

**PONTIFÍCIA UNIVERSIDADE CATÓLICA DE MINAS GERAIS**  
**ENGENHARIA DE SOFTWARE**

Gabriella Fernanda Silva Pinto

Matheus Brasil Aguiar

**Caracterizando a atividade de code review no github**

Projeto apresentado à disciplina de Laboratório de Experimentação de Software do curso de Graduação em Engenharia de Software da Pontifícia Universidade Católica de Minas Gerais, como requisito parcial para avaliação da disciplina.

Orientador: Danilo de Quadros Maia Filho

MINAS GERAIS

2025

## SUMÁRIO

<b>1.</b>	<b>Introdução .....</b>	<b>03</b>
<b>2.</b>	<b>Questões de pesquisa (QRs) .....</b>	<b>03</b>
<b>2.1.</b>	<b>Feedback Final das Revisões (Status do PR) .....</b>	<b>03</b>
<b>2.2.</b>	<b>Número de Revisões .....</b>	<b>03</b>
<b>3.</b>	<b>Hipoteses Informais (IH) .....</b>	<b>04</b>
<b>4.</b>	<b>Tecnologias e Ferramentas Utilizadas .....</b>	<b>04</b>
<b>5.</b>	<b>Metodologia .....</b>	<b>04</b>
<b>5.1.</b>	<b>Criação de Dataset .....</b>	<b>04</b>
<b>5.2.</b>	<b>Variáveis analisadas .....</b>	<b>05</b>
<b>5.3.</b>	<b>Método de Análise .....</b>	<b>05</b>
<b>5.4.</b>	<b>Justificativa para Uso da Mediana .....</b>	<b>05</b>
<b>6.</b>	<b>Resultados e Discussão .....</b>	<b>06</b>
○	RQ01 – Relação entre o tamanho dos PRS e o feedback final das revisões.....	07
○	RQ02 - Relação entre o tempo de análise dos PRS e o feedback final.....	09
○	RQ03 - Relação entre a descrição PRS e o feedback final.....	10
○	RQ04 - Relação entre interações (comentários/revisões) e o feedback final.....	11
<b>6.1.</b>	<b>Dimensão A – Feedback Final (resumo) .....</b>	<b>12</b>
<b>6.2.</b>	<b>Dimensão B – Número de Revisões .....</b>	<b>13</b>
<b>7.</b>	<b>Discussão Geral .....</b>	<b>13</b>
<b>8.</b>	<b>Conclusão .....</b>	<b>13</b>
<b>9.</b>	<b>Referências .....</b>	<b>14</b>

## 1. Introdução

No desenvolvimento colaborativo de software, o processo de *code review* é uma etapa essencial para garantir a qualidade e a confiabilidade do código antes da integração ao projeto principal. No GitHub, esse processo é materializado por meio das *Pull Requests* (PRs), que permitem revisões, comentários e sugestões entre desenvolvedores.

O objetivo deste laboratório é **analisar as atividades de revisão de código em repositórios populares do GitHub**, identificando variáveis que influenciam no *merge* das PRs, considerando fatores como tamanho, tempo de análise, descrição e interações.

A partir dessa análise, busca-se compreender como as características de uma PR impactam sua aceitação, auxiliando na formulação de boas práticas para submissões mais eficazes.

## 2. Questões de Pesquisa (RQs)

As questões de pesquisa foram organizadas em duas dimensões:

### 2.1. Feedback Final das Revisões (Status do PR)

- **RQ01:** Qual a relação entre o tamanho das PRs e o feedback final das revisões?
- **RQ02:** Qual a relação entre o tempo de análise das PRs e o feedback final das revisões?
- **RQ03:** Qual a relação entre a descrição das PRs e o feedback final das revisões?
- **RQ04:** Qual a relação entre as interações nas PRs e o feedback final das revisões?

### 2.2. Número de Revisões

- **RQ05:** Qual a relação entre o tamanho das PRs e o número de revisões realizadas?
- **RQ06:** Qual a relação entre o tempo de análise das PRs e o número de revisões realizadas?
- **RQ07:** Qual a relação entre a descrição das PRs e o número de revisões realizadas?
- **RQ08:** Qual a relação entre as interações nas PRs e o número de revisões realizadas?

