Análise de Repositórios Java Populares no GitHub

Lucas Cabral Soares
Lucas Hemétrio
Maria Eduarda Amaral Muniz

1. Introdução

Este relatório apresenta uma análise quantitativa da qualidade interna de sistemas Java populares disponíveis no GitHub, investigando como características como popularidade, maturidade, atividade e tamanho dos repositórios se relacionam com métricas importantes de qualidade de software, tais como acoplamento entre objetos (CBO), profundidade da árvore de herança (DIT) e falta de coesão entre métodos (LCOM). A análise abrange a sumarização desses dados através de medidas estatísticas de média, mediana e desvio padrão.

1.1. Hipóteses

Antes da análise dos dados, formulamos algumas hipóteses sobre os sistemas populares:

- **RQ 01:** Repositórios mais populares terão valores médios menores de CBO e LCOM, indicando melhor qualidade estrutural.
- **RQ 02:** Repositórios mais maduros apresentarão valores médios maiores de CBO e DIT, refletindo o aumento natural da complexidade com o tempo.
- **RQ 03:** Repositórios com mais releases terão valores médios menores de LCOM, indicando maior coesão dos métodos devido a constantes refatorações.
- **RQ 04:** Repositórios maiores, com mais linhas de código, terão valores médios mais altos de CBO, refletindo maior acoplamento devido à complexidade crescente do código.

2. Metodologia

Para responder às questões de pesquisa, utilizamos uma abordagem baseada na coleta, processamento e análise de dados de repositórios populares, com a linguagem principal sendo Java, do GitHub. A seleção dos repositórios foi feita considerando os 1000 projetos com maior número de estrelas, garantindo que

nossa análise se concentrasse nos sistemas mais reconhecidos e amplamente utilizados na plataforma.

A metodologia adotada pode ser dividida em três etapas principais: coleta dos dados, processamento das métricas e análise dos resultados.

2.1 Coleta dos Dados do GitHub

Os dados foram extraídos utilizando a GitHub GraphQL API, o que permitiu coletar informações detalhadas sobre cada repositório, incluindo idade, número de pull requests aceitos, releases, tempo desde a última atualização, linguagem primária e gestão de issues. Para garantir uma amostra representativa, utilizamos uma consulta paginada para coletar até 1000 repositórios de forma eficiente. Os dados foram armazenados em um arquivo JSON para facilitar o processamento posterior.

2.2 Coleta de dados do CK

Tendo os dados dos repositórios já selecionados, o próximo passo foi clonar esses repositórios para a máquina, realizar uma análise usando o CK e escrever os resultados em um arquivo .csv final com todas as métricas necessárias para responder a RQ 's.

2.3 Métricas

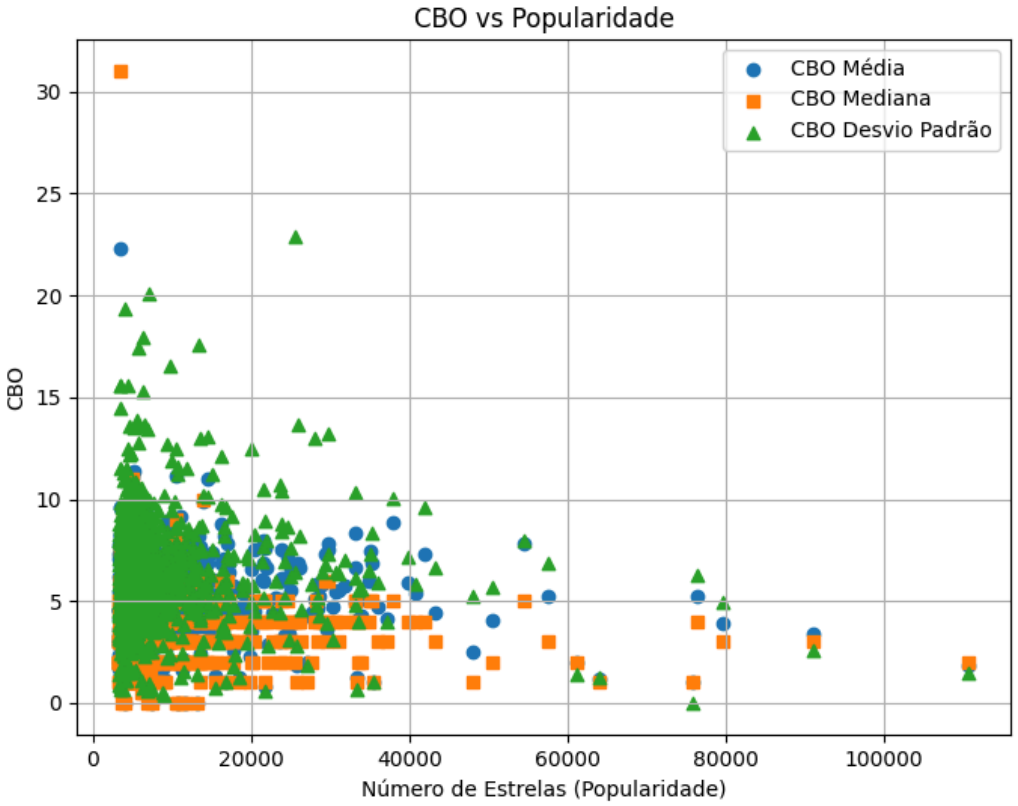
Após a coleta, os dados foram processados e organizados em um arquivo CSV, permitindo uma melhor estruturação para análise estatística. As métricas utilizadas para responder às questões de pesquisa foram as seguintes:

