CODE-WASHING:

EVIDENCE FROM OPEN-SOURCE BLOCKCHAIN STARTUPS*

Ofir Gefen† Daniel Rabetti‡§ Yannan Sun¶ Che Zhang‖

First Draft: Oct 2024 — This Draft: May 2026

Abstract

This study examines how startups strategically manage open-source code disclosures during fundraising. We introduce the concept of *code-washing*, the inflation of source code repositories through superficial coding activity, as a novel mechanism of information manipulation in entrepreneurial settings. Using a global dataset of blockchain startups and linking GitHub activity to detailed fundraising outcomes, we validate our code-washing proxies through several tests of coding timing, external developer engagement, disclosure informativeness, and responses to exogenous shocks affecting the payoffs of financial misconduct. We then estimate the effects of code-washing on fundraising success, controlling for extensive startup characteristics and market conditions. Our results show that startups with a high incidence of code inflation, which we refer to as *code-washers*, successfully mislead investors and enjoy a fundraising advantage during boom markets when scrutiny is low, but that this advantage disappears once investor optimism subsides or information availability increases. Post-fundraising, startups with a low incidence of code inflation, which we refer to as *code-producers*, sustain innovation and deliver significantly higher returns, whereas code-washers experience declines in code activity and negative performance. Our findings contribute to the literature on disclosure manipulation around capital market events by identifying open-sourcing as a new strategic channel through which firms may mislead investors, 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.

*We thank SFS Cavalcade Asia-Pacific for the 2025 Best Corporate Finance Paper Award and China Tech-Fin Research Conference in Shanghai for the 2024 Best Paper Award. This study benefits from the guidance of Gordon Phillips and Jack He at the 2026 Junior Faculty Mentoring Program at the American Finance Association (AFA) Meeting. We are thankful to Shai Bernstein, Lauren Cohen, Will Cong, Paul Decaire, Waverly Ding, Stephen Dimmock, Mengqiao Du, Chuck Eesley, Afonso Gambardella, John Griffin, Allen Huang, Shushu Jiang, Bjørn Jørgensen, Po-Hsuan Hsu, Poorya Kabir, Jonathan Karpoff, Wayne Landsman, Alfred Lehar, Chen Lin, Yibin Liu, Mei Luo, Evgeny Lyandres, Gustavo Manso, Moty Moravvej, Karsten Muller, Dessaint Olivier, Christine Parlour, David Reeb, David Shu, Tao Shu, Huan Tang, Xuan Tian, Tarik Umar, Baolian Wang, ChiShen Wei, Joanna Wu, Stephen Yang, Alminas Zaldokas, and Xiangyi Zhou for invaluable feedback. We also thank the seminar and conference participants at the 2026 American Finance Association (AFA) Annual Meeting, 2026 Hawai'i Accounting Research Conference, 2025 Global Entrepreneurship and Innovation Research Conference, 2025 Entrepreneurial Finance (ENTFIN) Conference, 2025 Global AI Finance Research Conference, 2025 Asian Finance Association (AsianFA) Annual Conference, 2025 Conference on Accounting and Finance Research and Education, 2024 China Tech-Fin Research Conference (CTFRC), Harvard Business School, the Hong Kong University, PBCSF Tsinghua University, and the National University of Singapore for their comments and suggestions. Rabetti thanks the Digital Economy and Financial Technology (DEFT) Lab at Cornell Fintech Initiative, the Research Center for Digital Financial Assets at SEM Tsinghua University, the Asian Institute of Digital Finance (AIDF) at NUS Business School, and the Asian Bureau of Finance and Economics Research (ABFER) for their extended discussions. Data and replication codes will be made publicly available in our repositories. All errors are our own. All authors are emerging scholars with a PhD granted after 2022.

†National University of Singapore, gefen@nus.edu.sg
‡National University of Singapore, rabetti@nus.edu.sg
§Harvard Business School (visiting), drabetti@hbs.edu
¶The University of Hong Kong, yannansun@connect.hku.hk
‖Tsinghua University, zhangch.19@pbcsf.tsinghua.edu.cn

1 Introduction

A core insight of the accounting literature is that voluntary disclosures are susceptible to strategic manipulation when external verification is weak (Ball, 2001; Healy and Wahlen, 1999). As entrepreneurial finance migrates beyond the traditional financial-reporting perimeter, firms increasingly communicate quality to investors through disclosures that are non-financial in nature. Public software repositories on platforms such as GitHub have become a particularly prominent channel, foregrounded in marketing materials, central to investor due diligence in technology ventures, and weighted by rating intermediaries as evidence of innovation effort.[1] Open-source disclosures, however, share three features that make them particularly vulnerable to strategic manipulation: they are cheap to display, broadly visible to retail investors, and difficult for outside investors to verify ex ante. We document a new form of capital-market disclosure manipulation that exploits these features, which we term *code-washing*.

Code-washing is the strategic inflation of a public software repository through superficial coding activity, designed to project an image of substantive technological progress when none exists. The practice exploits an asymmetry inherent to the open-source ecosystem: establishing a public repository is fast and costless, whereas building genuine software requires sustained developer effort, technical infrastructure, and organizational commitment. Code-washing ventures maintain a dormant repository through the pre-fundraising period, then, in proximity to fundraising, populate it with non-coding artifacts such as README files, hyperlinks, images, and marketing materials in lieu of substantive commits. The pattern is salient in observed cases. Sirin Labs raised $158 million in late 2017 after opening its repository less than three months earlier and contributing only README files and marketing documents. BitConnect maintained a single repository with seven lifetime commits, no commits in the year-long pre-offering window, and a peak market capitalization of $3.4 billion before defrauding investors of $2.69 billion in a Ponzi scheme that ultimately resulted in SEC enforcement.[2]

We study code-washing in the token offering market, the dominant fundraising channel for blockchain ventures during 2016 to 2022. The setting is well suited to the question along three dimensions. First, blockchain ventures are predominantly software-based, so code production maps directly onto the underlying technology and is observable to outside investors at high frequency. Second, token offerings raised over $30 billion in our sample window with minimal certification, weak intermediation, and limited regulatory oversight, generating sharp information asymmetries and strong incentives for opportunistic disclosure (Howell, Niessner, and Yermack, 2020; Bourveau, De George, Ellahie, and Macciocchi, 2022;

[1] As of 2025, GitHub hosts more than 630 million repositories and is used by over 180 million developers worldwide. See Lawrence (2023) for evidence that European and UK startups increasingly leverage open-source development to attract capital. DeepSeek's open-source release in 2025 illustrates the capacity of code disclosures to move investor expectations at the largest scales of technology.

[2] See Appendix Figures A1 and A2 for case-level documentation, and https://www.sec.gov/newsroom/press-releases/2021-172 for the BitConnect enforcement action.

Lyandres, Palazzo, and Rabetti, 2022). Third, the universe of public repositories is observable in full, which makes the disclosure channel measurable without selection on outcomes (Lyandres and Rabetti, 2024). These features create a quasi-laboratory environment in which to study how investors process unverified technology disclosures, when they are misled by them, and how capital is reallocated as substantive quality is revealed over time.

We frame code-washing in a standard signaling environment in which startups hold private information about their technological quality and investors observe only an imperfect, manipulable signal of that quality (Spence, 1973; Leland and Pyle, 1977; Connelly, Certo, Ireland, and Reutzel, 2011). Three predictions follow. First, at the moment of fundraising, when investors lack the technical sophistication or the cross-source data to verify the substance of an open-source signal, mimicry is unconstrained: code-washing ventures and genuine code producers both earn a fundraising premium over closed-source baselines, and the two premia are statistically indistinguishable (Ritter, 1984; Ljungqvist, Nanda, and Singh, 2006). Second, as the information environment improves through coverage by third-party intermediaries and the cross-validation of disclosures, investors screen the substance of the signal from its display, and a separating equilibrium emerges in which only code producers retain a premium (Derrien, 2005; Miller and Rock, 1985). Third, the short-term fundraising benefits of code-washing reverse over the post-offering horizon as true technological quality is revealed through observable performance, consistent with established disciplining mechanisms in capital markets (Teoh, Welch, and Wong, 1998; Karpoff, Lee, and Martin, 2008; Amiram, Lyandres, and Rabetti, 2025).

We operationalize code-washing using GitHub commit activity in the pre-offering window. A *code-washer* maintains a public repository but records zero substantive commits during the $[-365, -90]$-day window relative to the offering start date; a *code-producer* records at least one such commit; and *closed-source* ventures form the omitted baseline.[3] In a sample of 2,326 blockchain startups, 35 percent are code-washers and 31 percent are code-producers. Four validation tests, developed in detail in Section 3, support the proxy. Code-washers exhibit a 30 percent lower debug ratio, 33 percent fewer issues, and 28 percent fewer pull requests than code-producers, indicating a substantively weaker technical substrate. They concentrate disproportionate commit activity inside the offering window, with a 20.6 percentage-point higher within-window share than code-producers, often cease activity shortly after fundraising closes, and frequently delete their repository entirely when we re-examine GitHub years later. Their whitepapers are statistically indistinguishable from those of closed-source ventures in word count, technical content, and figure count, consistent with cheap talk (Li, 2008), and the team members behind these ventures are systematically less likely to disclose their affiliation on LinkedIn. Finally, following the SEC's November 2018 enforcement action against unregistered token offerings, the entry

[3]The 90-day exclusion window absorbs the anticipatory dressing-up that defines the strategic-timing channel and isolates the sustained development effort required for a credible coding signal.

rate of code-washers declines significantly while that of code-producers does not, and code-washers that continue to enter substantially improve their whitepaper informativeness. The pattern matches the predictions of a rational-misconduct framework (Becker, 1968) and is incompatible with a passive low-quality interpretation.

We turn next to the fundraising stage and ask whether investors successfully distinguish code-washers from code-producers in real time. They do not. Both groups raise capital at a significant premium over closed-source baselines, and the two premia are statistically indistinguishable. On the extensive margin, code-producers experience a 13.3 percentage-point increase in the probability of successful fundraising, and code-washers an 8.9 percentage-point increase. On the intensive margin, code-producers raise 1.61 log points more capital and code-washers 1.02 log points more, again with statistically indistinguishable coefficients. The pooling result is the central pricing finding of the paper: superficial signals of openness are misinterpreted as evidence of genuine technological substance, generating measurable capital misallocation. The pattern mirrors dynamics documented for inflated ESG narratives, cosmetic earnings releases, and pro forma metrics around equity issuance (Teoh et al., 1998; Rangan, 1998; Bradshaw and Sloan, 2003; Cohen, Gurun, and Nguyen, 2026), and is consistent with the broader insight that voluntary disclosure regimes are vulnerable to strategic manipulation when verification mechanisms are weak (Ball, 2001).

Pooling does not, however, persist uniformly. When we partition the sample by the quality of the information environment using two complementary proxies, namely the breadth of third-party data coverage and the internal consistency of cross-source disclosures (Lyandres et al., 2022), a sharp separating equilibrium emerges in the high-information subsamples. There, code-producers retain a large and significant fundraising premium while code-washers become statistically indistinguishable from closed-source ventures, and Wald tests reject the equality of the two coefficients. In the corresponding low-information subsamples, neither type can be distinguished from the closed-source baseline. The screening capacity of investors is bounded by the surrounding information infrastructure, and the empirical incidence of disclosure manipulation is endogenous to the quality of available third-party signals.

A natural concern is that the decision to maintain an empty repository is endogenous to unobserved venture characteristics that independently predict fundraising success. We address this concern with two complementary identification strategies. Propensity-score matching on the full vector of observable controls, including team size, token allocation, KYC and whitepaper disclosures, social-media activity, early-stage financing, and time, industry, and geography fixed effects, leaves the pooling coefficient essentially unchanged. We then exploit the rise of ICObench as the dominant token-offering rating platform in January 2018 as a quasi-natural experiment. ICObench's automated scoring algorithm assigned a fixed weight to the presence of a GitHub link without conditioning on the substance of the

linked repository, mechanically raising the marginal payoff to opening an empty repository without raising the payoff to producing substantive code. Consistent with this asymmetry, the shock differentially increases the fundraising premium of code-washers, while the coefficient on code-producers is statistically indistinguishable from zero. The asymmetric response is the signature of strategic disclosure: when an information intermediary commits to evaluating the display of a signal rather than its content, the empirical response is concentrated entirely among ventures whose signal lacked content to begin with.

If code-washing reflects opportunistic disclosure rather than intrinsic quality, its short-term benefits should attenuate as observable performance reveals true type. We document this reversal along two margins. First, post-listing financial performance bifurcates sharply between groups. Code-producers earn significantly higher post-listing returns and trading volume, and exhibit lower return volatility; code-washers are statistically indistinguishable from closed-source ventures on every financial metric, and Wald tests reject the equality of the two coefficients at conventional levels. Second, post-offering development effort tracks the financial pattern. Code-producers sustain commit activity, pull requests, and contributor engagement well past the fundraising window, while code-washers collapse to closed-source levels. Market discipline operates exactly as the signaling literature predicts (Karpoff et al., 2008; Teoh et al., 1998): cosmetic disclosures buy short-horizon mispricing, but the mispricing corrects as the realized productivity of the venture becomes observable.

At the extreme, code-washing predicts not merely underperformance but outright misconduct. Code-washers face more than twice the odds of being flagged in a post-offering scam relative to code-producers. The coefficient on *code-washer* is positive and significant at the one percent level for both the count of scam incidents and the probability of involvement, while the coefficient on *code-producer* is statistically indistinguishable from zero. Wald tests reject the equality of the two coefficients at the one percent level in both specifications. Code-washing is therefore not merely an unsuccessful disclosure strategy. It is a leading indicator of fraudulent outcomes, operationalizing a channel through which strategic information manipulation around capital-market events shades into outright fraud against investors.

Our study makes three contributions. First, we extend the literature on information manipulation around capital-market events to a new and increasingly important class of disclosures. A long tradition documents that managers strategically manage earnings, accruals, real activities, and non-GAAP metrics around IPOs, SEOs, and debt issuances to influence valuations (Teoh et al., 1998; Rangan, 1998; Healy and Wahlen, 1999; Bradshaw and Sloan, 2003). We show that the same logic, namely the incentive to inflate signals that are cheap to display and costly for outsiders to verify, governs the strategic use of technology disclosures in entrepreneurial fundraising.[4] Code-washing, like cosmetic earnings releases or

[4]Code-washing shares structural features with other forms of low-cost information manipulation studied in the literature: fake online reviews (Luca and Zervas, 2016; Mayzlin, Dover, and Chevalier, 2014), snowfall inflation by ski resorts (Zinman and Zitzewitz, 2016), wash trading on crypto exchanges (Amiram et al., 2025), and inflated credit ratings (Becker and Milbourn, 2011). See also Soltes (2016) on the qualitative origins of white-collar misconduct as opportunistic short-horizon rationalization.

inflated ESG narratives, is economically meaningful in shaping capital allocation. Unlike those settings, however, it operates entirely outside the audit perimeter, the financial-reporting framework, and the institutional gatekeepers that constrain comparable behavior in traditional capital markets, identifying a class of disclosure manipulation that current accounting institutions are structurally ill-equipped to police.

Second, we contribute to the literature on cheap talk versus costly signaling and the resulting capital misallocation in environments with weak verification. Prior work documents that superficial commitments distort capital flows in ESG ratings, diversity claims, and analogous unverifiable disclosures (Dyck, Lins, Roth, and Wagner, 2019; Christensen, Serafeim, and Sikochi, 2022; Baker, Larcker, McClure, Saraph, and Watts, 2024). We provide direct evidence of an analogous distortion in entrepreneurial finance: investor capital flows disproportionately to code-washers when the information environment is sparse and reverses toward genuine producers as third-party intermediation thickens. The dynamic identifies a specific mechanism through which weak verification regimes generate capital misallocation in high-technology ventures, and quantifies the magnitude of the misallocation along the extensive and intensive margins of fundraising as well as the long-run dispersion of post-listing performance.

Third, we contribute to the emerging accounting literature on the disclosure, auditing, and oversight of digital-asset ventures. Token-issuing ventures fall almost entirely outside the conventional financial-reporting perimeter, and the disclosures that investors observe consist disproportionately of voluntary, technology-related signals released through unaudited online channels. Our results show that these signals are systematically manipulated and that the manipulation is predictive of outright fraud. We further document a tractable mechanism through which credibility-enhancing institutions, including independent rating platforms with substance-based scoring, third-party data intermediaries, and regulatory regimes that impose costs on misrepresentation, mitigate the manipulation. The findings inform the ongoing debate over the appropriate disclosure, audit, and certification frameworks for digital-asset markets, and offer evidence-based guidance to investors and regulators on which disclosure channels are reliably informative and which are systematically gamed.

The remainder of the paper proceeds as follows. Section 2 develops the signaling framework and formalizes the three hypotheses. Section 3 describes the institutional setting, the dataset, and the construction and validation of the code-washing proxy. Section 4 presents the empirical analysis of the fundraising stage and reports the pooling and separating results, the matched-sample analysis, and the rating-platform experiment. Section 5 examines the long-run financial performance, development effort, and scam outcomes of code-washing ventures. Section 6 concludes.

