CODE-WASHING:

EVIDENCE FROM OPEN-SOURCE BLOCKCHAIN STARTUPS

First Draft: Oct 2024 — This Draft: July 2025

Abstract

This study examines how startups strategically manage open-source code disclosures during fundraising. We introduce the concept of *code-washing*—the superficial use of source code repositories to mimic authentic development—as a novel mechanism of information manipulation in the entrepreneurial setting. Using a global dataset of blockchain startups and linking GitHub activity to detailed fundraising outcomes, we classify startups as *code-producers* or *code-washers* based on the depth and timing of development activity. We estimate treatment effects on fundraising success, controlling for extensive startup characteristics and market conditions, and validate our proxies through tests of coding timing, external developer engagement, disclosure informativeness, and responses to the 2017 SEC crackdown on unregistered token offerings. Our main analysis stratifies startups by investor attention—proxied by Ether market returns—and information environment, testing whether markets shift from pooling to separating equilibria as scrutiny increases. Consistent with signaling theory, we find that code-producers outperform only in periods of low investor optimism or high information availability, while code-washers' fundraising advantage disappears. Post-fundraising, code-producers sustain innovation and deliver significantly higher returns, whereas code-washers experience declines in code activity and negative performance. Our findings extend the literature on disclosure manipulation around capital market events by documenting a new channel—open-source transparency—that can be strategically exploited, particularly in settings with high information asymmetry and weak external verification.

JEL classification: G15, G18, G24, K29, K42, L14, L26, M13, O31.
Keywords: Code-washing, strategic disclosure, open source, startups, and fundraising.

1 Introduction

A central question in accounting research is how information asymmetry shapes the quality and consequences of voluntary corporate disclosure. A rich literature documents that managers often exploit asymmetric information around capital market events—such as IPOs, seasoned equity offerings (SEOs), and debt issuances—to temporarily mislead investors through opportunistic disclosures, including accrual-based earnings management, real activities manipulation, and selective presentation of non-GAAP metrics (e.g., Teoh, Welch, and Wong (1998); Rangan (1998); Marquardt and Wiedman (2005); Cohen and Zarowin (2010); Cheng and Subramanyam (2008)). These practices are designed to inflate perceived firm performance, reduce cost of capital, or meet valuation thresholds at the time of issuance. While such strategies may yield short-term gains, markets tend to impose long-term costs once misreporting is revealed, with reputational penalties often exceeding any legal or regulatory sanctions (Karpoff, Lee, and Martin (2008a); Dechow, Sloan, and Sweeney (1995); Ecker, Francis, Kim, Olsson, and Schipper (2006)). This literature highlights the inherent tension between disclosure as a tool for reducing information asymmetry and its strategic use for opportunism in environments where verification is weak and monitoring is limited.

Our study builds on this literature by identifying and analyzing a novel form of strategic disclosure and information manipulation in the entrepreneurial setting. Specifically, we examine how and when code-washing influences fundraising success, the underlying drivers of this behavior, and its long-term implications. We do so by focusing on the distinction between "*code-producers*" and "*code-washers*."[1] Code-producers are ventures that genuinely share their source code and document ongoing developments, offering valuable transparency and meaningful insights into their technological capabilities. In contrast, code-washers superficially use open-source repositories to project an image of openness and innovation to appeal to investors and other stakeholders. These superficial practices—such as organizing minimal code in a repository, making trivial updates timed with fundraising efforts or other forms of 'window-dressing'—create a misleading perception of developmental progress and transparency. The practice mirrors classic information manipulation strategies in accounting and capital markets, including "cheap talk" in voluntary disclosure (Milgrom and Roberts, 1986), cosmetic reporting around IPOs (Lee, Shleifer, and Thaler, 1991), and greenwashing in ESG narratives (Cohen, Gurun, and Nguyen, 2020). While such strategies may yield short-term benefits during fundraising, they provide little substantive value to outsiders and can lead to long-term reputational costs.

While opening a GitHub repository is effectively costless, sustaining credible code activity over time is far from trivial. Public repositories expose a startup's codebase to scrutiny by a technically savvy community, including potential contributors, third-party developers, independent researchers, and specialized crypto media outlets. This

[1] Hereafter without quotes or italics.

public exposure imposes a significant discipline: it becomes risky to misrepresent development progress or submit low-quality code, as the broader community can evaluate, critique, or even publicly call out inconsistencies. For instance, in 2020, a project known as SushiSwap initially gained attention with aggressive marketing and GitHub visibility, but quickly faced backlash from blockchain developers and crypto analysts after independent experts flagged its code as hastily forked from Uniswap without adequate testing or security auditing.[2] Such episodes illustrate how public code exposure acts as a costly and credible signaling mechanism—and how startups engaging in superficial signaling face reputational consequences when their code is audited by third parties.

Although prior studies in the blockchain context (e.g., Amsden and Schweizer (2018), Fisch (2019), Momtaz (2020), Howell, Niessner, and Yermack (2020), Lyandres, Palazzo, and Rabetti (2022), and Davydiuk, Gupta, and Rosen (2023)) have documented that code activity plays a role in early-stage fundraising, it has typically been considered secondary to more prominent signals, such as founder commitment (e.g., skin-in-the-game), whitepaper quality, team characteristics, and social media presence. Outside the blockchain domain, Conti, Peukert, and Roche (2024) find that early-stage startups in the U.S. exhibit a significantly higher likelihood of securing funding after engaging with open-source communities on GitHub.[3] However, the strategic manipulation of code activity (code-washing) remains largely unexplored. Our study addresses this gap by not only advancing the accounting literature but also offering broader insights into how young ventures use open-source disclosures as a strategic signaling mechanism in environments characterized by high information asymmetry and the potential for misconduct akin to greenwashing.

To formalize these strategic dynamics, we develop a game-theoretic model in which blockchain startups possess private information about their technological quality and choose whether to engage in costly transparency or superficial signaling. Startups of high quality can credibly distinguish themselves by producing verifiable code at a higher cost—producing authentic coding and exposing it to public scrutiny, including competitors—whereas low-quality startups may attempt to mimic such signals at a lower cost by code-washing—code has little value to competitors (e.g., copied or dysfunctional). Investors observe signals (e.g., GitHub activity) but not the underlying type. They form beliefs and decide whether to fund a startup based on the perceived credibility of these signals. The model generates three key equilibria: a pooling equilibrium, where both types engage in superficial signaling and investors cannot distinguish them (likely in hot markets with high optimism and low attention); a separating equilibrium, where only high-quality startups produce meaningful code and low-quality startups opt out (more likely when reputational costs or scrutiny are high); and a dynamic refinement, where investor beliefs evolve through market feedback and reputational mechanisms eventually discipline deceptive behavior. This framework yields testable

[2] See, e.g., coverage by The Block (2020) and community discussions on Reddit and Twitter documenting developer concerns.

[3] Their analysis shows that the effect is particularly strong for firms developing novel technologies, while it is weaker for those using GitHub solely for internal development or operating in more competitive sectors.

predictions linking open-source activity to fundraising success, investor mispricing, and post-fundraising performance. These predictions echo foundational theories of signaling under asymmetric information (Spence (1973); Leland and Pyle (1977)).

We focus on the token offering setting (e.g., Howell et al. (2020)), a widely used fundraising method for startups in recent years.[4] This setting presents several advantages for our analysis. First, it provides granular fundraising data on thousands of young high-tech ventures globally. Additionally, it offers rich information on startup characteristics, product features, and investor engagement. Moreover, a highly asymmetric information environment creates strong incentives for information manipulation, enabling us to assess the impact of code-washing strategies. Finally, because blockchain startups are predominantly software-based, such as decentralized lending platforms, code production is a critical indicator of project development, offering a direct measure of innovation. These features create a quasi-laboratory environment for studying the interplay between open-source practices and fundraising success, the incentives to engage in information manipulation through code-washing, and related post-fundraising economic implications.

We begin our empirical analysis by exploring the relationship between blockchain startups' open-source decisions and fundraising success. Our findings show that having an open-source Github account is associated with a greater chance of successful fundraising (extensive margin) and larger amounts raised (intensive margin) during the fundraising phase. A percent increase in the number of commits recorded in GitHub is associated with a 0.31 percent increase in the total amount raised. This finding is also economically significant. Specifically, a one-standard deviation increase in code activity corresponds to an additional $18.86 million in funds raised by these startups. These results corroborate validating the relevance of open-source as a signaling mechanism during the fundraising phase of young ventures documented in the extant literature (e.g., Amsden and Schweizer (2018), Howell et al. (2020), Lyandres et al. (2022), Davydiuk et al. (2023), and Conti et al. (2024)).

Having validated the setting, we move next to explore the impact of code-washing on startups' fundraising success. We classify startups into two distinct groups: code-producers, those in the highest quartile of commit activity, and code-washers, those in the bottom quartile of commit activity. We follow prior studies in the blockchain literature in using the number of commits across all firm repositories as our primary proxy for open-source code production. The rationale is that a startup's code production represents a costly signal, as it involves the developers' effort to produce code (e.g., financial resources such as salaries, management, infrastructure, and time), making it difficult for opportunistic startups to mimic. In contrast, merely opening a GitHub account is costless. Therefore, startups with open-source accounts but minimal code production may be attempting to mislead early investors, particu-

[4] Blockchain-startups raised more than 30 billion dollars in token offerings (four times more than in venture capital markets) during our sample period. See section 4.1 for institutional background, and Lyandres and Rabetti (2024) for a literature review.

larly during booming markets when investors' attention is lower (e.g., Ritter (1984)). Note that code production may also reflect other factors such as startup size, business model, or prevailing market conditions. To mitigate confounding effects, we control for a range of startup characteristics, including the size of the development team, the number of tokens offered for sale, the presence of a fundraising hard cap, and whether the project implements Know-Your-Customer (KYC) procedures. We also account for business informativeness through whitepaper disclosures, community engagement across platforms such as Twitter, Reddit, Bitcointalk, and Medium, as well as financial capacity via indicators of early-stage fundraising rounds. In addition, all specifications include year-month, industry, and geographic fixed effects to control for time-varying trends, sector-specific dynamics, and jurisdictional differences that may influence code-washing behavior.

To validate our empirical proxy for code-washing, we conduct a series of tests consistent with construct validation practices in the accounting literature (e.g., Dechow et al. (1995); Healy and Wahlen (1999)). First, we document that suspicious coding behavior—characterized by skewed, clustered, or last-minute GitHub commits—is highly concentrated in the immediate pre-fundraising window, suggesting strategic timing rather than organic development. This mirrors evidence in accounting where real activities manipulation (e.g., overproduction or expense deferral) is timed around capital market events Cohen and Zarowin (2010). Second, we find that our proxy is negatively correlated with the informativeness of the firm's whitepaper, implying that code-washing tends to co-occur with intentionally opaque disclosures (e.g., cheap talk), much like firms that rely on vague MD&A or boilerplate language to obscure performance Li (2008). Third, we leverage an exogenous regulatory shock—the 2017 SEC crackdown on unregistered token sales—to examine firm behavior in dynamic equilibria. In the post-shock period, firms reduce code-washing and shift toward more substantive development activity, consistent with theoretical predictions that the cost of deception rises under increased scrutiny. Finally, we test whether code-washers produce less impactful code in *qualitative* terms than code-producers by analyzing the incidence of pull requests—a collaborative measure reflecting engagement by external developers—and issues, which capture bug-fixing discussions often used as proxies for commit quality (e.g., Fisch (2019)). Consistent with our proxies capturing meaningless or superficial coding, commits by code-washers are rarely associated with pull requests or issues, indicating limited collaboration and lower substantive contribution. Collectively, these validation tests strengthen the interpretation of our proxy as capturing opportunistic, superficial transparency rather than potential confounders such as startups' intrinsic development quality.

We start our analysis but examining the impact of code activity during the fundraising phase. We find that investors struggle to distinguish between code-producers and code-washers during hot markets, as both types of startups enjoy a higher likelihood of fundraising success compared to ventures without any open-source code presence. This pattern suggests that superficial signals of transparency—such as minimal or strategically timed GitHub activity—can be misinterpreted by investors as evidence of genuine innovation, resulting in capital misallocation. The economic implication is clear: during periods of exuberance or low attention, even low-quality firms can attract substantial funding by mimicking transparency, echoing dynamics seen in greenwashing, earnings management, and pro forma disclosure (Cohen et al., 2020; Healy and Wahlen, 1999; Bradshaw and Sloan, 2003). Much like inflated ESG narratives or cosmetic earnings disclosures around equity issuance (Teoh et al., 1998; Rangan, 1998), code-washing leverages the asymmetry between signal cost and investor verification capacity. Our findings point to a broader insight in the accounting literature: voluntary disclosure regimes are vulnerable to strategic manipulation when external monitoring is weak or absent (Ball, 2001), reinforcing the importance of credibility-enhancing mechanisms—such as audits, certification, or reputation systems—even in settings outside traditional financial reporting.

Since the effectiveness of code-washing depends on investors' ability (or lack thereof) to evaluate open-source code accurately, we investigate whether a separate equilibrium emerges when the quality of market information improves. In other words, effective signals must be sufficiently costly for code-washers to mimic, enabling code-producers to differentiate themselves (e.g., Spence (1973) and Miller and Rock (1985)). Notably, during market downturns and periods of high volatility, investors tend to become more risk-averse as uncertainty rises (Gennotte and Leland (1990); Froot and Obstfeld (1991)). Investors' heightened risk aversion is likely to make them more discerning and selective in their funding decisions. To formally assess whether reduced information asymmetry enhances investor ability to distinguish between code-washers and code-producers, we conduct a series of subsample analyses. Specifically, we re-estimate the baseline fundraising model within two key contexts: (i) settings characterized by high startup information quality and (ii) periods of heightened investor attention.

In the first set of tests, we construct two proxies for overall information availability. The *high information coverage* proxy classifies startups as high coverage if they appear in a greater-than-median number of third-party data sources during the fundraising phase. The *high information quality* proxy builds on Lyandres et al. (2022) and accounts for both the number of sources and the internal consistency of reported information across them. We then re-run our core regressions within each of these subsamples to evaluate whether the coefficient on *code-producer* remains significantly positive while the coefficient on *code-washer* becomes insignificant. The results align with this prediction: in both high-information environments, code-producers are significantly more likely to raise capital, while code-washers are not. Moreover, the difference in effect size between the two is statistically significant based on Wald tests, indicating that investors are indeed more capable of distinguishing between the two types when informational frictions are lower.

In a complementary set of tests, we examine whether investor attention—as proxied by cryptocurrency market sentiment—similarly facilitates discernment. Following Derrien (2005), we use monthly Ether (ETH) market returns as a proxy for investor attention. We define *low ETH return* periods as those below the sample median and interpret them as times when investor optimism is subdued and scrutiny is heightened. Re-estimating our baseline equation within these subsamples, we again find that only code-producers benefit from a significant fundraising advantage. The coefficient on *code-producer* is consistently positive and significant, while that on *code-washer* is not. The Wald test confirms the statistical difference between the two effects in most specifications. These findings suggest that, during market downturns, investors become more focused on fundamentals—such as code quality, transparency, and team credentials—and are thus better able to identify genuine signals of project viability.

Together, these analyses reinforce the hypothesis that the efficacy of code-washing as a signaling device diminishes when information asymmetry is lower or investor attention is higher. Under such conditions, separating equilibria emerge: high-quality startups credibly signal their type through meaningful code contributions, while low-quality startups are unable to mimic without detection.

We now move to assessing whether the short-term benefits of code-washing lead to long-term reputational harm for blockchain startups. Our analysis focuses on three aspects of a startup's long-term performance: exchange listings, financial outcomes, and technological innovation. The results reveal that code-washers are less likely to have their tokens listed on exchanges compared to code-producers. They also fail to deliver future token returns, exhibit lower return volatility, and face higher illiquidity. This indicates that investors tend to redirect their funds toward code-producing startups over time (i.e., market learning). We find that while code-producers experience a three-fold increase in code activity after fundraising, code-washers show a significant decline, reflecting their inability to create genuine products and services—revealing their strategic decision to engage in code-washing. These patterns suggest that investors are initially misled, but the truth emerges over time, consistent with the market learning and disciplining mechanism documented in the literature (Karpoff et al. (2008a); Teoh et al. (1998)).

To complement our long-term performance analysis, we examine the buy-and-hold returns (BHRs) of portfolios formed by code-washers and code-producers over the 18 months following exchange listing events. The results show that code-producers significantly outperform, with equal-weighted portfolio BHRs exceeding 600%, compared to negative returns for code-washers. This suggests that the market's inability to accurately value the innovation of code-producers underprices these firms at the fundraising stage, but also provides an opportunity for investors to earn abnormal returns by identifying and investing in firms with high code quality. In contrast, code-washers, initially overvalued during the fundraising stage, exhibit a reversal pattern over time. For these firms, the absence of substantial innovation behind code-washing results in market corrections and penalization as investors learn to reassess their true value in the long run.

To ensure that our findings are not driven by endogeneity, omitted variables, or model specification, we conduct a comprehensive set of additional robustness checks. First, we employ several alternative measures of open-source engagement, including a conservative indicator for zero-commit repositories, code revisions following Fisch (2019), the continuous form of the ratio of commits during the fundraising phase over total commits at the end of the fundraising phase, and more discretionary dimensions such as pull requests—which reflect community-driven feedback. Our results remain unchanged for all alternative specifications.

Second, although we control for an extensive set of observable startup characteristics, a potential concern remains that startups without a GitHub presence may differ systematically in unobserved ways from those with such accounts. To address this, we implement three empirical strategies. First, we alleviate concerns with heterogeneous controls by also conditioning our analysis on a subsample of open-sourced startups. Additionally, we apply a Heckman two-step correction (Heckman (1990)) to account for potential sample selection bias, recognizing that open-source startups may not be randomly drawn from the broader population. Finally, we construct matched samples of code-washers and control startups with similar observable characteristics to mitigate the influence of latent differences. Our results resist empirical strategies aimed at mitigating selection concerns.

Third, we further address the concern that our proxies may capture broader constructs—such as general startup quality—rather than specifically identifying misconduct behavior like code-washing. To test this, we examine whether code-washing correlates with other forms of low-cost signaling, or "cheap talk." Analyzing whitepaper content, we find that code-washers tend to produce more verbose documents, characterized by a greater number of repeated words, technical jargon, and visual elements. This pattern suggests that code-washers are more inclined to rely on superficial disclosures compared to code-producers. To further validate that code-washing captures startups' intention to influence investors rather than their intrinsic quality, we exploit a 2018 regulatory intervention by the U.S. Securities and Exchange Commission (SEC), which introduced heightened scrutiny over token offerings. While this policy plausibly reduced the incentives for opportunistic signaling, it should not have directly affected underlying startup quality. Consistent with this interpretation, our difference-in-differences analysis shows that code-washers experienced a statistically significant 18.3% decline in the likelihood of successful fundraising following the SEC's crackdown. These results suggest a shift in investor behavior toward more credible and transparent ventures. They are also consistent with the emergence of a separating equilibrium, in which code-washing becomes less effective as a signaling device due to its rising cost under increased scrutiny.

Our study provides several contributions. First, we extend a foundational stream in the accounting literature that examines how firms manipulate disclosures during equity issuance events to influence investor perceptions. Prior research shows that managers often inflate earnings, structure voluntary disclosures strategically, or window-dress financial reports to enhance valuations around IPOs and SEOs (Teoh et al. (1998); Rangan (1998); Healy and Wahlen (1999)). While most of this literature focuses on traditional financial statements or MD&A narratives, we document similar opportunistic behavior in a novel, non-financial disclosure setting: open-source software development. Our context of blockchain-based fundraising offers a natural laboratory to study voluntary, yet economically salient, forms of disclosure manipulation that mirror classic reporting games. Startups inflate superficial metrics of transparency (e.g., minimal GitHub activity) to signal innovation and attract funding. This form of misreporting is conceptually akin to cosmetic earnings management—costly to verify, cheap to fake, and difficult for unsophisticated investors to detect. We show that while such strategies help secure financing during hot markets, they result in negative investor outcomes and long-term reputational harm, consistent with the disciplining effects documented in the accounting literature on misreporting and investor learning.

Second, our study contributes to the literature on voluntary disclosure and signaling under asymmetric information (Healy and Wahlen (1999); Spence (1973); Leland and Pyle (1977)). We show that even in settings lacking traditional financial statements or auditors, the fundamental economics of disclosure credibility apply: when signals are cheap to imitate and costly to verify, market participants struggle to distinguish substance from mimicry. Open-source activity—like MD&A language or ESG reporting—can be strategically manipulated to influence perception, particularly when investor scrutiny is low. In this context, we highlight how a separating equilibrium only emerges under heightened investor attention or increased reputational costs, aligning with core predictions in signaling theory.

Third, we contribute to the emerging field of forensic finance by developing novel empirical proxies to detect strategic misrepresentation in nontraditional disclosure environments (Griffin and Kruger (2024)). Code-washing shares key features with other forms of financial misconduct—such as fake reviews (Luca and Zervas (2016); Mayzlin, Dover, and Chevalier (2014)), snow inflation by ski resorts (Zinman and Zitzewitz (2016)), wash trading in crypto exchanges (Amiram, Lyandres, and Rabetti (2024)), or inflated credit ratings (Becker and Milbourn (2011))—where firms engage in low-cost deception that can be profitable in the short term but harmful in the long run. Our framework and methodology offer new tools for detecting deception in digital and tech-intensive settings, with implications for regulators and auditors seeking to monitor firm behavior outside of standard financial reporting channels.

Finally, we contribute to a growing body of work on investor inattention and the valuation of innovation (Cohen, Diether, and Malloy (2013b); Hirshleifer, Hsu, and Li (2018); Shu, Tian, and Zhan (2022)). Like the literature documenting mispricing of innovative firms due to investor difficulty in interpreting R&D, patents, or ESG signals (Cohen, Gurun, and Kominers (2019); Cohen et al. (2020)), we show that investors struggle to accurately interpret open-source innovation. Startups that generate superficial signals are initially rewarded alongside genuine innovators. However, those that produce high-quality, verifiable code ultimately outperform, suggesting that market learning eventually restores pricing efficiency—a dynamic consistent with theories of post-disclosure market correction in accounting and finance.

The remainder of this study is structured as follows. Section 2 motivates and formalizes the signaling mechanism, presenting four core hypotheses grounded in information economics. Section 3 develops a game-theoretic model capturing the strategic incentives behind transparency and code-washing. Section 4 introduces the institutional setting and describes the construction of the dataset, including GitHub activity, fundraising outcomes, and post-launch performance. Section 5 presents the empirical analysis, testing the hypotheses across different market environments and validating results through robustness checks and regulatory shocks. Section 6 concludes by discussing implications for investors, regulators, and the entrepreneurial ecosystem more broadly.

