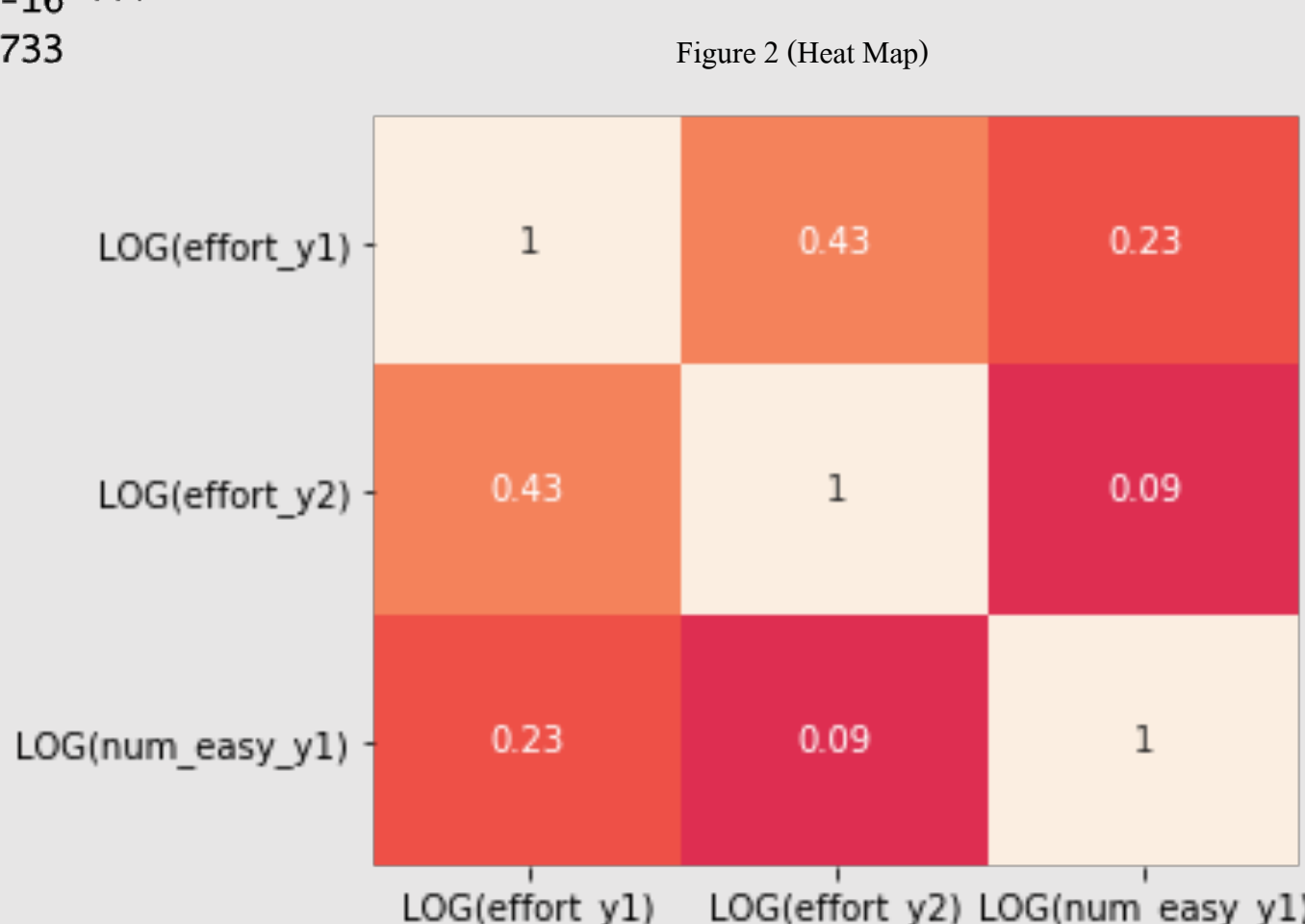