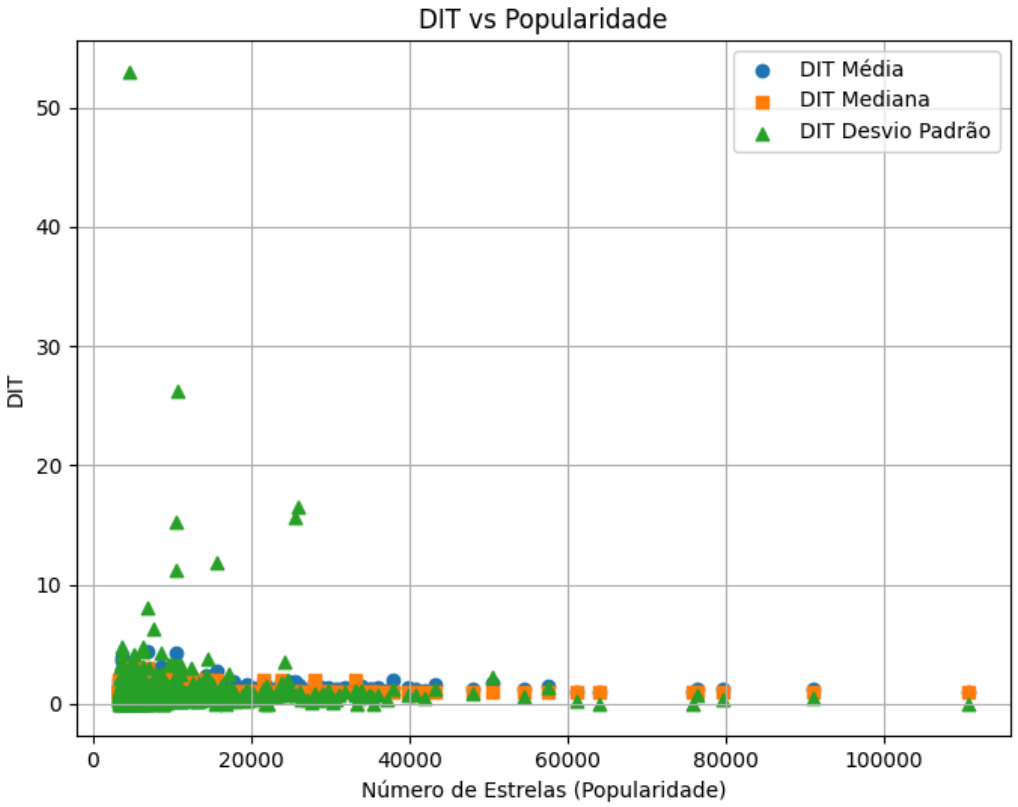
- **Popularidade:** número de estrelas - GitHub
- **Tamanho:** linhas de código (LOC) e linhas de comentários - CK
- **Atividade:** número de releases - GitHub
- **Maturidade:** idade (em anos) de cada repositório coletado - GitHub
- **CBO:** Coupling between objects - CK
- **DIT:** Depth Inheritance Tree - CK
- **LCOM:** Lack of Cohesion of Methods - CK

