

How to Not Get Rich: An Empirical Study of Donations in Open Source

著者

Cassandra Overney (Olin College, USA),
Jens Meinicke, Christian Kästner, Bogdan Vasilescu (Carnegie Mellon University, USA)

論文情報：ICSE '20, 42nd International Conference on Software Engineering,
May 23-29, Seoul, South Korea.

紹介者：堀口 日向

- ・オープンソースソフトウェア（OSS）は、企業の様々な製品開発に利用
しかし、多くの場合、利用者はOSSに貢献することはない
 - ・OSSの利用者が増えると、
 - ・バグ報告や機能リクエストなどが増える
 - ・バグや脆弱性の影響が大きくなる
- 開発作業や保守にコストが発生

- ・コストを賄う手段
 - ・企業や財団からの資金援助
 - ・有料版の配布
 - ・寄付を募る ← 手軽にできる手段として人気

OSSプロジェクトが寄付を募ることに関して、
普及の程度や成功しているのかといったことは知られていない

目的

OSSプロジェクトの寄付集めの目標，実情，効果などを理解する

リサーチクエスト

- RQ1：プロジェクトが寄付を募ることはどのくらい一般的か？
- RQ2：寄付を集めるために一般的に使われている寄付プラットフォームは何か？
- RQ3：寄付を募っているプロジェクトの特徴は何か？
- RQ4：寄付を多く受けているプロジェクトの特徴は何か？
- RQ5：寄付を募っている開発者は何にお金を使いたいのか？
- RQ6：寄付は開発・保守効率の向上に効果があるか
- RQ7：寄付はどのように使用されるか？

リサーチクエスト

RQ1：プロジェクトが寄付を募ることはどのくらい一般的か？

RQ2：寄付を集めるために一般的に使われている寄付プラットフォームは何か？

動機

寄付集めに関するOSSプロジェクトの全体的な傾向を理解するため

アプローチ

npmとGitHubにあるすべてのリポジトリを対象に、

1. 寄付を募っているリポジトリを特定
2. 使用している寄付プラットフォームの特定

データセット

2 種類用意（以降のRQでも同様）

- npm

GitHubと異なり，以下の情報が含まれる

- ダウンロード数
- 逆依存数（他のパッケージから参照されている数）

npmで公開されているすべて（約50万件）のリポジトリ

- GitHub

GHTorrentに含まれるすべて（約8,000万件）のリポジトリ



Node Package Manager (npm)
Node.jsのパッケージ管理システム

寄付を募っているリポジトリの特定

READMEに“donate”，”sponsor”，”donation”という単語が含まれている

使用している寄付プラットフォームの特定

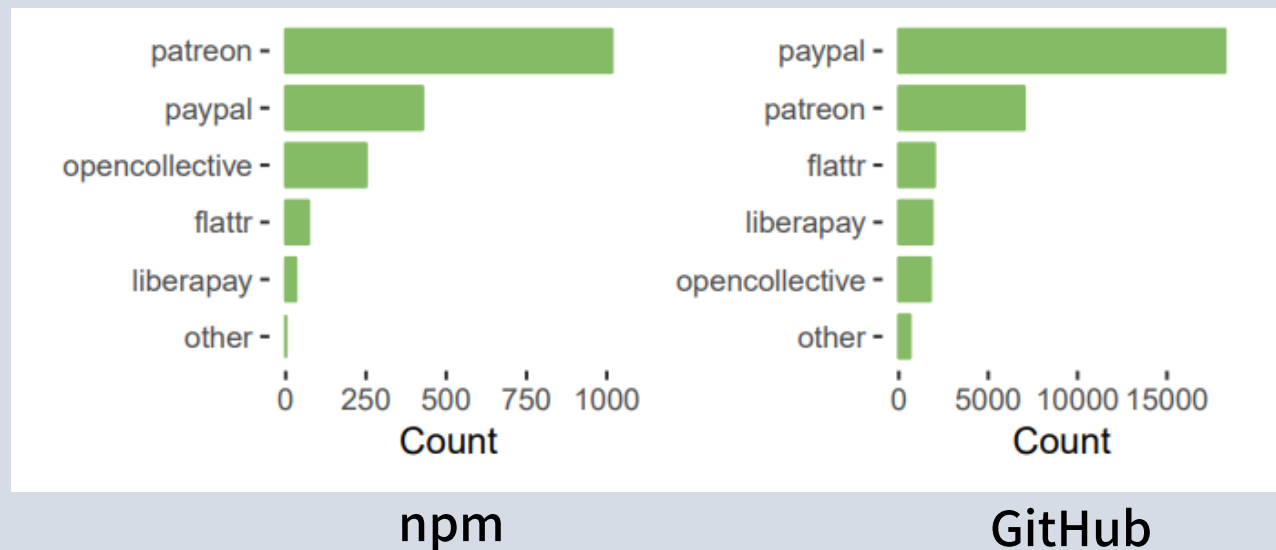
1. 著者らがREADMEから，目視でプラットフォーム名を抽出
 2. grepで検索し，ヒットしたリポジトリを検索から除外
 3. 残ったリポジトリから，新たなプラットフォームを抽出
- 1, 2, 3を繰り返す

