

# 重回帰分析モデルを利用した テストフレームワーク選定の定量的評価

富士通株式会社 森山泰亘 ymoriyama@jp.fujitsu.com

## 開発における問題点

テストフレームワークを利用した開発によりテスト品質の向上が期待される。本来テストフレームワークの選定はソフトウェア品質の観点で客観的・包括的な評価の基で選択されるべきであるが、現状は開発者の主観的かつ局所的な評価により選定されるという問題がある。

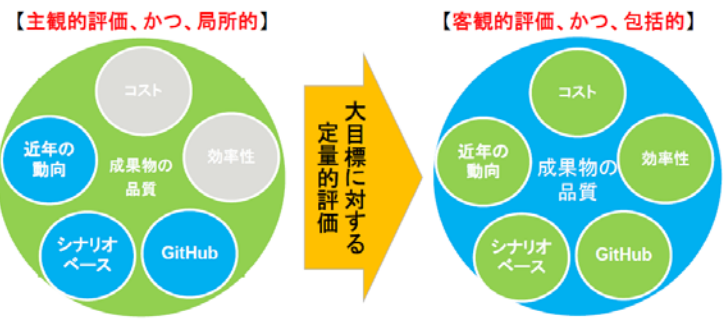
## 手法・ツールの適用による解決

先行研究(\*)の適用により、バグ影響因子を定量的に解析するモデルを作成。テストフレームワークの利用効果をバグ検出数との関係から測定することで、どのテストフレームワークを選定すべきかを定量的に評価する。

※RAY, Baishakhi, et al. A large scale study of programming languages and code quality in github. In: ACM, 2014.

## テストフレームワークの品質影響に対する定量的評価

- 選択したテストフレームワークは最適なツールか？
  - テストフレームワークの良し悪しに関する経験的な議論は多くなされているが、主観的かつ局所的な議論に留まる



テストフレームワーク選定を客観的かつ包括的に評価する為には、「成果物の定量的評価」が必要である

### ■ 分析対象およびデータセット

- ・JavaScriptを選定
- ・GitHubでStar数500以上のJavaScriptプロジェクトの内、4テストフレームワーク(Jasmine/ Mocha/ Qunit/ Tape)を検証。

### ■ テストフレームワークのバグ影響解析モデル

$$[\text{バグ検出数}] = \alpha[\text{Jasmine}] + \beta[\text{Mocha}] + \gamma[\text{Qunit}] + \delta[\text{Tape}] + p[\log(\text{commit})] + q[\log(\text{age})] + r[\log(\text{size})] + s[\log(\text{devs})] + [\text{base}]$$

- ・ [バグ検出数]: コミット履歴から得られたバグ修正のキーワードに該当するコミット数
- ・ [各テストフレームワーク]: テストフレームワークの使用/未使用を示す0または1の値
- ・  $\alpha, \beta, \gamma, \delta$ : 各テストフレームワークの使用/未使用を示す変数に対する係数

## 結果

説明変数	係数	P値
(base)	-2.155240	< 2e-16 ***
log commits	0.978867	< 2e-16 ***
log age	0.033567	1.21e-07 ***
log size	0.147106	< 2e-16 ***
Log devs	-0.011985	0.380881
Tape	-0.092252	0.001117 **
Qunit	0.171400	0.000179 ***
Mocha	-0.0002884	0.950395
Jasmine	0.072566	0.112353

・Tapeを利用しているプロジェクトは、データセット中の平均的なバグ数よりバグが少ない

- ・Qunitを利用しているプロジェクトは、データセット中の平均的なバグ数よりバグが多い
- ・Mochaを利用しているプロジェクトは、係数がほぼ0でかつp値が十分に高いので、データセット中の平均的なバグ数と同等
- ・Jasmineは係数が0より大きいですが、p値が高くどちらとも言えないため議論から除外

## まとめ

- 従来、主観的評価で利用選定していたテストフレームワークに対し、重回帰分析による統計的手法を適用し、ソフトウェア品質(バグ数)の観点での定量的評価を可能にした。
- 本手法をJavaScriptにおける4種類のテストフレームワークの比較に適用した結果、Tapeが最もバグを作り込みにくいテストフレームワークであることが分かった。



トップエスイー修了制作

# タイトル

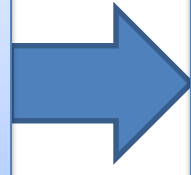
所属

名前

メールアドレス

開発における問題点

手法・ツールの適用による解決



ヘッダ

ヘッダ