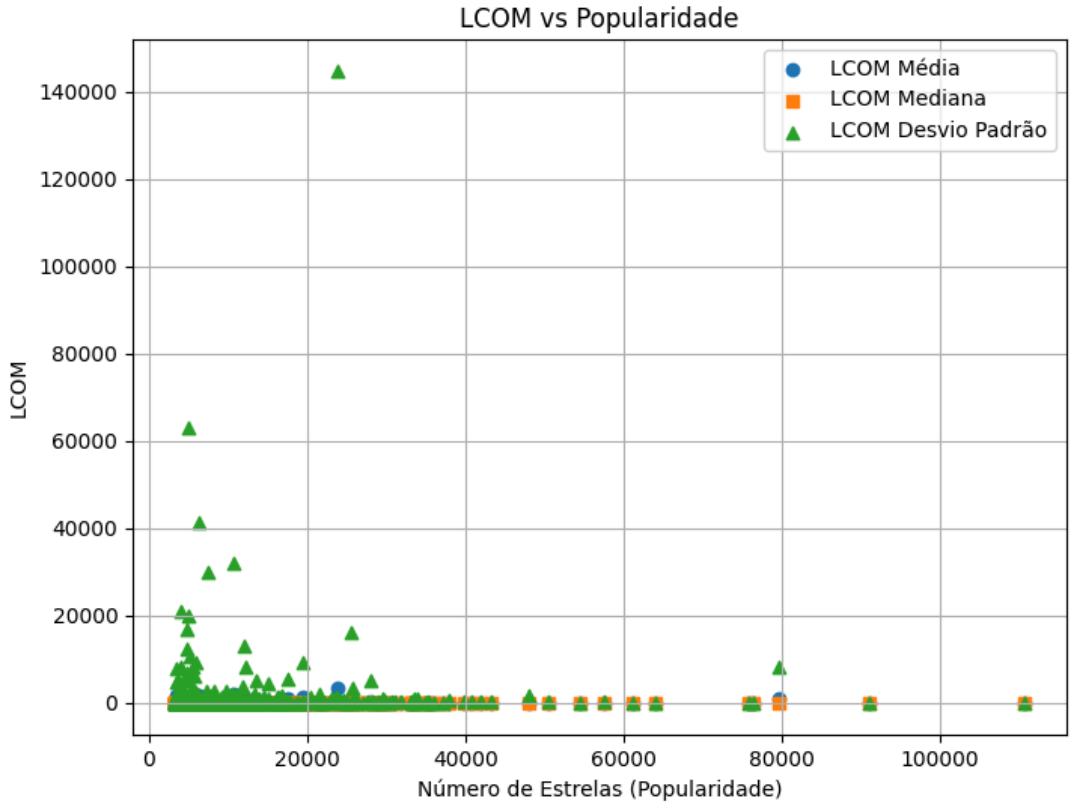
3. Resultados e Discussão

Todos os gráficos foram gerados com a biblioteca matplotlib e apresentam os valores da média, mediana e desvio padrão de cada uma das métricas de qualidade: CBO, DIT e LCOM.

3.1 RQ 01: Qual a relação entre a popularidade dos repositórios e as suas características de qualidade?


CBO vs Popularidade
CBO Média
CBO Mediana
CBO Desvio Padrão
CBO
Número de Estrelas (Popularidade)
0
5
10
15
20
25
30
0
20000
40000
60000
80000
100000

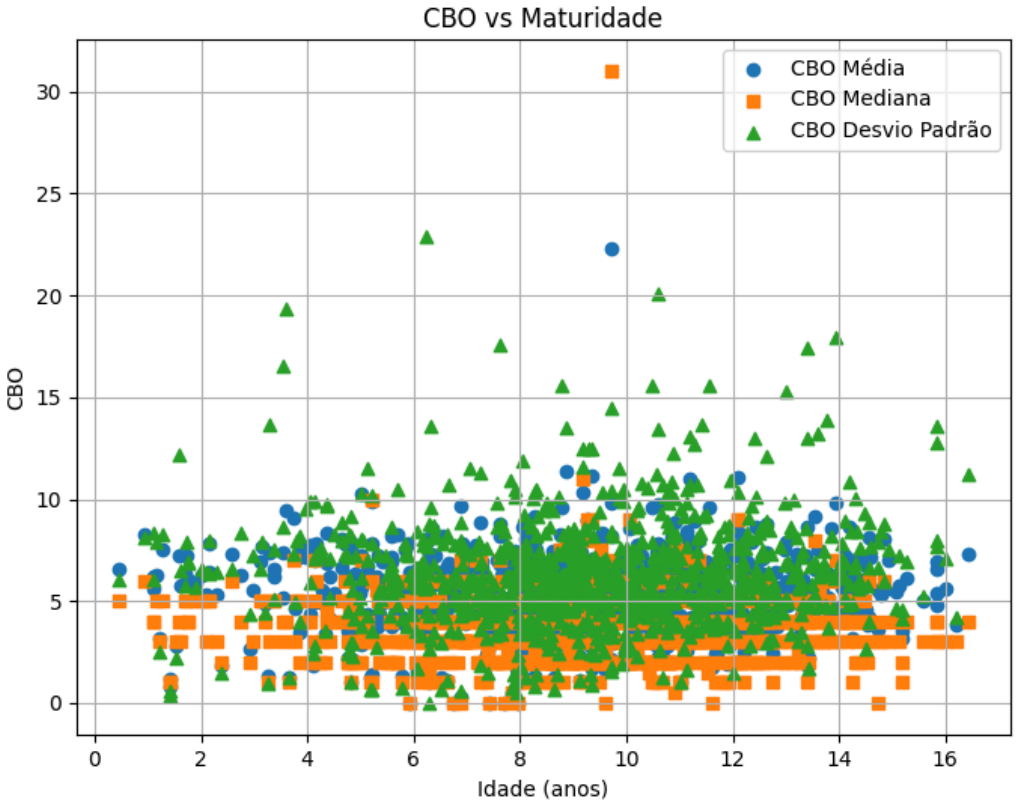

DIT vs Popularidade
DIT Média
DIT Mediana
DIT Desvio Padrão
DIT
Número de Estrelas (Popularidade)
0
10
20
30
40
50
0
20000
40000
60000
80000
100000