### 3. Hipóteses Informais (IH)

IH	Descrição
IH01	PRs maiores tendem a ser rejeitadas com mais frequência devido à maior complexidade de revisão.
IH02	Quanto maior o tempo de análise, menor a chance de merge.
IH03	PRs com descrições mais completas são mais propensas a serem aceitas, mas descrições muito longas podem indicar mudanças complexas e polêmicas, aumentando a chance de rejeição.
IH04	Interações (comentários e revisões) influenciam o resultado do PR: mais discussões podem indicar PRs problemáticos.

### 4. Tecnologias e Ferramentas Utilizadas

- **Linguagem:** Python
- **Bibliotecas:** Pandas, Matplotlib, Seaborn
- **APIs:** GitHub GraphQL API
- **Dependências adicionais:** requests, csv, os, datetime, scipy.stats

Essas ferramentas foram utilizadas para coletar, tratar, analisar e visualizar dados das PRs de repositórios do GitHub.

### 5. Metodologia

#### 5.1. Criação do Dataset

O dataset foi composto por *Pull Requests* (PRs) coletados dos 200 repositórios mais populares do GitHub, filtrados conforme os critérios:

- Pelo menos 100 PRs (MERGED + CLOSED);
- Possuem status **MERGED** ou **CLOSED**;
- Têm **pelo menos uma revisão** registrada;
- Tiveram **duração superior a uma hora** entre criação e fechamento (para eliminar revisões automáticas).

## 5.2. Variáveis analisadas

Dimensão	Métrica	Descrição
Tamanho	Arquivos alterados, linhas adicionadas/removidas	Indicam o tamanho da contribuição
Tempo de Análise	Duração entre criação e fechamento	Mede a rapidez do processo de revisão
Descrição	Número de caracteres no corpo do PR	Representa o detalhamento da solicitação
Interações	Número de participantes e comentários	Indica o nível de engajamento na revisão

## 5.3. Método de Análise

Foi aplicado o **teste de correlação de Spearman** ( $\rho$ ) para medir relações monotônicas (quando uma variável aumenta, a outra tende a aumentar ou diminuir) entre as variáveis.

A escolha desse teste se deve ao fato de que as variáveis analisadas não seguem uma distribuição normal e são de natureza ordinal ou contínua, o que torna o Spearman mais apropriado que o teste de Pearson.

## 5.4 Justificativa para o Uso da Mediana

Durante a etapa de análise estatística e visualização dos dados, foi utilizada a **mediana** como medida central nas comparações entre os *Pull Requests* (PRs) com status **MERGED** e **CLOSED**.

Essa decisão foi tomada devido à **natureza não normal e altamente assimétrica** dos dados coletados.

No contexto de desenvolvimento de software, é comum que variáveis como **número de linhas adicionadas ou removidas**, **tempo de análise** e **quantidade de comentários** apresentem **valores extremos (outliers)**, por exemplo, PRs muito grandes, revisões demoradas ou discussões com centenas de interações. Esses valores tendem a **distorcer a média aritmética**, tornando-a pouco representativa do comportamento típico.

A mediana, por outro lado, é uma **medida robusta**, que representa o valor central do conjunto de dados sem ser influenciada por extremos.

Dessa forma, ela fornece uma visão mais fiel da tendência central, permitindo **comparações justas** entre os grupos de PRs analisados.