2 Signaling Mechanism

2.1 Transparency as a Signal

High-quality startups may attempt to signal their quality to distinguish themselves from the pool of low-quality startups, especially in high-information asymmetry markets (Akerlof (1970), Modigliani and Miller (1958), Leland and Pyle (1977), Connelly, Certo, Ireland, and Reutzel (2011), and Lyandres et al. (2022)). Open-source code sharing on platforms like GitHub is one way to signal their commitment and capability in developing the projects.[5] The intrinsic complexity of blockchain projects, coupled with an unregulated environment and global investor's reach, complicates the certification of a firm's type. Additionally, although monitoring by certifying agents, such as underwriters, rating agencies, auditors, and venture capitalists, can help mitigate informational asymmetry (Diamond (1984), Megginson and Weiss (1991), Holmström and Tirole (1997), Besanko and Kanatas (1993), Admati and Pfleiderer (1994)), these third parties informational agents are often absent, have a conflict of interest preventing them from serving as effective monitoring agents, or even embroiled by financial misconduct themselves (Barth, Laturnus, Mansouri, and Wagner (2023)).

Therefore, coding platforms like GitHub may provide a way to certify the quality of a blockchain project (e.g., Amsden and Schweizer (2018), Howell et al. (2020), Lyandres et al. (2022), and Davydiuk et al. (2023)). The transparency from the openness of developing activities may give investors confidence in a startup's claims, addressing concerns about overpromising or delivering incomplete projects. In the inherently speculative crypto markets, such transparency may increase the chances of successful fundraising. Moreover, blockchain's technical complexity makes external project vetting quite tricky without access to evolving source code—precisely what transparency remedies provide (Cong, Prasad, and Rabetti (2024)). Open-source code sharing through platforms like GitHub

[5] See more on signaling theory in information economics in Spence (1973) and Rothschild and Stiglitz (1976), which is also prevalent in the IPO context (e.g., Wilson, 1977; Leland and Pyle, 1977; Brau and Fawcett, 2006).

can offer this transparency, helping to overcome the challenges of asymmetrically distributed expertise that are inevitably present during the review processes of early-stage firms (e.g., Amsden and Schweizer (2018), Fisch (2019), Momtaz (2020), Howell et al. (2020), Lyandres et al. (2022), and Davydiuk et al. (2023)).

2.2 Pooling in Hot Markets

On the other hand, information manipulation may be a rational strategy in a setting characterized by asymmetric information (e.g., Frankel and Kartik (2019), Ball (2022), and Sun (2022)). Since naive investors may not clearly understand code activity, startups could inflate coding on GitHub to signal fake quality. Opening an account on the platform and engaging in some coding activity around upcoming fundraising, if done strategically, may mislead potential investors about the project's future development. Given the lack of verifiable information at that point, investors may mistake this superficial activity for deeper and more meaningful work. This leads to investors being unable to distinguish between true signals of quality and mere attempts to manipulate the market. As the actual development is often blurry in these formative stages, where experimentation and failure rates are highly likely, we conjecture that a pooling equilibrium exists when the observed signal reveals no additional information about the blockchain startup's type (Bolton and Dewatripont (2005), MasColell, Whinston, and Green (1995)).

The situation described above is likely aggravated when investors' sentiment is highly optimistic (e.g., Loughran, Ritter, and Rydqvist (1994) and Lee et al. (1991)). For instance, Ljungqvist, Nanda, and Singh (2006) suggests that as investor sentiment grows, IPO offer size increases and lower-quality companies take public, decreasing average issuer quality.[6] During market booms, even startups of lower quality may be able to successfully raise funds, as investors' risk aversion tends to decrease and their appetite for new investment opportunities increases (Ritter and Welch (2002)). Thus, we hypothesize that low-quality blockchain startups may also succeed in fund-raising from a demand-side analysis: investors are less capable or less aware of the existence of low-quality startups when in a hot market, and high-quality startups would find it hard to distinguish themselves in the pool. Therefore, a natural conjecture is that a non-trivial number of blockchain startups pretend to be transparent but are instead engaging in code-washing, i.e., attempting to use GitHub as a signaling device to self-promote their business as open-source, leading to our first hypothesis:

H1: Investors are unable to distinguish between code-washers and code-producers during the fundraising phase.

[6] Information manipulation is prevalent in many settings. See, e.g., Luca and Zervas (2016) for the case of fake reviews of restaurants, Mayzlin et al. (2014) for the case of fake reviews of hotels, Zinman and Zitzewitz (2016) for the case of misreporting the amount of snow by ski resorts, and Becker and Milbourn (2011) for the case of credit rating inflation.

2.3 Separating with Attention

In contrast, a separating equilibrium, where startups' signaling choices can reveal their underlying quality, is more likely to occur when information asymmetry between entrepreneurs and investors is relatively low (MasColell et al. (1995), Bolton and Dewatripont (2005)). Especially when more available data and information resources exist, or information across different sources is consistent (Lyandres et al. (2022)), the investors can better distinguish the types of high-quality and low-quality startups. In other words, effective signals must be sufficiently costly for low-quality firms to mimic, enabling high-quality firms to differentiate themselves (e.g., Spence (1973) and Miller and Rock (1985)). Notably, during market downturns and periods of high volatility, investors tend to become more risk-averse as uncertainty rises (Gennotte and Leland (1990); Froot and Obstfeld (1991)). Investors' heightened risk aversion will likely make them more discerning and selective in their funding decisions. With that, our third hypothesis emerges:

H2: Investors can distinguish between code-washers and code-producers when asymmetric information is low and/or investor attention is high.

2.3.1 Reputational Costs

Although investors may not see code-washing practices during the fundraising phase in the short run, code-washers are likely revealed in the long run by signaling distrust in their inability to deliver promised products or services. Naturally, a blockchain firm's decision regarding the extent to which it attempts to mislead investors through code-washing involves a trade-off between higher short-term gains (e.g., raising more funds) and lower future profits due to a damaged reputation since the market can gradually discover misrepresentation of information (e.g., Karpoff, Lee, and Martin (2008b)). For instance, Amiram et al. (2024) finds that crypto exchanges engaged in volume inflation succeed in attracting demand in the short term but are later punished by traders due to damaged reputation as the real trading quality of these exchanges is revealed in the long run. Accordingly, our last hypothesis concludes:

H3: Code-washers are more likely to underperform than code-producers in the post-fundraising period.

3 Model

We formalize the signaling dynamics faced by blockchain startups during early-stage fundraising through a simple game-theoretic model. This framework allows us to examine how firms with private information about their true quality strategically communicate with outside investors through observable but potentially manipulable signals, such as activity on open-source coding platforms. The model captures both the incentives of startups to signal their quality credibly and the difficulty faced by investors in distinguishing genuine transparency from superficial mimicry.

Players. There are two types of blockchain startups:

- High-quality startups (H), which possess strong technological capabilities, dedicated development teams, and genuine plans to deliver functioning products.
- Low-quality startups (L), which lack substantive technological foundations or are opportunistic, aiming to capitalize on hype cycles without delivering long-term value.

The type of each startup is private information, known only to the startup. Investors cannot directly observe type, but they do observe signals generated by the startup. We model a representative investor who must decide whether to fund a startup based on its observable signal. The investor holds a prior belief $\pi \in (0, 1)$ that any given startup is of high quality.

Signal choices. Each startup chooses one of two possible signals:

- T: **Transparency**. This involves genuine open-source development activity. It may include meaningful code contributions, version updates, integration testing, and developer interactions, all publicly verifiable via platforms like GitHub.
- C: **Code-washing**. This entails superficial or manipulative signaling, such as placeholder repositories, empty commits, or scripted bot activity designed to feign active development.

Transparency (T) is more costly to produce, as it requires actual progress and engineering effort and also exposes genuine code to competitors. In contrast, code-washing (C) is inexpensive but potentially deceptive.

Sequence of the game. The model unfolds in the following sequence:

1. Nature draws the startup's type $\theta \in \{H, L\}$ with probability π for H and $1 - \pi$ for L.

2. The startup chooses a signal $s \in \{T, C\}$.

3. The investor observes s, updates their beliefs $\mu(\theta|s)$ using Bayes' rule, and decides whether to provide funding $F(s) \in \{0, F^*\}$.

4. Payoffs are realized for both the startup and the investor.

Payoffs. The utility of each player depends on the signal choice and investor action:

- **Startup of type θ:**

$$U_\theta(s) = \begin{cases} F^* - c_T & \text{if } s = T \\ F^* - c_C - \delta_\theta & \text{if } s = C \end{cases}$$

  where $c_T > c_C$ are the costs associated with signaling via transparency and code-washing, respectively. For high-quality startups, $\delta_H = 0$ as they have nothing to hide. Low-quality startups incur a reputational penalty $\delta_L > 0$ if they engage in code-washing and are later revealed to be deceptive.

- **Investor:**

$$\mathbb{E}[R|s] = \mu(H|s)R_H + (1 - \mu(H|s))R_L$$

  The investor chooses to fund the startup if the expected return from the investment exceeds the cost of funding, i.e., if $\mathbb{E}[R|s] \geq F^*$.

Equilibrium analysis. We analyze three types of equilibria that can emerge from this setting:

- **Pooling equilibrium (H1):** Both high- and low-quality startups choose the same signal, typically code-washing (C). This equilibrium arises when the cost of genuine transparency is high relative to the perceived benefit, and the reputational risk to low-quality startups is low (e.g., in euphoric markets). In this setting, the signal loses informativeness, and the investor cannot distinguish types:

$$F^* - c_C - \delta_L > -c_T \quad \text{and} \quad F^* - c_C \geq F^* - c_T$$

- **Separating equilibrium (H2):** High-quality startups choose transparency (T) to distinguish themselves, while low-quality startups choose code-washing (C). This equilibrium requires that the reputational cost for misrepresentation is sufficiently large, and that the benefits of honest signaling outweigh the higher cost for high-quality firms:

$$F^* - c_T > F^* - c_C \quad \text{and} \quad F^* - c_C - \delta_L > -c_T$$

- **Dynamic refinement (H3):** Over time, repeated exposure and post-funding performance allow investors to update beliefs, making code-washing increasingly risky for low-quality firms. Even if a pooling equilibrium prevails in the short run, market feedback can lead to separation in the long run as the reputation mechanism penalizes deceptive behavior:

$$\delta_L \uparrow \quad \Rightarrow \quad \text{Code-washing less attractive over time}$$

Numerical example. To illustrate these dynamics concretely, consider the following parameterization:

$$F^* = 100, \quad R_H = 150, \quad R_L = 50, \quad c_T = 20, \quad c_C = 5, \quad \delta_L = 30, \quad \pi = 0.6$$

- **Pooling equilibrium:** Suppose both types choose C. The investor's expected return is:

$$\mathbb{E}[R|C] = 0.6 \cdot 150 + 0.4 \cdot 50 = 110 > 100 = F^*$$

  The investor is willing to fund. This sustains a pooling equilibrium because the signal does not differentiate quality, yet both types benefit.

- **Separating equilibrium fails:** For H to choose T, it must receive higher utility:

$$U_H(T) = 100 - 20 = 80, \quad U_H(C) = 100 - 5 = 95$$

  The high-quality startup strictly prefers code-washing, so separation fails.

- **More skeptical market (lower $\pi = 0.3$):** Now, the expected return becomes:

$$\mathbb{E}[R|C] = 0.3 \cdot 150 + 0.7 \cdot 50 = 80 < 100$$

  The investor refuses to fund based on code-washing signals. Only genuine transparency can now elicit funding, allowing H to separate by choosing T.

This game-theoretic framework illustrates the economic incentives shaping blockchain startup behavior in early-stage fundraising. In optimistic markets, low-quality firms can mimic transparency through code-washing and still raise capital, resulting in pooling. However, as investor scrutiny intensifies or reputational costs rise (e.g., after scandals or failed projects), high-quality firms can credibly separate by engaging in real transparency, while low-quality firms find it harder to fake their signal without being penalized.

3.1 Dynamic Signaling Model with Reputation

We extend the static model in Section 3 to a dynamic setting that more realistic reflects the strategic environment of blockchain startups. In particular, we model two periods: the first captures the fundraising decision under asymmetric information, while the second reflects the consequences of signaling choices via reputational feedback and learning.

Timeline.

- **Period 1.** Nature draws startup type $\theta \in \{H, L\}$ with probability $\pi = \Pr(H)$. The startup chooses a signal $s \in \{T, C\}$:
  - T: transparent and verifiable open-source development;
  - C: code-washing (superficial signaling, e.g., dummy commits).

  Investors observe s, update beliefs $\mu(\theta|s)$, and choose whether to fund: $F(s) \in \{0, F^*\}$.

- **Period 2.** Startups that received funding produce a product, which may or may not succeed based on true type:

$$\text{Success probability: } \begin{cases} p_H & \text{if } \theta = H \\ p_L & \text{if } \theta = L, \quad p_H > p_L \end{cases}$$

  Investors observe success/failure and update beliefs about the startup's true quality. Reputational payoffs (or penalties) are realized accordingly.

Payoffs.

- **Startup (Type θ):**

$$U_\theta(s) = \begin{cases} F^* - c_T + \phi_\theta^T \cdot 1_{\text{success}} & \text{if } s = T \\ F^* - c_C - \delta_\theta + \phi_\theta^C \cdot 1_{\text{success}} & \text{if } s = C \end{cases}$$

where ϕ_θ^s captures future value conditional on success and signal s. Low-quality startups face a reputation penalty $\delta_L > 0$ if exposed as code-washers.

- **Investor:**

$$\mathbb{E}[R(s)] = \mu(H|s) \cdot R_H + (1 - \mu(H|s)) \cdot R_L - F^*$$

where R_H, R_L are returns conditional on success.

Key Dynamic Features.

1. Strategic Misrepresentation and Learning. In period 1, low-quality startups may mimic high-quality behavior if short-term gains outweigh expected long-term reputational penalties. Over time, however, failures in period 2 reveal the startup type. This structure captures the dynamic tension between short-term opportunism and long-term trust.

2. Investor Updating and Feedback. Investor beliefs evolve based on signal s and eventual project outcome. If a project signaled transparency but fails, beliefs about similar future startups shift downward. This allows for Bayesian learning:

$$\mu_{t+1}(H|T, \text{fail}) < \mu_t(H|T)$$

Signaling strategies and investor expectations co-evolve over time.

3. Endogenous Reputation Cost. The penalty δ_L is endogenous: it increases in the probability of being revealed post-fundraising:

$$\delta_L = \rho \cdot (1 - p_L) \cdot \Phi$$

where ρ is investor attention and Φ is the market discount applied to revealed misrepresentation. As investors become more discerning (e.g., in cold markets), ρ increases and discourages code-washing.

Numerical Example. Let:

$$F^* = 100, \quad c_T = 20, \quad c_C = 5, \quad \pi = 0.5, \quad p_H = 0.9, \quad p_L = 0.3, \quad \Phi = 80, \quad \rho = 0.5$$

Then:

$$\delta_L = 0.5 \cdot (1 - 0.3) \cdot 80 = 28$$

Startups evaluate:

$$U_H(T) = 100 - 20 + 0.9 \cdot 50 = 125, \quad U_H(C) = 100 - 5 + 0.9 \cdot 30 = 122$$

$$U_L(T) = -20 + 0.3 \cdot 10 = -17, \quad U_L(C) = 100 - 5 - 28 + 0.3 \cdot 10 = 70$$

Interpretation:

- High-quality startups prefer transparency: $U_H(T) > U_H(C)$.
- Low-quality startups prefer code-washing: $U_L(C) > U_L(T)$.

$\Rightarrow$ **Separating equilibrium** arises when investors are discerning and reputational risk is internalized.

Model Implications. The model implications are consistent with those discussed in Section 2, that is:

- **Framework (Transparency as a Signal):** In early stages, high-quality startups use transparency to credibly distinguish themselves (e.g., Amsden and Schweizer (2018), Howell et al. (2020), Lyandres et al. (2022), Davydiuk et al. (2023), and Conti et al. (2024)).
- **H1 (Pooling in Hot Markets):** When investor optimism (π or ρ low) is high, code-washing may go undetected, leading to pooling equilibria.
- **H2 (Separating with Attention):** With greater scrutiny and risk aversion, the cost of misrepresentation rises, enabling a separating equilibrium.
- **H3 (Reputational Costs):** Even if deception pays off in the short run, misrepresentation leads to lower second-period payoffs due to reputational damage.

Appendix C further extended the model to account for strategic intertemporal trade-offs, signal noise, reputation, and macro market forces.

4 Setting and Data

4.1 The blockchain-startup setting

Token offering (TO) is a new fundraising method where blockchain-based ventures sell their crypto tokens instead of traditional financial instruments like equity or debt.[7] These tokens can then later be sold in secondary markets such as crypto exchanges or used in exchange for services the blockchain startup provides (e.g., access to a digital platform). Since the first token offering in 2013, the market has grown significantly, with over 7,000 fundraising attempts by startups worldwide, raising more than $30 billion (Lyandres et al. (2022)). Many of these fundraising attempts occurred in 2018 when startups raised $20 billion. However, the market has since cooled due to substantial risks, including increased regulatory scrutiny from bodies like the SEC, heightened investor awareness of financial scams, and the emergence of other token issuance types, such as security and utility offerings.

Despite these challenges, token offerings presented a potential game changer for blockchain-based startups needing funding. However, the lack of regulation and the intangible nature of most blockchain-funded projects expose investors to significant risks. As a result, the success of fundraising efforts often depends on several signaling factors, including the credibility and reputation of the project team (Fisch (2019)), a well-crafted white paper that explains the project in detail and clearly outlines token distribution plans (Momtaz (2020)), social media activity (Lyandres et al. (2022)), skin-in-the-game commitments (e.g., Davydiuk et al. (2023), Chod and Lyandres (2021), and Gan, Tsoukalas, and Netessine (2021)), and prevailing market conditions (Amsden and Schweizer (2018)).[8] These signaling devices are voluntary but can disclose material information on the startup's performance.

Once listed on exchanges, the token prices of blockchain startups often show considerable volatility. For example, during early periods of TOs activity, Benedetti and Kostovetsky (2021) and Lee, Li, and Shin (2021) report average token offering returns of 179% and 112%, respectively. However, Lyandres et al. (2022) notes that these high average returns are driven by a small number of startups with exceptionally high returns, typically smaller ventures. Additionally, they find that approximately 60% of startups that successfully raise funds are never listed on a crypto exchange. As blockchain startups primarily target digital platforms, voluntarily disclosing the quantity and quality of their code production can enhance credibility by signaling the authenticity of their product development efforts. Although prior studies (e.g., Amsden and Schweizer (2018), Howell et al. (2020), and Lyandres et al. (2022))

[7] See Lyandres and Rabetti (2024) for a review of this market.

[8] Other studies (e.g., Benedetti and Kostovetsky (2021) and Howell et al. (2020)) also emphasize the importance of social media channels in the success of fundraising.

have suggested that code activity is a relevant factor during the fundraising phase, it has generally been viewed as secondary to other signals provided by blockchain firms.

Outside the blockchain domain, recent evidence shows a positive association between GitHub community engagement and fundraising outcomes in the U.S. startup ecosystem (Conti et al. (2024)). However, the extent to which startups strategically manipulate code activity—so-called "code-washing"—to influence fundraising success, the underlying drivers of this behavior, and its long-term implications remain underexplored. Our study addresses this gap by not only contributing to the blockchain literature but also offering broader insights into the strategic use of open-source code releases by young ventures, particularly in contexts where information asymmetry invites financial misconduct akin to greenwashing.

4.2 Collection and processing

To conduct this study, we compile a comprehensive dataset that includes information on fundraising, characteristics of blockchain startups, their code production, and performance after launch.

Our first empirical challenge is related to data collection. Fundraising data on blockchain startups is scattered across multiple online sources, with each source capturing only parts of the overall fundraising landscape (see Lyandres et al. (2022) for an extended discussion). The details can vary significantly even when different sources cover the same fundraising event. To address these challenges, we gather data from eleven fundraising aggregator websites to encompass nearly the entire fundraising landscape, following Lyandres et al. (2022) for data quality processing.

The data is compiled in two stages. First, we gather information from eleven blockchain startup fundraising websites to ensure comprehensive coverage, as each aggregator may only provide a portion of the data. We selected these aggregators based on their popularity, which was determined by the average historical web traffic during our sample period. The chosen websites are *Etherscan.io*, *CoinDesk.com*, *CoinGecko.com*, *CryptoCompare.com*, *ICObench.com*, *ICODrops.com*, *ICORating.com*, *ICOmarks.io*, *ICOdata.io*, *FoundICO.com*, and *TokenData.io*. Our final dataset includes 7,273 unique blockchain startups from over 100 countries that conducted fundraising attempts between 2013 and 2020, totaling nearly $33 billion in funds raised. The distribution of fundraising amounts by year is illustrated in Figure 1. As observed in the figure, the largest portion of funds raised occurred in 2018. This gradually declined toward the end of the period due to the rise of alternative fundraising methods, such as security offerings and exchange offerings, increased regulatory scrutiny, and crypto market frictions, which culminated in the Bitcoin winter following the pre-Covid phase.

Additionally, we collect other details about blockchain startups, including whether investors are required to register in advance (known as a "whitelist") and the occurrence of pre-fundraising rounds to angel investors or venture capital. We also gather information on the number of team members, industry type, and headquarters location. To identify potential code-washing behavior in blockchain startups, we collect more than one terabyte of project repository data from GitHub, the largest open-source platform, including rich information about code productivity, such as commits, reviews, comments, developers' details, and pull requests.

To examine the long-term implications of code-washing, we gather data on post-launch blockchain startups' performance in at least four areas. First, we collect data for post-launch token returns from *CoinMarketCap.com*. We use it to measure the financial performance of blockchain startups. We construct initial returns by collecting post-listing daily token prices and volume data. Second, we focus on on-chain activity, which helps measure blockchain startups' ongoing development and popularity. We collect data on the number of wallets and transactions from *Ethplorer.io*, which tracks the time-series evolution of distinct cryptographic wallets holding the tokens and the cumulative on-chain transactions involving the tokens. Additionally, we gather data from *Kaiko.com* on the off-chain trading volume of the blockchain startups' traded tokens. Third, we collect data on social media activities as these reflect the ongoing public attention to blockchain startups and the projects they fund. To analyze the time-series evolution of social media coverage, we monitor four popular channels used by blockchain startups during their fundraising phase: *Twitter*, *Reddit*, *Medium*, and *Bitcointalk*. Finally, to analyze financial misconduct, we collect data from *deadcoin.com,* which contains several hundred blockchain startups flagged with fundraising exit scams.[9]

4.3 Summary statistics

Table 1 presents the summary statistics of the data sample used in this study. We describe the definitions of variables in A1. Conditioned on the existence of all relevant variables, our sample includes 2,326 unique blockchain startups, with two-thirds of them having at least one GitHub repository at the end of the fundraising period. Among the startups with one or more GitHub repositories, the average number of technical commits recorded in their GitHub repositories is 1,015.10 at the end of the fundraising period. The coefficient of variation for *Code Production (log)* is 143.3% (calculated as 2.48/1.73), indicating high variability in code production among these startups.