RQ1：プロジェクトが寄付を募ることはどのくらい一般的か？

- npm : 0.2% (1,145 / 537,640)
- GitHub : 0.04% (25,885 / 77,934,441)

⇒ 寄付を募っているプロジェクトは全体のごく一部

RQ2：寄付を集めるために一般的に使われている寄付プラットフォームは何か？



npm, GitHubともに、
上位2つはPayPalとPatreon

リサーチクエスト

RQ3：寄付を募っているプロジェクトの特徴は何か？

RQ4：寄付を多く受けているプロジェクトの特徴は何か？

動機

寄付を募り，受け取っているプロジェクトの特徴を理解するため

アプローチ

寄付を募る，または多く受け取ることに関連するプロジェクト特性
（人気度，活動量，貢献者の数など）を確認するため，重回帰分析を行う

説明変数

コミット数，Issue数，スター数，リポジトリのサイズ，年数，
最近の活動の有無（過去9ヶ月間に1回以上のコミットがあったかどうか），
組織アカウントが所有しているリポジトリかどうか

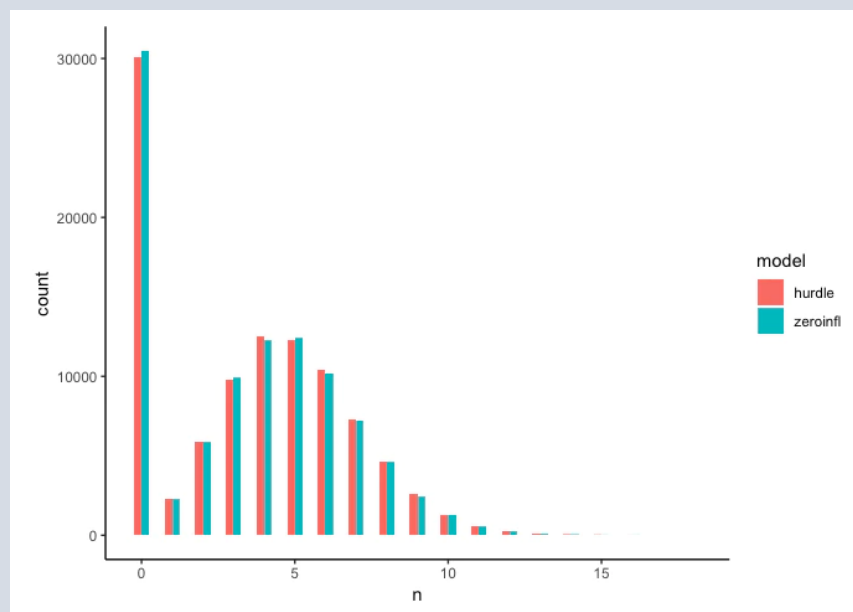
目的変数，回帰モデル

- RQ3：寄付を募っているプロジェクトの特徴は何か？
寄付を募っている or 募っていない（2値） → ロジスティック回帰
- RQ4：寄付を多く受けているプロジェクトの特徴は何か？
受け取った寄付金額 → ハードル回帰