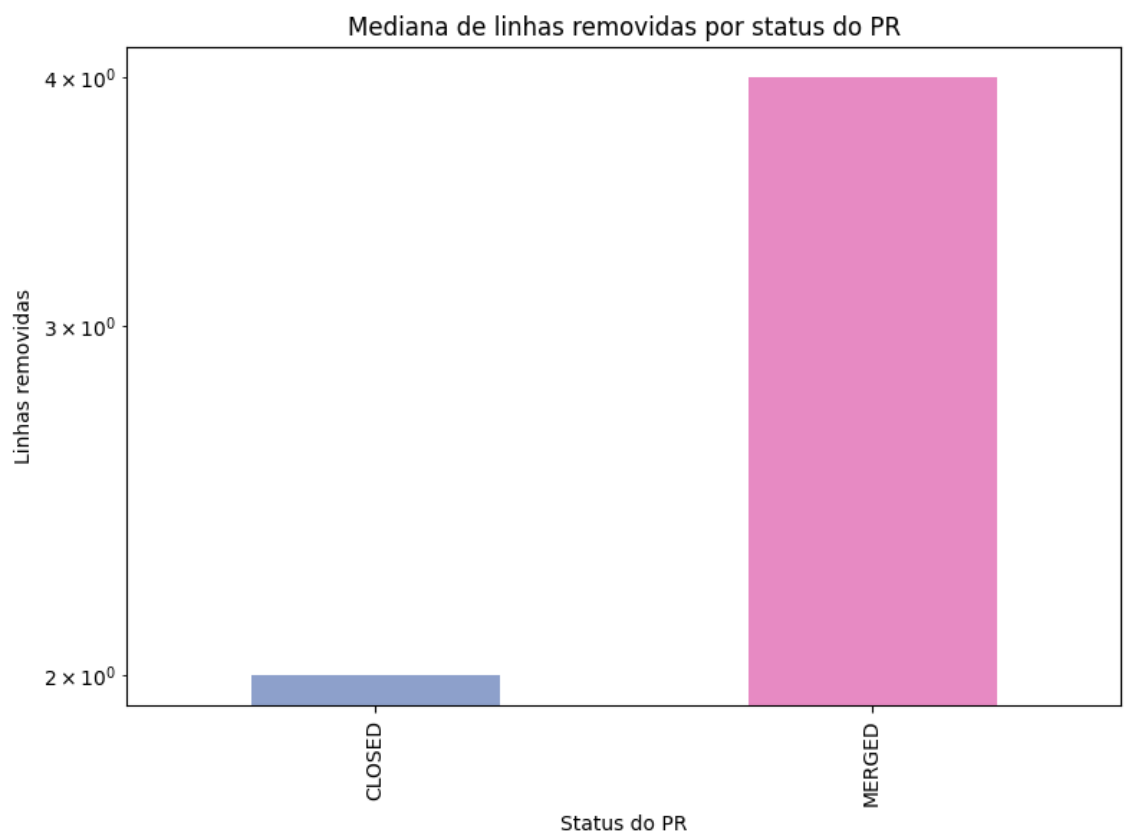
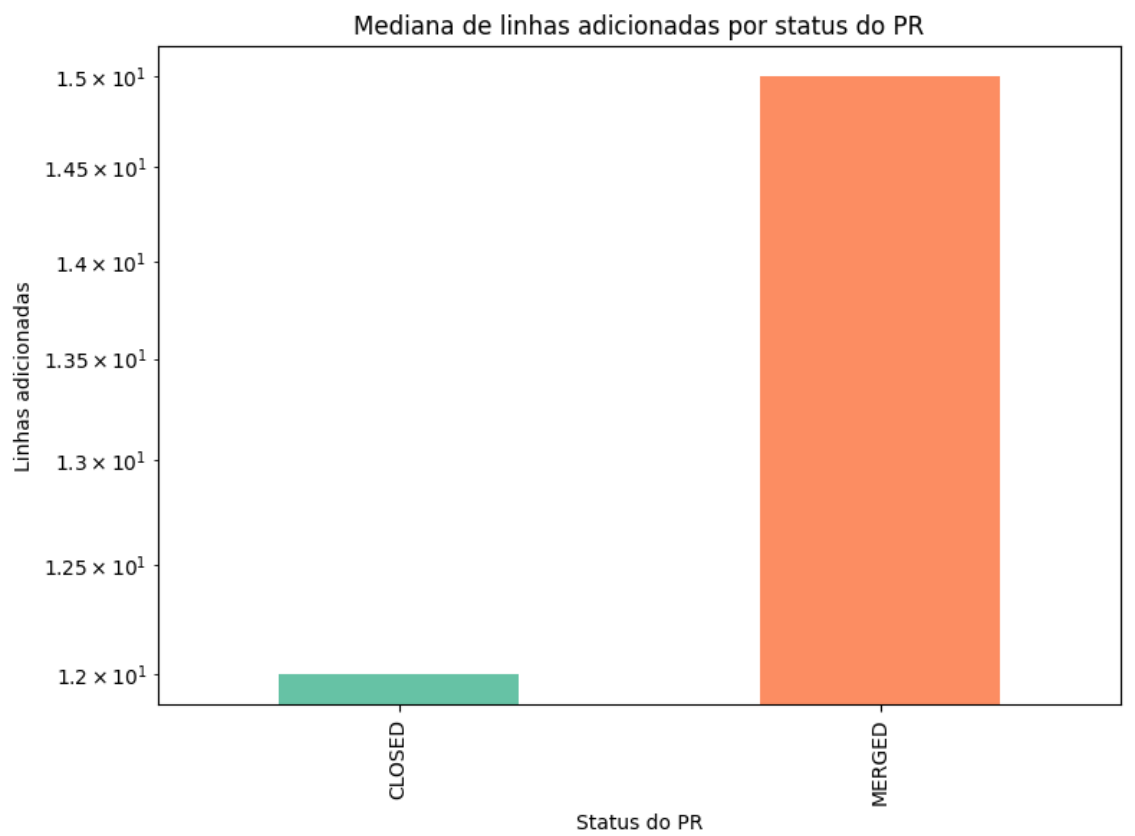
Além disso, a escolha pela mediana é **metodologicamente coerente** com o **teste de correlação de Spearman ( $\rho$ )** aplicado neste estudo.