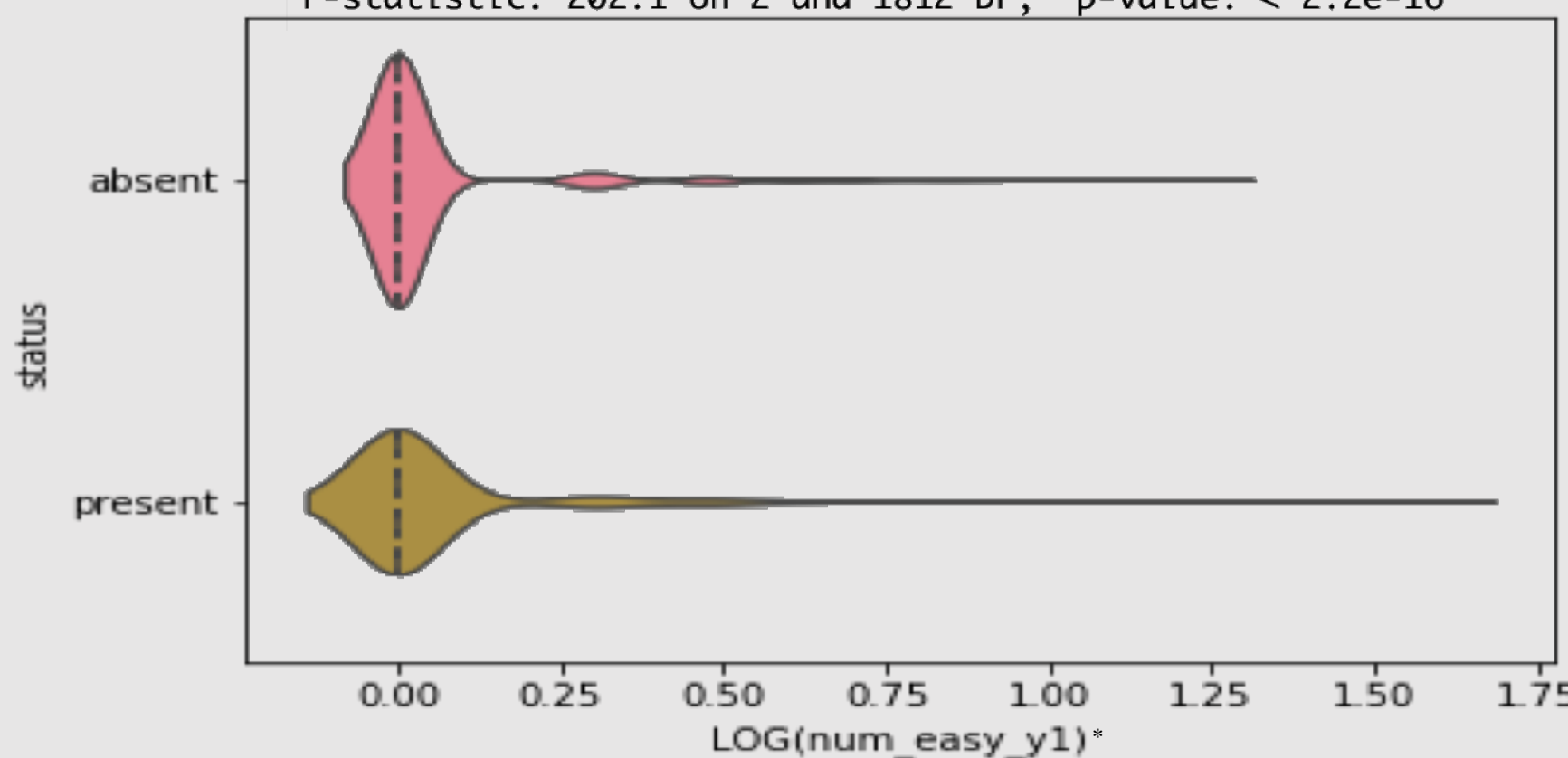
### Findings