ハードル回帰

0が多く含まれるデータを回帰分析する際に用いる
今回は，寄付金額が多くの場合0ドルであることを想定

ハードル回帰を使うデータの分布イメージ



画像出典：<https://qiita.com/nozma/items/52211b1bacaa8a898164>

RQ3：寄付を募っているリポジトリの特徴は何か？

npmでの分析結果．GitHubでも同様の結果

	Resp: Asks for donations	
	Coeffs (Err.)	Deviance
(Intercept)	−4.01 (0.19)***	
commits (log)	0.40 (0.05)***	72.95***
size (log)	−0.30 (0.03)***	125.74***
project age	0.02 (0.00)***	85.94***
is active	1.95 (0.09)***	502.20***
is org	−0.57 (0.10)***	33.63***
stars (log)	0.27 (0.02)***	129.89***
downloads (log)	−0.02 (0.02)	0.88
dependents (log)	0.01 (0.05)	0.02
Num. obs.	9137	
R ²	0.31	

*** $p < 0.001$, ** $p < 0.01$, * $p < 0.05$

正の相関（効果が大きい順）

- 最近活動しているかどうか
- コミット数
- 年齢
- スター数

⇒ 活動が活発で，人気があるプロジェクト
ほど寄付を求めている

RQ3：寄付を募っているリポジトリの特徴は何か？

npmでの分析結果．GitHubでも同様の結果

	Resp: Asks for donations Coeffs (Err.)	Deviance
(Intercept)	-4.01 (0.19)***	
commits (log)	0.40 (0.05)***	72.95***
size (log)	-0.30 (0.03)***	125.74***
project age	0.02 (0.00)***	85.94***
is active	1.95 (0.09)***	502.20***
is org	-0.57 (0.10)***	33.63***
stars (log)	0.27 (0.02)***	129.89***
downloads (log)	-0.02 (0.02)	0.88
dependents (log)	0.01 (0.05)	0.02
Num. obs.	9137	
R ²	0.31	

*** $p < 0.001$, ** $p < 0.01$, * $p < 0.05$

負の相関（効果が大きい順）

- ・リポジトリサイズ
- ・組織のリポジトリかどうか

⇒ 組織が保有するような大規模な
リポジトリは，寄付を必要としていない

RQ4：寄付を多く受けているプロジェクトの特徴は何か？

寄付金額の統計情報

- npmのリポジトリ

	受け取った金額[ドル]
最小値	0
中央値	55
最大値	589,382

- 1ドル以上の寄付：70%
- 月1000ドル以上の寄付：5%

⇒ ほとんどはフルタイムで開発者を雇えるほど寄付を得ていない

- GitHubのリポジトリでも同様
 - 1ドル以上寄付：58%
 - 月1000ドル以上寄付：10%

RQ4：寄付を多く受けているプロジェクトの特徴は何か？

	Hurdle model		Count model	
	Resp: <i>Received any</i>		Resp: <i>Amount received</i>	
	Coeffs (Err.)	Deviance	Coeffs (Err.)	Sum sq.
(Intercept)	0.12 (0.38)		4.17 (0.39)***	
commits (log)	−0.20 (0.12)	3.05	−0.26 (0.11)*	20.41*
size (log)	−0.10 (0.06)	2.80	0.06 (0.07)	2.67
project age	0.05 (0.01)***	58.63***	−0.01 (0.00)	10.93
is active	1.33 (0.22)***	38.73***	0.00 (0.21)	0.00
is org	0.84 (0.26)**	10.78**	0.12 (0.20)	1.37
stars (log)	0.14 (0.06)*	6.06*	0.39 (0.06)***	182.17***
downloads (log)	−0.11 (0.06)	3.51	0.13 (0.05)**	28.60**
dependents (log)	0.31 (0.11)**	8.98**	−0.04 (0.08)	0.85
Num. obs.	735		527	
R ²	0.29		0.30	