Regarding fundraising performance, over 50% of startups in our sample have successfully raised funds. Conditional on raising money, each startup raises an average of 11.68 million dollars, reaching 23.62% to *Hardcap (log)* on average—the maximum amount allowed to be raised. In 68% of the fundraising events, ventures attempt a presale (i.e., by angel, venture capital, and other early seed investors) before the fundraising stage. The average ratio of token supply to tokens for sale is 57%. More than half of the blockchain startups in our sample offer a whitelist to early investors. And 68% of startups have implemented *KYC* requirements where the project complies with the "know

[9]This comprehensive dataset on blockchain fundraising attempts is publicly available in our repositories.

your customer." Startups with whitelist and KYC demonstrate greater commitment during fundraising by fostering trust with investors and ensuring the involvement of legitimate participants. These measures also help startups comply with regulatory requirements. 47% of blockchain startups in our sample have a white paper outlining the project's goals, technology, use cases, and business model. The average team number is 11.63 (including founders and other key employees), indicating that these early-stage ventures are extremely small during the fundraising phase.

Social media channels are highly relevant in the blockchain startup space, as these ventures are often driven by specialized communities centered around the startup's digital platform. Since this market is predominantly retail-driven, social media also serves as a proxy for investor engagement. Accordingly, we include information extracted from key social media platforms (i.e., Twitter's tweets, Reddit's discussions, Medium's articles, and Bitcointalk's posts). These platforms play a critical role in informing and engaging potential investors. *Twitter (log)* exhibits a mean of 2.82 with a standard deviation of 2.24, indicating that it is widely utilized as a communication tool in fundraising campaigns. In contrast, *Reddit (log)* has a lower mean of 1.49 and a standard deviation of 1.69, reflecting its more niche role, likely geared toward fostering community-driven discussions. Unlike Twitter, where the startup primarily drives content, Reddit content is largely shaped by the communities interested in the startup's digital platform, providing a more nuanced channel to gauge investor sentiment. *Medium (log)*, with a mean of 0.65 and a standard deviation of 1.30, is used less extensively. While Medium articles are among the most informative pieces across social media platforms, only a minority of startups prioritize in-depth content sharing on this channel. Startups that do utilize Medium effectively, however, are likely more appealing to investors seeking comprehensive information. Finally, *BTCTalk (log)*, with the highest mean of 3.06 and a standard deviation of 2.73, emerges as a crucial platform for fundraising success in the blockchain space. Its focus on cryptocurrency and blockchain communities makes it an essential channel for credibility-building and fostering investor trust.

Since our analysis focuses on code-washers—startups in the bottom quartile of cumulative commits generated before the fundraising stage—we provide further details regarding the variables used in this study for this type of startup and compare them with the baseline group. The control group is constructed at a 3:1 ratio by matching each code-washer with the closest propensity score, estimated using *Tokens for sale (%)*, *Hardcap (log)*, *Whitelist*, *KYC*, *White paper*, *Team size (log)*, *Presale*, *Twitter (log)*, *Reddit (log)*, *BTCTalk (log)*, and *Medium (log)*. After controlling for other variables, our sample includes 313 unique code-washer startups and 236 code-producers. Among all blockchain startups with at least one commit at the GitHub repository (before matching control variables), we rank them into quartiles based on commit count before the fundraising end date. This yields 578 code-washers in the bottom quartile and 505 code-producers in the top quartile. After matching control variables in the regressions, we finally obtained 313 unique code-washer startups and 236 code-producers.

Appendix Table A3 reports the summary statistics for code-washers and code-producers. On average, a code-washer produces 0.14 commits on GitHub before the fundraising stage, while a code-producer produces a significantly higher number of commits, 3503.64 commits on average. Apart from the significant differences in code production, code-washers and code-producers are indistinguishable in several characteristics, such as the probability of raising funds, the amount raised, hardcap amounts, percentages of tokens for sale, team size, KYC requirements, and the disclosure of a white paper. An exception is the use of social media channels, which is more pronounced for code-producers than for code-washers in the unmatched sample. However, these differences are eliminated through the matching procedure, suggesting a more balanced sample that we use as one of our several robustness checks.

5 Empirical Results

5.1 Open-source and fundraising success

Many blockchain startups are still in the early stages of development, where most technology firms typically would not have developed enough intellectual property to warrant patenting. While these young startups may not have mature or formally documented research and development outputs like patents, GitHub is a valuable platform for investors to verify technical progress and credibility claims. By openly sharing their source code on GitHub, startups provide investors with a direct view of their capabilities and competence. This signaling mechanism makes GitHub an essential intermediary, attracting investor attention and interest and potentially improving a startup's chances of raising capital as their development work gains validation. In this section, we examine our first hypothesis, which states that "*Open-source startups are more likely to succeed in fundraising*." To test it, we propose the following empirical model:

$$Success_{it} = \alpha + \beta_1 GitHub_{it} + \Theta_{it} + \Lambda_{it} + \epsilon_{it} \tag{1}$$

This specification explores the relationship between blockchain startups' fundraising success and code activity in GitHub. The dependent variable *Success* captures a startup's ability to secure capital. This variable is measured in two ways: (1) *RaisedDummy*, a dummy variable that equals one if any funds have been raised at the end of the fundraising stage and zero otherwise, and (2) *FundsRaised (log)*, defined as the logarithm of the total amount raised at the end of the fundraising stage, capturing the scale of financing achieved including pre-sale amounts.

We employ two proxies to measure our key independent variable *GitHub*: (1) *OpenSource*, an indicator variable for whether the blockchain startup has a GitHub account at the beginning of the fundraising phase and zero otherwise, and (2) *Code Production (log)*, the logarithm of the total number of commits before the fundraising stage.

We follow extant literature (e.g., Amsden and Schweizer (2018), Fisch (2019), Lyandres et al. (2022), and Davydiuk et al. (2023)) to measure code production by using GitHub commits, which capture the volume and frequency of a startup's overall development performance. A "*commit*" refers to a fundamental feature of GitHub that facilitates collaboration by allowing team members to submit changes to a repository with accompanying messages that describe these changes. When a developer makes a commit, she saves a snapshot of the project's state at that particular point in time, enabling anyone with access to the public repository to track it. Θ is a vector of variables controlling for a startup's characteristics likely to influence the fundraising outcome, including the startup's project, fundraising, and social media characteristics, broadly used in the literature and discussed in Section 4.3.

Additionally, Λ captures fixed effects, including year-month fixed effects to account for intertemporal variation that affects the relation between code activity and startup's fundraising performance (i.e., market sentiment and market conditions), geographic region fixed effects to account for time-invariant cross-regional unobservable variations (i.e., regulatory environment across different regions), and industry fixed effects to control for omitted industry characteristics (i.e., technology shocks across different industries) that are constant over time.

Moreover, we cluster the standard errors by venture-fundraising completion year and month to address potential serial correlations in the residuals, which are commonly observed in panel data. Given the lightly regulated and highly asymmetric information environment in the blockchain startup setting documented in the literature (e.g., Lyandres et al. (2022)), we expect β_1 to be significantly positive, suggesting that greater transparency through open-source code sharing on GitHub is a credible signal to investors during fundraising. This transparency is expected to reduce information asymmetry and improve funding outcomes.

Table 2 reports the results of equation (1). In columns (i) and (ii), where the dependent variable is *Raised-Dummy*, we use logistic regression to estimate equation (1) and report the marginal effects of the estimated coefficients. In columns (iii) and (iv), where the dependent variable is *FundsRaised (log)*, we use ordinary least squares (OLS) regression. The coefficients of our key variables, *OpenSource* and *Code Production (log)*, are positive and statistically significant at the 1% level across all four columns. These results suggest that greater code activity on GitHub is associated with both a higher probability of successful fundraising (extensive margin) and a larger amount raised (intensive margin). These results are also economically significant. Startups with a GitHub repository have a 7.2% higher probability of successfully raising funds than those without a GitHub repository. At the intensive margin, a percent increase in the number of commits recorded in GitHub is associated with a 0.307% increase in the amount of money raised.

These results suggest that having a GitHub account is correlated with funding success, consistent with the notion that opening the source code may potentially serve as an effective informational channel for investors assessing these early ventures. This transparency from code sharing likely enhances both their willingness to invest and the amounts they contribute. Regarding control variables, startups with a lower percentage of tokens for sale, a higher hardcap for fundraising, a larger team size, KYC requirements, and an available white paper are more likely to raise money. Additionally, we find distinct effects of different types of social media on fundraising.

Together, these results suggest that blockchain startups that engage in open-source code sharing on GitHub are more likely to succeed in fundraising. This finding validates our proposed signaling mechanism, aligning well with earlier evidence documented in the literature (e.g., Amsden and Schweizer (2018), Howell et al. (2020), Lyandres et al. (2022), Davydiuk et al. (2023), and Conti et al. (2024)). Having validated the signaling mechanism in our setting, we are ready to move on to testing our main hypothesis stated in Sections 2 and 3.

5.2 Pooled equilibrium

During hot markets, investors may allocate capital based on hype, trends, or superficial signals (e.g., flashy websites, well-designed white papers, empty GitHub repositories, etc) rather than fundamental project quality. This creates an environment where low-quality projects can exploit investors' irrational exuberance by appearing credible without substantive progress or innovation, a behavior aligned with concerns over code-washing.[10]

5.2.1 Code-washing

We start our analysis by first outlining our empirical specifications. Our variable of interest, *Code-washer*, is a startup in the bottom quartile of cumulative commits before the fundraising stage, conditional on startups having at least one commit by the end of the fundraising stage[11]; and zero otherwise. Similarly, our counterpart variable of interest, *Code-producer*, is a startup in the top quartile of cumulative commits before the fundraising stage; and zero otherwise. The base category captures startups without open-source code activity on GitHub and startups within the middle quartiles of code activity.[12]

We focus on the period before the fundraising phase because code activity during this window is less likely to be influenced by fundraising expectations and more likely to reflect the underlying quality of the project itself. In contrast, code production during the fundraising window may be just code-washing aimed at misleading investors with

[10] "Irrational exuberance" is a term popularized by former U.S. Federal Reserve Chairman Alan Greenspan in a 1996 speech, referring to investor behavior marked by unwarranted optimism that drives asset prices far above their intrinsic value. This phenomenon often manifests during speculative bubbles, where enthusiasm for financial assets—like stocks, real estate, or cryptocurrencies—leads to valuations disconnected from assets' fundamental quality.

[11] The condition that the startup must have submitted at least one commit ensures that "code-washers," in their attempt to signal to investors, have incurred some costs, such as hiring developers, operating the project, and accumulating technology in line with signaling theory (e.g., Spence (1973); Rothschild and Stiglitz (1976); Miller and Rock (1985)). We relax this assumption in alternative specifications discussed in Section 5.5.

[12] We include both groups in the main specification to mitigate selection concerns. Further, we also test with different baseline groups, matched samples, Heckman correction, and exogenous shocks.

superficial activity. Figure 3 illustrates this behavior. The figure depicts the evolution of GitHub code commits surrounding two critical milestones for blockchain startups: the start dates of the fundraising (Panel A) and exchange listing (Panel B) events. The time window spans from 270 days before the event to 365 days after. The y-axis represents the number of commits within each time interval. As illustrated in the figure, code-washers show minimal commit activity—averaging close to zero—until 90 days before the fundraising starting date. Then, their commit activity increases during the fundraising stage, reaching an average of 40 commits. In contrast, code-producers display consistent and sustained commit activity as early as 270 days before the fundraising stage, with an average of 400 commits—ten times that of code-washers. As expected, commit activity declines during the fundraising stage for code-producers, consistent with these young ventures reallocating resources from development to marketing. These code production patterns suggest speculative, market-timing behavior by code-washers.[13]

5.2.2 Code-washers vs. code-producers

Having created, discussed, and validated our proxies, we now examine whether code-washers successfully mislead investors. In other words, we test our second hypothesis, which states that "*Investors are unable to distinguish between code-washers and code-producers during the fundraising phase*," with the following empirical model:

$$Success_{it} = \alpha + \beta_1 Code\text{-}washers_{it} + \beta_2 Code\text{-}producers_{it} + \Theta_{it} + \Lambda_{it} + \epsilon_{it} \quad (2)$$

The dependent variable *Success* follows the same specification as in equation 1. As described in section 5.1, we also include Θ, which represents a vector of characteristics likely influencing a startup's funding outcomes, and Λ, which captures time, geographic region, and industry fixed effects. Following the discussion in section 2, we expect both coefficients β_1 and β_2 for all specifications to be significantly positive and their difference to be statistically insignificant, suggesting that investors do not distinguish between code-washers and code-producers during the blockchain startup fundraising phase.

Table 3 reports the results of equation (2). Table 3 reports the results of equation (2). In columns (i) and (ii), the dependent variables are *RaisedDummy* and *FundsRaised (log)*, respectively. The coefficients for *Code-washer* and *Coder-producer* in column (i) are both positive and significant at the 5% level, suggesting that these two types of startups signal higher quality to potential investors compared to the baseline startups (i.e., those without a GitHub repository and those in the interquartile range of code production). Economically, code-washers and code-producers are associated with 36.2% and 53.9% higher probabilities of successfully raising funds than baseline startups, respectively. A Wald test for the difference in coefficients between code-producers and code-washers

[13] Section 5.5.4 further investigates code-washing behavior during the fundraising stage.

shows that we cannot reject the null hypothesis (H_0: $\beta_1 - \beta_2 = 0$), indicating that the differences are statistically insignificant.

These results suggest that investors are unable to distinguish between the true underlying quality of code-washers and code-producers. During the fundraising stage, they appear to focus on whether a startup publicly shares code rather than analyzing the content or quality of the code. As a result, code-washers successfully mislead investors, leading to a pooled equilibrium where resources are misallocated towards code-washers.

5.2.3 Robustness checks

To improve the robustness of our empirical analysis, we do the following.

First, our results may be sensitive to sample self-selection bias. Specifically, our observations are limited to startups that have chosen to create an open-source GitHub account. However, a subset of startups may refrain from disclosing code activity on GitHub due to unobserved characteristics (e.g., the incentive to protect proprietary information from competitors). In other words, startups without a GitHub account could exhibit systematically different unobserved characteristics compared to those with a GitHub account. To address this concern, we employ the Heckman (1979) two-step methodology. In the first step, the dependent variable is *OpenSource*, indicating whether a startup has a GitHub account. We run a Logit regression on the full sample, including control variables, as well as time, geographic region, and industry fixed effects, consistent with those used in equation 2. Notably, the *IMR* (inverse Mills ratio) is calculated as the ratio of the probability density function (PDF) to the cumulative distribution function (CDF) of the standard normal distribution derived from the first-stage regression results. Then, we run the second-stage outcome using the sample restricted to startups with a GitHub account. The outcome variables are the same as the regressions reported in Table 3, with the inverse Mills ratio included as the additional control variable to account for unobservable factors that influence both the probability of selection and the outcome of interest. Columns (i) to (iii) in Table 4 report the results of Heckman two-stage regression. Consistent with the results in Section 3, in columns (ii) and (iii), the coefficients of *code-producer* and *code-washer* are both significantly positive, and their difference is statistically insignificant. However, the economic magnitudes of estimated coefficients become smaller compared to those in Table 3, indicating that self-selection may cause upward biases in our baseline results in the pooled regression. This approach allows us to correct for potential biases arising from the systematic differences between startups that chose to have a GitHub account and those that did not. Qualitatively, the insights from this robustness test are similar to those discussed in the previous analysis.

Second, our results could be subject to confounding factors influencing the effects of code activity on fundraising outcomes. To mitigate the concern that our results may be driven by differences in unobserved characteristics between code-washers and control startups, we employ a propensity score matching (PSM) approach. Specifically, each code-washer is matched with three startups from the control group based on the closest propensity score with replacement. The propensity score is estimated using the variables captured at the fundraising start date, included in the baseline regressions, such as *Tokens for sale (%)*, *Hardcap (log)*, *Whitelist*, *KYC*, *White paper*, *Team size (log)* , *Presale*, *Twitter (log)*, *Reddit (log)*, *BTCTalk (log)*, and *Medium (log)*. Appendix Table A3 reports summary statistics for the matched sample. We obtain 651 unique control startups and 313 code-washers. The code-washers and matched control startups are indistinguishable across all characteristics, successfully achieving balance. Then, we re-run regressions of equation 2 and report results in columns (iv) and (v) in Table 4. Likewise previous specifications, the estimated coefficients of both *code-washer* and *coder-producer* are significantly positive. These results give us confidence that time-invariant characteristics are statistically unlikely to have confounded our previous findings.

5.2.4 Exogenous shock

Despite the discussion and tests used to mitigate endogeneity in the previous section 5.2.3, we still have concerns that omitted variables could be correlated with code quality and fundraising success, thus biasing our results. For example, new technological trends can lead to an influx of new entrants and thus increase industry competition, which probably incentivizes some low-quality startups to fake their code or imitate competitors on GitHub. Simultaneously, these new technological trends also attract more investors to this field, increasing the likelihood that startups will secure funding. In this section, we employ a difference-in-differences (DiD) design to mitigate these concerns, adopting an exogenous regulatory change introduced by the U.S. Securities and Exchange Commission (SEC) in 2018, which plausibly impacted the incentives of startups to engage in misconduct.

Due to the weekly regulated environment blockchain startups operate in, entrepreneurs' incentives to engage in code-washing are likely increasing in the inability of authorities to pinpoint and prosecute financial misconduct. Becker (1968) posits that individuals engage in criminal behavior based on a rational assessment of risks and rewards. In the unregulated token fundraising space, the lack of oversight diminishes the likelihood of detection and punishment for fraudulent activities, creating a fertile ground for financial misconduct (e.g., Amiram, Jørgensen, and Rabetti (2022), Cong, Landsman, Maydew, and Rabetti (2023), and Cong, Harvey, Rabetti, and Wu (2024)). Based on Becker (1968)'s framework, we suggest that as the probability of being caught for fraud increases—due to potential regulatory changes or enhanced enforcement mechanisms—the incentives for engaging in deceptive practices, such as code-washing, decrease. In other words, heightened scrutiny and enforcement will significantly reduce the incidence of fraud in the blockchain startup's fundraising market.

In November 2018, the SEC announced its first civil penalties against cryptocurrency founders who failed to register their coin offerings as part of a broader initiative to regulate the growing digital currency industry and curb fraudulent activities.[14] Notable cases included startups like *Airfox* and *Paragon*, which raised more than $10 million without proper registration. The SEC mandated these companies to register their tokens as securities, pay penalties, and return funds to affected investors. This marked a critical shift, signaling that all entities issuing tokens must comply with existing securities laws. The introduction of these regulations had a pronounced effect on the incentives to engage in fraudulent disclosures among blockchain startups. Before the SEC's enforcement actions, many startups capitalized on the regulatory ambiguity surrounding token offerings to mislead investors. However, following the SEC's regulatory actions, the motivation for firms to engage in fraudulent practices significantly weakened due to stricter enforcement and increased penalty costs. As a result, the SEC's regulatory crackdown generates a plausibly exogenous variation in the likelihood of startups washing their codes on GitHub.

To capture the immediate effects of the regulatory changes, we restrict our sample of blockchain startups from six months before the event date through six months afterward, specifically from May 2018 to May 2019. Figure 3 illustrates the monthly proportion of code-washers entering the market relative to the total number of startups entering the market around the event. The horizontal axis represents the period spanning six months before and after the SEC's crackdown event, indicated by a vertical dashed line in November 2018. Before regulations, the proportion of code-washers fluctuated slightly and generally hovers around 8%—10%. However, after the regulations, there is a marked decline in the proportion of code-washers, averaging about 6%. In Appendix Table A3, we report logistic regression results showing that SEC's regulations significantly negatively influenced the probability of blockchain startups engaging in code-washing activities. These results suggest a causal effect of the SEC's action on fraudulent and low-quality disclosures among blockchain startups.

Following the SEC's regulatory actions, the costs for code-washers to disguise themselves as high-quality startups increased substantially. They were required to invest more resources to maintain the appearance of active code development, all the while facing the heightened risk of exposure to fraudulent practices and severe penalties. Weighing the low returns against higher costs, many code-washers likely chose to exit the fundraising market. Consequently, investors found it easier to identify high-quality projects based on genuine code quality. For these reasons, we predict the probability of code-washers raising money successfully will be significantly lower in the post-regulation period. To test this conjecture, we propose the following empirical model:

$$\begin{aligned} \textit{Success}_{it} =& \alpha + \beta_1 \textit{Code-washer}_{it} + \beta_2 \textit{Coder-producer}_{it} \\ &+ \beta_3 Post \times \textit{Code-washer}_{it} + \beta_4 Post \times \textit{Coder-producer}_{it} + \Theta_{it} + \Lambda_{it} + \epsilon_{it} \end{aligned} \tag{3}$$

Based on equation (2), we incorporate interaction terms between *Code-washer* and *Post*, as well as between

[14] See https://www.cnbc.com/2018/11/16/in-crackdown-of-crypto-sec-goes-after-unregistered-coin-offerings.html.

Coder-producer and *Post*. *Post* is an indicator variable that equals one if the blockchain startup's fundraising occurred after the SEC's regulatory crackdown on new token offerings in November 2018 and zero otherwise. The dependent variable *Success* follows the same specification as in equation (2). Our primary focus is the estimated coefficient β_3 on the interaction term between *Code-washer* and *Post*, which captures the changes in the fundraising performance of code-washers after the SEC's regulatory implementation compared to before. As described in section 5.2, we also include Θ, which represents a vector of characteristics on blockchain startups, and Λ that captures time, geographic region, as well as industry fixed effects in equation 2.