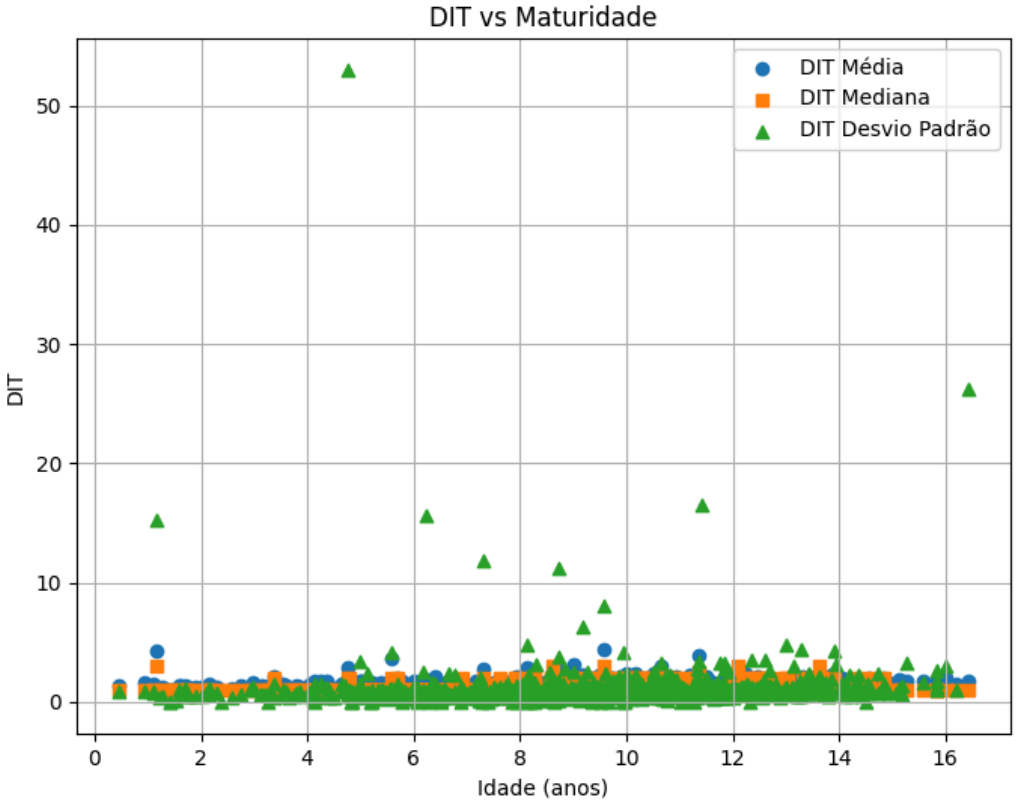


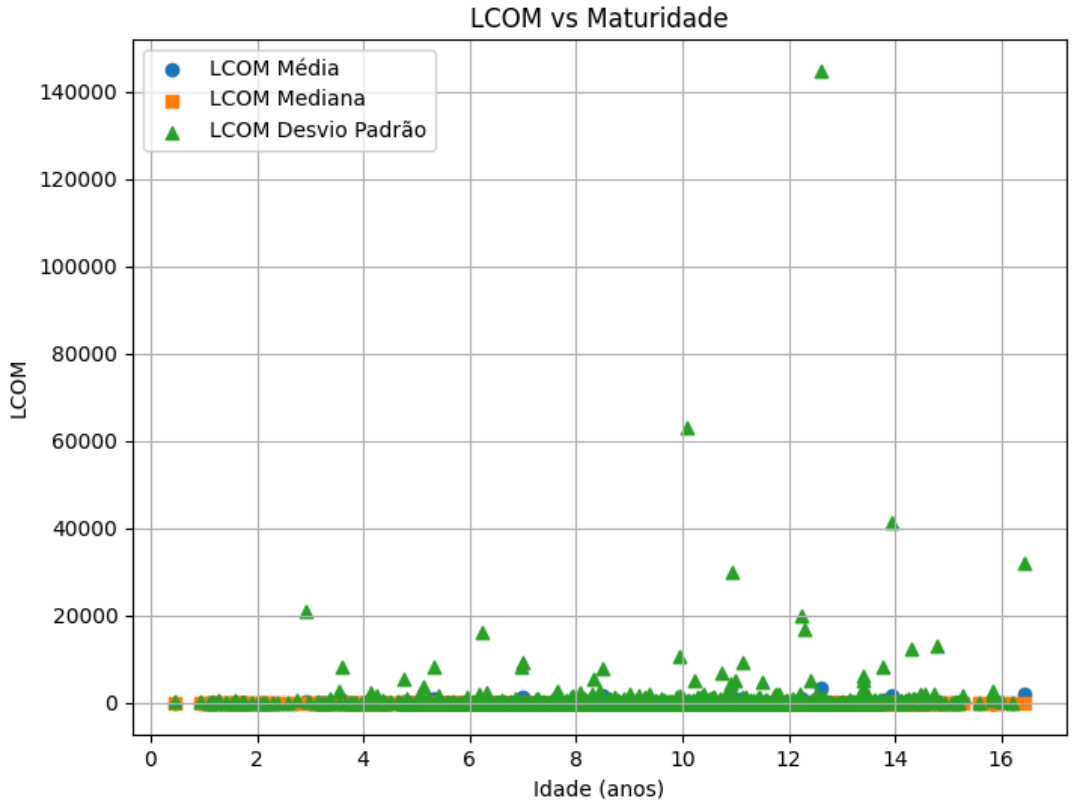
Nos gráficos de CBO Média vs Popularidade e LCOM Média vs Popularidade, percebe-se que, à medida que o número de estrelas aumenta, não há concentração de pontos em valores elevados de CBO ou LCOM; na maior parte, as médias de acoplamento e coesão permanecem relativamente baixas. Isso sugere uma tendência de repositórios mais populares manterem estruturas mais modulares. Já no gráfico de DIT Média vs Popularidade, as médias de herança (entre 1 e 2) se distribuem de forma mais uniforme, indicando que a profundidade de herança não varia tanto com a popularidade.