*** $p < 0.001$, ** $p < 0.01$, * $p < 0.05$

寄付を1ドル以上受けている
プロジェクトの特徴

正の相関（効果が大きい順）

- 年齢
- 最近活動しているかどうか
- 逆依存数
- スター数

⇒ 活発で，人気，開発者によく
利用されている

RQ4：寄付を多く受けているプロジェクトの特徴は何か？

	Hurdle model		Count model	
	Resp: <i>Received any</i>		Resp: <i>Amount received</i>	
	Coeffs (Err.)	Deviance	Coeffs (Err.)	Sum sq.
(Intercept)	0.12 (0.38)		4.17 (0.39)***	
commits (log)	-0.20 (0.12)	3.05	-0.26 (0.11)*	20.41*
size (log)	-0.10 (0.06)	2.80	0.06 (0.07)	2.67
project age	0.05 (0.01)***	58.63***	-0.01 (0.00)	10.93
is active	1.33 (0.22)***	38.73***	0.00 (0.21)	0.00
is org	0.84 (0.26)**	10.78**	0.12 (0.20)	1.37
stars (log)	0.14 (0.06)*	6.06*	0.39 (0.06)***	182.17***
downloads (log)	-0.11 (0.06)	3.51	0.13 (0.05)**	28.60**
dependents (log)	0.31 (0.11)**	8.98**	-0.04 (0.08)	0.85
Num. obs.	735		527	
R ²	0.29		0.30	

*** $p < 0.001$, ** $p < 0.01$, * $p < 0.05$

寄付を多く受け取っている
プロジェクトの特徴

正の相関（効果が大きい順）

- ・スター数
- ・ダウンロード数

⇒ 人気，ユーザによく
利用されている

リサーチクエスト

RQ5：寄付を募っている開発者は何にお金を使いたいのか？

動機

開発者の寄付に対する期待を確認し、
後のRQで寄付が効果的だったかどうかを確かめるため

アプローチ

寄付プラットフォーム上のREADMEやプロフィールページを収集し、
開発者が受け取った資金をどのように使うかについての言及を探す

定性的な分析

1. npmリポジトリのREADME.mdファイル，プロフィールページを収集

2. サンプリング

ファイルサイズで並べ替え，上位半分の中からランダムに109個サンプリング
(ファイルサイズが大きい方が，寄付に関する詳細な説明が含まれている可能性が高い)

3. カテゴリの分類

著者ら2人で，オープンコーディング

RQ5：寄付を募っている開発者は何にお金を使いたいのか？

用途	割合	
開発費	48%	機能の追加・改善，Issue解決などを行ったエンジニアに対する給与
コミュニティ費	18%	ドキュメントやチュートリアルを充実させるため
サーバ運用費	13%	
開発者自身への個人的な支援	9%	ストレス軽減，モチベーション維持，コーヒー代など

リサーチクエスション

RQ6：寄付は開発・保守効率の向上に効果があるか

動機

RQ5で開発者が最も多く期待していた開発活動への効果があるか確かめる

アプローチ

コミット数，Issue解決の速さに効果があるかを検証

寄付の前後で効果を確かめるため，中断時系列分析を行う

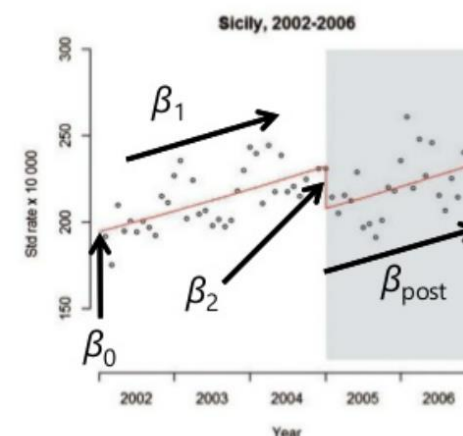
中断時系列分析 (interrupted time series analysis : ITSA)

介入（本研究の場合は寄付）前後で，傾向がどのように変化したか

- 目的変数： Y_t
- 説明変数： T （時刻）
- 介入の有無： X_t （2値）
- 説明変数の係数： β
 - 介入前の傾向： β_1
 - 介入の瞬間の影響度合： β_2
 - 介入前後の傾向の変化量： β_3
 - ※介入後の傾向そのものではない

$$Y_t = \beta_0 + \beta_1 T + \beta_2 X_t + \beta_3 T X_t$$

- ✓ β_0 : 切片 (介入前)
- ✓ β_1 : 傾き (介入前)
- ✓ β_2 : 切片変化量 (介入後 - 介入前)
- ✓ β_3 : 傾き変化量 ($\beta_{\text{post}} - \beta_1$)