2 Hypothesis Development

This section embeds code-washing in a standard signaling environment and develops three testable predictions. Section 2.1 characterizes the signaling problem confronting investors in token offerings and motivates open-source code as the candidate signal of project quality. Sections 2.2, 2.3, and 2.4 derive, in turn, the conditions under which a pooling equilibrium prevails at fundraising, the conditions under which separation emerges, and the dynamic revelation of type in the post-offering window.

2.1 The Signaling Problem in Token Offerings

The token-offering market exhibits the canonical conditions under which a costly signaling equilibrium is unlikely to emerge without external support. Founders observe their venture's technological quality; outside investors do not. The conventional certifying intermediaries that mitigate this asymmetry in traditional capital markets, namely auditors, underwriters, venture capitalists, and credit-rating agencies (Diamond, 1984; Megginson and Weiss, 1991; Holmström and Tirole, 1997; Besanko and Kanatas, 1993; Admati and Pfleiderer, 1994), are largely absent in this setting, operate under conflicts of interest, or have themselves been implicated in misconduct (Barth, Laturnus, Mansouri, and Wagner, 2023). Investors must therefore form expectations from the venture's own disclosures, the credibility of which depends on whether the underlying signal is sufficiently costly to deter mimicry by low-quality types (Akerlof, 1970; Spence, 1973; Leland and Pyle, 1977; Rothschild and Stiglitz, 1976; Connelly et al., 2011).

Open-source software activity on platforms such as GitHub is the most prominent candidate for a credible signal in this market. The platform is freely observable to investors, records developer effort at the commit level, and exposes a high-frequency record of project evolution that prior work in blockchain finance has used as a measure of technological substance (Amsden and Schweizer, 2018; Fisch, 2019; Momtaz, 2021; Howell et al., 2020; Lyandres et al., 2022; Davydiuk, Gupta, and Rosen, 2023; Cong, Prasad, and Rabetti, 2025). In principle, sustained code production is costly to fake: it requires technical talent, infrastructure, and continuous effort that low-quality ventures cannot easily replicate. In practice, however, the platform admits a low-cost form of mimicry. Establishing a repository is free, marking it public is free, and populating it with non-coding content such as marketing materials, hyperlinks, and image attachments is free. The signal that investors observe is therefore the joint product of two distinct activities, only one of which is costly to produce.

2.2 Pooling under Cheap Mimicry

When investors lack the technical sophistication or the cross-source data to separate substantive coding from cosmetic activity, the two activities pool into a single observable disclosure. The cheap-talk and disclosure-manipulation literatures establish that, under these conditions, low-quality types optimally mimic the observed disclosure of high-quality types, generating a pooling equilibrium in which the signal carries no incremental information about the underlying type (Frankel and Kartik, 2019; Ball, 2022; Sun, 2022; Bolton and Dewatripont, 2005; Mas-Colell, Whinston, and Green, 1995). The pooling outcome is amplified in periods of elevated investor sentiment, when the average willingness to fund unverified claims rises, the marginal informativeness of any individual signal falls, and a higher share of low-quality issuers enters the market (Loughran, Ritter, and Rydqvist, 1994; Ritter, 1984; Lee, Shleifer, and Thaler, 1991; Ljungqvist et al., 2006; Ritter and Welch, 2002).[5]

Two further features of the token-offering market reinforce the pooling prediction. First, the retail composition of the investor base implies that, even where in-principle verification is feasible through inspection of repository contents, the marginal investor lacks the technical expertise to perform it. Second, the absence of an audit perimeter or a comparable financial-reporting framework removes the institutional channel through which disclosures are routinely cross-checked against verifiable financial statements in traditional capital markets. Both features raise the relative benefit, and lower the relative cost, of cosmetic disclosure for low-quality types. These observations yield our first hypothesis.

> **H1.** *During fundraising, investors are unable to distinguish code-washers from code-producers, and both groups command a fundraising premium relative to closed-source ventures.*

2.3 Separating under a Richer Information Environment

A separating equilibrium emerges when the cost of mimicry rises relative to the cost of genuine signaling, or, equivalently, when the screening capacity of investors improves (Spence, 1973; Miller and Rock, 1985; Bolton and Dewatripont, 2005; Mas-Colell et al., 1995). In the token-offering market, the production cost of cosmetic commits is structurally low and does not vary meaningfully with investor scrutiny. What does vary is the screening side. As the surrounding information environment thickens, through a denser network of third-party data sources, the cross-validation of disclosures across those sources, and broader investor attention to project fundamentals (Lyandres et al., 2022; Derrien, 2005), inconsistencies between observed code activity and other markers of substantive development become

[5]The same mechanism operates across a wide range of settings in which a disclosure is cheap to display and costly to verify: fake online reviews of restaurants and hotels (Luca and Zervas, 2016; Mayzlin et al., 2014), snowfall inflation by ski resorts (Zinman and Zitzewitz, 2016), inflated credit ratings (Becker and Milbourn, 2011), and wash trading on cryptocurrency exchanges (Amiram et al., 2025).

detectable at low cost. The same logic applies during market downturns, when heightened risk aversion makes investors more discerning and more selective in their funding decisions (Gennotte and Leland, 1990; Froot and Obstfeld, 1991). Under these conditions, the pooled equilibrium of H1 gives way to separation, and the fundraising premium accrues only to ventures whose code-production record is corroborated by independent sources.

> **H2.** *Investors can distinguish code-washers from code-producers when the information environment is sufficiently rich and disclosures can be cross-validated across independent sources.*

2.4 Long-Run Discipline and Performance Reversal

The pooling equilibrium identified in H1, if it obtains at the moment of fundraising, is dynamically unstable. The defining feature of code-washing is that the underlying type is eventually revealed through observable post-offering performance. As the venture either delivers a functional product, sustains development effort, and generates trading activity, or fails to do so, the market gradually learns the type, and prices, trading activity, and capital allocation adjust. The accounting and finance literatures have long emphasized this discipline as the central reason that strategic mimicry generates short-term gains at the expense of long-run reputational and economic damage (Teoh et al., 1998; Karpoff et al., 2008; Amiram et al., 2025). The mechanism is sharpest in settings, such as ours, where the manipulated type is technological rather than purely financial: a venture cannot indefinitely sustain the appearance of innovation through cosmetic GitHub activity once investors expect a functional product. The same logic predicts that the most acute reversal should occur among ventures for which the gap between displayed signal and underlying substance is largest, and that this group should disproportionately populate the tail of outright misconduct.

> **H3.** *Code-washers underperform code-producers in the post-fundraising period along financial, developmental, and misconduct margins.*

3 Setting and Data

3.1 Data

We construct a venture-level dataset that links fundraising outcomes, venture characteristics, code production, post-listing financial performance, on-chain activity, social-media attention, and financial misconduct for blockchain ventures that conducted token-offering fundraising attempts between 2013 and 2020.

We assemble a comprehensive sample of token-offering fundraising attempts from eleven aggregators, since coverage of any single aggregator is incomplete. The aggregators are selected based on their average historical web traffic during the sample period (Lyandres et al., 2022): Etherscan (https://etherscan.io), CoinDesk (https://coindesk.com), CoinGecko (https://coingecko.com), CryptoCompare (https://cryptocompare.com), ICObench (https://icobench.com), ICODrops (https://icodrops.com), ICORating (https://icorating.com), ICOmarks (https://icomarks.io), ICOdata (https://icodata.io), FoundICO (https://foundico.com), and TokenData (https://tokendata.io). The aggregated sample contains 7,273 unique blockchain ventures from more than 100 countries, with cumulative fundraising of approximately $33 billion. We additionally collect venture-level characteristics, including industry classification, headquarters location, whitelist policies (which require investors to pre-register), and presale rounds raised from angel or venture-capital investors. For ventures that disclose a whitepaper, we extract both the text and the embedded images, which we later use to measure the informativeness of non-code disclosures.

To measure code production and identify code-washing behavior, we collect more than one terabyte of repository data from GitHub (https://github.com), the largest open-source software development platform. These data include commits, pull requests, code reviews, comments, issue activity, and developer identifiers within a 730-day event window spanning 365 days before to 365 days after each venture's offering start date. To reduce false matches between ventures and unrelated repositories, we manually verify each repository link disclosed in a venture's whitepaper, official website, or aggregator profile. We further retain the text of commit messages, which we use to identify debugging-related development activity.

To complement the venture-level footprint with information on the founding team, we hand-collect LinkedIn profiles for the team members disclosed by each venture. For each disclosed team member, we manually search by name and affiliation, retain the highest-confidence profile match, and record current and prior employment, educational background, and professional seniority. Of particular interest is whether the team member explicitly lists the venture in their disclosed work experience, which provides a measure of team-disclosure transparency. The LinkedIn-based dataset covers a subset of 565 ventures for which at least one verifiable team-member profile can be identified.

Finally, to examine the long-term consequences of code-washing, we collect post-listing outcomes from four sources. First, daily post-listing token returns and trading volumes are obtained from CoinMarketCap (https://coinmarketcap.com) and used to construct cumulative returns, trading volume, return volatility, and the Amihud illiquidity measure. Second, on-chain activity is measured using Ethplorer (https://ethplorer.io), which tracks the time-series evolution of distinct cryptographic wallets holding each token and cumulative on-chain token transactions, supplemented with off-chain

trading-volume data from Kaiko (https://kaiko.com). Third, social-media activity is collected from four platforms widely used in the blockchain ecosystem, namely Twitter, Reddit, Medium, and Bitcointalk, to capture public attention and investor engagement. Fourth, scam events are collected from DeadCoins (https://deadcoins.com), which catalogs several hundred blockchain ventures flagged as fundraising exit scams.[6]

3.2 Identifying Code-Washers

GitHub repository activity is the standard observable signal of technological progress for blockchain ventures (Lyandres and Rabetti, 2024), but the act of opening a repository is itself trivially cheap. A founder can create a public repository in minutes and populate it with non-technical content such as a README file, marketing slides, image attachments, or a PDF of the whitepaper, none of which requires engineering effort or conveys information about the underlying technology. The asymmetry between the cost of opening a repository and the cost of producing substantive code parallels the classical signaling distinction between observable but uninformative gestures and costly, separating actions (Spence, 1973; Connelly et al., 2011).

We exploit this asymmetry to identify market-timing code-washing, a strategy in which ventures maintain a publicly visible repository but concentrate commit activity in a narrow window around the token offering, when investor scrutiny is highest, rather than in the period when development should organically take place. The logic mirrors the strategic timing of voluntary disclosure around capital-market events documented in conventional settings (Aboody and Kasznik, 2000): managers allocate a costly, observable action to the moment its payoff is greatest, not to the moment it is most informative about fundamentals.

Following the literature in blockchain finance (Amsden and Schweizer, 2018; Fisch, 2019; Lyandres et al., 2022; Davydiuk et al., 2023), we measure code production using GitHub commits. A commit is the fundamental unit of the GitHub workflow: it records a snapshot of the project's state at a point in time, is accompanied by a commit message describing the change, and is observable to anyone with access to the public repository. The volume and frequency of commits capture a venture's substantive development effort and are widely treated as the canonical proxy for technological progress in this market.

To distinguish genuine development effort from offering-window inflation, we partition each venture's commit history into two non-overlapping intervals relative to the token offering. Figure 1 illustrates the partition. The pre-offering window, $[-365, -90]$ days prior to the offering start, captures the period when credible development effort would be observable in advance of fundraising. The offering window, between the offering start and end dates, captures any abnormal concentration of commit activity

[6]The dataset on blockchain fundraising attempts is publicly available in our online repositories.

consistent with market timing. The choice of pre-offering window is deliberate. Credible smart-contract and protocol work requires sustained effort over months rather than days, and reputable codebases in our sample exhibit commit activity well in advance of fundraising. We exclude the ninety days immediately preceding the offering because anticipatory dressing-up of the repository is most likely to occur in that interval and would contaminate any pre-offering measure that includes it.

Panel B of Figure 1 plots the distribution of cumulative pre-offering commits across open-source ventures. Two features stand out. A substantial mass of ventures records zero commits during the pre-offering window. Conditional on positive activity, the distribution is sharply right-skewed, with a long tail beyond 200 commits and a median of one. The median provides a natural cutoff that separates ventures exhibiting any observable pre-offering development from those exhibiting none. We therefore classify open-source ventures using a median split: a code-washer produces zero commits during $[-365, -90]$, while a code-producer produces at least one commit during the same window. Closed-source ventures, those without any publicly disclosed GitHub repository, form the omitted reference category in regressions that include both indicators.[7] The Online Appendix shows that the main empirical results are robust to alternative pre-offering windows (for example, $[-180, -90]$) and to alternative quantile thresholds (for example, the twenty-fifth percentile).

3.3 Summary Statistics

Panel A of Table 1 reports summary statistics for the analysis sample, conditional on the existence of all variables required by the baseline specification. The sample contains 2,326 unique blockchain ventures. Two-thirds of the sample (66 percent) maintains at least one public GitHub repository prior to the offering window, and the remaining one-third are closed-source. Within the open-source group, the two types are roughly evenly distributed: 35 percent of all sample ventures, or 53 percent of the open-source subsample, are classified as code-washers, and 31 percent, or 47 percent of the open-source subsample, are classified as code-producers. Variable definitions are reported in Appendix A1.

Just over half of the sample ventures successfully raise capital. Conditional on raising, the average venture raises approximately \$11.7 million, equivalent to 23.6 percent of the fundraising hardcap on average. Sixty-eight percent of ventures attempt a presale to angel, venture-capital, or other early-stage investors before the main fundraising stage. The average ratio of total token supply to tokens offered for sale is 57 percent. More than half of the sample offers a whitelist to early investors, 68 percent implement know-your-customer (KYC) requirements, and 47 percent disclose a whitepaper outlining the project's goals, technology, use cases, and business model. The average team size is 11.6 members, including

[7] Appendix Figure A1 presents Sirin Labs, a representative code-washer that posted its first GitHub activity less than three months before the offering start date, with content limited to README files and attached marketing documents and no production smart-contract code committed before fundraising.

founders and other key employees, indicating that these are very small ventures at the fundraising stage.

Social-media activity is widely observed in the blockchain fundraising ecosystem and is included as a vector of controls in the subsequent empirical analysis. Across the four platforms in our data, *Twitter (log)* exhibits a mean of 2.82 and a standard deviation of 2.24; *Reddit (log)* a mean of 1.49 and a standard deviation of 1.69; *Medium (log)* a mean of 0.65 and a standard deviation of 1.30; and *BTCTalk (log)* a mean of 3.06 and a standard deviation of 2.73. The four channels serve distinct roles in venture-investor communication and exhibit substantively different usage patterns across ventures.

Panel B of Table 1 compares the characteristics of code-producers and code-washers along the dimensions used as controls in equation (5). Two patterns emerge. First, the two groups are statistically indistinguishable on dimensions that proxy for fundamental capacity and resources. Team size, KYC compliance, whitepaper availability, and the hardcap of the offering exhibit no statistically significant differences between code-washers and code-producers. Second, the two groups differ systematically on dimensions that reflect investor screening and community engagement. Sixty percent of code-producers offer a whitelist, ten percentage points higher than the fifty percent rate for code-washers (Diff. $= -0.10$, $t = -3.96$), consistent with code-producers implementing more rigorous investor-screening processes that build the longer-run reputation of the project. Code-producers also maintain notably more active social-media presences than code-washers on Twitter (Diff. $= -0.56$, $t = -4.13$), Reddit (Diff. $= -0.84$, $t = -4.47$), and Medium (Diff. $= -0.32$, $t = -3.21$). These platforms attract technical developers and blockchain enthusiasts who are positioned to engage with the substance of a venture's code disclosures, and the higher activity of code-producers across them is consistent with the broader pattern that the code-producer signal is reinforced by substantive engagement across independent disclosure channels.

3.4 Proxy Validation

Before applying the proxy in the fundraising-performance analysis, we validate it through three complementary tests. We first conduct internal validation, examining whether code-washers exhibit other GitHub behaviors that distinguish fundraising-driven signaling from genuine software production, along two dimensions: the market-timing dynamics of repository activity and the substantive depth of pre-offering development (Section 3.4.1). We then conduct external validation to assess whether the same strategic gap extends to disclosure channels outside GitHub, where the cost of misrepresentation falls on the venture (whitepaper informativeness) and on individual team members (LinkedIn affiliation) (Section 3.4.2). Finally, we exploit the SEC's November 2018 enforcement action as an exogenous shock that raises the expected cost of strategic disclosure without altering the underlying distribution of venture quality, allowing us to distinguish a deliberate-misconduct interpretation from a passive low-quality one (Section 3.4.3).

3.4.1 Internal Validation: GitHub Behavior

A useful proxy for code-washing should separate ventures along the behavioral dimensions that distinguish a genuine software project from a repository maintained to support a fundraising signal. We examine two complementary sets of behaviors. The first concerns the market-timing dynamics of repository activity: when commits occur relative to the offering, and whether the repository continues to evolve once the raise concludes. The second concerns the substantive depth of pre-offering development: whether commits reflect maintenance and defect-fixing rather than cosmetic content, and whether the project sustains the peer review and task management that distinguish multi-developer engineering from individual scripting (Vasilescu, Yu, Wang, Devanbu, and Filkov, 2015; Syer, Nagappan, Adams, and Hassan, 2015; Fisch, 2019). If our proxy isolates strategic signaling, both sets should diverge sharply across types.