Table 5 reports the results for equation (3). The results indicate that before the SEC's regulatory crackdown, code-washers and code-producers had a higher probability of raising money than other startups. However, after the SEC's crackdown, code-washers exhibited an 18.3% significantly lower likelihood of successful fundraising. In contrast, the fundraising performance of code-producers remained statistically unaffected by the SEC's actions. These findings align with our expectations, demonstrating that after the regulatory crackdown, investors could better distinguish between low-quality and high-quality code, reducing the likelihood of providing funds to code-washers. The SEC's actions likely deterred low-quality code-washer projects from misleading investors and shifted resources towards more credible and transparent startups.

5.3 Separated equilibrium

We have established that investors cannot distinguish between code-washers and code-producers during the fundraising phase due to information asymmetry between startups and investors. As such information asymmetry decreases, we expect investors to better differentiate between these two types. We formally test our third hypothesis, which states that "*Investors can distinguish between code-washers and code-producers when asymmetric information is low and/or investor attention is high.*" by re-running equation (2) in the following sub-samples: (1) a subset with lower information asymmetry and (2) a subset with higher investor attention.

5.3.1 Code production, information quality, and fundraising

Our analysis starts by examining the relationship between startup code activity and its fundraising outcomes in the subset of high information quality. We employ two proxies to measure the extent of information asymmetry: (1) *high information coverage* indicates that a blockchain startup ranks above the median of all startups in terms of information coverage, as measured by the number of sources with available data characterizing the blockchain startup during the fundraising phase; and (2) *high information quality* indicates that a blockchain startup ranks above the median of all startups in terms of information quality, as measured by a function of the total number of sources with available data and average consistency across sources for each blockchain startup's fundraising attempt following Lyandres et al. (2022)—Appendix B details the data quality procedure.

Then, we repeat the analyses in Table 5.2 in subsamples of high information coverage and high information quality separately. We focus on whether the coefficients β_1 and β_2 exhibit a statistically significant difference in the subsample with lower information asymmetry. Following the discussion in section 2, we expect differences between β_1 and β_2 to be statistically significant and only the coefficient β_1 to be significantly positive for all specifications. We report the regression results in Table 6. In all columns, the coefficients of *code-producer* are significantly positive, while those for code-washer are not significant. In terms of economic magnitudes, the value of β_1 is significantly larger than that of β_2 as well. Additionally, the results of the Wald test with a p-value less than 0.1 (except for column (iii)) indicate that differences between β_1 and β_2 are statistically significant. These findings align with our expectations that when asymmetric information is lower, investors can effectively distinguish between code-washers and code-producers.

5.3.2 Code production, investor attention, and fundraising

In this section, we examine the relationship between blockchain startup code quality and its financial proceedings' success in the subset of higher investor attention. Following Derrien (2005), we use Ether (ETH) market returns as a proxy for investor attention. Specifically, *low Ether market returns* indicate that a blockchain startup's fundraising attempt occurs in a month when Ether market returns are below the median of all startups. Then, we re-estimate the models as equation (2) in the subsample of low ETH market returns. The regression results, shown in Table 7, reveal that in columns (i) and (ii), the coefficients of *code-producer* are significantly positive. In contrast, those for code-washer are not significant. Besides, the Wald test in column (ii) confirms a statistically significant difference between β_1 and β_2. These findings are consistent with our hypotheses: in a market downturn, when investors are more focused, they are more likely to analyze the fundamentals of projects more carefully, including factors such as code quality, team background, and transparency. Therefore, they are able to distinguish between coder washers and code-producers effectively.

5.4 Post-fundraising performance

Next, we aim to explore the long-term implications of blockchain startups' code-washing behavior by following the literature that suggests a trade-off of short-term gains from misleading investors to long-term implications due to damaged reputations. Following the discussion in section (2), the last Hypothesis proposes that "*code-washers are more likely to underperform than code-producers in the post-fundraising period.*" To test this assertion, we employ the following empirical model:

$$LTPerformance_{it+1} = \alpha + \beta_1 Code\text{-}washers_{it} + \beta_2 Code\text{-}producers_{it} + \Theta_{it} + \Lambda_{it} + \epsilon_{it} \quad (4)$$

The dependent variable *LTPerformance* captures the long-term performance of blockchain startups. We focus on three key aspects of a startup's long-term performance: (a) fundraising, (b) financial performance, and (c) technological innovation. (a) Fundraising: we use *Listing* as a proxy, an indicator variable equaling one if the blockchain startup is listed on at least one cryptographic exchange and zero otherwise. (b) Financial performance: we use the following three measurements: (1) *Returns (log)* is defined as the cumulative returns over the first 180 days since the first trading day expressed in logarithm, (2) *Volatility (log)* is defined as the daily return volatility calculated over the same period in logarithm, (3) *Amihud (log)* is defined as illiquidity calculated over the same period in logarithm. (c) Technological innovation: we use *Commits (log)* to measure a startup's long-term innovation outputs, defined as cumulative GitHub commits over the first 180 days after the token offering end date expressed in logarithm. The variables of interest *Code-washer* and *Coder-producer*, the vector of blockchain-level controls Θ, and fixed-effects Λ, follow the same specification as in equation (2). Following the discussion in section 2, we expect the coefficient in β_1 for all specifications to be significantly positive, suggesting that code-producers perform better than their peers in the post-fundraising period (i.e., are less likely than their peers to fail). Conversely, we expect the coefficient in β_2 for all specifications to be insignificant and even negative, suggesting that code-washers exhibit lower performance than their peers in the post-fundraising period (i.e., are more likely than their peers to fail).

The regression results reported in Table 8 indicate that the estimated coefficients for *code-producer* consistently exhibit significantly positive long-term performance across all columns. They tend to exhibit better financial performance, lower operational risks, and superior technology innovation outcomes. While code-washers demonstrate a higher probability of listing than other benchmark startups, their performance remains significantly lower than code-producers. Notably, in terms of technology innovation, code-washers exhibit even significantly worse performance than other benchmark startups. As we discussed in section 2, the damaged reputation of code-washers diminishes their competitiveness in the long term. Engaging in superficial signaling can result in a loss of trust among investors, customers, and industry peers.

We then follow the literature (e.g., Ritter (1991), Cohen, Malloy, and Pomorski (2012), and Cohen, Diether, and Malloy (2013a)) to examine the buy-and-hold returns (BHRs) of portfolios formed by code-washers and code-producers over the 18 months following exchange listing events. Figure 4 illustrates the evolution of BHRs for portfolios comprising code-washers and code-producers during this period. Startups are categorized into portfolios based on the quality of their code commits before the fundraising phase. We report the BHRs for equal-weighted portfolios in Panel A and value-weighted portfolios in Panel B. The red line represents the BHRs of the code-producer portfolio, while the blue line represents the code-washer portfolio. Our results indicate that the code-producer portfolio exhibits significantly higher BHRs over the 18 months following the exchange listing event. Notably, the equal-weighted portfolio BHR exceeds 600%, starkly contrasting the negative returns observed for the code-washer portfolio. Investors appear to evaluate the long-term innovation potential of code-producers based on their open-source activities, which contributes to higher market valuations. Interestingly, the code-washer portfolio exhibits a reversal pattern in the long term. This suggests that code-washers were likely misvalued during the fundraising stage. Over time, as investors increasingly focus on assessing the underlying technological innovation of these firms, the lack of substantial innovation behind code-washing leads to market corrections (i.e., through learning their type) and penalization.

5.5 Additional robustness and validation tests

5.5.1 Cheap talk

Our results suggest that code production is a credible signal of a blockchain startup's intrinsic quality; in this case, code-washers should also "wash" other aspects of the venture, such as voluntary disclosures to the public. To validate this conjecture, we investigate whether code-washers are more likely to engage in "cheap talk"; in other words, we examine the relationship between code quality and white paper informativeness. To test it, we estimate Equation (4), replacing the dependent variables with: (1) *Words per page (log)* is measured by the logarithm of word count per page in the blockchain start-up's white paper, (2) *Image count (log)* is measured by the logarithm of one plus the count of images in the blockchain start-up's white paper, (3) *Tech word count (log)* is measured by the logarithm of the count of technology-related words in the blockchain start-up's white paper, identified using natural language processing techniques. The variables of interest capture an indicator variable for whether the blockchain start-up is a code-producer (i.e., at the top quartile of cumulative commits generated before the fundraising stage) or a code-washer (i.e., at the bottom quartile of cumulative commits generated before the fundraising stage).

Table A4 presents the regression results. We document that code production is strongly associated with the quality of information disclosed in a startup's white paper. In general, code-producers create significantly higher-quality white papers compared to benchmark startups. However, in contrast to code-producers, code-washers tend to provide less informative content in their white papers, including a lower total word count, fewer images, and reduced technology-related word count. These characteristics collectively indicate that code-washers tend to engage more significantly in "cheap talk" than code-producers, further validating the link between open-source code production and intrinsic startup quality.

5.5.2 Pull requests and issues

In our baseline analysis, we follow extant literature (e.g., Amsden and Schweizer (2018), Fisch (2019), Lyandres et al. (2022), and Davydiuk et al. (2023)) to measure code production by using GitHub commits, which capture the volume and frequency of a startup's overall development performance. Complementary proxies for code development, such as pull requests and issues, have been proposed. Pull requests reflect collaborative coding practices and peer review, while issues represent ongoing project management, bug tracking, and feature development. Together, these activities indicate more complex technical requirements and larger-scale team collaboration efforts, suggesting a more costly signal than commits (as besides pull requests and issues, commits also include other software development tasks).

First, we construct four proxies to measure code quality: (1) *Pull (log)*, defined as the natural logarithm of one plus the cumulative number of pull requests before the fundraising conclusion; (2) *Issues (log)*, defined as the natural logarithm of one plus the cumulative number of issues created before the fundraising conclusion; (3) *PullDummy*, an indicator variable equal to one if the startup made at least one pull request before the fundraising conclusion and zero otherwise; (4) *IssuesDummy*, an indicator variable equal to one if the startup created at least one issue before the end of the fundraising phase and zero otherwise.

Next, to capture the relationship between code quality and pull request and issue activities on the GitHub platform, we re-run the specifications of Equation (4) using a sample of startups with at least one commit by the end of the fundraising phase. The results, reported in Appendix A5, show that the coefficients for *code-producer* are positive and significant at the 1% level in all columns, suggesting that code-producers demonstrate a higher capability to perform more complex technical operations on GitHub. In contrast, the coefficients for *code-washer* are not significant across all columns, indicating that while code-washers generate commits, they do not engage in more complex technical activities.

5.5.3 Zero commits

We also exploit a less restrictive signal of code production as it involves no production costs: when a startup opens a GitHub account but produces no code at all. We called this type of behavior *open-source washing*. Open-source washers, like code-washers, should be able to mislead investors during the fundraising phase, especially during moments of pooled equilibria where asymmetric information is too high to separate good code-producers from mimicking startups.

We rerun equation 1 but replace *Code-washer* and *Code-producer* with the following changes: *Open-source-washer,* which is defined as a startup that opens a Github account but produces zero commits before the fundraising starting date; and *Open-source-producer*, which is defined as a startup that produces at least one commit in the period before the fundraising starting date. Open-source-washer appears in 35 percent of the sample, while Open-source-producer appears in 31 percent of the sample. The remaining 34 percent of the ventures are those that have all variables of interest described in Table 1 but decided not to open the source code. We present the results in Table A6. We find that, although less likely than open-source producers, open-source washers succeed in misleading investors in the fundraising phase. Similar results are found when using the total amount of funds raised.

We also test this alternative specification for a separated equilibrium, focusing on the subsample of startups with high information coverage and quality, as outlined in 2. Results for the alternative specification, presented in Table A7, are similar to those from the main specification. This suggests that investors can distinguish between code-producers and code-washers during the fundraising stage when asymmetric information is reduced.

Overall, the results in this subsection suggest that this less restrictive specification could serve as a viable alternative to the current primary one. However, despite the potential utility of the alternative specification, we opt to retain a more conservative specification (which requires at least one commit produced) in the main analysis. This decision is driven by the inclusion of additional costs associated with code production, such as those related to human capital and infrastructure. These costs are a significant component in signaling effectiveness, as described in the seminal work by Spence (1973). In the context of token offerings, the costs of producing robust, functional, and transparent code can serve as a credible signal of project quality. High-quality projects are more likely to bear these costs as a means of differentiating themselves from low-quality or fraudulent offerings. Given the importance of credible signaling in mitigating information asymmetries, the inclusion of these costs in our primary specification ensures a more nuanced and realistic representation of the market dynamics. This decision is particularly relevant in a market where traditional regulatory oversight is minimal, and the burden of due diligence largely falls on retail investors, making the role of signals even more critical.

5.5.4 Fundraising stage

Producing code in the blockchain setting is equivalent to developing the main platform service, thus serving as a direct measure of an early venture's innovation. Code production is costly. It involves hiring developers, providing infrastructure such as office space and equipment, and managing the development through effective oversight. For legitimate startups, code production ideally begins several months before the fundraising event to demonstrate the capabilities of the digital platform to potential investors—for example, by showcasing a minimum viable product or conducting beta testing. Furthermore, these startups are expected to experience a natural decline in code production during the fundraising event as resources are reallocated from development activities to managing the fundraising process, particularly in startups with smaller teams. In contrast, code-washers are more likely to initiate or significantly increase their code production activity during the fundraising event.

To test these dynamics, we measure the extent of code production during the fundraising stage by constructing a continuous variable, *code-washing*, defined as the ratio of cumulative commits made during the fundraising stage to the total number of commits at the end of the fundraising. As shown in appendix A8, the average *code-washing* ratio for code-washers is 22.1%, compared to 6.4% for code-producers.

Next, we re-run the specifications of equation (2) using this variable as the dependent variable. In columns (i) and (ii), the coefficients for code-washer are both positive and significant at the 1% level, indicating that code-washers are more likely to engage in code activity during the fundraising compared to benchmark startups with at least one commit by the end of the fundraising stage. In columns (iii) and (iv), we use alternative proxies discussed in the previous subsection, *Open-code-washer*, and *Open-code-producer*, for additional robustness—results remain qualitatively unchanged.

Our results align with the economic intuition that code-producers are less likely to prioritize code production during the fundraising stage, as resources are reallocated from digital platform development to managing the fundraising effort. In contrast, code-washers inflate code production with meaningless coding as a signal to investors in an attempt to boost short-term financing outcomes. Overall, these findings support the economic interpretation presented throughout the manuscript.

5.5.5 Financial misconduct

Finally, as an extreme case of poor post-fundraising performance may be due to financial misconduct (e.g., Cong, Grauer, Rabetti, and Updegrave (2023)), we test a plausible link between code-washing and financial misconduct.[15] Following the discussion in section (2), we propose that code-washers are more likely to be perceived as scammers than code-producers in the post-fundraising period. To test this assertion, we employ the following empirical model:

$$\begin{aligned} Scam_{it+1} = & \alpha + \beta_1 Open\ Source_{it} \times Code\text{-}washer_{it} + \beta_2 Open\ Source_{it} \times Code\text{-}producer_{it} \\ & + \beta_3 Open\ Source_{it} + \Theta_{it} + \Lambda_{it} + \epsilon_{it} \end{aligned} \tag{5}$$

We use two proxies to measure *Scam*: (1) *Number of Scams* is defined as the total number of scam events involving the blockchain startup post fundraising. (2) *Scam Dummy* is an indicator variable equaling one if the blockchain startup is involved in at least one scam event and zero otherwise. *Open Source* is defined as whether the blockchain startup has a GitHub account. The variables of interest capture the interaction terms of *Open Source* and startup

[15] See appendix A4 for an example of an infamous token offering exit scam.

type, distinguishing whether the startup is a code-producer or a code-washer. The control variables are described in Table 1. Regarding the regression models, we estimate Poisson regressions for the dependent variable *Number of Scams* and Logistic regressions for the dependent variable *Scam Dummy*. As discussed in section 2, we expect the coefficient in β_1 for all specifications to be significantly positive, suggesting that code-washers exhibit a higher probability of being involved in a scam activity than other blockchain startups with a GitHub repository. The empirical results are shown in Table A9. In columns (i) and (ii), the coefficients of the interaction terms of *Source* and code-washer are both significantly positive, supporting our hypotheses. The main objective for code-washers is to secure quick funding, as they may lack the capability or motivation to fulfill their promises. Besides, there may be heightened pressure to show operating progress or returns on investment after fundraising. This combination of short-term focus and failure to deliver significantly raises the risk of scams among code-washers.

6 Conclusion

We examine the role of code production as a signaling mechanism influencing the fundraising outcomes of early-stage ventures in emerging technology sectors. Code production not only enhances a startup's credibility in delivering high-quality products and services—a critical component of entrepreneurial success for early ventures—but also plays a pivotal role in attracting investor interest. However, our analysis reveals a dual reality within this ecosystem. On the one hand, genuine code production fosters trust and supports sustainable resource allocation. On the other hand, deceptive practices such as code-washing undermine market integrity. Code-washing can mislead investors, resulting in misallocated resources and enabling financial misconduct. By analyzing outcomes for both code-producers and code-washers, we find that while code-washing strategies may generate short-term fundraising success, they impose significant long-term reputational costs. These costs include reduced access to capital markets, investor distrust, and pervasive inability to sustain innovation, ultimately hindering the growth of opportunistic startups.

Our findings offer several insights. First, they suggest a need for enhanced transparency in open-source platforms—including developing solutions to flag code-washing behavior and regulatory oversight to help investors discern between authentic and opportunistic coding activities by early high-tech ventures. Such measures can mitigate financial misconduct, ensure resources are allocated to genuinely innovative startups, and foster a more trustworthy entrepreneurial ecosystem. Moreover, we provide insights into broader discussions about how firms mislead stakeholders through fake signaling to boost short-term goals, paralleling concerns about practices like green-washing—currently on the agenda of policymakers and regulators.[16] Finally, we join a nascent literature examining the role of

[16] The US Federal Trade Commission (FTC) updated its Green Guides for the first time in more than a decade, with new recommendations on claims around sustainable materials (see https://www.reuters.com/business/sustainable-business/what-are-us-green-guides-can-

open-source coding disclosure. By disentangling genuine innovation from superficial signaling, we provide insights into the challenges and opportunities associated with open-source practices in entrepreneurial finance, particularly for early-stage ventures in the emerging technology sectors such as blockchain, artificial intelligence, and decentralized finance.

they-stamp-out-greenwashing-2023-04-27/).

References

Admati, A. R. and P. Pfleiderer (1994). Robust financial contracting and the role of venture capitalists. *The Journal of Finance 49*(2), 371–402.

Akerlof, G. A. (1970). The market for "lemons": Quality uncertainty and the market mechanism. *The Quarterly Journal of Economics 84*(3), 488–500.

Amiram, D., B. N. Jørgensen, and D. Rabetti (2022). Coins for bombs: The predictive ability of on-chain transfers for terrorist attacks. *Journal of Accounting Research 60*(2), 427–466.

Amiram, D., E. Lyandres, and D. Rabetti (2024). Trading volume manipulation and competition among centralized crypto exchanges. *Management Science*. Forthcoming. Available at https://dx.doi.org/10.2139/ssrn.3745617.

Amsden, R. and D. Schweizer (2018). Are blockchain crowdsales the new 'gold rush'? Success determinants of initial coin offerings. Unpublished manuscript. Available at https://dx.doi.org/10.2139/ssrn.3163849.

Ball, I. (2022). Scoring strategic agents. Unpublished manuscript. Available at https://arxiv.org/pdf/1909.01888.pdf.

Ball, R. (2001). Market and political/regulatory perspectives on the recent accounting scandals. *Accounting Horizons 15*(4), 437–450.

Barth, A., V. Laturnus, S. Mansouri, and A. F. Wagner (2023). Conflicted analysts and initial coin offerings. *Management Science 69*(11), 6641–6666.

Becker, B. and T. Milbourn (2011). How did increased competition affect credit ratings? *Journal of Financial Economics 101*(3), 493–514.

Becker, G. S. (1968). Crime and punishment: An economic approach. *Journal of Political Economy 76*(2), 169–217.

Benedetti, H. and L. Kostovetsky (2021). Digital tulips? Returns to investors in initial coin offerings. *Journal of Corporate Finance 66*, 1–20.

Besanko, D. and G. Kanatas (1993). Credit market equilibrium with bank monitoring and moral hazard. *The Review of Financial Studies 6*(1), 213–232.

Bolton, P. and M. Dewatripont (2005). *Contract Theory*. Cambridge, MA: The MIT Press.

Bradshaw, M. T. and R. G. Sloan (2003). The relevance of conservative accounting to analysts' forecast accuracy. *The Accounting Review 78*(1), 19–45.

Brau, C., J. and S. Fawcett, E. (2006). Initial public offerings: An analysis of theory and practice. *The Journal of Finance 61*, 399–436.

Cheng, Q. and K. Subramanyam (2008). Analyst coverage and discretionary accruals: Evidence from equity offerings. *Contemporary Accounting Research 25*(3), 1007–1041.

Chod, J. and E. Lyandres (2021). A theory of icos: Diversification, agency, and information asymmetry. *Management Science 67*(10), 5969–89.

Cohen, D. and P. Zarowin (2010). Accrual-based and real earnings management around seasoned equity offerings. *The Accounting Review 85*(1), 191–225.

Cohen, L., K. Diether, and C. Malloy (2013a). Misvaluing innovation. *Review of Financial Studies 26*(3), 635–666.

Cohen, L., K. B. Diether, and C. J. Malloy (2013b). Misvaluing innovation. *The Review of Financial Studies 26*(3), 635–666.

Cohen, L., U. G. Gurun, and S. D. Kominers (2019). Patent trolls: Evidence from targeted firms. *Management Science 65*(12), 5461–5486.

Cohen, L., U. G. Gurun, and Q. H. Nguyen (2020). The esg-innovation disconnect: Evidence from green patents. *The Review of Financial Studies 33*(1), 1–42.

Cohen, L., C. Malloy, and L. Pomorski (2012). Decoding inside information. *The Journal of Finance 67*(3), 1009–1043.

Cong, L. W., K. Grauer, D. Rabetti, and H. Updegrave (2023). Blockchain forensics and crypto-related cybercrimes. Book Chapters. Available at http://dx.doi.org/10.2139/ssrn.4358561.

Cong, L. W., W. Landsman, E. Maydew, and D. Rabetti (2023). Tax-loss harvesting with cryptocurrencies. *Journal of Accounting and Economics 76*(2), 101607.

Cong, L. W., E. Prasad, and D. Rabetti (2024). Financial and informational integration through decentralized oracle networks. Unpublished manuscript. Available at https://dx.doi.org/10.2139/ssrn.4495514.

Cong, W., C. Harvey, D. Rabetti, and Z.-Y. Wu (2024). An anatomy of crypto-enabled cybercrimes. *Management Science*. Forthcoming. Available at https://www.nber.org/papers/w30834.