画像出典：<https://www.slideshare.net/Shuheilchikawa/ss-107977109>

データセット

寄付の履歴が追えるのが9か月のため，寄付（介入）があった前後9か月間
データポイントは月ごと

目的変数

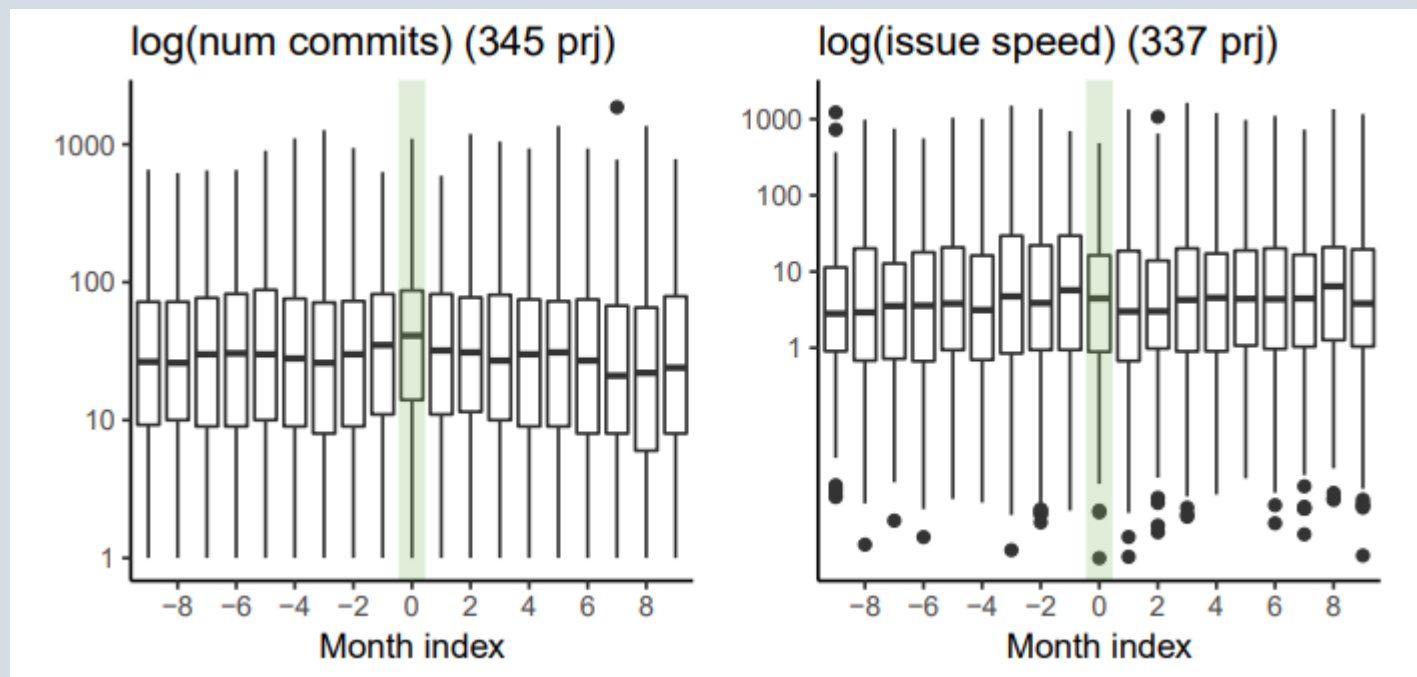
- ・ひと月間のコミット数
- ・ひと月の間にクローズされたIssueが開いていた平均時間