We begin with the market-timing dynamics. For genuine ventures, code production should accumulate over many months, plateau modestly during the offering as engineering resources are reallocated to managing the raise, and resume after the raise concludes. For ventures that open a repository to support a fundraising signal, the dynamics should be the mirror image: little or no activity in the long pre-offering window, a concentrated burst around the offering, and a sharp decay, often to zero, once funds are secured. Figure 2 provides the visual evidence. Code-washers exhibit virtually no code production before the offering and begin committing only at the offering start, with activity that continues to accumulate through the fundraising window, consistent with market-timed code disclosure that becomes visible to investors precisely when scrutiny is highest. Code-producers, by contrast, display substantially higher and more persistent code production throughout the event window, with no analogous concentration around the fundraising phase.

We formalize this pattern with three lifecycle variables. *Code Concentration* is the share of cumulative commits made during the offering window relative to all commits as of offering end. *RepoAbandoned* equals one if the venture makes zero commits in the 180 days following the offering end. *RepoDeleted* equals one if the GitHub URL recorded for the venture in our 2020 source index returns HTTP status 404 when re-queried in April 2026, indicating that the public repository has been removed from GitHub. We estimate the following model on the open-source subsample with code-producers as the omitted reference:

$$Y_i = \alpha + \beta\, \textit{Code-washer}_i + \Theta_i + \Lambda_i + \varepsilon_i, \tag{1}$$

where Θ is the standard vector of venture, fundraising, and social-media controls described in Section 3.3; Λ contains region, industry, and offering-month fixed effects; and standard errors are clustered at the offering-month level. We use OLS for the continuous *Code Concentration* outcome and Logit for the

two binary outcomes.

Panel (a) of Table 2 reports the results. Code-washers concentrate 20.6 percentage points more of their cumulative commits inside the offering window than code-producers ($\hat{\beta} = 0.21$, $t = 10.17$); against an unconditional code-producer concentration of roughly six percent, this represents more than a threefold difference. Code-washers are also far more likely to fall silent immediately after the raise (*RepoAbandoned*: $\hat{\beta} = 1.97$, $z = 15.57$), and significantly more likely to delete the repository altogether years later. The raw deletion rate is 17.4 percent for code-washers versus 9.6 percent for code-producers ($\chi^2 = 17.2$, $p < 0.001$). Conditional on the full set of controls, the Logit coefficient is $\hat{\beta} = 0.78$ ($z = 5.95$), implying that code-washers are roughly 7.7 percentage points more likely to have erased their public repository by 2026. The three outcomes trace a single behavior: effort clustered to support the fundraising phase, silence afterward, and eventual erasure of the public trace.

We next turn to the substantive depth of pre-offering development, beginning with the content of individual commits. If pre-offering commits represent genuine engineering effort rather than cosmetic content, they should include a meaningful share of the unglamorous debugging, maintenance, and refactoring tasks that characterize real software projects (Fisch, 2019). We construct *Debug Ratio* as the share of a venture's pre-offering commits whose commit message contains debugging-related terms ("fix," "bug," "error," or "patch"), and re-estimate equation (1) with this measure as the dependent variable. Panel (b) of Table 2 shows that the *Debug Ratio* is 6.2 percentage points lower for code-washers than for code-producers ($\hat{\beta} = -0.062$, $t = -6.10$), a more than 30 percent reduction relative to the code-producer mean. The commits code-washers do produce are far less likely to address defects in a working codebase.

Beyond the content of individual commits, we examine the collaborative and project-management activity that distinguishes multi-developer software engineering from individual scripting (Vasilescu et al., 2015; Syer et al., 2015). A pull request initiates peer review of proposed code changes, and an issue tracks a task, bug, or feature request through resolution. Both require a non-trivial team and an active codebase, and neither can be generated cheaply at the moment of fundraising. We construct *Issues (log)* and *Pulls (log)* as the natural logarithms of one plus the cumulative number of issues opened and pull requests submitted before the offering end, and re-estimate equation (1) with each as the dependent variable. Panel (c) of Table 2 reports the results. Code-washers open significantly fewer issues ($\hat{\beta} = -0.40$, $t = -8.75$) and submit significantly fewer pull requests ($\hat{\beta} = -0.33$, $t = -8.29$) than code-producers, implying roughly 33 percent and 28 percent fewer collaborative-development artifacts, respectively. The activity that code-washers do generate during the offering window does not resemble the iterative debugging and peer-reviewed integration that characterize real software projects.

A natural concern with a quantity-based proxy is that commit counts mechanically reflect team size

and human capital rather than strategic choice. Smaller or less-credentialed teams may simply lack the capacity to commit as often, in which case the proxy would conflate code-washing with firm size or weak human capital. We address this concern in Appendix Table A2, which reports balance tests of team characteristics across code-production status. Code-washers and code-producers are statistically indistinguishable from each other on team size and on the share of team members holding a PhD; the Wald test fails to reject $\beta_1 = \beta_2$ in every column. Code-washers therefore look like code-producers on the dimensions that proxy for fundamental team capacity, but direct those resources away from substantive development on every behavioral dimension we measure.

3.4.2 External Validation: Voluntary Disclosure

A more demanding test is whether the same strategic gap appears in voluntary disclosure channels outside GitHub. When an issuer faces a choice over how informative its public communications should be, models of strategic disclosure imply that issuers with weaker fundamentals release less informative content, dilute hard information with boilerplate, and avoid channels in which insiders' identities can be verified (Bushee and Leuz, 2005; Li, 2008; Baker et al., 2024). In the unregulated token-offering market, where investor protection is minimal and intermediaries do not certify disclosures, this prediction is sharpened further: cheap-to-produce content dominates, and voluntary self-disclosure carries strong selection (Bourveau et al., 2022). If our proxy captures a strategy of mimicking high-quality issuers cheaply, then code-washers should also produce thinner whitepapers and avoid LinkedIn channels in which team members must publicly stake their professional reputations on the venture.

We test the whitepaper prediction first. Whitepapers are the central voluntary-disclosure document in token offerings; they are unaudited and unstandardized, which makes their informational content endogenous to the issuer's strategy. We measure whitepaper informativeness with three variables: *Word count* is the total words in the whitepaper; *Tech word count* is the count of technology-related terms identified by a domain-specific NLP classifier; and *Image count* is the count of figures. We estimate

$$Y_i = \alpha + \beta_1 \, \textit{Code-producer}_i + \beta_2 \, \textit{Code-washer}_i + \Theta_i + \Lambda_i + \varepsilon_i, \tag{2}$$

on the full sample with closed-source ventures as the omitted category, using Poisson regression for the count levels. The wedge between β_1 and β_2 captures the differential disclosure quality of the two open-source types relative to the same closed-source benchmark.

Table 3 reports the results. Relative to the closed-source benchmark, code-producers disclose whitepapers with more total content ($\hat{\beta}_1 = 0.09$, $z = 1.72$), significantly more technology-specific content ($\hat{\beta}_1 = 0.12$, $z = 3.59$), and a directionally larger but statistically insignificant volume of illustrative figures ($\hat{\beta}_1 = 0.16$, $z = 1.36$). Code-washers, by contrast, are statistically indistinguishable

from closed-source ventures in word count and tech word count ($\hat{\beta}_2 \approx 0$ in both columns), and produce significantly fewer figures than closed-source benchmarks ($\hat{\beta}_2 = -0.30$, $z = -3.29$), corresponding to roughly 25.6 percent fewer figures. Wald tests reject $\beta_1 = \beta_2$ in all three specifications. The visible signal of code-washers stops at the GitHub repository: when a disclosure channel asks for substantive technical content, code-washers either match closed-source benchmarks or fall below them.

We next examine a disclosure channel where the marginal cost of misrepresentation falls on individuals rather than on the venture: LinkedIn. Team members of a legitimate venture have a private incentive to disclose their affiliation, because their human capital appreciates with the venture's success. Team members of a venture they expect will fail, or, more pointedly, a venture they recognize as a strategic disclosure vehicle, have the opposite incentive: visible association is a personal liability. In a related vein, the willingness of the venture to attract verifiable elite human capital is itself a costly signal that mimicking issuers cannot easily replicate (Conti, Peukert, and Roche, 2025). We hand-collect LinkedIn profiles for 8,759 individuals listed across 1,108 ventures and aggregate the information to the venture level. *Any member disclosed* equals one if at least one team member explicitly identifies the venture on their public profile, and *Share of disclosed* is the fraction of team members who do so. We re-estimate equation (2) replacing the dependent variables accordingly, using Logit for the binary outcomes and Tobit for the censored fractions.

The results in Table 4 separate the two open-source types. Code-producer team members are far more likely to disclose their venture affiliation on LinkedIn ($\hat{\beta}_1 = 1.36$, $z = 4.17$) and disclose at higher intensity, while code-washer teams are statistically indistinguishable from closed-source teams on either margin. The elite human capital prediction holds in the same direction: code-producers attract significantly more elite-educated team members on both the extensive margin ($p = 0.027$) and the intensive margin ($p = 0.044$), while code-washers do not. Wald tests reject $\beta_1 = \beta_2$ in all specifications. Together with the whitepaper evidence, these findings show that the strategic gap we identify on GitHub is mirrored in two independent voluntary-disclosure channels, one where the venture itself controls the content, and one where individuals weigh the personal reputational cost of association.

3.4.3 External Shock: SEC Enforcement

A skeptic might still argue that what we are measuring is venture quality rather than strategic behavior: low-quality ventures may simply produce less code, write thinner whitepapers, and attract weaker teams. We address this alternative with an exogenous shock that increases the expected cost of strategic disclosure while leaving observable venture characteristics largely unchanged.

Becker (1968) provides the framework. Strategic disclosure is a calculated act whose expected cost rises with the probability of detection and the magnitude of the penalty conditional on detection; a passive

low-quality firm has nothing analogous to optimize. The two interpretations therefore make different predictions on two distinct margins. On the composition margin, if code-washers are strategic disclosers, their share among new entrants should fall after the shock, while the entry rate of closed-source ventures, which never staked a costly open-source signal, should be unaffected. On the disclosure margin, the code-washers who choose to enter the market post-shock should be a more selected subset that produces more informative voluntary disclosures than their pre-shock counterparts; the corresponding pre/post difference for ventures whose disclosure choices were never strategic should be absent. A passive low-quality interpretation cannot rationalize either response: weak ventures should neither selectively exit nor differentially raise the informativeness of their disclosures.

In November 2018, the SEC announced its first civil penalties against token-offering issuers, including *Airfox* and *Paragon*, and signaled that all issuers were subject to existing securities laws, including registration requirements and disclosure liability.[8] Although industry observers had anticipated some form of SEC action since the 2017 DAO Report, the November 2018 announcement was the first concrete enforcement against token-offering issuers and the first observable signal of both the magnitude of penalties and the breadth of SEC scope. The action applied uniformly across project types and discretely raised both the probability of detection and the expected penalty conditional on fraudulent disclosure.

To formalize this composition test, we restrict the sample to ventures completing a token offering between May 2018 and April 2019 and estimate the following specification separately for each type k:

$$\mathbf{1}\{\text{Type}_i = k\} \;=\; \alpha + \beta\, Post_i + \Theta_i + \Lambda_i + \varepsilon_i, \quad k \in \{\text{Code-washer, Code-producer, Closed-source}\}, \tag{3}$$

where *Post* equals one for offerings completed after November 16, 2018. We estimate Probit specifications with the standard control vector Θ and region/industry fixed effects in Λ; standard errors are clustered by calendar month.[9]

Panel (a) of Table 5 reports the composition results. The probability that a new entrant is classified as a code-washer falls significantly after the shock ($\hat{\beta} = -0.225$, $z = -2.75$), while the corresponding probability for code-producers rises by a comparable magnitude ($\hat{\beta} = 0.300$, $z = 4.08$), and the closed-source entry rate remains statistically unchanged. The selective response is precisely what the strategic-disclosure interpretation predicts: ventures that previously found it profitable to manufacture a cheap open-source signal withdraw from the market, while ventures that never produced a public technical signal are unaffected. A passive low-quality interpretation cannot rationalize the asymmetric response between open-source code-washers and closed-source ventures: the two groups should move together if

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

[9] Because *Post* is itself a month-level treatment indicator, we omit offering-month fixed effects in this specification.

code-washing reflected only venture quality.

We next ask how the code-washers who remain after the shock, those who still find it worthwhile to enter despite the heightened cost of strategic disclosure, differ in disclosure quality from their pre-shock counterparts. We compare the whitepapers of post-shock code-washers to those of pre-shock code-washers, holding the specification of equation (2) otherwise constant. We re-estimate that specification on the SEC-window sample using Poisson regression, with *Post (SEC)* and its interactions with the two open-source type indicators added:

$$\begin{aligned} Y_i \;=\; & \alpha + \beta_1\, \textit{Code-producer}_i + \beta_2\, \textit{Code-washer}_i + \gamma\, \textit{Post (SEC)}_i \\ & + \delta_1\, \textit{Post (SEC)}_i \times \textit{Code-producer}_i + \delta_2\, \textit{Post (SEC)}_i \times \textit{Code-washer}_i + \Theta_i + \Lambda_i + \varepsilon_i . \end{aligned} \tag{4}$$

Panel (b) of Table 5 reports the results. Whitepapers produced by post-shock code-washers exhibit significantly higher per-page word density ($\hat{\delta}_2 = 0.22$, $z = 2.89$) and significantly more technology-specific content ($\hat{\delta}_2 = 0.16$, $z = 2.84$) than those of pre-shock code-washers, while the corresponding pre/post difference for code-producers is statistically indistinguishable from zero. The asymmetry mirrors the composition-margin pattern: the group whose pre-shock disclosures were strategically thin is the same group whose post-shock disclosures move toward greater informativeness, while the group whose pre-shock disclosures already reflected genuine information shows no analogous shift. The shock removes the marginal code-washer from the entry pool, and the inframarginal code-washers who continue to enter are observably more informative on the dimensions where their pre-shock peers had been thinnest. Code-washing is therefore a deliberate disclosure strategy responsive to the expected cost of misrepresentation, not a label for low-quality ventures.

4 Code-Washing and Fundraising

Section 2 developed three predictions about how investors price code-washing relative to substantive code production. This section tests the first two in the cross-section of token offerings. Section 4.1 tests H1, which states that investors fail to distinguish code-washers from code-producers at the moment of fundraising. Section 4.2 tests H2, which states that this pooling breaks down as the information environment around the offering becomes richer. Section 4.3 addresses the endogeneity of the code-washing decision through propensity-score matching and a quasi-natural experiment that exogenously raises the marginal payoff to displaying, rather than producing, a GitHub signal.

4.1 Pooling Equilibrium

H1 predicts that, at the fundraising stage, code-washers and code-producers both command a premium over closed-source ventures and that the two premia are statistically indistinguishable. We test the prediction with the following specification:

$$Success_i = \alpha + \beta_1 \, Code\text{-}producer_i + \beta_2 \, Code\text{-}washer_i + \Theta_i + \Lambda_i + \varepsilon_i, \tag{5}$$

where *Success* is measured either as *RaisedDummy*, an indicator equal to one if the venture raises any capital by the close of the token offering, or as *FundsRaised (log)*, the natural logarithm of the total amount raised. The two classification indicators are defined as in Section 3.2; closed-source ventures form the omitted reference category. The vector Θ collects the project-, fundraising-, and social-media controls described in Section 3.3, and Λ absorbs year-month, industry, and geographic-region fixed effects. Standard errors are clustered at the venture-fundraising completion month to allow for serial correlation in residuals around contemporaneous market conditions. A pooling equilibrium implies $\beta_1 > 0$, $\beta_2 > 0$, and $\beta_1 = \beta_2$. The alternative, in which investors price the substance of development rather than its display, implies $\beta_1 > \beta_2 \geq 0$.

Table 6 reports the estimates. On the extensive margin (column (i)), both classification indicators are positive and significant at the five percent level: $\hat{\beta}_1 = 0.539$ ($t = 2.46$) for *Code-producer* and $\hat{\beta}_2 = 0.360$ ($t = 2.25$) for *Code-washer*. Evaluated at the sample-mean success rate of 0.56, these coefficients imply marginal effects of 13.3 percentage points for code-producers and 8.9 percentage points for code-washers, both relative to the closed-source baseline. The intensive-margin estimates in column (ii) deliver the same qualitative pattern: $\hat{\beta}_1 = 1.61$ ($t = 3.21$) and $\hat{\beta}_2 = 1.02$ ($t = 2.55$), corresponding to amounts raised that are 1.61 and 1.02 log points larger than the baseline.