Connelly, B. L., S. T. Certo, R. D. Ireland, and C. R. Reutzel (2011). Signaling theory: A review and assessment. *Journal of Management 37*(1), 39–67.

Conti, A., C. Peukert, and M. Roche (2024). Beefing it up for your investor? Engagement with open source communities, innovation and startup funding: Evidence from github. Unpublished manuscript. Available at https://ssrn.com/abstract=3883936.

Davydiuk, T., D. Gupta, and S. Rosen (2023). De-crypto-ing signals in initial coin offerings: Evidence of rational token retention. *Management Science 69*(11), 6584–6624.

Dechow, P. M., R. G. Sloan, and A. P. Sweeney (1995). Detecting earnings management. *The Accounting Review 70*(2), 193–225.

Derrien, F. (2005). Ipo pricing in "hot" market conditions: Who leaves money on the table? *The journal of finance 60*(1), 487–521.

Diamond, D. W. (1984). Financial intermediation and delegated monitoring. *The Review of Economic Studies 51*(3), 393–414.

Ecker, F., J. Francis, I. Kim, P. Olsson, and K. Schipper (2006). Predicting earnings management: The case of earnings restatements. *Contemporary Accounting Research 23*(1), 1–37.

Fisch, C. (2019). Initial coin offerings (ICOs) to finance new ventures. *Journal of Business Venturing 34*(1), 1–22.

Frankel, A. and N. Kartik (2019). Muddled information. *Journal of Political Economy 4*(127), 1739–1776.

Froot, K. A. and M. Obstfeld (1991). Intrinsic bubbles: The case of stock prices. *The American Economic Review 81*(5), 1189–1214.

Gan, J., G. Tsoukalas, and S. Netessine (2021). Initial coin offerings, speculators, and asset tokenization. *Management Science 67*(2), 914–931.

Gennotte, G. and H. Leland (1990). Market liquidity, hedging, and crashes. *The American Economic Review 80*(5), 999–1021.

Griffin, J. M. and S. Kruger (2024). What is forensic finance? *Foundations and Trends® in Finance 14*(3), 137–243.

Healy, P. M. and J. M. Wahlen (1999). A review of the earnings management literature and its implications for standard setting. *Accounting Horizons 13*(4), 365–383.

Heckman, J. (1979). Sample selection bias as a specification error. *Econometrica 47*(1), 153–161.

Heckman, J. (1990). Varieties of selection bias. *The American Economic Review 80*(2), 313–318.

Hirshleifer, D., P.-H. Hsu, and D. Li (2018). Innovative originality, profitability, and stock returns. *The Review of Financial Studies 31*(7), 2553–2605.

Holmström, B. and J. Tirole (1997). Financial intermediation, loanable funds, and the real sector. *The Quarterly Journal of Economics 112*(3), 663–691.

Howell, S. T., M. Niessner, and D. Yermack (2020). Initial coin offerings: Financing growth with cryptocurrency token sales. *The Review of Financial Studies 33*(9), 3925–3974.

Karpoff, J. M., D. S. Lee, and G. S. Martin (2008a). The consequences to managers for financial misrepresentation. *Journal of Financial Economics 88*(2), 193–215.

Karpoff, J. M., D. S. Lee, and G. S. Martin (2008b). The cost to firms of cooking the books. *Journal of Financial and Quantitative Analysis 43*(3), 581–611.

Lee, C. M. C., A. Shleifer, and R. H. Thaler (1991). Investor sentiment and the closed-end fund puzzle. *The Journal of Finance 46*(1), 75–109.

Lee, J., T. Li, and D. Shin (2021). The wisdom of crowds in fintech: Evidence from initial coin offerings. *Review of Corporate Finance Studies*. Forthcoming. Available at https://ssrn.com/abstract=3195877.

Leland, H. and D. Pyle (1977). Informational asymmetries, financial structure, and financial intermediation. *The Journal of Finance 32*(2), 371–387.

Li, F. (2008). Annual report readability, current earnings, and earnings persistence. *Journal of Accounting and Economics 45*(2-3), 221–247.

Ljungqvist, A., V. Nanda, and R. Singh (2006). Hot markets, investor sentiment, and IPO pricing. *The Journal of Business 79*(4), 1667–1702.

Loughran, T., J. R. Ritter, and K. Rydqvist (1994). Initial public offerings: International insights. *The Journal of Finance 49*(5), 1755–1791.

Luca, M. and G. Zervas (2016). Fake it till you make it: Reputation, competition, and Yelp review fraud. *Management Science 62*(12), 3412–3427.

Lyandres, E., B. Palazzo, and D. Rabetti (2022). Initial coin offering (ICO) success and post-ICO performance. *Management Science 68*(12), 8658–8679.

Lyandres, E. and D. Rabetti (2024). Initial coin offerings. In D. Cumming and B. Hammer (Eds.), *The Palgrave Encyclopedia of Private Equity*. Cham: Palgrave Macmillan.

Marquardt, C. A. and C. I. Wiedman (2005). Earnings management: The role of accruals and real activities manipulation. *Journal of Accounting and Economics 39*(3), 313–342.

MasColell, A., M. Whinston, and J. Green (1995). *Microeconomic Theory*. Oxford University Press.

Mayzlin, D., Y. Dover, and D. Chevalier (2014). Promotional reviews: An empirical investigation of online review manipulation. *American Economic Review 104*(8), 2421–2455.

Megginson, W. L. and K. A. Weiss (1991). Venture capitalist certification in initial public offerings. *The Journal of Finance 46*(3), 879–903.

Milgrom, P. R. and J. Roberts (1986). Relying on the information of interested parties. *RAND Journal of Economics 17*(1), 18–32.

Miller, M. H. and K. Rock (1985). Dividend policy under asymmetric information. *The Journal of Finance 40*(4), 1031–1051.

Modigliani, F. and M. H. Miller (1958). The cost of capital, corporation finance and the theory of investment. *The American Economic Review 48*(3), 261–297.

Momtaz, P. P. (2020). The pricing and performance of cryptocurrency. *The European Journal of Finance 27*(4-5), 367–80.

Rangan, S. (1998). Earnings management and the performance of seasoned equity offerings. *Journal of Financial Economics 50*(1), 101–122.

Ritter, J. R. (1984). The "hot issue" market of 1980. *Journal of Business*, 215–240.

Ritter, J. R. (1991). The long-run performance of initial public offerings. *The Journal of Finance 46*(1), 3–27.

Ritter, J. R. and I. Welch (2002). A review of IPO activity, pricing, and allocations. *The Journal of Finance 57*(4), 1795–1828.

Rothschild, M. and J. E. Stiglitz (1976). Equilibrium in competitive insurance markets: An essay on the economics of imperfect information. *The Quarterly Journal of Economics 90*(4), 629–649.

Shu, T., X. Tian, and X. Zhan (2022). Patent quality, firm value, and investor underreaction: Evidence from patent examiner busyness. *Journal of Financial Economics 143*(3), 1043–1069.

Spence, M. (1973). Job market signaling. *The Quarterly Journal of Economics 87*, 335–374.

Sun, J. (2022). Algorithmic transparency. Unpublished manuscript. Available at https://ink.library.smu.edu.sg/lkcsb_research/7109/.

Teoh, S. H., I. Welch, and T. Wong (1998). Earnings management and the long-run market performance of initial public offerings. *Journal of Finance 53*(6), 1935–1974.

Wilson, C. (1977). A model of insurance markets with incomplete information. *Journal of Economic Theory 16*(2), 167–207.

Zinman, J. and E. Zitzewitz (2016). Wintertime for deceptive advertising? *American Economic Journal: Applied Economics 8*(1), 177–192.

Figure 1. Funds raised by blockchain startups. This figure illustrates the total amounts raised in fundraising attempts by blockchain startups from 2013 to 2020. Fundraising amounts in all currencies have been converted to U.S. dollars and are expressed in millions.

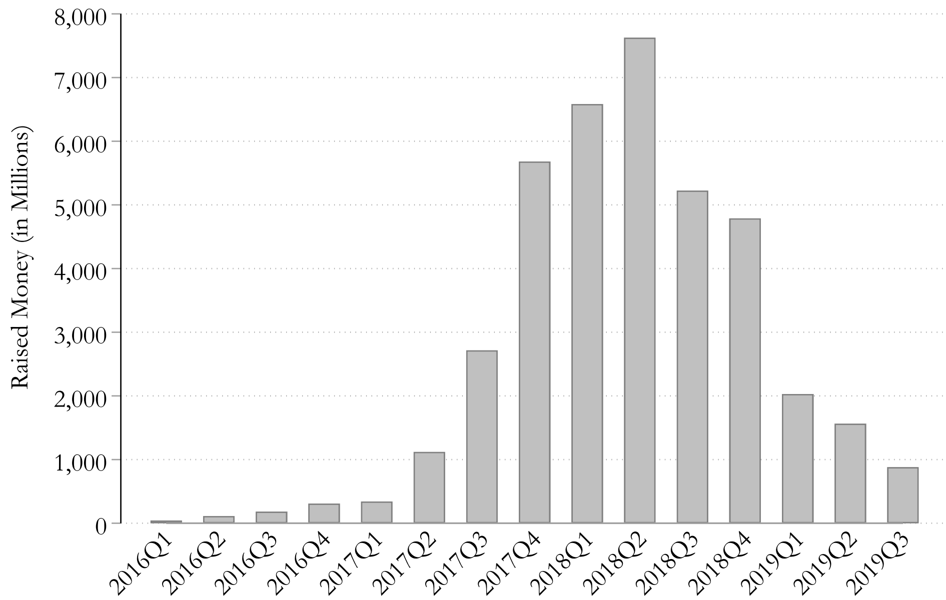
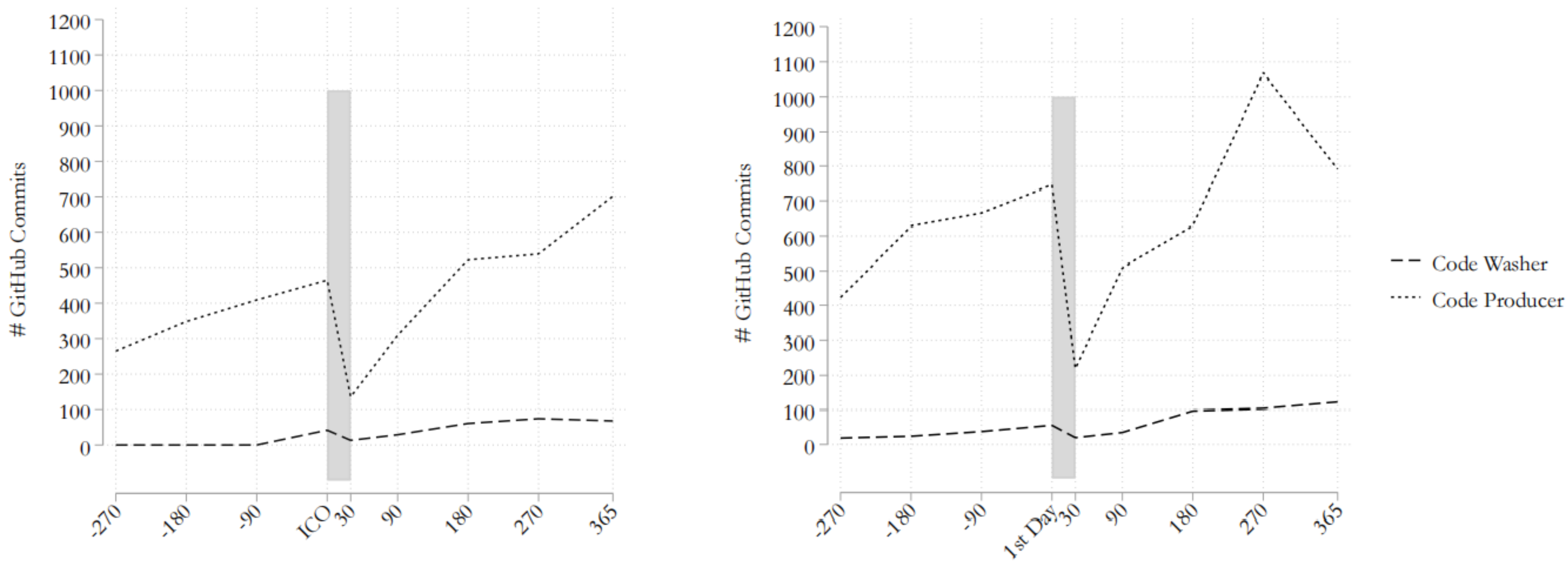


Figure 2. Code production around fundraising and exchange listing events. This figure illustrates the evolution of code production, captured as GitHub code commits, around the fundraising start date and the first exchange trading date of blockchain startups in our sample. The time window spans from 270 days before the event to 365 days after the event. The y-axis represents the number of commits made within each time interval. For example, the point labeled ”st” represents the number of commits made in the interval [-30 days, token offering start date]. Panel A shows the code production for code-producers and code-washers around the fundraising event. Panel B shows the code production of code-producers and code-washers around the exchange listing event.



(a) Panel A: Code production around token offering.

(b) Panel B: Code production around token listing.

Figure 3. Evolution of code-washers. This figure illustrates the percentage of code-washers relative to the total number of blockchain startups. A "code-washer" is defined as a startup in the bottom quartile of cumulative commits generated before the fundraising stage, calculated across the entire sample period. The vertical axis represents the proportion of code-washers entering the market relative to the total number of startups entering the market, while the horizontal axis represents the time period spanning six months before and after the SEC's crackdown event. The red vertical line marks the SEC's regulatory crackdown on blockchain startups for abuses and fraud in November 2018 (See https://www.cnbc.com/2018/11/16/in-crackdown-of-crypto-sec-goes-after-unregistered-coin-offerings.html.).

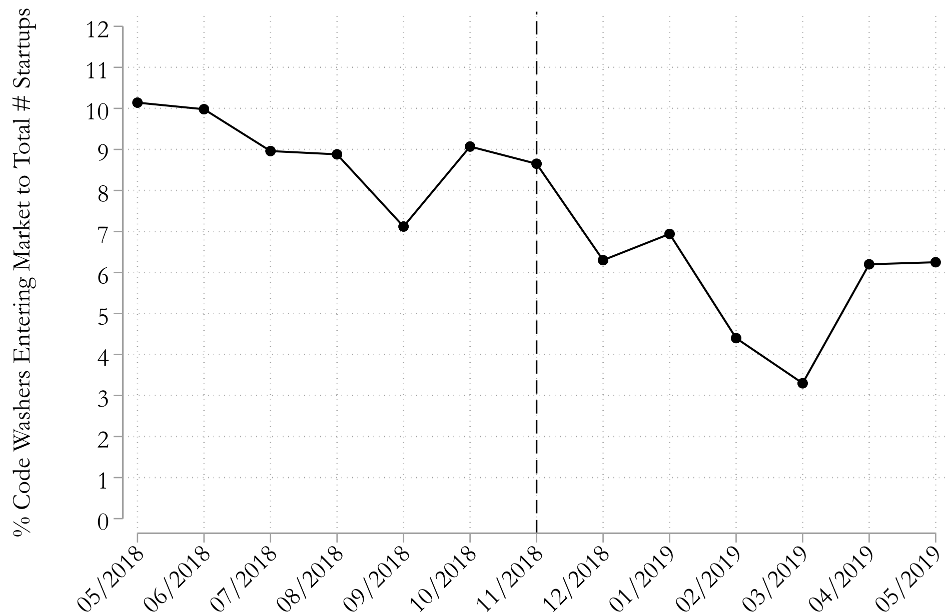
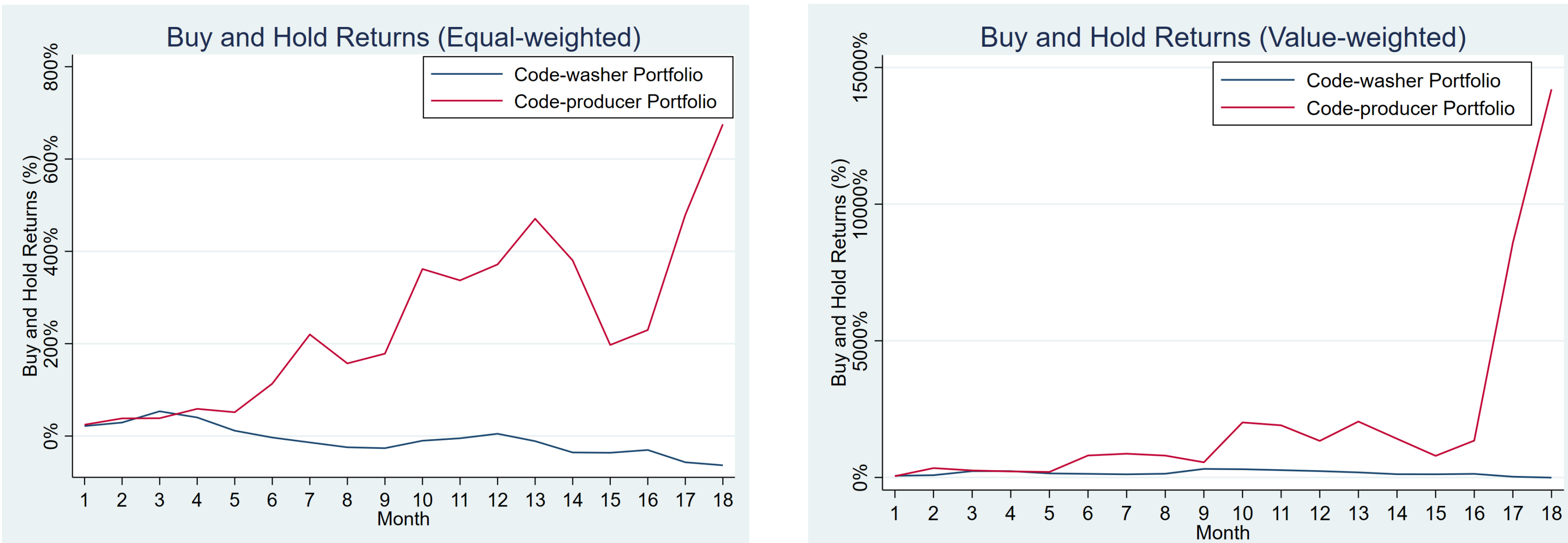


Figure 4. Buy and hold returns around exchange listing events. This figure illustrates the evolution of buy-and-hold returns (BHRs) for portfolios comprising code-washers and code-producers over the 18 months following exchange listing events. Startups are categorized into portfolios based on their code commit quality before fundraising phase. The red line represents the BHRs of the code-producer portfolio, while the blue line represents the code-washer portfolio. Panel A shows the BHRs for equal-weighted portfolios, where each firm is equally weighted. Panel B displays the BHRs for value-weighted portfolios, using market capitalization at the end of each month as the weighting factor.



(a) Panel A: Equal-weighted portfolios.

(b) Panel B: Value-weighted portfolios.

Table 1. Summary statistics. This table reports the summary statistics of the full sample used in this study. The main variables of interest are described as follows. *Open Source* is an indicator variable for whether the blockchain startup has a GitHub account before the fundraising stage. *Code Production* is the logarithm of the total number of commits before the fundraising stage. *Code-producer* is a startup in the top quartile of cumulative commits generated before the fundraising stage, and *Code-washer* is a startup in the bottom quartile of cumulative commits generated before the fundraising stage. *RaisedDummy* is a dummy variable that equals one if any funds have been raised at the end of the fundraising stage and zero otherwise. *FundsRaised (log)* is defined as the logarithm of the total amount raised at the end of the fundraising stage. *Tokens for sale (%)* is defined as the ratio of token supply to tokens for sale. *Hardcap (log)* is defined as the maximum amount allowed to be raised. *Whitelist* is a dummy variable that equals one if the project offers a whitelist to early investors and zero otherwise. *KYC* is a dummy variable that equals one if the project complies with the "know your customer" requirement and zero otherwise. *White paper* is a dummy variable that equals one if the project disclosed a white paper and zero otherwise. *Team size (log)* is defined as the logarithm of the total number of team members collected from LinkedIn. *Presale* is a dummy variable that equals one if the project attempted a presale (i.e., by an angel, venture capital fund, or other early seed investors) before the fundraising stage. *Twitter (log)* is defined as the logarithm of the total amount of tweets posted on the project's official account before the fundraising stage. *Reddit (log)* is defined as the logarithm of the total amount of Reddit discussions before the fundraising stage. *BTCTalk (log)* is defined as the logarithm of the total amount of Bitcointalk posts before the fundraising stage. *Medium (log)* is defined as the logarithm of the total amount of medium articles before the fundraising stage.

	Obs.	Mean	SD	p25	p50	p75
Dependent variables:						
RaisedDummy	2,326	0.56	0.50	0	1	1
FundsRaised (log)	2,326	8.80	7.29	0.69	12.79	15.64
Variables of interest:						
Open Source	2,326	0.66	0.48	0	0	1
Code Production (log)	2,326	1.73	2.48	0	0	3.09
Code-producer	2,326	0.13	0.34	0	0	0
Code-washer	2,326	0.10	0.30	0	0	0
Controls:						
Tokens for sale (%)	2,326	0.57	0.23	0.40	0.59	0.70
Hardcap (log)	2,326	16.71	1.19	16.12	16.81	17.37
Whitelist	2,326	0.51	0.50	0	1	1
KYC	2,326	0.68	0.47	0	1	1
White paper	2,326	0.47	0.50	0	0	1
Team size (log)	2,326	2.18	0.82	1.61	2.30	2.77
Presale	2,326	0.68	0.47	0	1	1
Twitter (log)	2,326	2.82	2.24	0	3.33	4.68
Reddit (log)	2,326	1.49	2.69	0	0	1.79
BTCTalk (log)	2,326	3.06	2.73	0	3.50	5.39
Medium (log)	2,326	0.65	1.30	0	0	0

Table 2. The relevance of code activity for fundraising success. This table reports the regression results of equation (1), capturing the blockchain startup characteristics associated with fundraising success. The dependent variable in columns (i) and (ii) is *RaisedDummy*—an indicator variable equaling one if the blockchain startup successfully raised in and zero otherwise. The dependent variable in columns (iii) and (iv) is *FundsRaised (log)*—the logarithm of the total funds raised by the blockchain startup. The variables of interest are *Open Source*, which is an indicator variable for whether the blockchain startup has a GitHub account before the fundraising stage, and *Code Production (log)*, which is the logarithm of the total number of commits before the fundraising stage. Table 1 describes the remaining control variables. We estimate the Logit regressions in columns (i) and (ii) and OLS regressions in columns (iii) and (iv). All regression specifications include year-month, industry, and geographic region fixed effects. The standard errors are clustered at the venture-fundraising completion date, and t-statistics are reported in parentheses. The symbols *, **, and *** denote significance at the 10%, 5%, and 1% levels, respectively.