Em suma, os resultados apontam que repositórios com maior número de estrelas tendem a não ter CBO ou LCOM muito altos, o que vai ao encontro da hipótese de que projetos mais populares apresentariam melhor qualidade estrutural.

3.2 RQ 02: Qual a relação entre a maturidade dos repositórios e as suas características de qualidade ?


CBO vs Maturidade
CBO Média
CBO Mediana
CBO Desvio Padrão
CBO
Idade (anos)
0
5
10
15
20
25
30
0
2
4
6
8
10
12
14
16

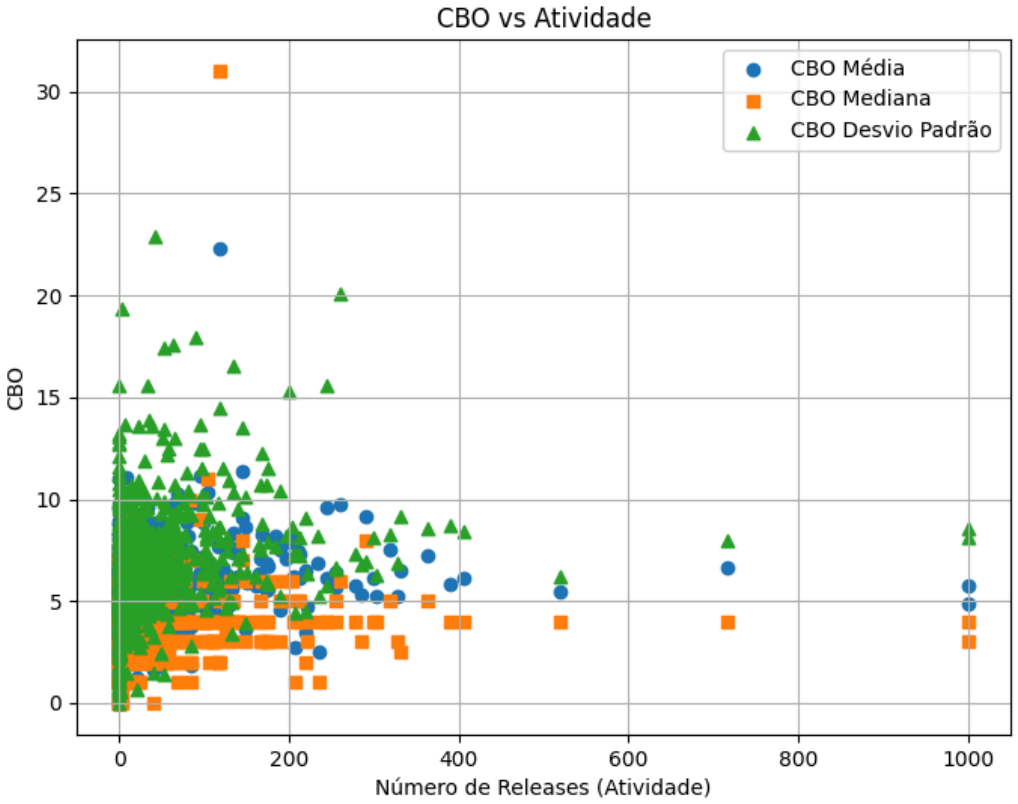

DIT vs Maturidade
DIT Média
DIT Mediana
DIT Desvio Padrão
DIT
Idade (anos)
0
10
20
30
40
50
0
2
4
6
8
10
12
14
16