The decisive test for H1 is whether the two coefficients are statistically distinguishable. A Wald test fails to reject the null of $\beta_1 = \beta_2$ in either column, indicating that the fundraising premium attached to a public GitHub repository is statistically the same across ventures that produce substantive code and ventures that produce only cosmetic activity. Investors price the presence of a public repository, not the content of what is committed to it. Code-washers free-ride on the credibility of code-producers, and a pooling equilibrium obtains in which superficial open-sourcing is rewarded at a rate the data cannot distinguish from substantive development. The pattern is consistent with the broader insight in the accounting literature that voluntary disclosure regimes are vulnerable to strategic manipulation when external verification is weak (Ball, 2001; Healy and Wahlen, 1999).

4.2 Separating Equilibrium

The pooling result in Section 4.1 is a sample-average statement: on average, investors do not distinguish code-washers from code-producers. Theory predicts, however, that this pooling should attenuate as the information environment around the offering becomes richer. When more data on the venture are available and when reported values are consistent across independent sources, investors can verify the GitHub signal against an independent body of information rather than evaluating it on its surface form alone. H2 formalizes this prediction. The hypothesis is testable against equation (5): when the information environment is rich, $\beta_1 > \beta_2$ and the Wald test rejects the equality of the two coefficients; when the information environment is sparse, the equality cannot be rejected.

We use two complementary proxies for the surrounding information environment, both following Lyandres et al. (2022). The first, *information coverage*, captures the breadth of the available information set, measured as the number of independent sources reporting fundraising details for the venture. The second, *information quality*, combines coverage with the average consistency of reported values across sources, capturing both breadth and reliability. Appendix B details the construction of both indices. For each proxy, we partition the sample at the median into a high- and a low-information subsample, and re-estimate equation (5) separately within each.

Table 7 reports the results. Under the *information coverage* split, the contrast across subsamples is sharp. In the high-coverage subsample (columns (i) and (ii)), *Code-producer* retains a large and significant fundraising premium, while *Code-washer* collapses toward zero and becomes statistically indistinguishable from the closed-source baseline; a Wald test rejects the equality of the two coefficients in both columns. In the low-coverage subsample (columns (iii) and (iv)), neither classification indicator is individually significant and the Wald test fails to reject equality, consistent with pooling persisting where the information environment is sparse. The *information quality* split in columns (v) through (viii) delivers the same pattern. In the high-quality subsample, *Code-producer* retains a significant premium while *Code-washer* attenuates substantially toward zero. In the low-quality subsample, neither type can be distinguished from the closed-source baseline.

Taken together, the evidence supports the predicted transition from a pooling to a separating equilibrium. When investors can cross-validate the GitHub signal against an independent body of information, they price code-producers and code-washers differently. When the signal stands alone, they cannot tell the two apart. The empirical incidence of disclosure manipulation is therefore not a fixed feature of the market but is endogenous to the quality of the surrounding information infrastructure, a point we develop further when we exploit the rise of a substance-blind rating intermediary as a quasi-natural experiment in Section 4.3.

4.3 Endogeneity Concerns and Identification

The baseline pooling result in Table 6 may reflect selection on observables or on unobservables rather than the causal effect of the code-washing signal. The decision to maintain a public repository without producing substantive code is a strategic choice. Ventures that elect this strategy may differ from closed-source ventures in unobservable dimensions, including managerial sophistication, marketing capability, and access to resources, that independently predict fundraising success. To the extent that any of these factors drive the estimated coefficient on *Code-washer*, the pooling result could reflect omitted-variable bias rather than the strategic-disclosure mechanism we propose. We address this concern with two complementary identification strategies. Section 4.3.1 reports a propensity-score-matched comparison that absorbs selection on observable venture characteristics. Section 4.3.2 reports a triple-difference design that exploits a quasi-natural experiment, namely the rise of an algorithmic rating platform whose scoring rule rewarded the display of a GitHub URL without conditioning on the substance of the linked repository. The shock altered the marginal payoff to mimicry while leaving the underlying distribution of venture quality unchanged.

4.3.1 Propensity Score Matching

To absorb selection on observables, we construct a propensity-score-matched sample in which each code-washer is paired one-to-one with the observably most similar closed-source venture. The propensity to code-wash is estimated by Logit on the full vector of controls together with geographic-region and industry fixed effects, and each code-washer is matched to its nearest closed-source neighbor without replacement. Appendix Table A3 reports the post-matching balance: none of the eleven control variables exhibits a statistically significant difference between matched code-washers and closed-source ventures, confirming that the procedure eliminates systematic observable differences between the two groups. We then reintroduce the code-producers into the matched sample and re-estimate equation (5).

Columns (iii) and (iv) of Table 6 report the matched-sample estimates. On the extensive margin (column (iii)), both *Code-producer* ($\hat{\beta}_1 = 0.43$, $t = 3.82$) and *Code-washer* ($\hat{\beta}_2 = 0.33$, $t = 2.21$) are associated with a significantly higher probability of fundraising success relative to observably comparable closed-source ventures. The intensive-margin estimates in column (iv) deliver the same qualitative pattern. Crucially, a Wald test fails to reject $\beta_1 = \beta_2$ in either specification. The pooling result is therefore not driven by observable differences between code-washers and closed-source ventures; investors remain unable to distinguish the two types after the matching adjustment.

4.3.2 A Quasi-Natural Experiment: Algorithmic Rating Shock

The matched-sample evidence controls for selection on observables but cannot rule out the possibility that an unobserved factor jointly determines a venture's decision to code-wash and its fundraising success. An unobserved component of marketing capability, for instance, may both help a venture manufacture a credible-looking GitHub presence and persuade investors directly through other channels. To address this concern, we exploit a quasi-natural experiment that altered the marginal payoff to displaying a GitHub signal while leaving the underlying venture characteristics that drive fundraising success unchanged: the rise of algorithmic rating platforms that rewarded GitHub presence in their automated scoring.

ICObench was the dominant rating platform during our sample period, providing ratings for more than 5,000 token-offering projects.[10] The platform combined algorithmic scoring with human expert assessment. The algorithm evaluated approximately thirty criteria, which included the presence of a GitHub link but not the quality of code commits, the frequency of development activity, or the substantive content of the linked repository. Its scoring rule rewarded any visible GitHub URL with the same weight, regardless of whether the linked repository contained one commit or one thousand. This feature provides the variation our design exploits. The platform's rise raised the marginal payoff to displaying a GitHub URL but offered no corresponding payoff for the substantive code activity behind it. The prediction that follows is asymmetric: the rating shock should differentially raise the fundraising premium of code-washers, whose signal benefits fully from the algorithmic weight, while having no effect on the premium of code-producers, who already supplied verifiable substance behind the link.

We test the prediction with a triple-difference design that exploits variation along three dimensions: venture type (code-washer or code-producer, with closed-source as the baseline), time (before or after ICObench became the dominant rating platform in January 2018), and exposure (high or low coverage on rating platforms). The specification is

$$\begin{aligned} Y_i \;=\; \alpha \;+\; & \beta_1 \, \textit{Code-washer}_i \times \textit{Post}_t \times \textit{HighCov}_i \\ & + \; \beta_2 \, \textit{Code-producer}_i \times \textit{Post}_t \times \textit{HighCov}_i \\ & + \; \boldsymbol{\gamma}' \mathbf{Z}_{i,t} \;+\; \Theta_i \;+\; \Lambda_i \;+\; \varepsilon_i, \end{aligned} \tag{6}$$

where $\mathbf{Z}_{i,t}$ collects all lower-order interactions of *Code-washer*, *Code-producer*, *Post*, and *HighCov*, and the remaining controls and fixed effects are as defined for equation (5). The exposure indicator *HighCov* equals one if a venture's coverage across rating platforms exceeds the sample median, constructed following Lyandres et al. (2022) as a function of the number of data sources reporting information on

[10] ICObench launched on August 2, 2017 and released its automated rating algorithm "Benchy" on September 10, 2017. See https://icobench.com/about-us/.

the venture; ventures with higher coverage are more likely to be listed and rated by ICObench, and so more exposed to the algorithmic rating shock. The triple-interaction coefficients β_1 and β_2 capture the differential change in the code-washing and code-producing premia attributable to intensified exposure to algorithmic ratings after the platform shock. The asymmetric channel articulated above predicts $\beta_1 > 0$, β_2 statistically indistinguishable from zero, and a Wald test rejecting $\beta_1 = \beta_2$.

Panel A of Table 8 reports the main results. The triple-interaction coefficient on *Code-washer* is positive and statistically significant for both *RaisedDummy* ($\hat{\beta}_1 = 0.29$, $t = 2.47$) and *FundsRaised (log)* ($\hat{\beta}_1 = 4.00$, $t = 2.72$). The analogous coefficient on *Code-producer* is statistically indistinguishable from zero in both columns, and a Wald test rejects $\beta_1 = \beta_2$ at conventional levels. The asymmetry is the empirical signature our identification predicts: when an information intermediary commits to evaluating the display of a signal rather than its content, the empirical fundraising response is concentrated entirely among ventures whose signal lacked content to begin with. Code-producers, whose repositories already contained substantive development activity, derived no additional fundraising premium from the rise of algorithmic rating. The pattern is incompatible with explanations in which an omitted venture characteristic, such as marketing capability or managerial sophistication, drives both code-washing and fundraising success: such an unobserved factor would lift the fundraising premia of code-washers and code-producers symmetrically, but the data show no such lift on the code-producer side.

Panel B of Table 8 reports robustness tests along three dimensions. First, we assess the parallel-trends assumption by re-estimating equation (6) on the pre-treatment subsample of offerings ending before January 2018, using placebo treatment dates of September, October, and November 2017. The triple-interaction coefficient is statistically indistinguishable from zero across all three placebos, supporting the absence of a differential pre-trend between code-washers and the comparison groups in the months preceding the rating-platform shock. Second, the main coefficient is stable across alternative definitions of rating-platform exposure: top-tercile coverage, top-quartile coverage, and a continuous coverage measure all yield statistically significant triple-interaction coefficients of similar magnitude. Third, the result holds in a subsample restricted to Ethereum-based projects, which constitute the majority of blockchain ventures with public GitHub repositories. Taken together, the robustness battery indicates that the asymmetric response in Panel A reflects the differential exposure of code-washers to a substance-blind rating algorithm, rather than a coincidental shift in market conditions or an artifact of a specific cut of the data.

5 Consequences of Code-Washing

The preceding sections established that code-washers and code-producers earn statistically indistinguishable fundraising premia at the moment of fundraising, that this pooling attenuates as the surround-

ing information environment becomes richer, and that the asymmetric response to the algorithmic rating shock is incompatible with explanations rooted in omitted venture characteristics. A natural question is whether the short-term fundraising success of code-washers carries any subsequent cost. The signaling literature predicts that strategic mimicry generates short-term gains at the expense of long-run reputational damage, with the trade-off sharpest in settings where the underlying type is eventually revealed through observable performance (Spence, 1973; Teoh et al., 1998; Karpoff et al., 2008; Amiram et al., 2025). This section documents the trade-off along the three margins that become observable in the post-offering window. Section 5.1 examines financial performance in the post-listing market and the trajectory of substantive development effort on GitHub. Section 5.2 examines the tail of post-offering misconduct and shows that code-washers are systematically over-represented among ventures that ultimately defraud investors.

5.1 Long-Term Performance

If code-washing reflects opportunistic disclosure rather than intrinsic technological quality, the short-term fundraising premium documented in Section 4.1 should reverse in the post-offering window as the underlying type is revealed through observable performance. Two channels of revelation are available in our setting. First, once tokens are listed on secondary markets, prices, trading volume, and return volatility provide a high-frequency public signal of how investors and traders revise their assessment of the venture's quality. Second, once the fundraising stage closes, GitHub development activity becomes a forward-looking indicator of whether the venture is delivering on its technological premise or has abandoned it. H3 predicts that code-producers should outperform code-washers along both channels.

We test the prediction with the following specification:

$$LTPerformance_i = \alpha + \beta_1 \, Code\text{-}producer_i + \beta_2 \, Code\text{-}washer_i + \Theta_i + \Lambda_i + \varepsilon_i, \tag{7}$$

where the controls Θ and fixed effects Λ follow the specification of equation (5) and standard errors are clustered at the venture-fundraising completion month. We measure *LTPerformance* along two dimensions. *Financial performance* is measured over the 180 days following exchange listing using four standard market-based proxies: *Returns (log)*, the natural logarithm of cumulative returns; *Volume (log)*, the logarithm of one plus the mean daily trading-volume rate; *Volatility (log)*, the logarithm of return volatility; and *Illiquidity (log)*, the logarithm of the Amihud illiquidity measure. The financial-performance sample is the full set of listed token offerings, with closed-source ventures as the omitted reference category. *Development performance* is measured over the 180 days following the close of the fundraising stage using three GitHub-based proxies: *Commits (log)*, *Pulls (log)*, and *Contributors (log)*, each defined as the natural logarithm of one plus the cumulative count of, respectively, commits,

pull requests, and distinct contributors. Because development variables are observed only for ventures with public GitHub repositories, the development-performance sample is restricted to the open-source subsample and uses code-washers as the omitted reference category, so that the reported β_1 captures the post-fundraising development gap between code-producers and code-washers directly. H3 implies that β_1 should be positive on metrics for which higher values indicate stronger performance (returns, volume, commits, pulls, contributors) and negative on metrics for which higher values indicate weaker performance (volatility, illiquidity). In the financial-performance specifications, β_2 should additionally be statistically indistinguishable from zero, reflecting the prediction that code-washers earn financial outcomes no better than ventures that never disclosed any code at all.

Panel (a) of Table 9 reports the financial-performance results. The coefficient on *Code-producer* is positive and significant for post-listing returns ($\hat{\beta}_1 = 0.35$, $t = 1.72$) and trading volume ($\hat{\beta}_1 = 0.76$, $t = 2.67$), and is negative and significant for return volatility ($\hat{\beta}_1 = -0.02$, $t = -1.97$). The coefficient on *Code-washer*, by contrast, is statistically indistinguishable from zero on every financial metric. A Wald test rejects $\beta_1 = \beta_2$ at conventional levels for returns, volatility, and illiquidity ($\Pr > \chi^2 = 0.017$, 0.019, and 0.096, respectively). The secondary market therefore prices the underlying type that investors could not separate at the fundraising stage: code-producers deliver economically meaningful post-listing returns, attract higher trading volume, and exhibit lower volatility, while code-washers earn outcomes statistically indistinguishable from those of ventures that never disclosed any code.

Panel (b) of Table 9 reports the development-performance results. Code-producers exhibit substantially higher post-fundraising development activity than code-washers along all three GitHub-based proxies, including commits ($\hat{\beta}_1 = 3.33$, $t = 34.4$), pull requests ($\hat{\beta}_1 = 0.62$, $t = 7.56$), and distinct contributors ($\hat{\beta}_1 = 0.49$, $t = 6.83$). The economic magnitudes are large: the point estimate on commits implies that, conditional on observables and fixed effects, code-producers record roughly an order of magnitude more substantive coding activity in the 180 days following the close of fundraising than code-washers. The development-performance results therefore reinforce the financial-performance evidence: code-washers not only deliver weaker financial outcomes, they also fail to sustain any substantive technical effort once the fundraising window closes. The fundraising-period GitHub signal of code-producers anticipates ongoing development; the analogous signal of code-washers does not.

Taken together, the long-term performance results indicate that the short-term fundraising premium documented in Section 4.1 is more than offset by significantly worse post-offering outcomes for code-washers. Once tokens begin trading and the venture's development trajectory becomes publicly observable, both the secondary market and the developer community withdraw their endorsement of code-washers. Code-producers, whose pre-fundraising signal was backed by substantive development effort, continue to deliver on the technological premise of the offering. The pattern is the precise reversal

that H3 predicts.

5.2 Financial Misconduct

An extreme manifestation of the post-offering performance reversal documented in Section 5.1 is outright financial misconduct. A salient case is *BitConnect*, the high-profile blockchain venture illustrated in Figure A2. BitConnect created its sole GitHub repository in the immediate run-up to its main fundraising event, recorded zero commits in the year-long pre-offering window of $[-365, -90]$ days, and accumulated a lifetime total of seven commits. Yet the venture went on to reach a peak market capitalization of $3.4 billion before defrauding investors of $2.69 billion through a Ponzi-style "lending program" and being charged by the SEC.[11] The pattern, a hastily created GitHub presence concealing the absence of any substantive technology, is the empirical signature of code-washing as we define it. We now ask whether the BitConnect case generalizes across the cross-section: do code-washers systematically populate the tail of post-offering misconduct?

We re-estimate equation (7) using two proxies for post-offering scam involvement as the dependent variable. *Number of Scams* is the count of scam events in which the venture was implicated after fundraising, collected from `deadcoins.com` and estimated by Poisson regression. *Scam Dummy* is an indicator equal to one if the venture was flagged in at least one such scam event, estimated by Logit. Closed-source ventures form the omitted reference category in both specifications. H3 predicts that code-washers carry a higher probability and intensity of post-offering misconduct than both closed-source ventures and code-producers, formally $\beta_2 > 0$, β_1 statistically indistinguishable from zero, and a Wald test rejecting $\beta_1 = \beta_2$.