	(i) ***RaisedDummy***	(ii) ***RaisedDummy***	(iii) ***FundsRaised (log)***	(iv) ***FundsRaised (log)***
Variables of interest:				
Open Source	0.072***		1.003***	
	(3.998)		(3.412)	
Code Production (log)		0.113***		0.307***
		(4.555)		(4.904)
Controls:				
Tokens for sale (%)	−0.071*	−0.350	−1.438**	−1.290*
	(−1.677)	(−1.503)	(−2.201)	(−1.949)
Hardcap (log)	0.003	0.010	0.378***	0.373***
	(0.308)	(0.214)	(3.128)	(3.007)
Whitelist	−0.014	−0.115	0.117	−0.020
	(−0.675)	(−1.013)	(0.330)	(−0.058)
KYC	0.156***	0.856***	2.452***	2.483***
	(4.616)	(4.408)	(5.395)	(5.303)
White paper	0.040**	0.207*	0.688**	0.661**
	(2.087)	(1.950)	(2.660)	(2.528)
Team size (log)	0.083***	0.455***	1.293***	1.312***
	(6.171)	(6.300)	(6.103)	(6.041)
Presale	−0.013	−0.046	−0.262	−0.200
	(−0.604)	(−0.391)	(−0.880)	(−0.666)
Twitter (log)	−0.001	−0.009	−0.018	−0.023
	(−0.275)	(−0.423)	(−0.288)	(−0.365)
Reddit (log)	−0.009**	−0.055**	−0.130**	−0.150**
	(−2.071)	(−2.435)	(−2.090)	(−2.376)
BTCTalk (log)	0.028***	0.155***	0.402***	0.415***
	(8.858)	(9.477)	(8.129)	(8.896)
Medium (log)	0.012	0.048	0.213*	0.173
	(1.510)	(1.048)	(1.888)	(1.461)
Year-month FE	Yes	Yes	Yes	Yes
Industry FE	Yes	Yes	Yes	Yes
Geographic region FE	Yes	Yes	Yes	Yes
Obs.	2,326	2,326	2,368	2,368
Pseudo-r^2 / Adj.r^2	0.198	0.198	0.282	0.277

Table 3. Code production and fundraising (pooled equilibrium). This table reports the regression results of equation (2), capturing the relation between code-washers and code-producers and fundraising success. The dependent variable in column (I) is *RaisedDummy*—an indicator variable equaling one if the blockchain startup successfully raised in and zero otherwise. The dependent variable in column (ii) is *FundsRaised (log)*—the logarithm of the total funds raised by the blockchain startup. The variables of interest are as follows: Code-producer is a startup in the top quartile of cumulative commits generated before the fundraising stage, and Code-washer is a startup in the bottom quartile of cumulative commits generated before the fundraising stage. Table 1 describes the remaining control variables. We estimate Logit regressions in column (i) and OLS regressions in column (ii). All regression specifications include year-month, industry, and geographic region fixed effects. The standard errors are clustered at the venture-fundraising completion date, and t-statistics are reported in parentheses. The symbols *, **, and *** denote significance at the 10%, 5%, and 1% levels, respectively.

	(i) ***RaisedDummy***	(ii) ***FundsRaised (log)***
Variables of interest:		
β_1(*Code-producer*)	0.539**	1.612***
	(2.458)	(3.213)
β_2(*Code-washer*)	0.362**	1.015**
	(2.254)	(2.554)
Controls:		
Tokens for sale (%)	−0.362	−1.371**
	(−1.560)	(−2.073)
Hardcap (log)	0.005	0.355***
	(0.106)	(2.953)
Whitelist	−0.086	0.074
	(−0.731)	(0.205)
KYC	0.862***	2.493***
	(4.376)	(5.282)
White paper	0.201*	0.636**
	(1.926)	(2.523)
Team size (log)	0.459***	1.305***
	(6.358)	(6.211)
Presale	−0.049	−0.191
	(−0.412)	(−0.629)
Twitter (log)	−0.002	−0.008
	(−0.112)	(−0.126)
Reddit (log)	−0.049**	−0.138**
	(−2.186)	(−2.219)
BTCTalk (log)	0.160***	0.426***
	(9.346)	(9.014)
Medium (log)	0.059	0.197
	(1.251)	(1.646)
Year-month FE	Yes	Yes
Industry FE	Yes	Yes
Geographic region FE	Yes	Yes
Obs.	2,326	2,368
Pseudo-r^2 / Adj.r^2	0.198	0.284
$H_0 : \beta_1 = \beta_2$	Pr. $> \chi^2 = 0.497$	Pr. $> \chi^2 = 0.317$

Table 4. Robustness checks: Heckman sample correction and matching. This table reports the results of the Heckman two-step regressions in the first three columns and the results of equation (2) using a propensity-score matched sample in the last two columns. The dependent variable in column (i) is *OpenSource*—an indicator variable equaling one if the blockchain startup has at least one repository on GitHub and zero otherwise. The dependent variable in columns (ii) and (iv) is *RaisedDummy*—an indicator variable equaling one if the blockchain startup successfully raised in and zero otherwise. The dependent variable in columns (iii) and V is *FundsRaised (log)*—the total funds raised by the blockchain startup expressed in logarithm. The variables of interest are as follows: *Code-produce*r is a startup in the top quartile of cumulative commits generated before the fundraising stage, and *Code-washer* is a startup in the bottom quartile of cumulative commits generated before the fundraising stage. Table 1 describes the remaining control variables. *IMR* is the inverse Mills ratio, calculated as the ratio of the probability density function (PDF) to the cumulative distribution function (CDF) of the standard normal distribution, using an estimated linear predictor from the first-stage model. In columns IV and V, the control group is 3:1 formed by matching each code-washer startup with the closest propensity score, where the closest propensity score is estimated using *Tokens for sale (%)*, *Hardcap (log)*, *Whitelist*, *KYC*, *White paper*, *Team size (log)*, *Presale*, *Twitter (log)*, *Reddit (log)*, *BTCTalk (log)*, *Medium (log)*. We report the *t*-statistics for the differences in mean values between the code-washers and matched control firms in Appendix A3. All regression specifications include year-month, industry, and geographic region fixed effects. The standard errors are clustered at the venture-fundraising completion date, and *t*-statistics are reported in parentheses. The symbols *, **, and *** denote significance at the 10%, 5%, and 1% levels, respectively.

	(i) *OpenSource*	(ii) *RaisedDummy*	(iii) *FundsRaised (log)*	(iv) *RaisedDummy*	(v) *FundsRaised (log)*
	First Stage	**Second Stage**		**PSM sample**	
Variables of interest:					
β_1(*Code-producer*)		0.042*	0.896*	0.092**	1.627***
		(1.703)	(2.454)	(2.475)	(3.529)
β_2(*Code-washer*)		0.053**	0.914**	0.062**	0.997**
		(2.002)	(3.262)	(2.199)	(2.530)
Controls:					
Tokens for sale (%)	−0.229*	−0.006	−0.566	−0.102*	−1.775**
	(−1.915)	(−0.120)	(−0.561)	(−1.826)	(−2.099)
Hardcap (log)	−0.090***	0.023	0.614	0.007	0.483***
	(−3.889)	(0.889)	(1.853)	(0.670)	(3.379)
Whitelist	0.051	−0.027	−0.016	−0.001	0.390
	(0.830)	(−1.011)	(−0.044)	(−0.043)	(0.915)
KYC	0.335***	0.064	1.522	0.158***	2.268***
	(5.298)	(0.667)	(1.258)	(4.210)	(4.764)
White paper	0.069	0.011	0.269	0.011	0.316
	(1.214)	(0.461)	(0.810)	(0.471)	(0.925)
Team size (log)	0.193***	0.023	0.554	0.088***	1.375***
	(5.390)	(0.513)	(1.112)	(4.455)	(4.331)
Presale	0.029	−0.011	−0.118	−0.007	−0.227
	(0.477)	(−0.350)	(−0.266)	(−0.239)	(−0.603)
Twitter (log)	0.038***	−0.012	−0.140	0.000	0.015
	(2.906)	(−1.527)	(−1.485)	(0.004)	(0.193)
Reddit (log)	0.015	−0.014***	−0.210***	−0.004	−0.077
	(1.269)	(−4.200)	(−8.738)	(−0.814)	(−0.953)
BTCTalk (log)	0.105***	−0.003	0.013	0.026***	0.408***
	(9.760)	(−0.109)	(0.038)	(8.042)	(7.746)
Medium (log)	0.065***	−0.004	−0.024	0.011	0.215
	(2.717)	(−0.183)	(−0.091)	(1.004)	(1.487)
IMR		−0.603	−7.445		
		(−1.304)	(−1.431)		
Year-month FE	No	Yes	Yes	Yes	Yes
Industry FE	No	Yes	Yes	Yes	Yes
Geographic region FE	No	Yes	Yes	Yes	Yes
Obs.	2,326	1,185	1,218	1,180	1,221
Pseudo-r^2 / Adj.r^2	0.198	0.198	0.290	0.223	0.311
$H_0 : \beta_1 = \beta_2$		Pr. $> \chi^2$ = 0.7519	Pr. $> \chi^2$ = 0.9640	Pr. $> \chi^2$ = 0.8209	Pr. $> \chi^2$ = 0.9772

Table 5. Exogenous shock: SEC crackdown. This table reports the regression results of equation (3), examining how exogenous shocks of the SEC's regulatory crackdown on code quality affect fundraising success. The dependent variable in column (i) is *RaisedDummy*—an indicator variable equaling one if the blockchain startup successfully raised funds and zero otherwise. The dependent variable in column (ii) is *FundsRaised (log)*—the total funds raised by the blockchain startup expressed in the logarithm. The variables of interest are as follows: *Code-producer* is a startup in the top quartile of cumulative commits generated before the fundraising stage, and *Code-washer* is a startup in the bottom quartile of cumulative commits generated before the fundraising stage. Table 1 describes the remaining control variables. *Post* is an indicator variable equaling one if the blockchain startup's fundraising attempt occurred after the SEC's regulatory crackdown (November 2018) and zero otherwise. The variables of interest capture the interaction terms of *Post* and startup type (i.e., *Code-producer* or *Code-washer*). All regression specifications include year-month, industry, and geographic region fixed effects. The standard errors are clustered at the venture-fundraising completion date, and *t*-statistics are reported in parentheses. The symbols *, **, and *** denote significance at the 10%, 5%, and 1% levels, respectively.

	(i) *RaisedDummy*	(ii) *FundsRaised (log)*
Variables of interest:		
β_1(*Post* × *Code-washer*)	−0.183**	−2.488*
	(−2.432)	(−2.054)
β_2(*Post* × *Code-producer*)	−0.104	−1.507
	(−0.744)	(−0.816)
β_3(*Code-washer*)	0.093**	1.498**
	(2.394)	(2.589)
β_4(*Code-producer*)	0.099*	1.715*
	(1.658)	(2.052)
β_5(*Post*)	−0.213***	−1.710***
	(−8.886)	(−6.187)
Controls:		
Tokens for sale (%)	−0.092	−1.831
	(−1.277)	(−1.682)
Hardcap (log)	0.002	0.337**
	(0.219)	(2.749)
Whitelist	−0.003	0.174
	(−0.105)	(0.408)
KYC	0.148***	2.121***
	(3.213)	(3.456)
White paper	0.041	0.599
	(1.608)	(1.738)
Team size (log)	0.074***	1.082***
	(4.857)	(4.496)
Presale	0.028	0.217
	(0.869)	(0.480)
Twitter (log)	−0.006	−0.127
	(−1.015)	(−1.442)
Reddit (log)	−0.011*	−0.166*
	(−1.685)	(−1.833)
BTCTalk (log)	0.031***	0.455***
	(6.924)	(6.387)
Medium (log)	0.018	0.258
	(1.471)	(1.429)
Year-month FE	Yes	Yes
Industry FE	Yes	Yes
Geographic region FE	Yes	Yes
Obs.	1,354	1,354
Pseudo-r^2 / Adj.r^2	0.127	0.178

Table 6. Code production, information quality, and fundraising (separated equilibrium). This table reports the regression results for equation (2) in the subsample, capturing the relationship between blockchain startup code quality and its financial proceedings' success in the subset of high information quality. *High information coverage* indicates that a blockchain startup ranks above the median of all startups in terms of information coverage, as measured by the number of sources with available data characterizing the startup fundraising details. *High information quality* indicates that a blockchain startup ranks above the median of all startups in terms of information quality, as measured by a function of the total number of sources with available data and average consistency across sources following Lyandres et al. (2022)—See Appendix B. The dependent variable in columns (i) and (iii) is *RaisedDummy*—an indicator variable equaling one if the blockchain startup successfully raided funds and zero otherwise. The dependent variable in columns (ii) and (iv) is *FundsRaised (log)*—the total funds raised by the blockchain startup expressed in logarithm. The variables of interest are as follows: *Code-producer* is a startup in the top quartile of cumulative commits generated before the fundraising stage, and *Code-washer* is a startup in the bottom quartile of cumulative commits generated before the fundraising stage. Table 1 describes the remaining control variables. We also report the Chi-Squared test results for the equality: $H_0: \beta_1 = \beta_2$. The control variables are described in Table 1. All regression specifications include year-month, industry, and geographic region fixed effects. The standard errors are clustered at the venture-fundraising completion date, and *t*-statistics are reported in parentheses. The symbols *, **, and *** denote significance at the 10%, 5%, and 1% levels, respectively.

	(i) ***RaisedDummy***	(ii) ***FundsRaised (log)***	(iii) ***RaisedDummy***	(iv) ***FundsRaised (log)***
	High information coverage		**High information quality**	
Variables of interest:				
β_1(*Code-producer*)	1.317***	1.748***	0.929**	1.490***
	(4.241)	(4.415)	(2.467)	(3.247)
β_2(*Code-washer*)	0.076	0.229	0.250	0.344
	(0.311)	(0.715)	(1.109)	(1.020)
Controls:				
Tokens for sale (%)	−0.641	−1.428*	−0.614	−1.464**
	(−1.279)	(−2.049)	(−1.473)	(−2.280)
Hardcap (log)	0.041	0.643***	0.017	0.611***
	(0.676)	(5.953)	(0.328)	(6.146)
Whitelist	−0.031	0.354	−0.033	0.343
	(−0.135)	(0.833)	(−0.121)	(0.698)
KYC	0.602**	1.423***	0.612**	1.502***
	(2.289)	(3.121)	(2.003)	(2.999)
White paper	0.409**	0.740**	0.418**	0.744**
	(2.152)	(2.723)	(2.046)	(2.426)
Team size (log)	0.346***	0.710***	0.505***	1.002***
	(4.433)	(3.790)	(4.964)	(4.332)
Presale	−0.258	−0.374	−0.125	−0.180
	(−1.095)	(−1.144)	(−0.662)	(−0.673)
Twitter (log)	−0.029	−0.053	−0.015	−0.021
	(−0.599)	(−0.734)	(−0.356)	(−0.319)
Reddit (log)	0.008	0.015	−0.015	−0.040
	(0.300)	(0.289)	(−0.480)	(−0.704)
BTCTalk (log)	0.159***	0.263***	0.138***	0.238***
	(4.640)	(4.933)	(4.210)	(5.332)
Medium (log)	0.051	0.196*	0.096	0.218
	(0.575)	(1.804)	(0.932)	(1.692)
Year-month FE	Yes	Yes	Yes	Yes
Industry FE	Yes	Yes	Yes	Yes
Geographic region FE	Yes	Yes	Yes	Yes
Obs.	1,039	1,163	1,081	1,163
Pseudo-r^2 / Adj.r^2	0.198	0.270	0.198	0.280
$H_0: \beta_1 = \beta_2$	Pr. $> \chi^2$ = 0.006**	Pr. $> \chi^2$ = 0.010**	Pr. $> \chi^2$ = 0.187	Pr. $> \chi^2$ = 0.071*

Table 7. Code production, investor attention, and fundraising (separated equilibrium). This table reports the regression results for equation (2) in a subsample, capturing the relationship between blockchain startup code quality and its financial proceedings' success in the subset of low Ether (ETH) returns. *Low market returns (Ether)* indicate that a blockchain startup's fundraising attempt occurs in a month when ETH returns are below the entire sample median. The dependent variable in column (i) is *RaisedDummy*—an indicator variable equal to one if the blockchain startup successfully raised funds and zero otherwise. The dependent variable in column (ii) is *FundsRaised (log)*—the total funds raised by the blockchain startup expressed in the logarithm. The variables of interest are as follows: *Code-producer* is a startup in the top quartile of cumulative commits generated before the fundraising stage, and *Code-washer* is a startup in the bottom quartile of cumulative commits generated before the fundraising stage. Table 1 describes the remaining control variables. We also report the Chi-Squared test results for the equality: $H_0 : \beta_1 = \beta_2$. All regression specifications include year-month, industry, and geographic region fixed effects. The standard errors are clustered at the venture-fundraising completion date, and t-statistics are reported in parentheses. The symbols *, **, and *** denote significance at the 10%, 5%, and 1% levels, respectively.

	(i) ***RaisedDummy***	(ii) ***FundsRaised (log)***
	Low market returns (Ether)	
Variables of interest:		
β_1(*Code-producer*)	0.534*	1.767**
	(1.897)	(2.353)
β_2(*Code-washer*)	0.145	0.319
	(0.757)	(0.540)
Controls:		
Tokens for sale (%)	−0.490	−1.464
	(−1.437)	(−1.435)
Hardcap (log)	−0.022	0.243**
	(−0.574)	(2.519)
Whitelist	−0.279*	−0.621
	(−1.814)	(−1.318)
KYC	0.790**	2.211***
	(2.576)	(2.944)
White paper	0.162	0.482
	(0.937)	(1.070)
Team size (log)	0.363***	0.973***
	(3.399)	(3.236)
Presale	0.035	−0.024
	(0.159)	(−0.042)
Twitter (log)	−0.021	−0.053
	(−0.586)	(−0.452)
Reddit (log)	−0.061**	−0.161*
	(−2.193)	(−1.904)
BTCTalk (log)	0.145***	0.391***
	(5.513)	(4.724)
Medium (log)	0.081**	0.249**
	(2.086)	(2.257)
Year-month FE	Yes	Yes
Industry FE	Yes	Yes
Geographic region FE	Yes	Yes
Obs.	1,039	1,186
Pseudo-r^2 / Adj.r^2	0.198	0.238
$H_0 : \beta_1 = \beta_2$	Pr. > χ^2 = 0.159	Pr. > χ^2 = 0.0893*

Table 8. Code production and startup long-term performance. This table reports the regression results for equation (4), capturing the relation between blockchain startup code production and long-term performance, including fundraising, financial performance, and technology development. The dependent variable in column (i) is *Listing*—an indicator variable equaling one if the blockchain startup succeeds in listing on at least one exchange and zero otherwise. The dependent variable in column (ii) is *Returns (log)*—the logarithm of cumulative return over the first 180 days from listing. The dependent variable in column (iii) is *Volatility (log)*—the logarithm of return volatility calculated over the first 180 days from listing. The dependent variable in column (iv) is *Illiquidity (log)*—the logarithm of illiquidity calculated over the 180 days from listing. The dependent variable in column (v) is *Commits (log)*—the logarithm of cumulative GitHub commits over the 180 days after the fundraising stage. The variables of interest are as follows: *Code-producer* is a startup in the top quartile of cumulative commits generated before the fundraising stage, and *Code-washer* is a startup in the bottom quartile of cumulative commits generated before the fundraising stage. Table 1 describes the remaining control variables. We also report the Chi-Squared test results for the equality: $H_0 : \beta_1 = \beta_2$. All regression specifications include year-month, industry, and geographic region fixed effects. The standard errors are clustered at the venture-fundraising completion date, and t-statistics are reported in parentheses. The symbols *, **, and *** denote significance at the 10%, 5%, and 1% levels, respectively.

	(i)	(ii)	(iii)	(iv)	(v)
	Listing	***Returns (log)***	***Volatility (log)***	***Illiquidity (log)***	***Commits (log)***
	Fundraising	**Financial performance**			**Technology**
Variables of interest:					
β_1(*Code-producer*)	0.162***	0.369**	−0.036***	−0.983**	3.677***
	(4.000)	(2.123)	(−3.157)	(−2.353)	(23.795)
β_2(*Code-washer*)	0.084***	0.055	−0.021	−0.492	−0.494***
	(2.951)	(0.542)	(−1.662)	(−1.171)	(−3.584)
Controls:					
Tokens for sale (%)	−0.208***	0.026	0.028	1.489**	−0.674***
	(−5.825)	(0.105)	(1.329)	(2.318)	(−4.143)
Hardcap (log)	0.008	−0.055	−0.007	−0.276	−0.037
	(0.612)	(−1.157)	(−1.561)	(−1.640)	(−0.761)
Whitelist	0.039	−0.057	−0.017*	−0.675*	0.318**
	(1.070)	(−0.425)	(−1.707)	(−1.846)	(2.275)
KYC	0.220***	0.042	−0.040***	−1.408***	0.505***
	(7.058)	(0.240)	(−3.179)	(−4.084)	(3.030)
White paper	0.057**	−0.025	−0.001	−0.014	0.281**
	(2.207)	(−0.207)	(−0.067)	(−0.049)	(2.474)
Team size (log)	0.066***	−0.113	−0.013	−0.518**	0.031
	(3.822)	(−1.338)	(−1.453)	(−2.218)	(0.394)
Presale	0.032**	−0.166	−0.007	0.468	−0.220**
	(2.026)	(−1.192)	(−0.615)	(1.091)	(−2.403)
Twitter (log)	−0.009*	0.046**	−0.004*	−0.057	0.013
	(−1.900)	(2.252)	(−1.807)	(−0.994)	(0.518)
Reddit (log)	−0.035***	0.019	0.002	0.017	−0.003
	(−5.413)	(0.488)	(0.956)	(0.211)	(−0.145)
BTCTalk (log)	0.012***	−0.012	0.001	0.053	0.007
	(3.370)	(−0.402)	(0.440)	(1.308)	(0.342)
Medium (log)	0.021***	−0.000	−0.004*	−0.091	0.106***
	(3.775)	(−0.005)	(−1.990)	(−0.877)	(3.057)
Year-month FE	Yes	Yes	Yes	Yes	Yes
Industry FE	Yes	Yes	Yes	Yes	Yes
Geographic region FE	Yes	Yes	Yes	Yes	Yes
Obs.	1,292	639	639	638	1,075
Pseudo-r^2 / Adj.r^2	0.335	0.289	0.227	0.256	0.591
$H_0 : \beta_1 = \beta_2$ Pr. > χ^2 =	0.064*	0.067*	0.392	0.430	0.000***

Appendix A: Additional Figures and Tables

Figure A1. An example of a code-washer startup. This figure illustrates the GitHub repository page of a blockchain startup named "Bitconnect". The GitHub repository shows that BitConnect created one repository in 2016 and had nearly zero code production (only seven commits) right before its main fundraising event. Bitconnect reached a peak market capitalization of $3.4 billion and defrauded investors of $2.69 billion through a "lending program" that operated as a Ponzi scheme. See https://www.sec.gov/newsroom/press-releases/2021-172.