制御変数

- ・ひと月間の寄付金額
- ・ひと月間の間にクローズされたIssue数
増えればIssueが開いている時間が長くなる可能性

RQ6：寄付は開発・保守効率の向上に効果があるか

月ごとの
コミット数の
分布



月ごとの
Issueが開いていた
平均時間の分布

介入があっても変化がなさそう・・・？

RQ6：寄付は開発・保守効率の向上に効果があるか

コミット数

	Commits model		Issue speed model	
	Resp: num commits (log)		Resp: issue speed (log)	
	Coeffs (Err.)	Chisq	Coeffs (Err.)	Chisq
(Intercept)	0.79 (0.42)		1.79 (0.29)***	
earnings (log)	0.30 (0.06)***	21.25***	-0.14 (0.04)**	9.75**
closed issues (log)			0.38 (0.03)***	134.65***
month_index	0.15 (0.02)***	59.59***	0.05 (0.02)**	10.69**
intervention	0.03 (0.16)	0.05	-0.37 (0.12)**	9.72**
month_after	-0.25 (0.03)***	79.70***	-0.01 (0.02)	0.27
Num. obs.	6210		4912	
R ²	0.36		0.33	

*** $p < 0.001$, ** $p < 0.01$, * $p < 0.05$

介入前：わずかな増加傾向
介入の瞬間：相関なし
介入後：減少傾向に変化

⇒ 寄付があっても
活動量は増えない

寄付金額が多いと
活動量は増える

RQ6：寄付は開発・保守効率の向上に効果があるか

Issueクローズまでの平均時間

	Commits model		Issue speed model	
	Resp: num commits (log)		Resp: issue speed (log)	
	Coeffs (Err.)	Chisq	Coeffs (Err.)	Chisq
(Intercept)	0.79 (0.42)		1.79 (0.29)***	
earnings (log)	0.30 (0.06)***	21.25***	-0.14 (0.04)**	9.75**
closed issues (log)			0.38 (0.03)***	134.65***
month_index	0.15 (0.02)***	59.59***	0.05 (0.02)**	10.69**
intervention	0.03 (0.16)	0.05	-0.37 (0.12)**	9.72**
month_after	-0.25 (0.03)***	79.70***	-0.01 (0.02)	0.27
Num. obs.	6210		4912	
R ²	0.36		0.33	