Table 10 reports the results. The coefficient on *Code-washer* is positive and significant at the one percent level for both *Number of Scams* ($\hat{\beta}_2 = 0.57$, $z = 3.02$) and *Scam Dummy* ($\hat{\beta}_2 = 0.71$, $z = 4.43$). The coefficient on *Code-producer*, by contrast, is statistically indistinguishable from zero in both columns, and a Wald test rejects $\beta_1 = \beta_2$ at the one percent level in both specifications ($\Pr > \chi^2 = 0.007$ and 0.005). The point estimate on *Scam Dummy* implies that code-washers face more than twice the odds of being flagged in a post-offering scam than code-producers, evaluated at the sample-mean baseline.

The misconduct evidence completes the dynamic picture that our framework predicts. The pooling equilibrium documented in Section 4.1 allows ventures that never intended to deliver on their technological promise to secure capital alongside genuine code-producers. The performance evidence in Section 5.1 shows that this misallocation is partially corrected after listing, as the secondary market and the developer community both withdraw their endorsement of code-washers. The misconduct evidence in this section identifies the tail of the distribution for which the correction arrives too late to recover:

[11] See `https://www.sec.gov/newsroom/press-releases/2021-172`.

a subset of code-washers raises capital, fails to deliver any technological substrate, and exits with the proceeds before the absence of code becomes publicly observable. Code-washing therefore does more than misallocate capital. It operationalizes a channel through which strategic information manipulation around capital-market events shades into outright fraud against investors.

6 Conclusion

This paper identifies and studies a new form of strategic disclosure manipulation around capital-market events, which we term *code-washing*: the inflation of a public software repository through superficial coding activity, designed to project an image of substantive technological progress when none exists. The practice exploits an asymmetry inherent to voluntary technology disclosure, in which the cost of displaying a signal is trivial relative to the cost of producing the substance behind it. Code-washing pools with genuine code production at the moment of fundraising, attenuates as the surrounding information environment becomes denser, and reverses sharply in the post-offering window, where code-washers underperform along financial, developmental, and misconduct margins as the underlying type becomes observable. An asymmetric quasi-natural experiment that exploits the rise of an algorithmic rating platform isolates the strategic-disclosure mechanism from omitted-variables alternatives.

The findings carry direct implications for the design of accounting institutions in digital-asset markets. Token-issuing ventures fall almost entirely outside the conventional financial-reporting perimeter, and the voluntary, technology-related signals that investors observe are systematically manipulated and predictive of fraud. The manipulation responds elastically to the design of credibility-enhancing institutions: it attenuates when third-party intermediaries provide cross-validation, and intensifies when intermediaries score the display of a signal rather than its content. These results inform the ongoing debate over the appropriate audit, certification, and disclosure-regulation frameworks for digital-asset markets, and suggest that the marginal regulatory dollar is better spent on institutions that verify the substance of voluntary disclosures than on those that score their surface form.

Code-washing joins greenwashing in ESG reporting, diversity-washing in workforce disclosures (Baker et al., 2024), and cosmetic earnings management around capital-market events (Rangan, 1998; Teoh et al., 1998) as part of a broader class of disclosure phenomena in which firms exploit the gap between the minimal cost of producing signals and the higher cost of verifying them (Healy and Wahlen, 1999; Ball, 2001). The phenomenon we document operates outside the audit perimeter and the institutional gatekeepers that constrain comparable behavior in traditional capital markets, and identifies a class of disclosure manipulation that current accounting institutions are structurally ill-equipped to police. Closing that gap is the central design problem of the emerging accounting of digital assets.

References

Aboody, D. and R. Kasznik (2000). CEO stock option awards and the timing of corporate voluntary disclosures. *Journal of Accounting and Economics 29*(1), 73–100.

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., E. Lyandres, and D. Rabetti (2025). Trading volume manipulation and competition among centralized crypto exchanges. *Management Science 71*(10), 8604–8622.

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.

Baker, A. C., D. F. Larcker, C. G. McClure, D. Saraph, and E. M. Watts (2024). Diversity washing. *Journal of Accounting Research 62*(5), 1359–1406.

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.

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.

Bourveau, T., E. T. De George, A. Ellahie, and D. Macciocchi (2022). The role of disclosure and information intermediaries in an unregulated capital market: Evidence from initial coin offerings. *Journal of Accounting Research 60*(1), 129–167.

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

Bushee, B. J. and C. Leuz (2005). Economic consequences of SEC disclosure regulation: Evidence from the OTC bulletin board. *Journal of Accounting and Economics 39*(2), 233–264.

Christensen, D. M., G. Serafeim, and A. Sikochi (2022). Why is corporate virtue in the eye of the beholder? the case of ESG ratings. *The Accounting Review 97*(1), 147–175.

Cohen, L. H., U. G. Gurun, and Q. H. Nguyen (2026). The ESG-innovation disconnect: Evidence from green patenting. *Management Science*. Articles in Advance.

Cong, L. W., E. S. Prasad, and D. Rabetti (2025). Financial and informational integration through oracle networks. NBER Working Paper 33639, National Bureau of Economic Research.

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. P. Roche (2025). Beefing IT up for your investor? engagement with open source communities, innovation, and startup funding: Evidence from GitHub. *Organization Science 36*(4), 1551–1573.

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.

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.

Dyck, A., K. V. Lins, L. Roth, and H. F. Wagner (2019). Do institutional investors drive corporate social responsibility? International evidence. *Journal of Financial Economics 131*(3), 693–714.

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 127*(4), 1739–1776.

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

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

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.

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 (2008). The consequences to managers for financial misrepresentation. *Journal of Financial Economics 88*(2), 193–215.

Lawrence, C. (2023). European and UK startups lead the way in open source AI despite regulation.

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.

Leland, H. E. and D. H. 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. *Pacific-Basin Finance Journal 2*(2–3), 165–199.

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.

Mas-Colell, A., M. D. Whinston, and J. R. Green (1995). *Microeconomic Theory*. New York: Oxford University Press.

Mayzlin, D., Y. Dover, and J. 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.

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

Momtaz, P. P. (2021). The pricing and performance of cryptocurrency. *The European Journal of Finance 27*(4–5), 367–380.

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 57*(2), 215–240.

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.

Soltes, E. (2016). *Why They Do It: Inside the Mind of the White-Collar Criminal*. New York: PublicAffairs.

Spence, M. (1973). Job market signaling. *The Quarterly Journal of Economics 87*(3), 355–374.

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

Syer, M. D., M. Nagappan, B. Adams, and A. E. Hassan (2015). Studying the relationship between source code quality and mobile platform dependence. *Software Quality Journal 23*(3), 485–508.

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

Vasilescu, B., Y. Yu, H. Wang, P. Devanbu, and V. Filkov (2015). Quality and productivity outcomes relating to continuous integration in GitHub. In *Proceedings of the 2015 10th Joint Meeting on Foundations of Software Engineering*, pp. 805–816.

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

Figure 1. Measuring Pre-TO Code Production and Token-Offering Code-Washing. Figure (a) illustrates the event-time windows used to construct our code-washing measure. The pre-TO window, defined as $[-365, -90]$ days relative to the token-offering start date, captures technical code output that predates the immediate fundraising period and is therefore less likely to reflect fundraising-timed disclosure. The token-offering window captures code production during the fundraising period, when startups have stronger incentives to create an appearance of technological development. Figure (b) reports the distribution of cumulative commits during the pre-TO window. The median startup generates one cumulative commit in this window, providing a natural cutoff based on the presence of observable pre-TO code production. We classify startups with zero pre-TO commits as *Code-washers* and those with at least one pre-TO commit as *Code-producers*.



Figure (a): Event-time windows for measuring code-washing

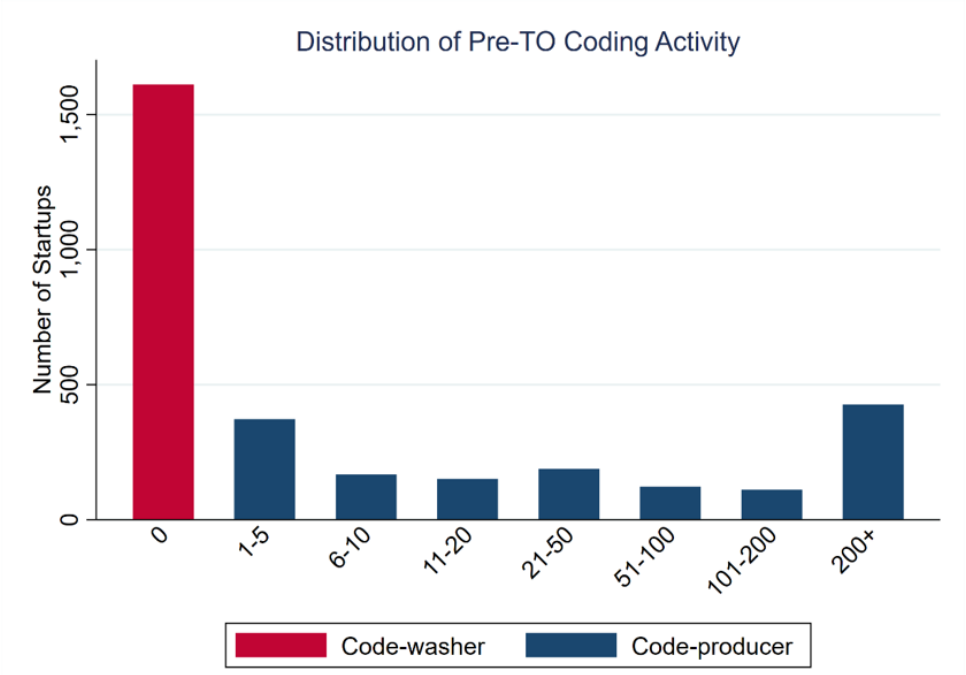


Figure (b): Distribution of pre-TO coding activity

Figure 2. Code Production Around Token-Offering Events. This figure plots the evolution of code production around the token-offering (TO) start date for *Code-washers* and *Code-producers*. Code production is measured as the number of new GitHub commits generated within each event-time interval. The event window spans from 360 days before to 360 days after the TO start date. The x-axis labels mark event time relative to the TO start date, and each point represents the number of new commits generated within the interval ending at that event-time point and beginning at the previous point. For example, the point labeled "0" corresponds to commits made during the interval $[-90, 0]$ days.

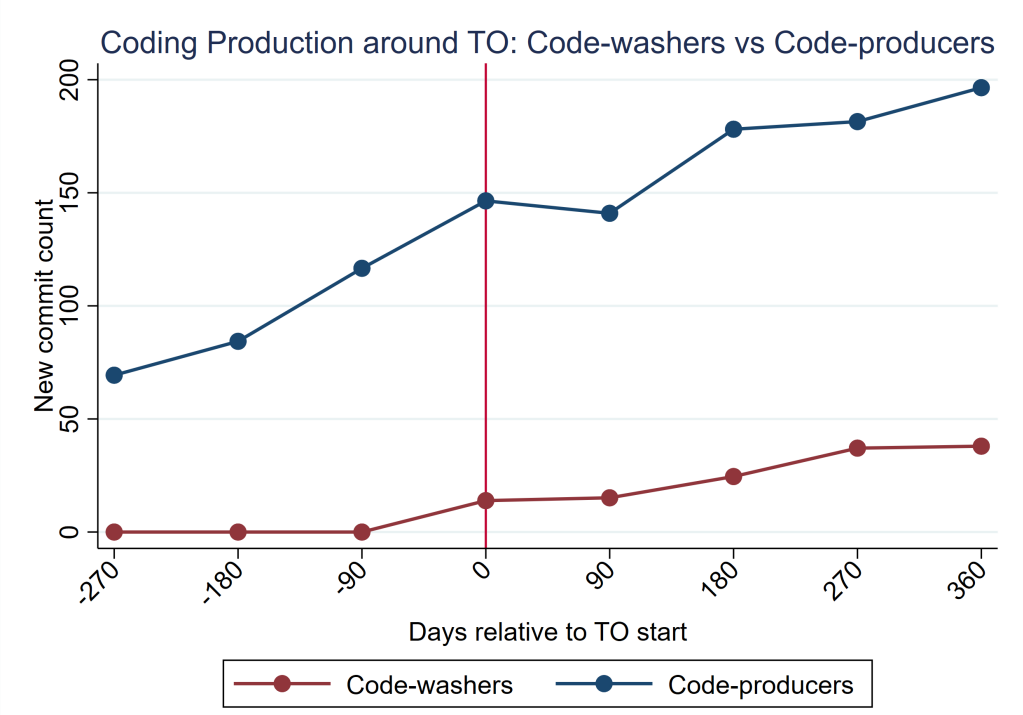


Table 1. Summary Statistics. This table reports summary statistics for the sample used in our main analyses. Panel A reports summary statistics for the full sample. Panel B compares startup characteristics between *Code-washers* and *Code-producers* and reports differences in means and corresponding *t*-statistics. *Open Source* is an indicator variable equal to one if the blockchain startup has a public GitHub repository before the token-offering (TO) period, and zero otherwise. *Code-washer* is an open-source startup with zero commits during the pre-TO window $[-365, -90]$ days relative to the TO start date, while *Code-producer* is an open-source startup with at least one commit during the same window. *RaisedDummy* is an indicator variable equal to one if the startup raised any funds by the end of the TO period, and zero otherwise. *FundsRaised (log)* is the logarithm of the total amount raised by the end of the TO period. *Tokens for sale (%)* is the percentage of tokens offered for sale to the total token supply. *Hardcap (log)* is the logarithm of the maximum amount the startup allows itself to raise. *Whitelist* is an indicator variable equal to one if the startup requires prospective investors to register on a whitelist, and zero otherwise. *KYC* is an indicator variable equal to one if the startup requires investors to satisfy know-your-customer procedures, and zero otherwise. *White paper* is an indicator variable equal to one if the startup discloses a white paper, and zero otherwise. *Team size (log)* is the logarithm of the number of disclosed team members. *Presale* is an indicator variable equal to one if the startup conducts a presale before the main fundraising stage, and zero otherwise. *Twitter (log)* is the logarithm of one plus the number of tweets posted by the startup's official account before the TO start date. *Reddit (log)* is the logarithm of one plus the number of Reddit discussions before the TO start date. *BTCTalk (log)* is the logarithm of one plus the number of Bitcointalk posts before the TO start date. *Medium (log)* is the logarithm of one plus the number of Medium articles before the TO start date. Appendix A1 provides detailed definitions for all variables used in our empirical analysis. The symbols *, **, and *** denote significance at the 10%, 5%, and 1% levels, respectively.

Panel A: Summary statistics for full sample

	Obs.	Mean	SD	p25	p50	p75
(a) Open-source activity:						
Open Source	2,326	0.66	0.48	0	0	1
Code-producer	2,326	0.31	0.46	0	0	1
Code-washer	2,326	0.35	0.48	0	0	1
(b) Fundraising outcomes:						
RaisedDummy	2,326	0.56	0.50	0	1	1
FundsRaised (log)	2,326	8.80	7.29	0.69	12.79	15.64
(c) Control variables:						
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

Panel B: Code-washers vs. code-producers

	Code-washers	Code-producers	Difference	*t*-value
Tokens for sale (%)	0.55	0.56	–0.00	0.10
Hardcap (log)	16.69	16.68	0.01	0.05
Whitelist	0.50	0.60	–0.10***	–3.96
KYC	0.75	0.73	0.01	0.56
White paper	0.51	0.47	0.03	1.36
Team size (log)	2.29	2.31	–0.02	–0.48
Presale	0.68	0.72	–0.04*	–1.79
Twitter (log)	2.78	3.34	–0.56***	–4.13
Reddit (log)	1.34	2.17	–0.84***	–4.47
BTCTalk (log)	3.47	3.71	–0.24*	–1.82
Medium (log)	0.62	0.94	–0.32***	–3.21
Observations	804	740		

Table 2. Validation tests: market-timing code-washing, debug development, and developer engagement during the fundraising phase. **Panel (a)** reports validation tests for *market-timing code-washing*, examining commit timing during the token offering (TO) phase, repository abandonment after the offering, and outright deletion of the public repository. The dependent variable in column (i) is *CodeConcentration*—the ratio of cumulative commits submitted during the TO to the total cumulative commits at the end of the TO. The dependent variable in column (ii) is *RepoAbandoned*—a dummy equal to one if the startup has zero commits in the 180 days following the end of TO. The dependent variable in column (iii) is *RepoDeleted*—a dummy equal to one if the public repository has been removed from GitHub by the founder, which means the GitHub URL recorded for the startup returns HTTP status 404 when re-queried 3 years following the end of TO. **Panel (b)** reports the validation test for *genuine development* activity during the TO fundraising phase, examining bug-fixing activity. The dependent variable in column (iv) is *Debug Ratio*—the ratio of debugging-related commits to total commits made before the end of TO; a commit is classified as debugging-related if its message contains terms such as "fix", "bug", "error", or "patch". **Panel (c)** reports validation tests for *developer engagement* activity during the TO fundraising phase, examining issues and pull requests. The dependent variable in column (v) is *Issues (log)*—the natural logarithm of one plus the cumulative number of GitHub issues created before the end of TO; an issue is a tool used to track tasks, enhancements, or bugs in a repository. The dependent variable in column (vi) is *Pulls (log)*—the natural logarithm of one plus the cumulative number of GitHub pull requests submitted before 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 variable of interest is as follows: *Code-washer* is an open-source startup with zero commits during the pre-TO window $[-365, -90]$ days relative to the TO start date. The sample includes only open-source startups, with *Code-producer* as the baseline. Columns (i), (iv), (v), and (vi) are estimated using OLS; columns (ii) and (iii) are estimated using a Logit model. Table 1 includes the control variables. All specifications include year-month, industry, and geographic region fixed effects. The standard errors are clustered at the venture-fundraising completion date, and t-statistics (z-statistics for Logit columns) are reported in parentheses. The symbols *, **, and *** denote significance at the 10%, 5%, and 1% levels, respectively.