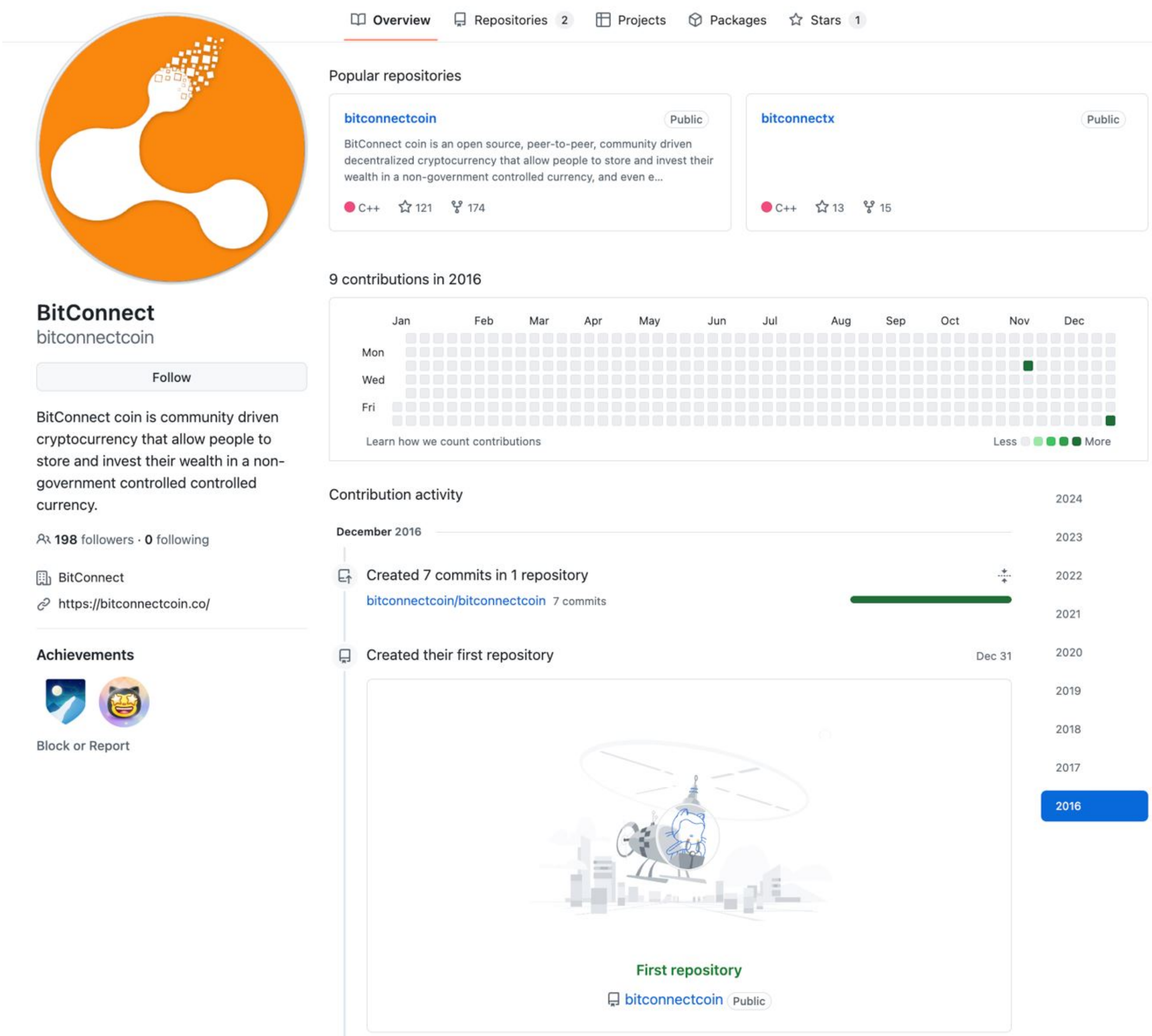


Figure A2. Distribution of firms by code production. This figure illustrates the distribution of firms by code production. The height of each bin represents the total number of startups, while each bin corresponds to a range of code production, measured by the total number of cumulative commits generated prior to the fundraising stage.

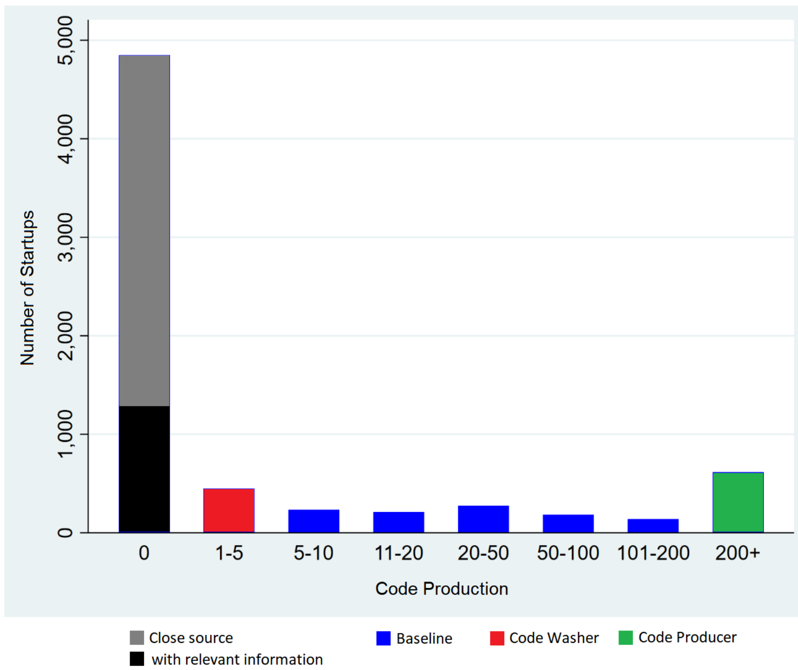
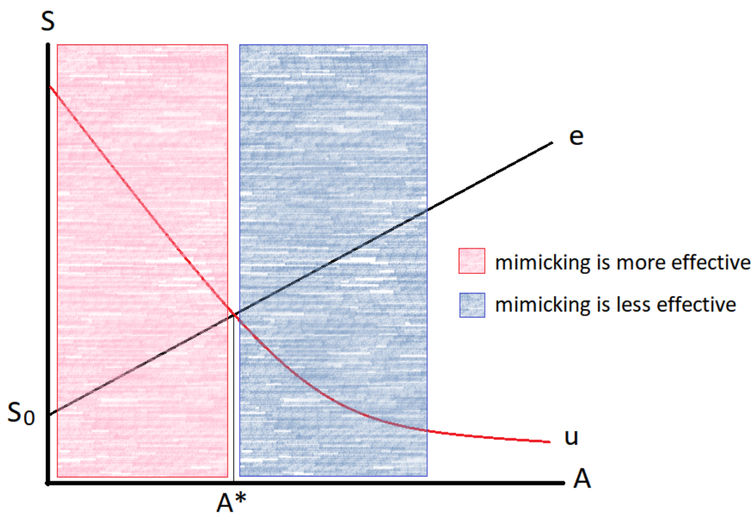


Figure A3. Multiple equilibria and investors' attention. This figure illustrates the economic framework of the interplay between code production and its function as a credible signal of venture quality under different market conditions, accounting for investors' ability to interpret the signal accurately.



S = Cost of Signal
S0 = Inital cost (Github)
e = equilibrium
A = investors attention

Dynamics:
A < A* (pooled equilibria)
A > A* (separated equilibria)

State A < A*, good type produce more code trying to diffenrentiate from mimicking type as attentive investors proportion is low. Extreme case: at zero-information set, even the firm with largest code production cannot differentiate from bad type, it's valued at discount.

State A > A*, it's enough for good type to produce marginally above S0 to differentiate from mimicking type as attentive investors proportion is high. Extreme case: at full-information, shy code production is enough to reveal firms type, it's valued at premium.

Notes:

Code production is costly: it involves the developers' effort to produce code (e.g., financial resources such as salaries, management, infrastructure, and time)

S0 = denotes some proprietary costs of open sourcing

u = denotes the inability of code production to signal firm's type

We assume investors attention to be parallel to understanding code production information

Figure A4. A token offering scam case. This figure illustrates a scam case involving a token offering (TO) by Sirin Labs. Sirin Labs gained widespread attention with its high-profile promise to deliver the world's first fully blockchain-powered smartphone, raising $158 million in December 2017. However, plaintiffs claim that it never developed the products it promised investors, alleging the defendants misappropriated investors' funds for personal use. In our sample, Sirin Labs is classified as a code-washer, as it opened a GitHub account and posted its first commit on Sep 26, 2017—less than three months prior to the TO start date. This timing suggests an intentional attempt to create a facade of active development and mislead investors about the legitimacy of their project. For more details, see: `https://www.investing.com/news/cryptocurrency-news/3-israeli-icos-that-raised-250m-were-allegedly-fraud-2519946`

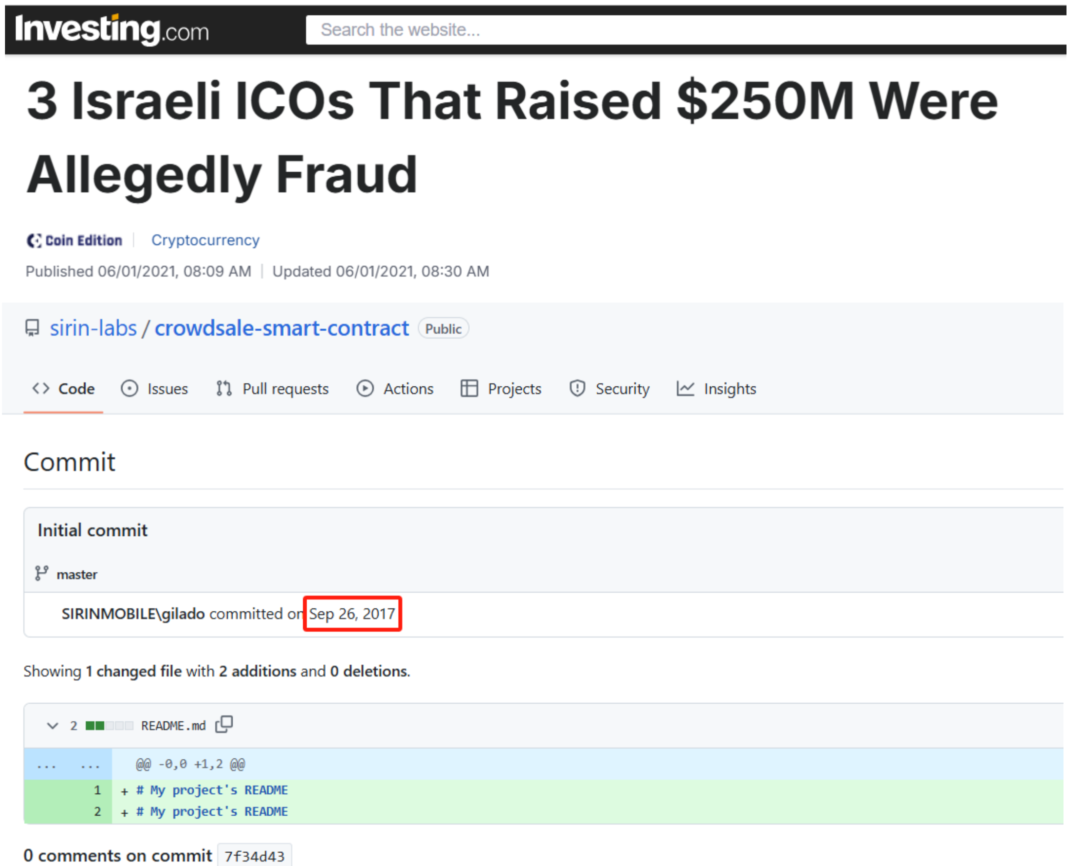


Table A1. Definitions of the variables used in the empirical analysis.

Variable	Description
RaisedDummy	a dummy variable that equals one if any funds have been raised at the end of the fundraising stage and zero otherwise
FundsRaised (log)	the logarithm of the total amount raised at the end of the fundraising stage
OpenSource	an indicator variable for whether the blockchain startup has a GitHub account at the beginning of the fundraising phase
Code Production (log)	the logarithm of the total number of commits before the fundraising stage
Code-producer	a startup in the top quartile of cumulative commits generated before the fundraising stage
Code-washer	a startup in the bottom quartile of cumulative commits generated before the fundraising stage
Tokens for sale (%)	the ratio of token supply to tokens for sale
Hardcap (log)	the maximum amount allowed to be raised
Whitelist	a dummy variable that equals one if the project offers a whitelist to early investors and zero otherwise
KYC	a dummy variable that equals one if the project complies with the ”know your customer” and zero otherwise
White paper	a dummy variable that equals one if the project disclosed a white paper, and zero otherwise
Team size (log)	the logarithm of the total number of team members collected from LinkedIn
Presale	a dummy variable that equals one if the project attempted a presale (i.e., by angel, venture capital, and other early seed
	investors) before the fundraising stage
Twitter (log)	the logarithm of the total amount of tweets posted in the project's official account before the fundraising stage
Reddit (log)	the logarithm of the total amount of Reddit discussions before the fundraising stage
BTCTalk (log)	the logarithm of the total amount of Bitcointalk posts before the fundraising stage
Medium (log)	the logarithm of the total amount of medium articles before the fundraising stage
Listing	an indicator variable equaling one if the blockchain startup succeeds listing in at least one exchange and zero otherwise
Returns (log)	the logarithm of cumulative return over the first 180 days from listing
Volatility (log)	the logarithm of return volatility calculated over the first 180 days from listing
Illiquidity (log)	the logarithm of illiquidity calculated over the 180 days from listing
Commits (log)	the logarithm of cumulative GitHub commits over the 180 days after the fundraising stage

Table A2. Exogenous shocks on coding quality. This table reports the effects of the SEC's increased enforcement of penalties against crypto founders for abuses and fraud on the probability of blockchain startups engaging in fraudulent code-commit activity. The dependent variable in column (i) *code-washer*—indicates whether the blockchain startup is a *code-washer* (i.e., at the bottom quartile of cumulative commits generated before the fundraising stage). The dependent variable in column (ii) *code-producer*—indicates whether the blockchain startup is a *code-producer* (i.e., at the top quartile of cumulative commits generated before the fundraising stage). The variables of interest *Post* is an indicator variable equaling one if the blockchain startup's TO occurred after the SEC's regulatory crackdown on new coin offerings in November 2018 and zero otherwise. The control variables are described in Table 1. We estimate Logit regressions in columns (i) and (ii). All regression specifications include year-month, industry, and geographic region fixed effects. The standard errors are clustered at the venture-fundraising completion date, and *t*-statistics are reported in parentheses. The symbols *, **, and *** denote significance at the 10%, 5%, and 1% levels, respectively.

	(i) ***code-washer***	(ii) ***code-producer***
Variables of interest:		
Post	−1.259***	0.053***
	(−11.623)	(4.059)
Controls:		
Tokens for sale (%)	−0.052	−0.042
	(−1.329)	(−1.209)
Hardcap (log)	−0.008	−0.006
	(−1.006)	(−1.030)
Whitelist	0.007	0.037**
	(0.280)	(2.096)
KYC	0.058*	−0.012
	(1.749)	(−0.674)
White paper	0.021	−0.002
	(1.190)	(−0.121)
Team size (log)	0.008	0.036***
	(0.477)	(3.017)
Presale	−0.014	−0.005
	(−0.691)	(−0.159)
Twitter (log)	−0.001	0.000
	(−0.249)	(0.025)
Reddit (log)	0.001	0.010***
	(0.265)	(2.993)
BTCTalk (log)	0.005*	0.002
	(1.745)	(0.642)
Medium (log)	−0.002	0.020***
	(−0.596)	(4.927)
Year-month FE	Yes	Yes
Industry FE	Yes	Yes
Geographic region FE	Yes	Yes
Obs.	1,313	1,354
Pseudo-r^2	0.051	0.141

Table A3. Summary statistics for the propensity matching sample. This table reports the characteristics at the startup level for the subsample of code-washers and control groups. Panel A presents the unmatched sample, and Panel B presents the matched sample. A *code-washer* is defined as a startup in the bottom quartile of cumulative commits generated before the fundraising stage, and a *code-producer* is defined as a startup at the top quartile of cumulative commits generated before the fundraising stage. The control group is 3:1 formed by matching each *code-washer* with the closest propensity score, where the closest propensity score is estimated using *Tokens for sale (%)*, *Hardcap (log)*, *Whitelist*, *KYC*, *White paper*, *Team size (log)*, *Presale*, *Twitter (log)*, *Reddit (log)*, *BTCTalk (log)*, and *Medium (log)*. We report the mean of each variable for the control group in column (i), for code-washers in column (ii), and the t-statistics for the differences in mean values between code-washers and matched startups. The symbols *, **, and *** denote significance at the 10%, 5%, and 1% levels, respectively.

Panel A: Unmatched Sample

	Code-washer		**Code-producer**		**Difference**	
	#	**Mean**	**#**	**Mean**	**Diff**	**t-value**
Code production	313	0.14	236	3503.64	3503.50***	6.92
Code production (log)	313	2.75	236	7.13	4.38***	32.88
RaisedDummy	313	0.71	236	0.71	0.01	0.15
FundsRaised (log)	313	10.85	236	11.22	0.37	0.60
Tokens for sale (%)	313	0.54	236	0.52	−0.02	−0.790
Hardcap (log)	313	16.72	236	16.74	0.80	0.401
Whitelist	313	0.51	236	0.67	0.16***	3.761
KYC	313	0.79	236	0.78	−0.01	−0.177
White paper	313	0.54	236	0.57	0.03	0.673
Team size (log)	313	2.35	236	2.39	0.04	0.631
Presale	313	0.63	236	0.66	0.03	0.737
Twitter (log)	313	2.88	236	3.18	0.30*	1.506
Reddit (log)	313	1.50	236	2.46	0.96***	3.739
BTCTalk (log)	313	3.58	236	3.65	0.07	0.306
Medium (log)	313	0.69	236	1.24	0.55***	4.381

Panel B: Matched Sample

	(i) **Control group**	(ii) **Code-washer**	(iii) **Difference**
Tokens for sale (%)	0.543	0.539	−0.005
	(0.237)	(0.233)	(0.016)
Hardcap (log)	16.770	16.711	−0.059
	(1.164)	(1.159)	(0.080)
Whitelist	0.512	0.503	−0.008
	(0.500)	(0.501)	(0.034)
KYC	0.760	0.789	0.029
	(0.427)	(0.408)	(0.029)
White paper	0.505	0.550	0.045
	(0.500)	(0.498)	(0.034)
Team size (log)	2.273	2.336	0.063
	(0.780)	(0.853)	(0.055)
Presale	0.637	0.616	−0.021
	(0.481)	(0.487)	(0.033)
Twitter (log)	2.964	2.871	−0.093
	(2.233)	(2.208)	(0.152)
Reddit (log)	1.513	1.498	−0.015
	(2.669)	(2.681)	(0.183)
BTCTalk (log)	3.333	3.585	0.252
	(2.752)	(2.667)	(0.186)
Medium (log)	0.672	0.698	0.027
	(1.325)	(1.297)	(0.090)
Obs.	651	313	964

Table A4. Validation test: Whitepaper information. This table reports the relationship between *code-washers*, *code-producers*, and information disclosure in the white paper. The dependent variable in column (i) is *Words per Page (log)*—the count of words per page in the start-up's white paper expressed in the logarithm. The dependent variable in column (ii) is *Image count (log)*—the logarithm of one plus the count of images in the start-up's white paper. The dependent variable in column (iii) is *Tech word count (log)*—the logarithm of the count of technology-related words in the start-up's white paper, identified using natural language processing techniques. The variables of interest capture an indicator variable for whether the start-up is a *code-producer* (i.e., at the top quartile of cumulative commits generated before the fundraising stage) or a *code-washer* (i.e., at the bottom quartile of cumulative commits generated before the fundraising stage). Table 1 describes the remaining control variables. We estimate OLS regressions in all regression specifications, including year-month, industry, and geographic region fixed effects. The standard errors are clustered at the venture-fundraising completion date, and *t*-statistics are reported in parentheses. The symbols *, **, and *** denote significance at the 10%, 5%, and 1% levels, respectively.

	(i) ***Words per page (log)***	(ii) ***Image count (log)***	(iii) ***Tech word count (log)***
Variables of interest:			
β_1(*Coder-producer*)	0.066*	0.180***	0.105**
	(1.715)	(3.194)	(2.345)
β_2(*Code-washer*)	0.019	−0.034	0.042
	(0.949)	(−0.547)	(0.827)
Controls:			
Tokens for sale (%)	−0.019	0.019	−0.165**
	(−0.414)	(0.241)	(−2.157)
Hardcap (log)	0.013	0.048***	0.077***
	(1.064)	(3.395)	(6.491)
Whitelist	0.050*	0.051	0.096***
	(1.715)	(0.876)	(3.377)
KYC	0.038	0.090**	0.114***
	(1.622)	(2.101)	(4.010)
Team size (log)	0.023	0.081***	0.085***
	(1.612)	(3.853)	(4.497)
Presale	0.017	−0.033	−0.019
	(0.756)	(−0.581)	(−0.576)
Twitter (log)	0.005	0.006	0.013*
	(1.080)	(0.646)	(1.792)
Reddit (log)	0.005**	0.003	0.009*
	(1.950)	(0.334)	(1.740)
BTCTalk (log)	−0.003	0.012*	0.006
	(−0.552)	(1.936)	(0.982)
Medium (log)	0.010	−0.004	0.024**
	(1.093)	(−0.327)	(2.466)
Year-month FE	Yes	Yes	Yes
Industry FE	Yes	Yes	Yes
Geographic region FE	Yes	Yes	Yes
Obs.	1,125	1,125	1,125
Adj.r^2	0.117	0.114	0.230
$H_0 : \beta_1 = \beta_2$	Pr. > χ^2 = 0.348	Pr. > χ^2 = 0.014**	Pr. > χ^2 = 0.374

Table A5. Validation test: Pull requests and issues. This table reports the validation tests for the relationship between code quality and pull request and issue activities on the GitHub platform. The dependent variable in column (i) is *IssuesDummy*—an indicator variable equal to one if the startup created at least one issue prior to the end of token offering (TO); an issue is a tool used to track tasks, enhancements, or bugs in a repository. The dependent variable in column (ii) is *Issues (log)*—defined as the natural logarithm of one plus the cumulative number of issues created before the end of TO. The dependent variable in column (iii) is *PullDummy*—an indicator variable equal to one if the startup made at least one pull request prior to the end of TO; a pull request is a request to merge code changes from one branch into another (typically into the main branch). The dependent variable in column (iv) is *Pull (log)*—defined as the natural logarithm of one plus the cumulative number of pull requests prior to the end of TO. The sample includes only startups with at least one commit by the end of the TO. Table 1 describes the remaining control variables. We estimate the Logit regressions in columns (i) and (iii) and OLS regressions in columns (ii) and (iv). A *code-producer* is defined as a startup falling in the top quartile of cumulative commits generated before the fundraising stage. A *code-washer* is defined as a startup falling in the bottom quartile of cumulative commits generated before the fundraising stage. All regression specifications include year-month, industry, and geographic region fixed effects. The standard errors are clustered at the venture-fundraising completion date, and t-statistics are reported in parentheses. The symbols *, **, and *** denote significance at the 10%, 5%, and 1% levels, respectively.