- Are contributors who propose more pull requests (PR) that reference issues that contain the label “E-Easy” more likely to produce effort in their second year?

**Result:** We **do not** have statistical significant evidence ( $p > .05$ ) to believe that the number of E-Easy issues you complete helps your chance of putting in effort in the second year (figure 1). In addition figure 2 show us  $r = .09$  resulting in a very weak positive correlation. Less than 1% (R-squared\*100) of the variability in  $effort_{y2}$  can be explained by the linear association between  $effort_{y1}$

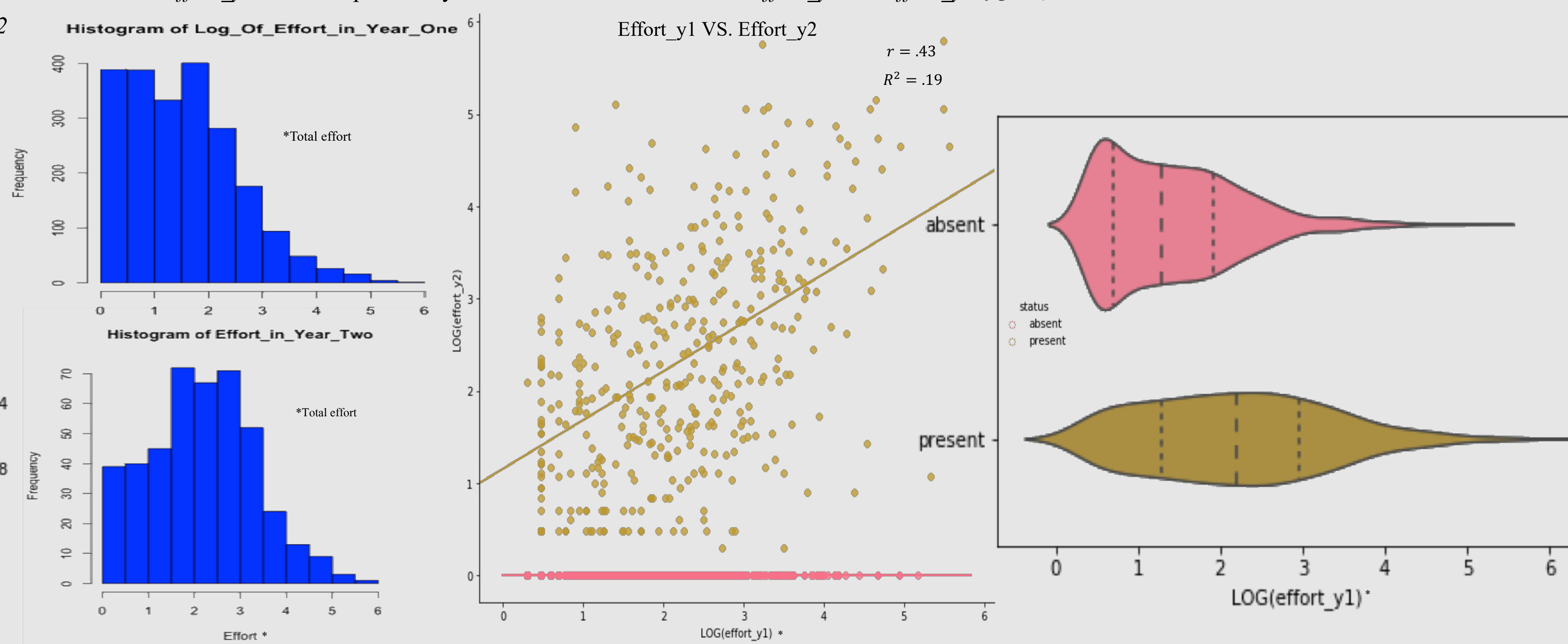
and  $num\_easy\_y1$  (figure 3).  
 Estimate Std. Error t value Pr(>|t|)  
 (Intercept) -0.27171 0.04566 -5.951 3.18e-09 \*\*\*  
 inputlinear1\$LOG\_effort\_y1 0.49569 0.02522 19.656 < 2e-16 \*\*\*  
 inputlinear1\$LOG\_num\_easy\_y1 -0.04370 0.12803 -0.341 0.733  
 ---  
 Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.012 on 1812 degrees of freedom  
 Multiple R-squared: 0.1824, Adjusted R-squared: 0.1815  
 F-statistic: 202.1 on 2 and 1812 DF, p-value: < 2.2e-16



- Are contributors that put in more effort in their first year more likely to put effort in their second year?

**Result:** In our multiple linear regression model in figure (1), the model predicts that  $effort_{y2}$  will change approximately by 3.13 ( $3.13 = 10^{.49}$ ) as  $effort_{y1}$  increases by 1 (units) on average. In addition, we have statistical significant evidence ( $p < .05$ ) to believe that effort in year one increases you chance of putting effort in year two (figure 1). In our sample 19.0% (R-squared\*100) of the variability in  $effort_{y2}$  can be explained by the linear association between  $effort_{y2}$  and  $effort_{y1}$  (figure 3).



### Future Plans

Conduct interviews with random contributors who proposed PR's referencing issues that left after their first year. Identify the reasons for why they left. Also, run same tests with “E-Mentor” label and add more variables in model for our original test. The bigger picture in our research is to identify *why* people are leaving and more importantly, *how* we can get them to stay.

### Acknowledgments

This research was possible by the REU - SE program at CMU. Thank you to my advisors for their time and knowledge. Thank you to my significant others for their support.

### References

- [1] “Rust is a systems programming language”, Rust-lang.org. Retrieved 2017-07-17.
- [2] Hoare, Graydon (2016-12-28). “Rust is mostly safety”. Graydon2. Dreamwidth Studios. Retrieved 2019-05-13.
- [3] “FAQ – The Rust Project”, Rust-lang.org. Retrieved 27 June 2019.
- [4] “Rust vs. C++ Comparison”. Retrieved 20 November 2018. Rust is syntactically similar to C++, but it provides increased speed and better memory safety