	(a) Market-timing code-washing			**(b) Genuine development**	**(c) Developer engagement**	
	(i)	(ii)	(iii)	(iv)	(v)	(vi)
	CodeConcentration	*RepoAbandoned*	*RepoDeleted*	*Debug Ratio*	*Issues (log)*	*Pulls (log)*
Variable of interest:						
Code-washer	0.206***	1.967***	0.776***	–0.062***	–0.404***	–0.333***
	(10.170)	(15.565)	(5.950)	(–6.100)	(–8.750)	(–8.290)
Controls:						
Tokens for sale (%)	–0.040	0.987***	0.058	–0.000	0.062	0.105
	(–1.148)	(3.583)	(0.140)	(–0.010)	(0.490)	(0.890)
Hardcap (log)	–0.002	0.070	0.035	–0.000	–0.016	–0.016
	(–0.255)	(1.358)	(0.480)	(–0.120)	(–0.910)	(–0.990)
Whitelist	0.000	–0.296*	–0.205	0.007	0.051	0.043
	(0.029)	(–1.797)	(–1.220)	(0.490)	(0.990)	(0.890)
KYC	0.012	–0.615***	–0.333*	0.028**	0.103	0.101
	(0.691)	(–3.646)	(–1.760)	(2.020)	(1.430)	(1.680)
White paper	–0.011	–0.432***	–0.154	0.009	–0.005	–0.007
	(–0.693)	(–4.787)	(–0.990)	(0.870)	(–0.100)	(–0.130)
Team size (log)	–0.002	–0.029	–0.098	0.004	–0.021	–0.026
	(–0.186)	(–0.356)	(–1.020)	(0.660)	(–0.890)	(–1.130)
Presale	0.002	0.269*	0.090	–0.014	–0.085	–0.088
	(0.072)	(1.763)	(0.520)	(–1.090)	(–1.200)	(–1.330)
Twitter (log)	0.002	–0.013	–0.012	0.003*	0.028***	0.028***
	(0.458)	(–0.394)	(–0.300)	(1.660)	(3.380)	(3.780)
Reddit (log)	–0.003	–0.030	0.023	0.001	0.034***	0.031**
	(–1.166)	(–1.091)	(0.810)	(0.390)	(2.770)	(2.530)
BTCTalk (log)	–0.001	–0.000	0.015	0.004**	–0.020**	–0.019**
	(–0.219)	(–0.017)	(0.380)	(2.020)	(–2.210)	(–2.060)
Medium (log)	–0.008	–0.193***	0.107*	0.005	0.035	0.029
	(–1.468)	(–5.635)	(1.710)	(1.160)	(1.440)	(1.350)
Year-month FE	Yes	Yes	Yes	Yes	Yes	Yes
Industry FE	Yes	Yes	Yes	Yes	Yes	Yes
Geographic region FE	Yes	Yes	Yes	Yes	Yes	Yes
Model	OLS	Logit	Logit	OLS	OLS	OLS
Obs.	1,007	1,544	1,353	1,020	1,544	1,544
(Pseudo) Adj.r^2	0.183	0.228	0.061	0.096	0.125	0.111

Table 3. Validation tests: whitepaper informativeness. This table reports the relation between code washing and information disclosure in the startup's whitepaper. The dependent variable in column (i) is *Word count*—the total count of words per page in the startup's whitepaper. The dependent variable in column (ii) is *Tech word count*—the count of technology-related words in the startup's whitepaper, identified using natural language processing techniques. The dependent variable in column (iii) is *Image count*—the count of images in the startup's whitepaper. The variables of interest are as follows: *Code-washer* is an open-source startup with zero commits during the pre-TO window $[-365, -90]$ days relative to the TO start date, while *Code-producer* is an open-source startup with at least one commit during the same window. We report the Chi-Squared test results for the hypothesis $H_0 : \beta_1 = \beta_2$. We estimate Poisson regressions in all specifications. Table 1 describes the remaining control variables. All 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) ***Word count***	(ii) ***Tech word count***	(iii) ***Image count***
Variables of interest:			
β_1(*Code-producer*)	0.085*	0.115***	0.156
	(1.720)	(3.587)	(1.364)
β_2(*Code-washer*)	0.002	0.040	–0.296***
	(0.046)	(1.468)	(–3.293)
Controls:			
Tokens for sale (%)	0.054	–0.094	0.089
	(0.482)	(–1.440)	(0.526)
Hardcap (log)	0.095***	0.078***	0.135***
	(4.656)	(7.181)	(3.984)
Whitelist	0.117*	0.099***	0.149
	(1.943)	(3.382)	(1.102)
KYC	0.147***	0.124***	0.267**
	(2.941)	(4.268)	(2.417)
Team size (log)	0.070	0.069***	0.147***
	(1.109)	(2.960)	(3.219)
Presale	–0.051	–0.041	–0.105
	(–1.251)	(–1.286)	(–0.677)
Twitter (log)	0.014	0.009	0.009
	(1.281)	(1.437)	(0.505)
Reddit (log)	0.013	0.007	–0.007
	(0.987)	(1.177)	(–0.379)
BTCTalk (log)	0.001	0.003	0.039**
	(0.088)	(0.407)	(2.448)
Medium (log)	0.011	0.018	0.015
	(0.705)	(1.594)	(0.495)
Year-month FE	Yes	Yes	Yes
Industry FE	Yes	Yes	Yes
Geographic region FE	Yes	Yes	Yes
Model	Poisson	Poisson	Poisson
Obs.	1,099	1,099	1,099
Pseudo r^2	0.164	0.214	0.106
$H_0 : \beta_1 = \beta_2$	Pr. > χ^2 = 0.029**	Pr. > χ^2 = 0.012**	Pr. > χ^2 = 0.000***

Table 4. Validation tests: voluntary disclosure on LinkedIn. This table reports the validation tests for the relation between code-washing and team disclosure behavior on LinkedIn. The dependent variable in column (i) is *Any member disclosed*—an indicator variable equal to one if at least one team member explicitly identifies the blockchain startup on their LinkedIn profile. The dependent variable in column (ii) is *Share of disclosed*—the fraction of team members who disclose their involvement with the blockchain startup on LinkedIn. The variables of interest are as follows: *Code-washer* is an open-source startup with zero commits during the pre-TO window $[-365, -90]$ days relative to the TO start date, while *Code-producer* is an open-source startup with at least one commit during the same window. We report the Chi-Squared test results for the hypothesis $H_0 : \beta_1 = \beta_2$. Column (i) is estimated using Logit; column (ii) is estimated using Tobit. Table 1 describes the remaining control variables. All 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) ***Any member disclosed***	(ii) ***Share of disclosed***
Variables of interest:		
β_1(*Code-producer*)	1.359***	0.062**
	(4.165)	(2.158)
β_2(*Code-washer*)	0.163	–0.007
	(0.586)	(–0.278)
Controls:		
Tokens for sale (%)	–0.931	–0.106
	(–1.289)	(–1.634)
Hardcap (log)	–0.366**	–0.010
	(–2.301)	(–0.518)
Whitelist	–0.401	0.003
	(–1.018)	(0.047)
KYC	1.440***	0.187***
	(3.113)	(2.758)
Whitepaper	0.244	–0.010
	(0.457)	(–0.269)
Team size (log)	0.670***	0.046
	(3.263)	(1.576)
Presale	–0.601	–0.067**
	(–1.257)	(–2.319)
Twitter (log)	–0.016	0.006
	(–0.181)	(0.920)
Reddit (log)	0.007	0.012
	(0.071)	(1.419)
BTCTalk (log)	0.047	–0.001
	(0.891)	(–0.142)
Medium (log)	0.378*	0.022*
	(1.694)	(1.892)
Year-month FE	Yes	Yes
Industry FE	Yes	Yes
Geographic region FE	Yes	Yes
Model	Logit	Tobit
Obs.	391	414
Pseudo r^2	0.232	0.289
$H_0 : \beta_1 = \beta_2$	Pr. > χ^2 = 0.017**	Pr. > χ^2 = 0.021**

Table 5. Validation tests: SEC enforcement. This table reports the effects of the SEC's first civil enforcement actions against unregistered token offerings, announced on November 16, 2018. The sample window spans May 2018 to May 2019. *Post (SEC)* equals one for token offerings completed after November 16, 2018, and zero otherwise. Panel (a) examines the composition shift among new token-offering entrants: the dependent variable is the indicator *Code-washer* in column (i), *Code-producer* in column (ii), and *Closed-source* in column (iii), each estimated by Probit. Panel (b) examines the change in whitepaper informativeness around the shock and is restricted to startups that disclose a whitepaper; the dependent variable is *Word count* in column (i), *Tech word count* in column (ii), and *Image count* in column (iii), each estimated by Poisson. *Code-washer* is an open-source startup with zero commits during the pre-TO window $[-365, -90]$ days relative to the token-offering start date; *Code-producer* is an open-source startup with at least one commit during the same window; closed-source startups form the omitted reference category. Table 1 describes the remaining control variables. All specifications include industry and geographic region fixed effects. Standard errors are clustered by ICO completion month, and z-statistics are reported in parentheses. The symbols *, **, and *** denote significance at the 10%, 5%, and 1% levels, respectively.

Panel (a): Composition shift around the SEC enforcement

	(i) ***Code-washer***	(ii) ***Code-producer***	(iii) ***Closed-source***
Variable of interest:			
Post (SEC)	–0.225***	0.300***	–0.103
	(–2.751)	(4.075)	(–0.949)
Controls	Yes	Yes	Yes
Industry FE	Yes	Yes	Yes
Geographic region FE	Yes	Yes	Yes
Model	Probit	Probit	Probit
Obs.	1,484	1,484	1,484
Pseudo-r^2	0.038	0.097	0.134

Panel (b): Whitepaper informativeness around the SEC enforcement

	(i) ***Word count***	(ii) ***Tech word count***	(iii) ***Image count***
Variables of interest:			
Code-washer	–0.075**	–0.052	–0.310***
	(–2.039)	(–1.598)	(–2.675)
Code-producer	0.094*	0.050	0.366**
	(1.694)	(1.078)	(2.407)
Post (SEC)	–0.014	–0.007	0.102
	(–0.243)	(–0.145)	(0.476)
Post (SEC) × *Code-washer*	0.222***	0.163***	–0.272
	(2.893)	(2.842)	(–1.427)
Post (SEC) × *Code-producer*	–0.008	–0.001	–0.452
	(–0.093)	(–0.021)	(–1.428)
Controls	Yes	Yes	Yes
Industry FE	Yes	Yes	Yes
Geographic region FE	Yes	Yes	Yes
Obs.	665	665	665
Pseudo-r^2	0.169	0.184	0.103

Table 6. Code production and fundraising: pooled equilibrium. This table reports estimates of equation (5), which examines the association between code production and fundraising outcomes. Panel (a) presents baseline regressions using the full sample. Panel (b) reports results using a 1:1 propensity score matched (PSM) sample, in which each *Code-washer* is matched to its closest closed-source startup; Appendix Table A3 reports the matching procedure and covariate balance. The dependent variable is either an indicator for whether the startup successfully raised any funds by the end of the fundraising stage, *RaisedDummy*, or the natural logarithm of total funds raised, *FundsRaised (log)*. The variables of interest are *Code-washer*—an open-source startup with zero commits during the pre-TO window $[-365, -90]$ days—and *Code-producer*—an open-source startup with at least one commit during the same window; closed-source startups form the omitted reference category. Columns (i) and (iii) are estimated using Logit models, while columns (ii) and (iv) are estimated using OLS. Table 1 describes the remaining control variables. All 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)
	RaisedDummy	***FundsRaised (log)***	***RaisedDummy***	***FundsRaised (log)***
	(a) Full sample		**(b) PSM sample**	
Variables of interest:				
β_1(*Code-producer*)	0.539**	1.612***	0.428***	1.195***
	(2.458)	(3.213)	(3.820)	(3.710)
β_2(*Code-washer*)	0.362**	1.015**	0.332**	0.944**
	(2.254)	(2.554)	(2.210)	(2.280)
Controls:				
Tokens for sale (%)	–0.362	–1.371**	–0.424*	–1.494**
	(–1.560)	(–2.073)	(–1.755)	(–2.218)
Hardcap (log)	0.005	0.355***	0.010	0.379***
	(0.106)	(2.953)	(0.222)	(2.969)
Whitelist	–0.086	0.074	–0.119	0.011
	(–0.731)	(0.205)	(–1.041)	(0.033)
KYC	0.862***	2.493***	0.901***	2.590***
	(4.376)	(5.282)	(5.165)	(6.263)
White paper	0.201*	0.636**	0.256**	0.791***
	(1.926)	(2.523)	(2.337)	(2.905)
Team size (log)	0.459***	1.305***	0.406***	1.193***
	(6.358)	(6.211)	(5.271)	(5.438)
Presale	–0.049	–0.191	–0.033	–0.149
	(–0.412)	(–0.629)	(–0.262)	(–0.473)
Twitter (log)	–0.002	–0.008	–0.012	–0.033
	(–0.112)	(–0.126)	(–0.502)	(–0.465)
Reddit (log)	–0.049**	–0.138**	–0.045*	–0.125*
	(–2.186)	(–2.219)	(–1.918)	(–1.883)
BTCTalk (log)	0.160***	0.426***	0.153***	0.408***
	(9.346)	(9.014)	(8.259)	(7.792)
Medium (log)	0.059	0.197	0.072	0.237*
	(1.251)	(1.646)	(1.566)	(1.986)
Year-month FE	Yes	Yes	Yes	Yes
Industry FE	Yes	Yes	Yes	Yes
Geographic region FE	Yes	Yes	Yes	Yes
Obs.	2,326	2,368	1,682	1,682
Pseudo-r^2 / Adj. r^2	0.198	0.284	0.199	0.268
$H_0 : \beta_1 = \beta_2$	Pr. > χ^2 = 0.497	Pr. > χ^2 = 0.317	Pr. > χ^2 = 0.475	Pr. > χ^2 = 0.481

Table 7. Code production, information environment, and fundraising (separated equilibrium). This table reports estimates of equation (5) in subsamples split by the information environment surrounding the token offering. *High (Low) information coverage* indicates that the startup ranks above (at or below) the sample median in information coverage, measured by the number of independent sources reporting fundraising details for the startup. *High (Low) information quality* indicates the analogous split using the information quality index following Lyandres et al. (2022), which combines the number of available sources with the average consistency of reported values across sources—see Appendix B for construction details. The dependent variable is *RaisedDummy* in columns (i), (iii), (v), and (vii), and *FundsRaised (log)* in columns (ii), (iv), (vi), and (viii). The variables of interest are *Code-washer*—an open-source startup with zero commits during the pre-TO window $[-365, -90]$ days—and *Code-producer*—an open-source startup with at least one commit during the same window; closed-source startups form the omitted reference category. We estimate Logit regressions for *RaisedDummy* and OLS regressions for *FundsRaised (log)*. Table 1 describes the remaining control variables. All specifications include year-month, industry, and geographic region fixed effects, and standard errors are clustered at the venture-fundraising completion month. t-statistics (z-statistics for Logit) are reported in parentheses. The symbols *, **, and *** denote significance at the 10%, 5%, and 1% levels, respectively.