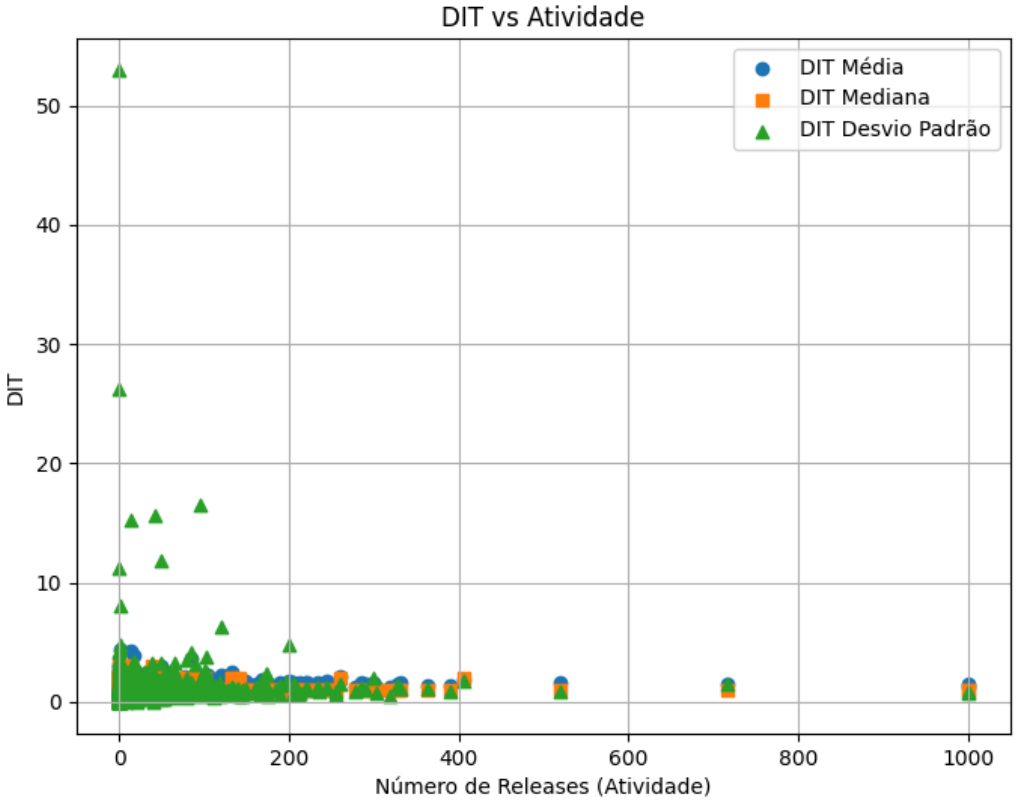


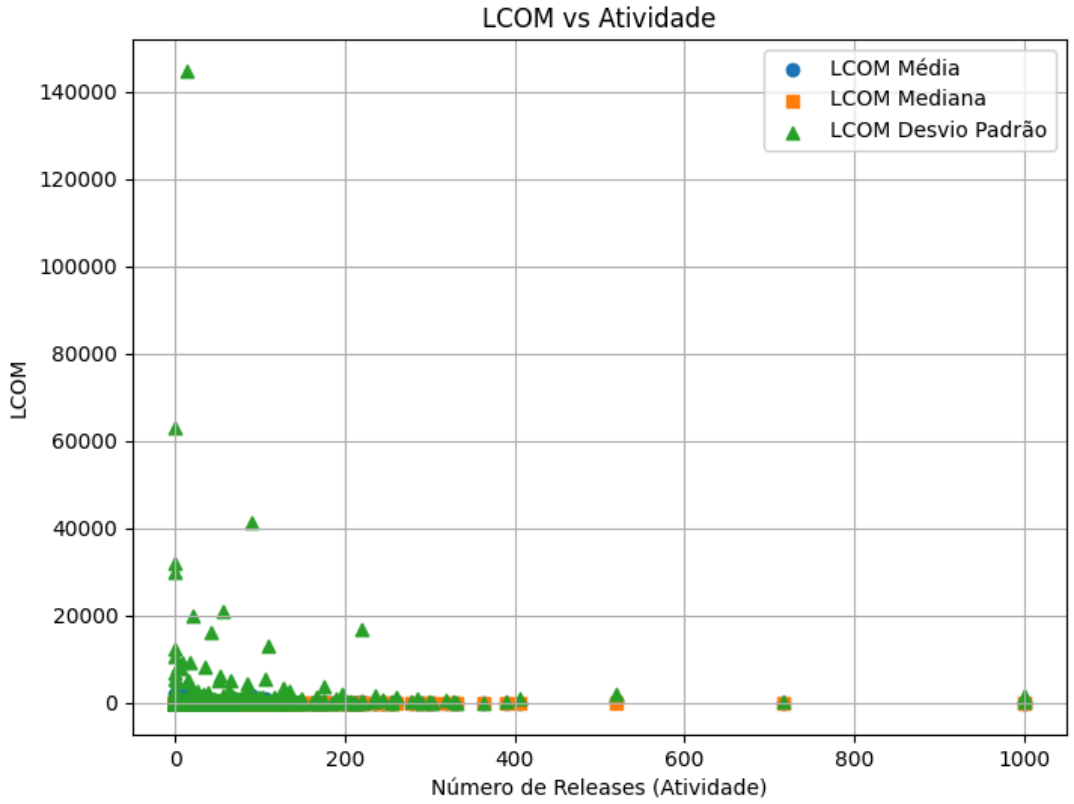
O gráfico de CBO Média vs Maturidade demonstra que não há uma tendência clara de aumento das métricas de acoplamento conforme os repositórios envelhecem. No gráfico de DIT Média vs Maturidade tem um leve aumento de profundidade à medida que a idade do repositório aumenta. No gráfico de LCOM Média vs Maturidade, observa-se que a coesão dos métodos se mantém predominantemente em níveis baixos.

A hipótese de que repositórios mais maduros apresentariam maiores valores de CBO e DIT é parcialmente confirmada, pois enquanto CBO não tem uma variação clara, DIT mostra um leve aumento.