*** $p < 0.001$, ** $p < 0.01$, * $p < 0.05$

介入前：時間が長くなる傾向
介入の瞬間：時間が短くなる
介入後：変化なし

⇒ 寄付の影響期間は短い

寄付金額が多いと
クローズされるまでの時間が短い

リサーチクエストション

RQ7：寄付はどのように使用されるか？

動機

RQ6で寄付の効果が限定的であった理由を調査するため

アプローチ

READMEやプロフィールページを目視

目視用データセット

寄付金額の規模でプロジェクトを分ける

- ・ 月1000ドル以上
- ・ 月100ドル未満
- ・ その中間

+ 累計の寄付金額が多い
上位15プロジェクト

各15プロジェクトをランダムサンプリング

分析

README, プロファイルページを3人の著者らが目視
寄付の用途のパターンを特定, 各パターンで簡単な定量化

支出家 vs 貯蓄家

- 目視での発見
積極的に寄付金を使うプロジェクトもあれば，貯蓄するプロジェクトもある
- GitHub全体での定量化

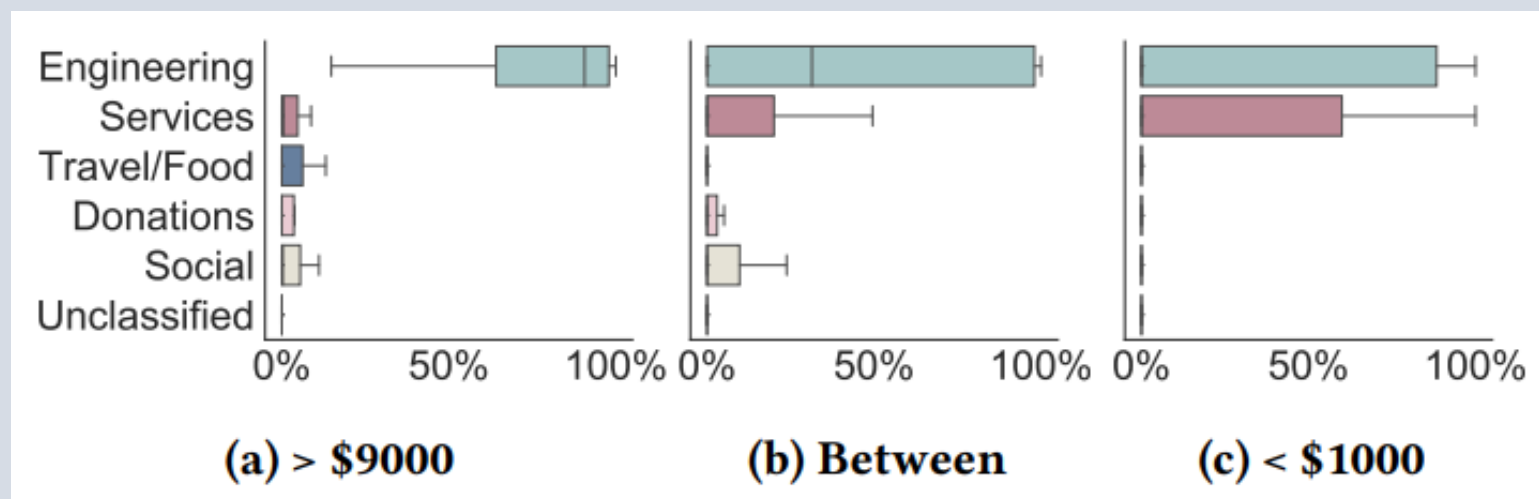
	Spending < 25 %	Spending > 75 %
Donations > 1000 USD/month	37 %	17 %
Donations 100–1000 USD/month	59 %	10 %
Donations < 100 USD/month	68 %	11 %

貯蓄するプロジェクトが多い

⇒ 寄付金を使ってないから寄付の効果が出にくい？

寄付の用途

- 目視での発見
開発費やサーバ運用費に使われている
- GitHub全体での定量化



⇒ 寄付金額が少ないほど，開発費に充てられていない

寄付の受領者

- 目視での発見
 - 多く活動している貢献者が受け取る：72%
 - 新人，賞金付きIssue，コミュニティ活動への投資：35%
- GitHub全体での定量化
分析を自動化できず

- OSSプロジェクトの中で寄付を募っているのは**ごく一部** (RQ1)
- 寄付を募っている, 受け取っているプロジェクトは,
活動が活発で, 成熟していて, 人気のある (RQ3, 4)
- 開発者が寄付を募る理由 (RQ5)
開発費, 運用費のカバー, コミュニティ育成, ストレス軽減
- 寄付を受け取っても, **開発活動への効果は限定的** (RQ6)
(ただし, いっぱいもらうと活発にはなる)

寄付の効果が限定的であった理由の調査

- ・ 寄付金がどのように使われているのか (RQ7)

⇒ 多くのプロジェクトでは、寄付を募った**目的に沿って利用**

しかし、実際には**ごく一部の資金**しか使われていない

また、**寄付金額が少ないと、運用にお金**が集中し、開発に回らない

効果的な寄付金の集め方、使い方などを理解することが今後の研究課題

- 研究の手続き的な話

分析 → 結果 → **裏付けのための分析** → … を繰り返しているのですが、結果に納得感しかし、1人じゃ絶対手が回らないぐらい作業量が多い、さすICSE

- 寄付する側の期待するところは？

投げ銭感覚？プロジェクトの存続？

- 寄付の対象が個人なのかプロジェクトなのか、切り分けて分析したら？

個人に対する支援とプロジェクトに対する支援では、性格が違ふような気がする

- OSS活動で生計を立ててるフリーランス開発者について

数は多くないだろうから、研究にはならなさそうだけど、
そんな生き方を知りたい・・・

データセット

寄付を募っているすべてのリポジトリと、
寄付を募っていないリポジトリをランダムサンプリング

- npm : 1,145件 / 8,124件
- GitHub : 25,885件 / 109,289件

RQ4では寄付金額のデータを使うので、
寄付金額が過去9か月分公開されているプラットフォーム

- Patreon
- OpenCollective

を利用しているリポジトリのみを分析 (npm : 817件, GitHub : 6,516件)

- 分析スクリプト，データセット

<https://github.com/CMUSTRUDEL/oss-donations>