	(i)	(ii)	(iii)	(iv)	(v)	(vi)	(vii)	(viii)
	High info coverage		**Low info coverage**		**High info quality**		**Low info quality**	
Variables of interest:								
β_1(*Code-producer*)	0.589***	1.102***	0.239	0.504	0.657***	1.197***	0.199	0.433
	(3.153)	(2.824)	(1.225)	(0.897)	(3.152)	(3.017)	(1.133)	(0.826)
β_2(*Code-washer*)	0.029	0.306	0.124	0.252	0.259	0.699*	0.094	0.183
	(0.130)	(0.683)	(0.671)	(0.478)	(1.469)	(1.812)	(0.573)	(0.378)
Variables of interest:								
Tokens for sale (%)	–0.745	–1.629**	0.133	0.279	–0.606	–1.507**	–0.363	–1.124
	(–1.410)	(–2.226)	(0.420)	(0.321)	(–1.374)	(–2.181)	(–1.096)	(–1.208)
Hardcap (log)	0.057	0.673***	–0.020	0.162	0.035	0.648***	–0.045	0.100
	(0.862)	(5.409)	(–0.307)	(0.890)	(0.615)	(5.852)	(–0.678)	(0.571)
Whitelist	–0.018	0.382	–0.347**	–0.971***	–0.034	0.371	–0.329**	–0.889**
	(–0.077)	(0.854)	(–2.571)	(–2.595)	(–0.128)	(0.740)	(–2.111)	(–2.230)
KYC	0.601**	1.478***	0.409**	1.166**	0.631**	1.531***	0.468**	1.345**
	(2.352)	(3.114)	(2.098)	(2.232)	(2.096)	(2.954)	(2.415)	(2.540)
White paper	0.419**	0.826***	–0.009	0.077	0.433**	0.832***	0.037	0.194
	(2.193)	(2.866)	(–0.059)	(0.188)	(2.114)	(2.653)	(0.254)	(0.497)
Team size (log)	0.352***	0.752***	0.381***	1.055***	0.495***	1.028***	0.327***	0.898***
	(4.559)	(3.610)	(3.089)	(2.960)	(5.117)	(4.145)	(3.042)	(2.943)
Presale	–0.245	–0.410	–0.149	–0.501	–0.135	–0.225	–0.090	–0.323
	(–1.109)	(–1.188)	(–0.877)	(–1.040)	(–0.744)	(–0.796)	(–0.613)	(–0.823)
Twitter (log)	–0.038	–0.076	0.008	0.043	–0.024	–0.043	0.009	0.016
	(–0.811)	(–1.006)	(0.310)	(0.529)	(–0.601)	(–0.648)	(0.373)	(0.202)
Reddit (log)	0.008	0.019	–0.062**	–0.154*	–0.015	–0.037	–0.054*	–0.137*
	(0.284)	(0.359)	(–2.072)	(–1.918)	(–0.436)	(–0.598)	(–1.940)	(–1.758)
BTCTalk (log)	0.154***	0.263***	0.066**	0.143	0.126***	0.228***	0.080***	0.190**
	(4.550)	(4.903)	(2.023)	(1.482)	(3.794)	(5.268)	(2.805)	(2.143)
Medium (log)	0.069	0.226*	0.016	0.025	0.102	0.241*	0.022	0.067
	(0.794)	(1.925)	(0.265)	(0.164)	(1.046)	(1.790)	(0.465)	(0.541)
Year-month FE	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Industry FE	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Geographic region FE	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Obs.	1,039	1,163	1,163	1,163	1,081	1,163	1,163	1,163
Pseudo-r^2 / Adj.r^2	0.208	0.231	0.094	0.070	0.220	0.244	0.104	0.081
$H_0 : \beta_1 = \beta_2$, Pr. $> \chi^2$=	0.035**	0.078*	0.484	0.601	0.128	0.204	0.592	0.667

Table 8. A quasi-natural experiment: rating platform algorithm shock. This table reports the results of a triple difference-in-differences (DID) design exploiting the launch of ICObench's algorithmic rating system ("Benchy") as an exogenous shock to the information environment. ICObench, the largest ICO rating platform, introduced an automated algorithm that evaluated startups on approximately 30 criteria, including the presence of a GitHub account, without assessing the quality of code commits. *Post (ICObench)* is an indicator for fundraising attempts occurring after January 2018, when ICObench became the dominant rating platform. *High coverage* indicates that a blockchain startup ranks above the median in data coverage across rating platforms, following Lyandres et al. (2022). The triple interaction *Code-washer* × *Post (ICObench)* × *High coverage* captures the differential change in the code-washing premium for startups listed on rating platforms after the algorithmic rating shock. Panel (a) reports the main DID estimates, where the dependent variable is *RaisedDummy* in column (i) and *FundsRaised (log)* in column (ii). Panel (b) reports robustness checks across placebo tests, alternative coverage definitions and subsample restrictions, summarizing the coefficient on the triple interaction term across each specification. Table 1 describes the remaining control variables. All specifications include year, industry, and geographic region fixed effects. Standard errors are clustered at the venture-fundraising completion month, and t-statistics are reported in parentheses. The symbols *, **, and *** denote significance at the 10%, 5%, and 1% levels, respectively.

Panel (a): Triple difference-in-differences

	(i) ***RaisedDummy***	(ii) ***FundsRaised (log)***
Variables of interest:		
Code-washer × *Post (ICObench)* × *High coverage*	0.286**	3.997***
	(2.470)	(2.724)
Code-producer × *Post (ICObench)* × *High coverage*	0.150	1.847
	(1.054)	(1.018)
Lower-order interactions:		
Code-washer × *Post (ICObench)*	–0.143	–2.363*
	(–1.291)	(–1.682)
Code-washer × *High coverage*	–0.269**	–3.430**
	(–2.388)	(–2.443)
Post (ICObench) × *High coverage*	–0.063	–0.979
	(–1.040)	(–1.180)
Code-washer	0.146	2.175*
	(1.381)	(1.652)
Code-producer × *Post (ICObench)*	–0.023	–1.176
	(–0.164)	(–0.677)
Code-producer × *High coverage*	–0.117	–1.109
	(–0.859)	(–0.650)
Code-producer	0.049	1.240
	(0.377)	(0.753)
Controls	Yes	Yes
Year & Industry & Region FE	Yes	Yes
Obs.	2,326	2,326
R^2	0.394	0.434

Panel (b): Robustness checks on the triple-interaction coefficient

Specification	**CW × Post × Cov**	***t*-statistic**	**N**
Placebo (fake shock dates within pre-treatment window):			
Fake shock = Sep 2017	–0.111	(–0.91)	308
Fake shock = Oct 2017	–0.232	(–1.39)	308
Fake shock = Nov 2017	0.107	(0.48)	308
Alternative coverage definitions:			
Top tercile	0.195**	(2.29)	2,326
Top quartile	0.151**	(2.11)	2,326
Continuous coverage	0.145***	(2.69)	2,326
Subsample robustness:			
Ethereum projects only	0.223*	(1.76)	2,029
2017–2019 sample	0.286**	(2.47)	2,326
DV = FundsRaised (log)	3.997***	(2.72)	2,326

Table 9. Code production and startup long-term performance. This table reports estimates of equation (7), capturing the relation between code-production and post-fundraising performance along two dimensions. Panel (a) examines financial performance over the first 180 days following exchange listing: the dependent variable is *Returns (log)* in column (i)—the logarithm of cumulative returns; *Volume (log)* in column (ii)—the logarithm of one plus the mean daily trading-volume rate; *Volatility (log)* in column (iii)—the logarithm of return volatility; and *Illiquidity (log)* in column (iv)—the logarithm of the Amihud illiquidity measure. Panel (b) examines post-fundraising development performance over the 180 days after the fundraising stage: the dependent variable is *Commits (log)* in column (v); *Pulls (log)* in column (vi); and *Contributors (log)* in column (vii)—each defined as the natural logarithm of one plus the cumulative count of commits, pull requests, and distinct GitHub contributors, respectively. *Code-washer* is an open-source startup with zero commits during the pre-TO window $[-365, -90]$ days relative to the token-offering start date; *Code-producer* is an open-source startup with at least one commit during the same window. Columns (i)–(iv) use the full sample with closed-source startups as the omitted reference category and report the Wald test for $H_0 : \beta_1 = \beta_2$ at the foot of the table. Columns (v)–(vii) restrict the sample to open-source startups with code-washers as the omitted reference. Table 1 describes the remaining control variables. All specifications include year-month, industry, and geographic region fixed effects. Standard errors are clustered at the venture-fundraising completion month, and t-statistics are reported in parentheses. The symbols *, **, and *** denote significance at the 10%, 5%, and 1% levels, respectively.

	Panel (a): Financial performance				**Panel (b): Development performance**		
	(i) ***Returns***	(ii) ***Volume***	(iii) ***Volatility***	(iv) ***Illiquidity***	(v) ***Commits***	(vi) ***Pulls***	(vii) ***Contributors***
Variables of interest:							
β_1(*Code-producer*)	0.352*	0.755**	–0.024**	0.000	3.328***	0.622***	0.486***
	(1.722)	(2.674)	(–1.966)	(0.161)	(34.379)	(7.558)	(6.829)
β_2(*Code-washer*)	0.003	0.245	–0.005	0.002	—	—	—
	(0.016)	(1.289)	(–0.419)	(1.171)	—	—	—
Controls:							
Tokens for sale (%)	0.037	–0.453	0.031**	0.006**	–0.984***	–0.242	–0.153
	(0.156)	(–1.085)	(2.070)	(2.462)	(–7.711)	(–1.488)	(–1.021)
Hardcap (log)	–0.020	0.349***	–0.009***	0.000	–0.015	–0.003	–0.004
	(–0.452)	(3.144)	(–2.597)	(0.589)	(–0.307)	(–0.134)	(–0.169)
Whitelist	–0.063	0.798***	–0.016*	0.002	0.533***	0.242***	0.209***
	(–0.487)	(3.412)	(–1.939)	(1.338)	(4.007)	(3.178)	(3.171)
KYC	0.018	0.171	–0.024***	–0.002	0.551***	0.330***	0.309***
	(0.094)	(0.564)	(–2.779)	(–1.341)	(3.368)	(5.074)	(4.820)
White paper	–0.004	0.274	–0.005	–0.001	0.499***	0.133*	0.075
	(–0.033)	(1.208)	(–0.559)	(–1.212)	(4.368)	(2.018)	(1.337)
Team size (log)	–0.119	0.101	–0.010	–0.002*	0.090	–0.049	–0.034
	(–1.216)	(0.982)	(–1.523)	(–1.822)	(1.026)	(–1.308)	(–0.978)
Presale	–0.157	–0.202	–0.004	0.000	–0.243**	–0.152**	–0.142**
	(–1.080)	(–0.703)	(–0.440)	(0.037)	(–2.183)	(–2.484)	(–2.634)
Twitter (log)	0.037*	0.009	–0.001	–0.000	–0.003	0.052***	0.041***
	(1.766)	(0.207)	(–0.873)	(–0.341)	(–0.143)	(3.469)	(3.264)
Reddit (log)	0.011	–0.028	0.003	0.000	0.032	0.014	0.006
	(0.283)	(–0.535)	(1.386)	(1.034)	(1.346)	(1.016)	(0.585)
BTCTalk (log)	–0.006	–0.032	0.001	0.000	–0.024	–0.038***	–0.030**
	(–0.209)	(–1.097)	(0.829)	(0.602)	(–1.083)	(–2.794)	(–2.330)
Medium (log)	–0.007	0.074	–0.004*	–0.000*	0.179***	0.083**	0.075**
	(–0.179)	(0.930)	(–1.763)	(–1.697)	(5.284)	(2.314)	(2.346)
Year-month FE	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Industry FE	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Geographic region FE	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Sample	Full	Full	Full	Full	OS	OS	OS
Obs.	599	599	599	598	1,526	1,526	1,526
Adj. r^2	0.164	0.222	0.175	0.068	0.472	0.166	0.152
$H_0 : \beta_1 = \beta_2$ Pr. > χ^2 =	0.017**	0.118	0.019**	0.096*	—	—	—

Table 10. Code production and post-fundraising financial misconduct. This table reports estimates of equation (7), capturing the association between code-production status 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, collected from deadcoins.com; estimated by Poisson. The dependent variable in column (ii) is *ScamDummy*—an indicator equal to one if the startup is flagged in at least one scam event post-fundraising; estimated by Logit. *Code-washer* is an open-source startup with zero commits during the pre-TO window $[-365, -90]$ days relative to the token-offering start date; *Code-producer* is an open-source startup with at least one commit during the same window; closed-source startups form the omitted reference category. Both columns use the full sample and report the Wald test for $H_0 : \beta_1 = \beta_2$ at the foot of the table. Table 1 describes the remaining control variables. All specifications include year-month, industry, and geographic region fixed effects. Standard errors are clustered at the venture-fundraising completion month, and z-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:		
β_1(*Code-producer*)	0.209	0.227
	(1.081)	(1.173)
β_2(*Code-washer*)	0.570***	0.706***
	(3.021)	(4.430)
Controls:		
Tokens for sale (%)	–0.315	–0.035
	(–0.777)	(–0.084)
Hardcap (log)	0.251***	0.174*
	(3.571)	(1.852)
Whitelist	–0.052	–0.058
	(–0.272)	(–0.275)
KYC	0.519**	0.552*
	(2.287)	(1.938)
White paper	–0.245*	–0.169
	(–1.787)	(–0.954)
Team size (log)	0.107	0.142
	(1.252)	(1.502)
Presale	0.042	0.099
	(0.153)	(0.317)
Twitter (log)	–0.036	–0.049
	(–1.060)	(–1.378)
Reddit (log)	–0.025	–0.068
	(–0.605)	(–1.519)
BTCTalk (log)	0.065	0.071
	(1.480)	(1.622)
Medium (log)	0.061	0.094
	(1.017)	(1.582)
Year-month FE	Yes	Yes
Industry FE	Yes	Yes
Geographic region FE	Yes	Yes
Obs.	2,326	2,032
Pseudo-r^2	0.166	0.121
$H_0 : \beta_1 = \beta_2$	Pr. > χ^2 = 0.007***	Pr. > χ^2 = 0.005***

Appendix A: Additional Figures and Tables

Figure A1. An example of a code-washer startup. This figure illustrates a scam case involving a token offering 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

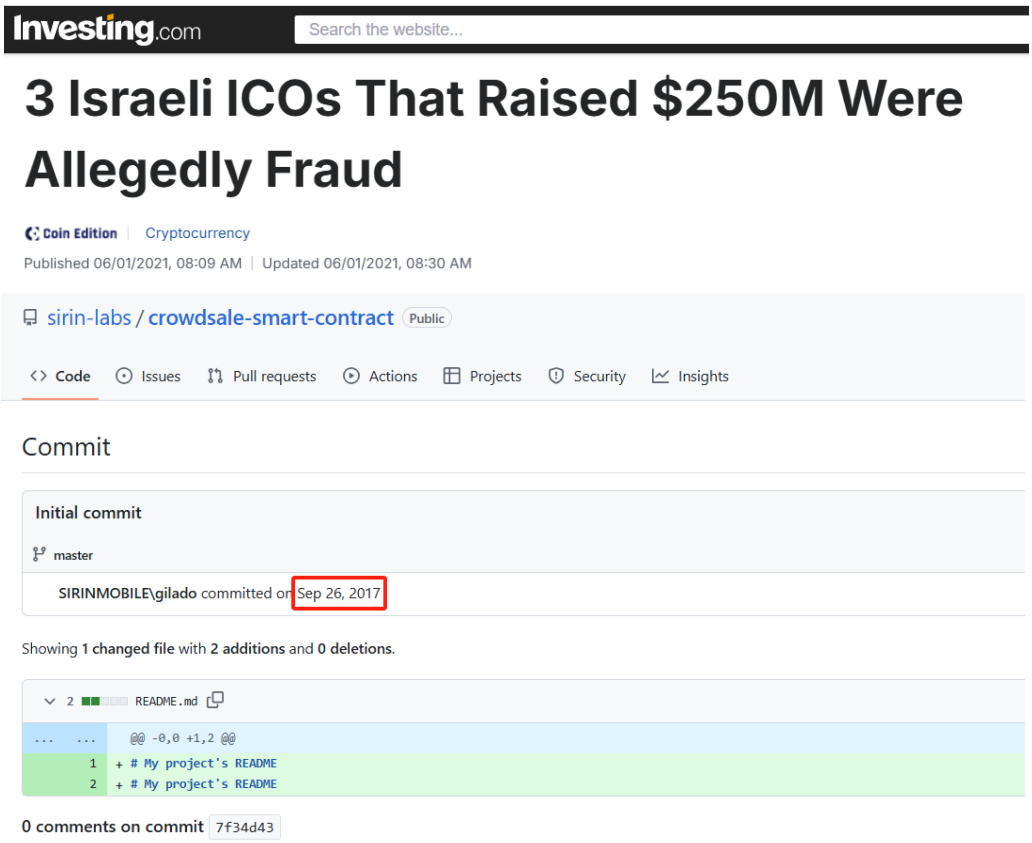


Figure A2. A token offering scam case. This figure illustrates the GitHub repository page of a blockchain startup named "Bitconnect". The GitHub repository shows that BitConnect created one repository in 2016 and had nearly zero code production (only seven commits) right before its main fundraising event. 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.