3.3 RQ 03: Qual a relação entre a atividade dos repositórios e as suas características de qualidade?

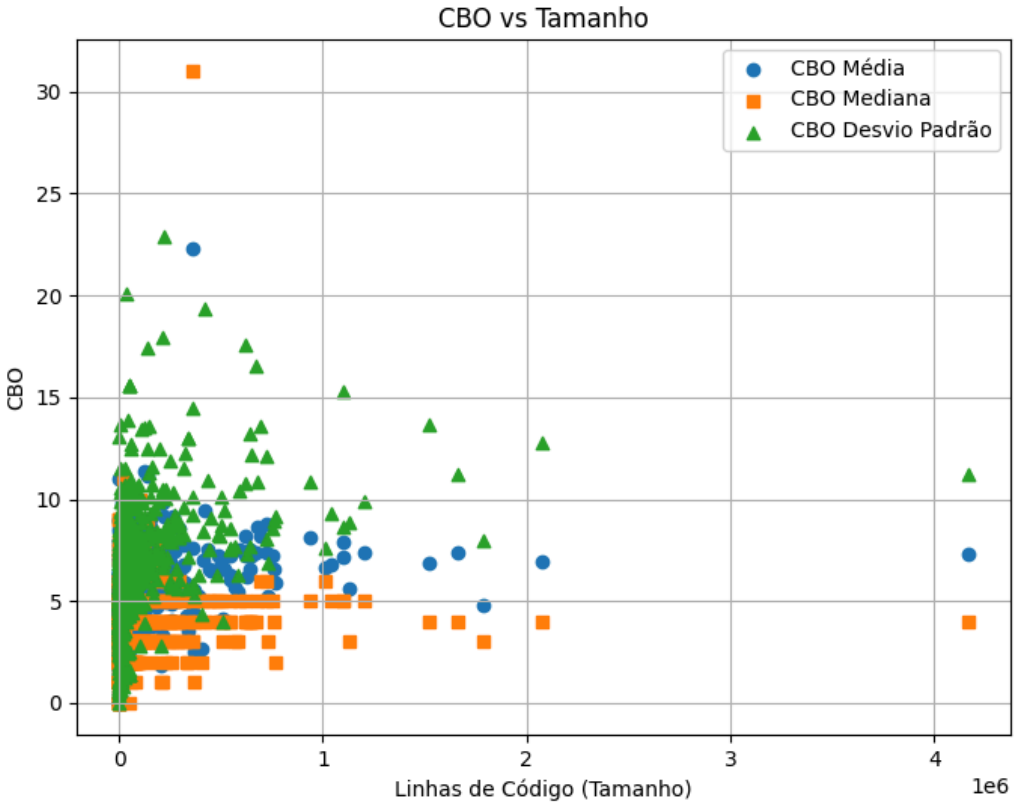

CBO vs Atividade
CBO Média
CBO Mediana
CBO Desvio Padrão
CBO
Número de Releases (Atividade)
0
5
10
15
20
25
30
0
200
400
600
800
1000

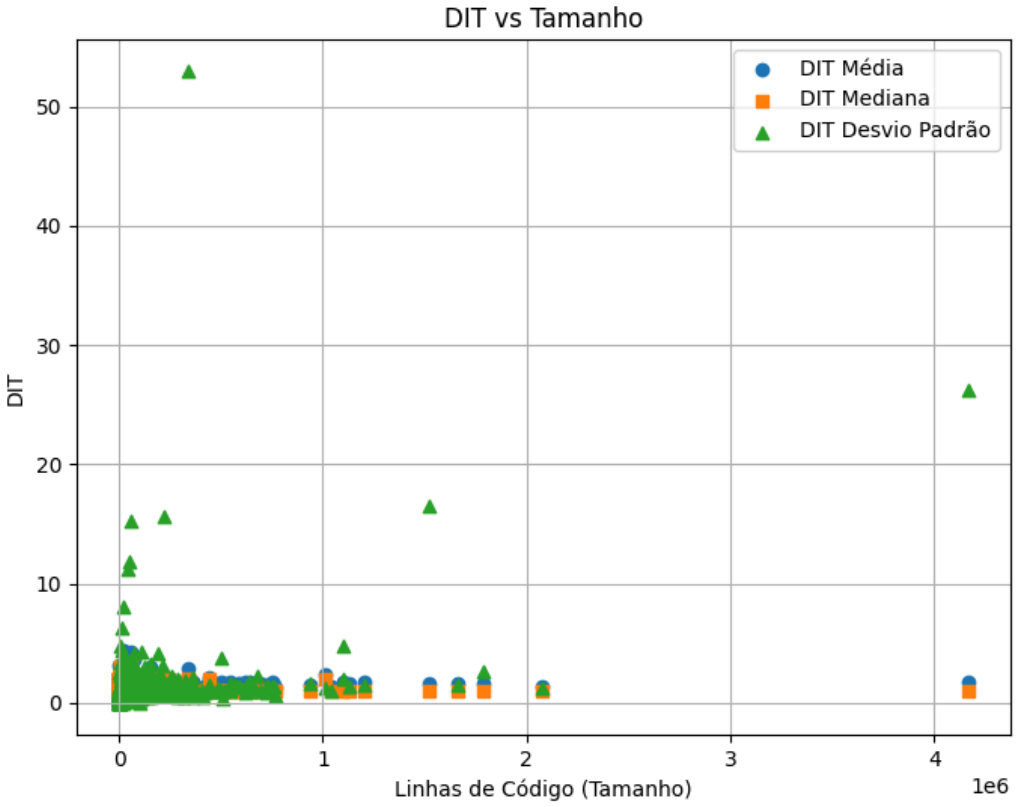

DIT vs Atividade
DIT Média
DIT Mediana
DIT Desvio Padrão
DIT
Número de Releases (Atividade)
0
10
20
30
40
50
0
200
400
600
800
1000