	#	**Issues (log)**	**Pull (log)**
Code-producer	505	1.808	1.602
Code-washer	587	0.277	0.197

	(i) ***IssuesDummy***	(ii) ***Issues (log)***	(iii) ***PullDummy***	(iv) ***Pull (log)***
Variables of interest:				
Code-producer	0.182***	1.269***	0.169***	1.145***
	(4.968)	(5.892)	(5.758)	(6.109)
Code-washer	0.018	−0.033	−0.018	−0.039
	(0.706)	(−0.517)	(−0.749)	(−0.670)
Controls:				
Tokens for sale (%)	0.015	0.232	0.034	0.298
	(0.249)	(1.126)	(0.730)	(1.504)
Hardcap (log)	−0.015	−0.037	−0.015*	−0.033
	(−1.488)	(−1.633)	(−1.796)	(−1.361)
Whitelist	0.037	0.069	0.020	0.058
	(0.967)	(0.658)	(0.681)	(0.680)
KYC	0.064**	0.264**	0.106***	0.275***
	(2.015)	(2.547)	(3.738)	(3.272)
White paper	0.004	−0.006	0.011	0.015
	(0.106)	(−0.053)	(0.379)	(0.143)
Team size (log)	−0.010	−0.056	−0.008	−0.046
	(−0.615)	(−1.667)	(−0.560)	(−1.425)
Presale	−0.068***	−0.260***	−0.051**	−0.237***
	(−3.066)	(−3.071)	(−2.332)	(−2.803)
Twitter (log)	0.016**	0.074***	0.012**	0.065***
	(2.276)	(3.683)	(2.037)	(3.497)
Reddit (log)	0.004	0.016	0.003	0.015
	(0.959)	(1.069)	(0.662)	(1.051)
BTCTalk (log)	−0.004	−0.029**	−0.006	−0.027*
	(−0.883)	(−2.058)	(−1.248)	(−2.021)
Medium (log)	0.010	0.039	0.011	0.036
	(1.157)	(1.037)	(1.158)	(1.006)
Year-month FE	Yes	Yes	Yes	Yes
Industry FE	Yes	Yes	Yes	Yes
Geographic region FE	Yes	Yes	Yes	Yes
Obs.	1,025	1,035	1,015	1,035
Pseudo-r^2 / Adj.r^2	0.132	0.281	0.175	0.266

Table A6. Alternative specification: Zero Commits (pooled equilibrium). This table reports the robustness checks for the pooled equilibrium shown in Table 3, capturing the relation between *code-washers* and *code-producers* and fundraising success. The *Open-source-washer* is defined as a startup that opens a Github account but produces zero commits before the fundraising stage. An *Open-source-producer* is defined as a startup that produces at least one commit before the fundraising stage. The dependent variable in column (i) is *RaisedDummy*—an indicator variable that equals one if any funds have been raised at the end of the fundraising stage and zero otherwise. The dependent variable in column (ii) is *FundsRaised (log)*—the logarithm of the total amount raised at the end of the fundraising stage. Table 1 describes the remaining control variables. We estimate the Logit regressions in columns (i) and OLS regressions in columns (ii). All regression specifications include year-month, industry, and geographic region fixed effects. The standard errors are clustered at the venture-fundraising completion date, and *t*-statistics are reported in parentheses. The symbols *, **, and *** denote significance at the 10%, 5%, and 1% levels, respectively.

	#	Mean	SD	p25	p50	p75
Open-source-producer	2,326	0.31	0.46	0	0	1
Open-source-washer	2,326	0.35	0.48	0	0	1

	(i)	(ii)
	RaisedDummy	***FundsRaised (log)***
Variables of interest:		
β_1(*Open-source-producer*)	0.086***	1.158***
	(4.023)	(3.392)
β_2(*Open-source-washer*)	0.062***	0.894**
	(2.871)	(2.635)
Controls:		
Tokens for sale (%)	−0.073*	−1.432**
	(−1.707)	(−2.167)
Hardcap (log)	0.003	0.384***
	(0.319)	(3.107)
Whitelist	−0.015	0.088
	(−0.710)	(0.251)
KYC	0.157***	2.490***
	(4.694)	(5.460)
White paper	0.041**	0.708**
	(2.099)	(2.646)
Team size (log)	0.083***	1.310***
	(6.182)	(5.997)
Presale	−0.013	−0.270
	(−0.608)	(−0.901)
Twitter (log)	−0.001	−0.021
	(−0.367)	(−0.332)
Reddit (log)	−0.009**	−0.135**
	(−2.136)	(−2.104)
BTCTalk (log)	0.028***	0.404***
	(8.878)	(7.992)
Medium (log)	0.011	0.211*
	(1.429)	(1.841)
Year-month FE	Yes	Yes
Industry FE	Yes	Yes
Geographic region FE	Yes	Yes
Obs.	2,326	2,326
Pseudo-r^2 / Adj.r^2	0.199	(0.272)
$H_0 : \beta_1 = \beta_2$	Pr. $> \chi^2$ = 0.282	Pr. $> \chi^2$ = 0.421

Table A7. Alternative specification: Zero Commits (separated equilibrium). This table reports the robustness checks for the separated equilibrium shown in Table 6, capturing the relationship between code quality and its financial proceedings' success in the subset of high information quality. *Open-source-washer* is defined as a startup that opens a Github account but discloses zero before the fundraising stage. *Open-source-producer* is defined as a startup that discloses at least one commit before the fundraising stage. *High information coverage* indicates that a blockchain startup ranks above the median of all startups in terms of information coverage, as measured by the number of sources with available data characterizing the TO project. *High information quality* indicates that a blockchain startup ranks above the median of all startups in terms of information quality, as measured by a function of the total number of sources with available data and average consistency across sources for each TO following Lyandres et al. (2022). The dependent variable in columns (i) and (iii) is *RaisedDummy*— a dummy variable that equals one if any funds have been raised at the end of the fundraising stage and zero otherwise. The dependent variable in columns (ii) and (iv) is *FundsRaised (log)*—the logarithm of the total amount raised at the end of the fundraising stage. We also report the Chi-Squared test results for the equality: $H_0 : \beta_1 = \beta_2$. The control variables are described in Table 1. All regression specifications include year-month, industry, and geographic region fixed effects. The standard errors are clustered at the venture-fundraising completion date, and *t*-statistics are reported in parentheses. The symbols *, **, and *** denote significance at the 10%, 5%, and 1% levels, respectively.

	(i) ***RaisedDummy***	(ii) ***FundsRaised (log)***	(iii) ***RaisedDummy***	(iv) ***FundsRaised (log)***
	High information coverage		**High information quality**	
Variables of interest:				
β_1(*Open-source-producer*)	1.317***	1.748***	0.929**	1.490***
	(3.451)	(2.839)	(3.412)	(2.991)
β_2(*Open-source-washer*)	0.004	0.271	0.032	0.649*
	(0.152)	(0.646)	(1.501)	(1.764)
Controls:				
Tokens for sale (%)	−0.092	−1.604**	−0.075	−1.498**
	(−1.447)	(−2.295)	(−1.428)	(−2.263)
Hardcap (log)	0.007	0.674***	0.005	0.651***
	(0.943)	(5.754)	(0.726)	(6.260)
Whitelist	−0.002	0.374	−0.004	0.362
	(−0.080)	(0.874)	(−0.133)	(0.751)
KYC	0.072**	1.441***	0.075**	1.490***
	(2.394)	(3.190)	(2.142)	(3.022)
White paper	0.051**	0.791***	0.052**	0.798**
	(2.193)	(2.879)	(2.141)	(2.682)
Team size (log)	0.043***	0.726***	0.059***	0.989***
	(4.517)	(3.619)	(5.034)	(4.159)
Presale	−0.031	−0.395	−0.017	−0.217
	(−1.134)	(−1.195)	(−0.771)	(−0.797)
Twitter (log)	−0.005	−0.073	−0.003	−0.041
	(−0.802)	(−1.002)	(−0.588)	(−0.638)
Reddit (log)	0.001	0.017	−0.002	−0.037
	(0.296)	(0.328)	(−0.418)	(−0.621)
BTCTalk (log)	0.019***	0.250***	0.015***	0.216***
	(4.903)	(4.929)	(4.004)	(5.318)
Medium (log)	0.008	0.216*	0.012	0.229*
	(0.758)	(1.941)	(1.018)	(1.787)
Year-month FE	Yes	Yes	Yes	Yes
Industry FE	Yes	Yes	Yes	Yes
Geographic region FE	Yes	Yes	Yes	Yes
Obs.	1,039	1,163	1,081	1,163
Pseudo-r^2 / Adj.r^2	0.209	0.266	0.221	0.278
$H_0 : \beta_1 = \beta_2$	Pr. > χ^2 = 0.030**	Pr. > χ^2 = 0.075*	Pr. > χ^2 = 0.110	Pr. > χ^2 = 0.188

Table A8. Additional validation test: Code-washing during the fundraising phase. This table reports the validation tests for *code-washing* activity during the Token Offering (TO) phase. The dependent variable is continuous *code-washing*—defined as the ratio of cumulative commits submitted during the TO phase to the total number of commits at the end of TO. Table 1 describes the remaining control variables. The sample includes only startups with at least one commit by the end of the TO. We estimate OLS regressions in all columns. A *code-producer* is a startup in the top quartile of cumulative commits generated before the fundraising stage. A *code-washer* is a startup in the bottom quartile of cumulative commits generated before the fundraising stage. *Open-source-washer* is an *open-source* startup with zero commits before the fundraising stage. *Open-source-producer* is an *open-source* startup with at least one commit, 90 days before the fundraising stage. In columns (ii) and (iv), the regression specifications include year-month, industry, and geographic region fixed effects. The standard errors are clustered at the venture-fundraising completion date, and *t*-statistics are reported in parentheses. The symbols *, **, and *** denote significance at the 10%, 5%, and 1% levels, respectively.

	#	**Commits during fundraising / All commits**
Code-producer	505	0.064
Code-washer	587	0.221
Open-source-producer	1,432	0.057
Open-source-washer	473	0.257

	(i)	(ii)	(iii)	(iv)
		Code-washing		
Variables of interest:				
Code-producer	0.004	0.001		
	(0.674)	(0.131)		
Code-washer	0.161***	0.173***		
	(13.384)	(8.693)		
Open-source-producer			−0.061***	0.001
			(−5.018)	(0.050)
Open-source-washer			0.139***	0.209***
			(6.513)	(5.814)
Controls:				
Tokens for sale (%)		−0.043		−0.037
		(−1.276)		(−1.096)
Hardcap (log)		−0.000		−0.001
		(−0.031)		(−0.175)
Whitelist		0.004		0.000
		(0.275)		(0.012)
KYC		0.018		0.011
		(0.986)		(0.672)
White paper		−0.004		−0.011
		(−0.247)		(−0.733)
Team size (log)		−0.001		−0.001
		(−0.098)		(−0.117)
Presale		0.003		0.002
		(0.146)		(0.099)
Twitter (log)		0.000		0.001
		(0.079)		(0.257)
Reddit (log)		−0.003		−0.003
		(−1.169)		(−1.180)
BTCTalk (log)		−0.000		0.000
		(−0.003)		(0.011)
Medium (log)		−0.007		−0.008
		(−1.389)		(−1.431)
Year-month FE	No	Yes	No	Yes
Industry FE	No	Yes	No	Yes
Geographic region FE	No	Yes	No	Yes
Obs.	2,023	1,035	2,023	1,035
Pseudo-r^2	0.101	0.159	0.134	0.190

Table A9. Financial misconduct and exit scams. This table reports the regression results for equation (5), capturing the relation between blockchain startup code quality and post-fundraising scam activity. The dependent variable in column (i) is *Number of Scams*—the total number of scam events involving the blockchain startup post-fundraising. The dependent variable in column (ii) is *ScamDummy*—an indicator variable equaling one if the blockchain startup is involved in at least one scam event post-fundraising and zero otherwise. The variables of interest capture an indicator variable indicating whether the blockchain startup has a GitHub account, as well as interaction terms of *Open Source* and startup type, distinguishing whether the startup is a c*ode-producer* or a *code-washer*. The control variables are described in Table 1. We estimate Poisson regressions in column (i) and Logit regressions in column (ii). All regression specifications include year-month, industry, and geographic region fixed effects. The standard errors are clustered at the venture-fundraising completion date, and *t*-statistics are reported in parentheses. The symbols *, **, and *** denote significance at the 10%, 5%, and 1% levels, respectively.

	(i) ***Number of Scams***	(ii) ***ScamDummy***
Variables of interest:		
Open Source	−0.469*	−0.071**
	(−1.738)	(−2.070)
Open Source × *Coder-producer*	0.513	0.061
	(1.181)	(1.145)
Open Source × *Coder-washer*	0.584*	0.076*
	(1.920)	(1.692)
Controls:		
Tokens for sale (%)	−0.523	−0.041
	(−1.166)	(−0.580)
Hardcap (log)	0.305***	0.033***
	(4.164)	(2.714)
Whitelist	−0.400	−0.064*
	(−1.569)	(−1.652)
KYC	0.059	0.014
	(0.247)	(0.401)
White paper	−0.490***	−0.057**
	(−3.020)	(−2.338)
Team size (log)	−0.062	−0.000
	(−0.789)	(−0.019)
Presale	0.095	0.025
	(0.386)	(0.624)
Twitter (log)	−0.024	−0.004
	(−0.521)	(−0.522)
Reddit (log)	0.060	−0.001
	(0.874)	(−0.046)
BTCTalk (log)	0.048	0.006
	(0.954)	(0.832)
Medium (log)	0.038	0.008
	(0.718)	(1.020)
Year-month FE	Yes	Yes
Industry FE	Yes	Yes
Geographic region FE	Yes	Yes
Obs.	671	626
Pseudo-r^2 / Adj.r^2	0.335	0.069

Appendix B: Deriving Measures of Information Quality

Information quality at the data-source Level

A persistent challenge in analyzing data on Token Offerings (TOS) is the inconsistency in the reported values of key characteristics across data aggregators—see Lyandres et al. (2022) for an extended discussion. These inconsistencies pertain to critical variables such as the amount raised, the hardcap, the number of tokens available for sale, and the overall token supply. To address this issue, Lyandres et al. (2022) develops a systematic procedure to identify the most reliable value of a variable x in cases where it is reported by multiple sources but with varying degrees of disagreement. This measure is inversely related to the average disagreement between the value of a variable reported by a given source and the mean value reported by all sources for the same variable. Specifically, for each observation k and source i, the "relative distance" of the reported value $x_{i,k}$ from the consensus (mean) value $\bar{x}_k$ across all sources reporting that variable is calculated.

The relative distance is defined as:

$$\text{Relative Distance}_{i,k} = \left| \frac{x_{i,k} - \bar{x}_k}{x_{i,k} + \bar{x}_k} \right|$$

Where, if $x_{i,k} = \bar{x}_k$, the relative distance equals zero. Conversely, as $x_{i,k}$ approaches zero or diverges significantly from $\bar{x}_k$, the relative distance approaches one.

In the next step, we calculate the average relative distance for each variable x across all observations k reported by source i, referred to as the "mean deviation." This mean deviation serves as a proxy for the quality of a source's data for a given variable, with lower deviations indicating higher information quality. To operationalize these quality assessments, we compute the inverse of each source's mean deviation to serve as a measure of information quality. To normalize these quality scores across variables with differing scales of reliability, we calculate the relative quality of a source's data for a given variable by dividing its inverse mean deviation by the highest inverse mean deviation across all sources reporting that variable. For example, if *CoinGecko* and *CryptoCompare* report mean deviations of 0.084 and 0.130, respectively, the relative quality of their data for "amount raised" is 1 (*CoinGecko*) and 0.646 (*CryptoCompare*), as normalized against the highest quality score.

To select the most reliable value for a given variable, we incorporate two key factors: (a) the degree of agreement among sources and (b) the quality of these sources. For each value reported by a source, we sum the source-variable-level quality measures for all sources reporting that value. This produces a "quality-weighted" count of sources supporting each reported value. The value with the highest quality-weighted count is selected as the most trustworthy. It is crucial to note that our selection process does not simply choose the value closest to the mean across all sources. Instead, we prioritize values reported by sources with the highest cumulative quality measures for that variable. This approach ensures that our method remains robust across all observations and is not biased by any particular observation's characteristics.

Information quality at the venture level

In addition to determining the most reliable values for individual variables, Lyandres et al. (2022) procedure facilitates the construction of an information environment quality measure at the venture level. This measure accounts for three critical considerations:

1. **Coverage:** Information quality should increase with the number of sources reporting details on the venture fundraising event.

2. **Source Quality:** Higher-quality sources should contribute more to the overall measure.

3. **Consistency:** Information quality should decrease with greater disagreement among sources.

We follow Lyandres et al. (2022) and first identify all sources reporting values for the four main token offerings variables. For each variable, we define its consistency as one minus the mean relative difference of this variable across all sources reporting it. The total quality of a variable for a given token offering is then computed as the sum of the quality scores of all sources reporting that variable, multiplied by the consistency of the data for that variable.[17] We use the information environment quality measure at the venture level to analyze the effects of code-washing in a subsample with lower asymmetric information. Section 5.2 provides the empirical analysis. See Lyandres et al. (2022) for further details about the construction of these proxies and extended discussion related to information quality during the fundraising stage of blockchain startups.

[17] Lyandres et al. (2022) comprehensive validates information quality measures.

Appendix C: Model Extensions

This dynamic model in Section 3 can be further extended to multiple funding rounds where startups repeatedly choose signals. Incorporating outside certifiers (auditors, developers, token reviewers) or endogenizing ρ based on macro conditions (e.g., market volatility) would further enhance realism.

Stochastic Signals and Repeated Interaction

To more realistically reflect the signaling behavior in blockchain ecosystems, we extend the dynamic model along four dimensions: (1) a stochastic signal space, (2) a repeated interaction setting across multiple periods, (3) endogenous investor attention driven by market sentiment, and (4) empirical implications for estimation and calibration.

Stochastic Signaling with Continuous Signal Space

Let the signal $s \in \mathbb{R}_+$ be continuous, representing measurable coding activity on open-source platforms (e.g., number of commits, stars, forks, contributor diversity). We define the signal distribution conditional on type $\theta \in \{H, L\}$ as:

$$s \sim \begin{cases} \mathcal{N}(\mu_H, \sigma_H^2) & \text{if } \theta = H \\ \mathcal{N}(\mu_L, \sigma_L^2) & \text{if } \theta = L \end{cases}, \quad \mu_H > \mu_L$$

Investors form posterior beliefs using Bayes' Rule:

$$\mu(H|s) = \frac{\pi \cdot \phi(s; \mu_H, \sigma_H)}{\pi \cdot \phi(s; \mu_H, \sigma_H) + (1 - \pi) \cdot \phi(s; \mu_L, \sigma_L)}$$

Here, $\phi(\cdot)$ is the Gaussian density. This framework admits partial informativeness of signals and avoids unrealistic discrete jumps in beliefs.

Interpretation. High signal realizations are more likely from high-quality firms, but noise (e.g., superficial commits, forked code) allows some low-quality firms to mimic. Investors apply thresholds s^* to decide funding. Over time, distributions can shift due to learning or strategic behavior.

Repeated Game with Multi-Round Funding

Let time be indexed by $t = 1, 2, ..., T$, where each startup seeks funding in each round. At each t, the startup chooses a signal s_t, incurs a signaling cost $c(s_t)$, and receives funding F_t if investor belief $\mu_t(H|s_t) \geq \bar{\mu}$.

Payoff:

$$U_\theta = \sum_{t=1}^{T} \delta^{t-1} \left(F_t(s_t) - c(s_t) + \phi_\theta \cdot 1_{\text{success}_t}\right) - 1_{\text{caught}} \cdot \Delta_\theta$$

Where:

- δ is the discount factor,
- Δ_θ is the reputational cost if misrepresentation is revealed,
- ϕ_θ is the continuation value from retained reputation or adoption.

Interpretation. High-quality firms may invest in costly signals early on to build a reputation and enjoy higher funding in later rounds. Low-quality firms face an intertemporal trade-off between short-term gains and future exclusion or detection. This leads to potential *reputation-building equilibria* where signals are initially mimicked but later diverge.

Endogenous Investor Attention and Market Conditions

Investor scrutiny (denoted by ρ_t) and prior belief π_t evolve over time. Let:

$$\rho_t = f(v_t), \quad \pi_t = g(m_t)$$

where v_t is market volatility (inverse of investor attention), and m_t is a proxy for sentiment (e.g., token prices, ICO volume, funding rounds).

Dynamic Feedback. During hot markets (low v_t, high m_t), ρ_t is low and π_t is high:

- Code-washing becomes more attractive for L types;
- Funding thresholds loosen, leading to pooling equilibria.

During cold markets (high v_t, low m_t), attention increases, leading to:

- Stricter due diligence;
- Greater risk of exposure $\Rightarrow$ separating equilibrium re-emerges.

These extensions allow the model to account for strategic intertemporal trade-offs, signal noise, reputation, and macro market forces.