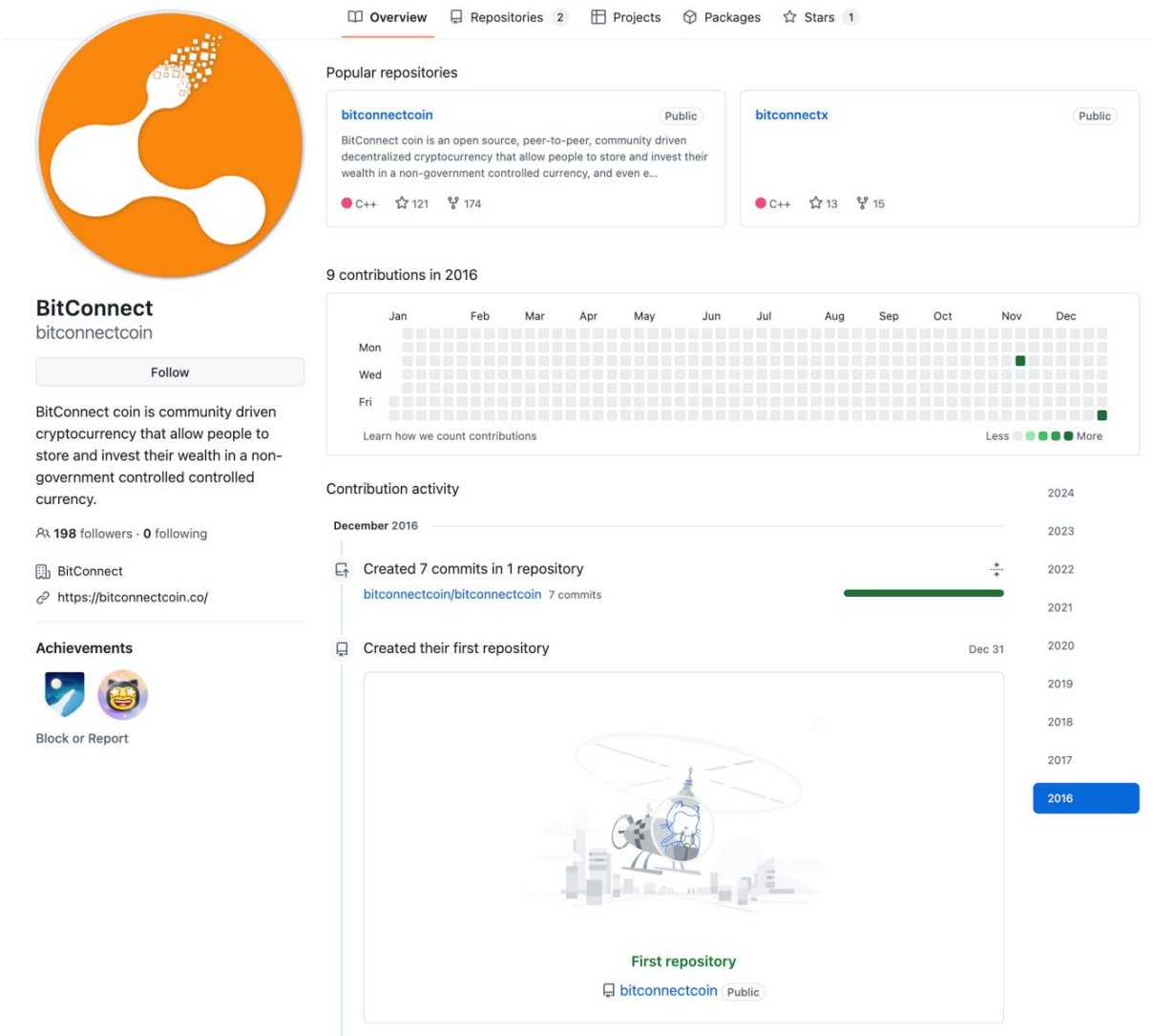
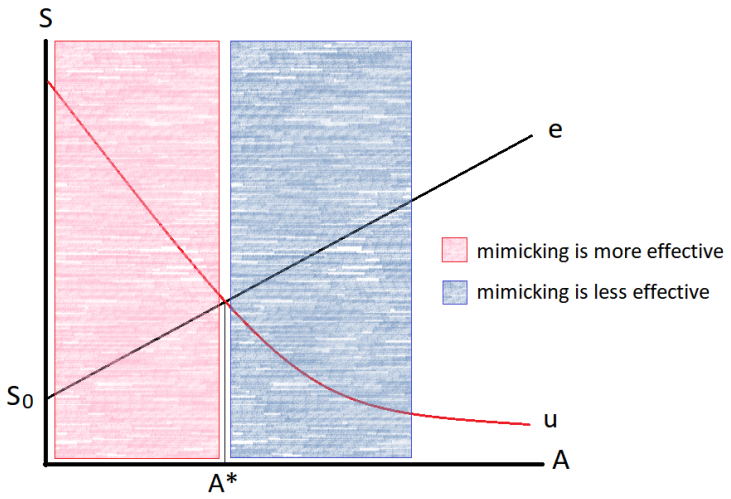


Figure A3. Multiple equilibria and investors' attention. This figure illustrates the economic framework of the interplay between code production and its function as a credible signal of venture quality under different market conditions, accounting for investors' ability to interpret the signal accurately.



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.

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

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

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

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

Variable	Description
(a) Fundraising outcomes	
RaisedDummy	A dummy variable equal to 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.
(b) Code production	
OpenSource	An indicator variable equal to one if the blockchain startup has a public GitHub repository before the fundraising stage, and zero otherwise.
Code-washer	An indicator variable equal to one if the startup is open-source and has zero commits during the pre-TO window $[-365, -90]$ days relative to the token-offering start date, and zero otherwise.
Code-producer	An indicator variable equal to one if the startup is open-source and has at least one commit during the pre-TO window $[-365, -90]$ days, and zero otherwise.
Closed-source	An indicator variable equal to one if the startup has no public GitHub repository before the fundraising stage, and zero otherwise. Closed-source startups serve as the omitted reference category in regressions including both *Code-washer* and *Code-producer*.
Code Concentration	The share of cumulative commits made during the TO window relative to all commits as of the TO end.
RepoAbandoned	An indicator variable equal to one if the startup makes zero commits in the 180 days following the TO end, and zero otherwise.
RepoDeleted	An indicator variable equal to one if the GitHub URL recorded for the startup in our 2020 source index returns HTTP status 404 when re-queried on April 2026, indicating that the public repository has been removed from GitHub.
Debug Ratio	The share of a startup's pre-TO commits whose commit message contains debugging-related terms ("fix," "bug," "error," or "patch").
Issues (log)	The natural logarithm of one plus the cumulative number of GitHub issues opened by the startup.
Pulls (log)	The natural logarithm of one plus the cumulative number of GitHub pull requests submitted to the startup's repository.
(c) Validation tests	
Word count	The count of words per page in the startup's whitepaper.
Tech word count	The count of technology-related terms in the whitepaper, identified using natural-language-processing techniques.
Image count	The count of figures in the whitepaper.
Any member disclosed	An indicator variable equal to one if at least one team member explicitly identifies the ICO startup as a current or past affiliation on their LinkedIn profile, and zero otherwise.
Share of disclosed	The fraction of team members who explicitly identify the ICO startup as a current or past affiliation on LinkedIn.
Any PhD	An indicator variable equal to one if at least one team member holds a PhD degree, constructed from manually collected LinkedIn profiles.
PhD share	The fraction of team members holding a PhD degree, constructed from LinkedIn profiles.
(d) Long-term performance (post-listing, 180-day window)	
Returns (log)	The logarithm of cumulative returns over the first 180 days from listing.
Volume (log)	The logarithm of one plus the mean daily trading-volume rate 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 the Amihud illiquidity measure calculated over the first 180 days from listing.
Commits (log)	The logarithm of one plus cumulative GitHub commits over the 180 days after the fundraising stage.
Contributors (log)	The logarithm of one plus the cumulative number of distinct GitHub contributors over the 180 days after the fundraising stage.
(e) Financial misconduct	
Number of Scams	The total number of scam events involving the blockchain startup post-fundraising, collected from deadcoins.com.
ScamDummy	An indicator variable equal to one if the startup is flagged in at least one scam event post-fundraising, and zero otherwise.

Continued on next page

(Table A1 continued)

Variable	Description
(f) Events	
Post (SEC)	An indicator variable equal to one for token offerings completed after the SEC's enforcement actions on November 16, 2018, and zero otherwise.
Post (ICObench)	An indicator variable equal to one for fundraising attempts occurring after January 2018, when ICObench's algorithmic rating system became the dominant ICO rating platform, and zero otherwise.
(g) Information environment	
High information coverage	An indicator variable equal to one if the startup ranks above the sample median in information coverage—measured by the number of independent sources reporting fundraising details for the startup—and zero otherwise. Also denoted *High coverage* in the triple difference-in-differences specification.
High information quality	An indicator variable equal to one if the startup ranks above the sample median in the information-quality index of Lyandres et al. (2022), which combines the number of available sources with the average consistency of reported values across sources, and zero otherwise. See Appendix B for construction details.
(h) Startup characteristics (controls)	
Tokens for sale (%)	The ratio of tokens offered for sale to total token supply.
Hardcap (log)	The logarithm of the maximum amount allowed to be raised in the token offering.
Whitelist	A dummy variable equal to one if the project offers a whitelist to early investors, and zero otherwise.
KYC	A dummy variable equal to one if the project complies with "know your customer" requirements, and zero otherwise.
White paper	A dummy variable equal to one if the project disclosed a whitepaper, and zero otherwise.
Team size (log)	The logarithm of one plus the total number of team members collected from LinkedIn.
Presale	A dummy variable equal to one if the project attempted a presale (i.e., by angel investors, venture capital, or other early seed investors) before the fundraising stage, and zero otherwise.
Twitter (log)	The logarithm of one plus the total number of tweets posted on the project's official account before the fundraising stage.
Reddit (log)	The logarithm of one plus the total number of Reddit discussions before the fundraising stage.
BTCTalk (log)	The logarithm of one plus the total number of Bitcointalk posts before the fundraising stage.
Medium (log)	The logarithm of one plus the total number of Medium articles before the fundraising stage.

Table A2. Robustness check: balance tests on team size and quality. This table reports balance tests of team characteristics across code-production status to assess whether the main results are confounded by pre-existing team-quality differences between code-producers and code-washers. The dependent variable in column (i) is *Team size (log)*—the natural logarithm of one plus the number of disclosed team members at the blockchain startup. The dependent variable in column (ii) is *Any PhD*—an indicator variable equal to one if at least one team member holds a PhD degree, constructed from manually collected LinkedIn profiles. The dependent variable in column (iii) is *PhD share*—the fraction of team members holding a PhD degree, constructed from the same LinkedIn profiles. The variables of interest are as follows: *Code-producer* is a startup in the top half of cumulative commits prior to the fundraising stage, while *Code-washer* is a startup in the bottom half, based on a median cutoff of cumulative commits during the pre-fundraising window $[-365, -90]$ days relative to the ICO starting date. We report the Chi-Squared test results for the hypothesis $H_0 : \beta_1 = \beta_2$. Column (i) is estimated using OLS on the full main sample; column (ii) is estimated using a Logit model on the LinkedIn subsample; column (iii) is estimated using a Tobit model with two-sided censoring at 0 and 1 on the LinkedIn subsample. Table 1 describes the remaining control variables. All specifications include year-month, industry, and geographic region fixed effects. The standard errors are clustered at the venture-fundraising completion date, and t-statistics (z-statistics in column (ii)) are reported in parentheses. The symbols *, **, and *** denote significance at the 10%, 5%, and 1% levels, respectively.

	(i) ***Team size (log)***	(ii) ***Any PhD***	(iii) ***PhD share***
Variables of interest:			
β_1(*Code-producer*)	0.189***	0.361*	0.029
	(5.114)	(1.777)	(0.817)
β_2(*Code-washer*)	0.205***	0.424**	0.028
	(4.067)	(2.497)	(1.191)
Controls:			
Tokens for sale (%)	–0.042	0.350	0.047
	(–0.567)	(0.860)	(0.997)
Hardcap (log)	0.064***	0.038	0.006
	(3.799)	(0.417)	(0.601)
Whitelist	0.092**	–0.025	0.006
	(2.450)	(–0.157)	(0.339)
KYC	0.278***	0.621***	0.066***
	(5.731)	(3.518)	(3.368)
White paper	0.020	–0.270	–0.029
	(0.836)	(–0.978)	(–0.834)
Presale	0.140***	0.057	0.011
	(5.762)	(0.229)	(0.365)
Twitter (log)	0.015*	0.043	0.006
	(1.724)	(0.927)	(1.161)
Reddit (log)	0.019***	–0.085	–0.011
	(3.336)	(–1.314)	(–1.605)
BTCTalk (log)	0.039***	0.039	0.000
	(5.228)	(0.918)	(0.098)
Medium (log)	0.061***	0.102**	0.011*
	(5.020)	(2.020)	(1.881)
Year-month FE	Yes	Yes	Yes
Industry FE	Yes	Yes	Yes
Geographic region FE	Yes	Yes	Yes
Model	OLS	Logit	Tobit
Obs.	2,326	552	565
Adj./Pseudo r^2	0.191	0.067	0.134
$H_0 : \beta_1 = \beta_2$	Pr. > χ^2 = 0.632	Pr. > χ^2 = 0.717	Pr. > χ^2 = 0.956

Table A3. Summary statistics for the propensity matching sample. This table reports the balance between code-washers and matched closed-source firms after propensity score matching (PSM). We match each code-washer to its closest closed-source firm using 1:1 nearest-neighbor matching with a caliper of 0.001 on the propensity score. The control group is 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.

	Code-washer Mean	**Matched control Mean**	**Difference (t-value)**
Tokens for sale (%)	0.553	0.571	−0.018
			(1.20)
Hardcap (log)	16.698	16.720	−0.022
			(0.27)
Whitelist	0.493	0.475	0.018
			(−0.56)
KYC	0.738	0.697	0.041
			(−1.42)
White paper	0.504	0.475	0.029
			(−0.92)
Team size (log)	2.258	2.176	0.082
			(−1.71)
Presale	0.677	0.696	−0.020
			(0.67)
Twitter (log)	2.752	2.671	0.081
			(−0.57)
Reddit (log)	1.325	1.148	0.177
			(−1.13)
BTCTalk (log)	3.382	3.192	0.190
			(−1.13)
Medium (log)	0.607	0.496	0.111
			(−1.48)
Observations	593	357	

Table A4. Open-source disclosure and fundraising success. This table reports the association between blockchain-startup code disclosure and fundraising success. The dependent variable is *RaisedDummy* in columns (i) and (ii)—an indicator equal to one if the startup raised any funds by the end of the fundraising stage—and *FundsRaised (log)* in columns (iii) and (iv)—the natural logarithm of total funds raised. The variables of interest are *Open Source*, an indicator equal to one if the startup has a public GitHub repository before the fundraising stage, and *Code Production (log)*, the natural logarithm of one plus the cumulative number of commits before the fundraising stage. Columns (i) and (ii) are estimated by Logit; columns (iii) and (iv) by OLS. Table 1 describes the remaining control variables. All specifications include year-month, industry, and geographic region fixed effects. Standard errors are clustered at the venture-fundraising completion month, and t-statistics (z-statistics for Logit) 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 A5. Robustness: Pool equilibrium under quartile-based proxies. This table reports estimates of equation (5) using quartile-based alternatives to our baseline code-washing and code-producing proxies. *Code-washer (Q1)* is an open-source startup in the bottom quartile of cumulative commits generated before the fundraising stage; *Code-producer (Q4)* is an open-source startup in the top quartile of cumulative commits generated before the fundraising stage; closed-source startups and open-source startups in the interquartile range form the omitted reference category. The dependent variable is *RaisedDummy* in column (i)—an indicator equal to one if the startup raised any funds by the end of the fundraising stage—and *FundsRaised (log)* in column (ii)—the natural logarithm of total funds raised. Column (i) is estimated by Logit; column (ii) by OLS. Table 1 describes the remaining control variables. All specifications include year-month, industry, and geographic region fixed effects. Standard errors are clustered at the venture-fundraising completion month, 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 (Q4)*)	0.539**	1.612***
	(2.458)	(3.213)
β_2(*Code-washer (Q1)*)	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. > χ^2 = 0.497	Pr. > χ^2 = 0.317

Table A6. Robustness: Separated equilibrium under quartile-based proxies. This table reports estimates of equation (5) using quartile-based alternatives to our baseline code-washing and code-producing proxies, in subsamples split by the information environment surrounding the token offering. *Code-washer (Q1)* is an open-source startup in the bottom quartile of cumulative commits generated before the fundraising stage; *Code-producer (Q4)* is an open-source startup in the top quartile of cumulative commits generated before the fundraising stage; closed-source startups and open-source startups in the interquartile range form the omitted reference category. *High information coverage* indicates that the startup ranks above the sample median in information coverage, measured by the number of independent sources reporting fundraising details for the startup. *High information quality* indicates the analogous split using the information quality index following Lyandres et al. (2022), which combines the number of available sources with the average consistency of reported values across sources—see Appendix B for construction details. The dependent variable is *RaisedDummy* in columns (i) and (iii) and *FundsRaised (log)* in columns (ii) and (iv). Logit regressions are estimated for *RaisedDummy* and OLS regressions for *FundsRaised (log)*. We also report the Chi-Squared test for $H_0 : \beta_1 = \beta_2$ at the foot of the table. Table 1 describes the remaining control variables. All specifications include year-month, industry, and geographic region fixed effects, and standard errors are clustered at the venture-fundraising completion month. t-statistics (z-statistics for Logit) 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 (Q4)*)	1.317***	1.748***	0.929**	1.490***
	(4.241)	(4.415)	(2.467)	(3.247)
β_2(*Code-washer (Q1)*)	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. > χ^2 = 0.006***	Pr. > χ^2 = 0.010**	Pr. > χ^2 = 0.187	Pr. > χ^2 = 0.071*

Appendix B: Deriving Measures of Information Quality

Information quality at the data-source Level

A persistent challenge in analyzing data on token offerings 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.[12] 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 4.2 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.

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