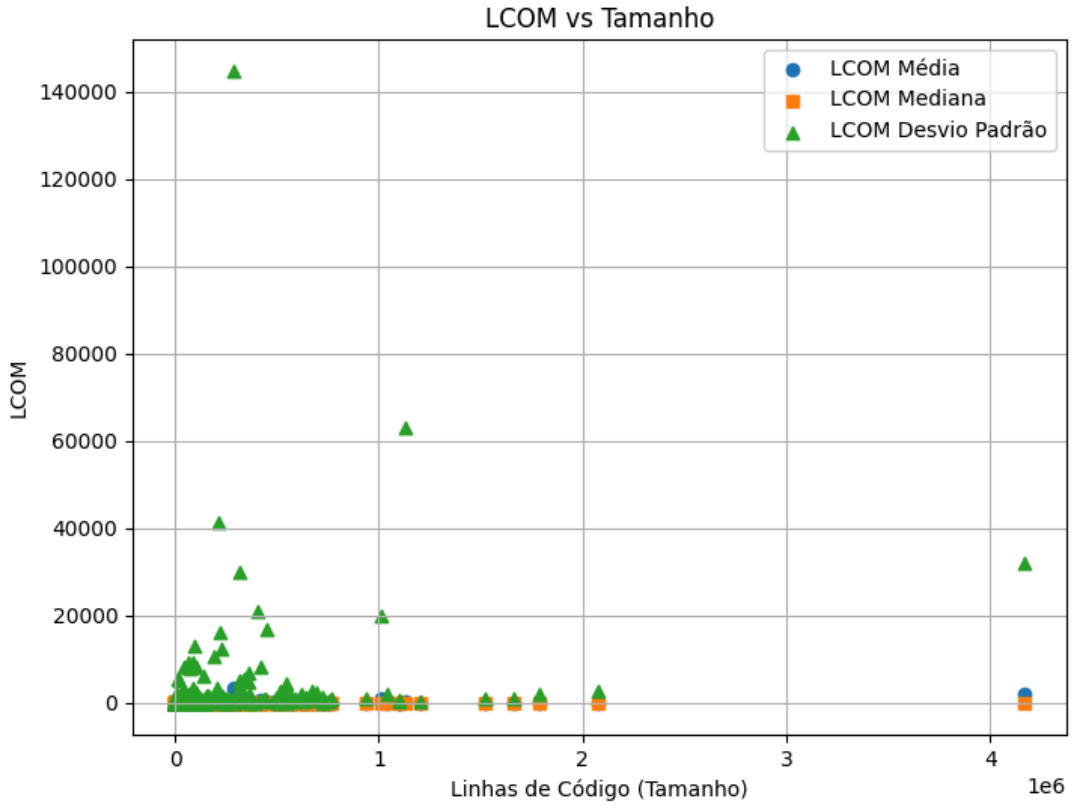


Os gráficos indicam que, no geral, a maioria dos repositórios com maior número de releases tende a manter valores de CBO e LCOM em faixas relativamente baixas. Em particular, a distribuição de LCOM sugere que repositórios com mais releases podem, de fato, apresentar métodos mais coesos (LCOM menor).
Portanto, a hipótese é parcialmente confirmada por falta de uma relação forte e linear. Muitos repositórios com poucas release também mantém um COM baixo.

3.4 RQ 04: Qual a relação entre o tamanho dos repositórios e as suas características de qualidade?


CBO vs Tamanho
CBO Média
CBO Mediana
CBO Desvio Padrão
CBO
Linhas de Código (Tamanho)
1e6


DIT vs Tamanho
DIT Média
DIT Mediana
DIT Desvio Padrão
DIT
Linhas de Código (Tamanho)
1e6



A maior parte dos repositórios concentra-se em até algumas centenas de milhares de linhas de código, com CBO variando principalmente entre 0 e 10. Embora não exista uma relação claramente linear, há indícios de que repositórios muito grandes podem apresentar acoplamento um pouco maior. observa-se que tanto DIT quanto LCOM não mostram uma correlação forte com o tamanho, existindo projetos extensos com valores moderados dessas métricas e projetos menores que exibem valores significativamente altos.

A hipótese foi confirmada apenas parcialmente. Os gráficos indicam que repositórios maiores tendem a apresentar acoplamento (CBO) um pouco mais elevado, mas a relação não é linear nem consistente para todos os casos.

4. Conclusão

Em geral, o relatório mostrou que os repositórios mais populares tendem a apresentar melhores índices de qualidade estrutural, com menores valores médios de CBO e LCOM, o que indica maior modularidade e coesão. A métrica DIT, entretanto, permaneceu relativamente constante, sugerindo que a profundidade da hierarquia de classes não é fortemente influenciada pela popularidade. Esses resultados reforçam a ideia de que a visibilidade e o contínuo feedback da

comunidade podem contribuir para a manutenção de um código mais limpo e organizado.

Por outro lado, as análises relativas à maturidade, atividade e tamanho dos repositórios evidenciaram tendências menos consistentes. Repositórios mais maduros e maiores apresentaram, em alguns casos, maiores valores de CBO e DIT, mas com grande dispersão dos dados, indicando que outros fatores também influenciam a complexidade do código. Da mesma forma, embora repositórios com mais releases geralmente tenham demonstrado melhores níveis de coesão (LCOM menor), essa relação não se mostrou linear. Assim, as hipóteses foram confirmadas parcialmente, ressaltando a complexidade inerente aos processos de desenvolvimento e manutenção de software.