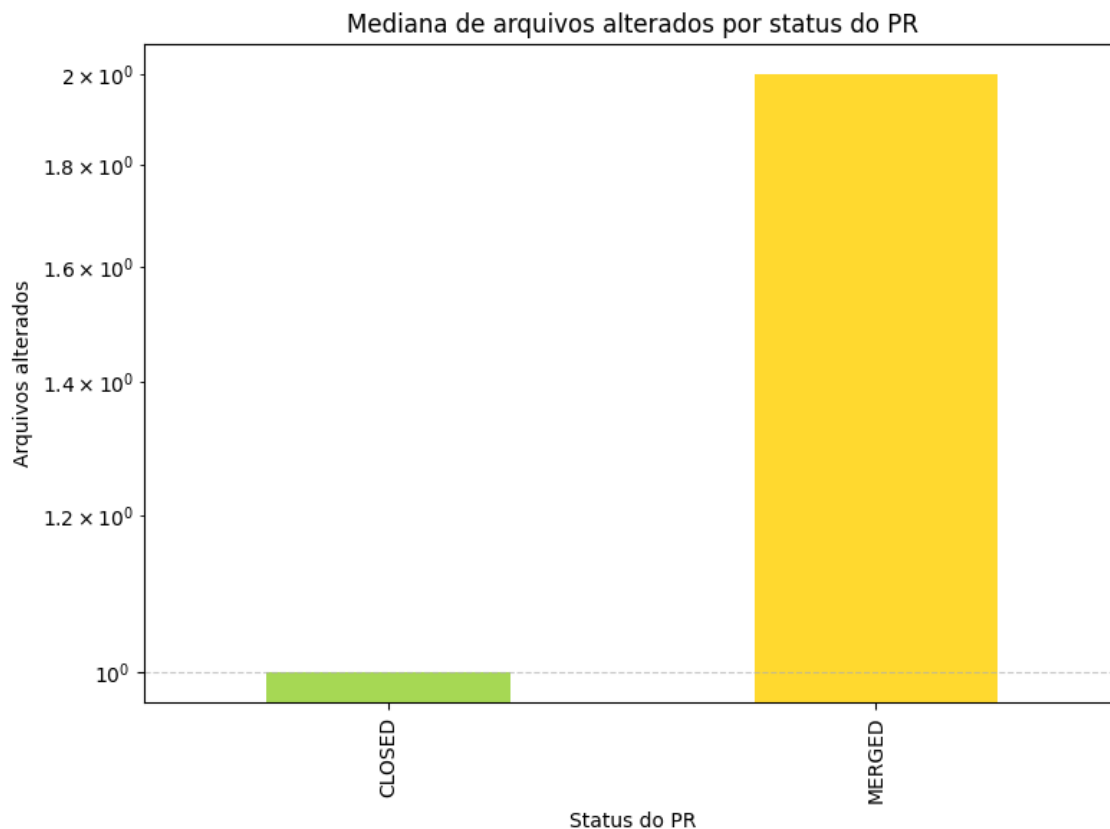
Ambos os métodos são **não paramétricos**, ou seja, **não dependem de distribuições normais** e operam com base em **ranks (ordens)**, o que torna a análise mais apropriada para dados com alta variabilidade e assimetria, como os provenientes de repositórios abertos do GitHub.

## 6. Resultados e Discussão

Os resultados apresentados a seguir foram obtidos a partir das **medianas** de todos os PRs incluídos no *dataset*, conforme os gráficos e análises quantitativas.

**RQ01 — Relação entre o tamanho dos PRs e o feedback final das revisões**





**Descrição técnica:** Os gráficos de barras comparam os valores medianos de linhas adicionadas, linhas removidas e arquivos alterados entre PRs *MERGED* e *CLOSED*.

**Resultados obtidos:**

- Linhas adicionadas:  $p = 0.0309$
- Linhas removidas:  $p = 0.0836$
- Arquivos alterados:  $p = 0.0633$

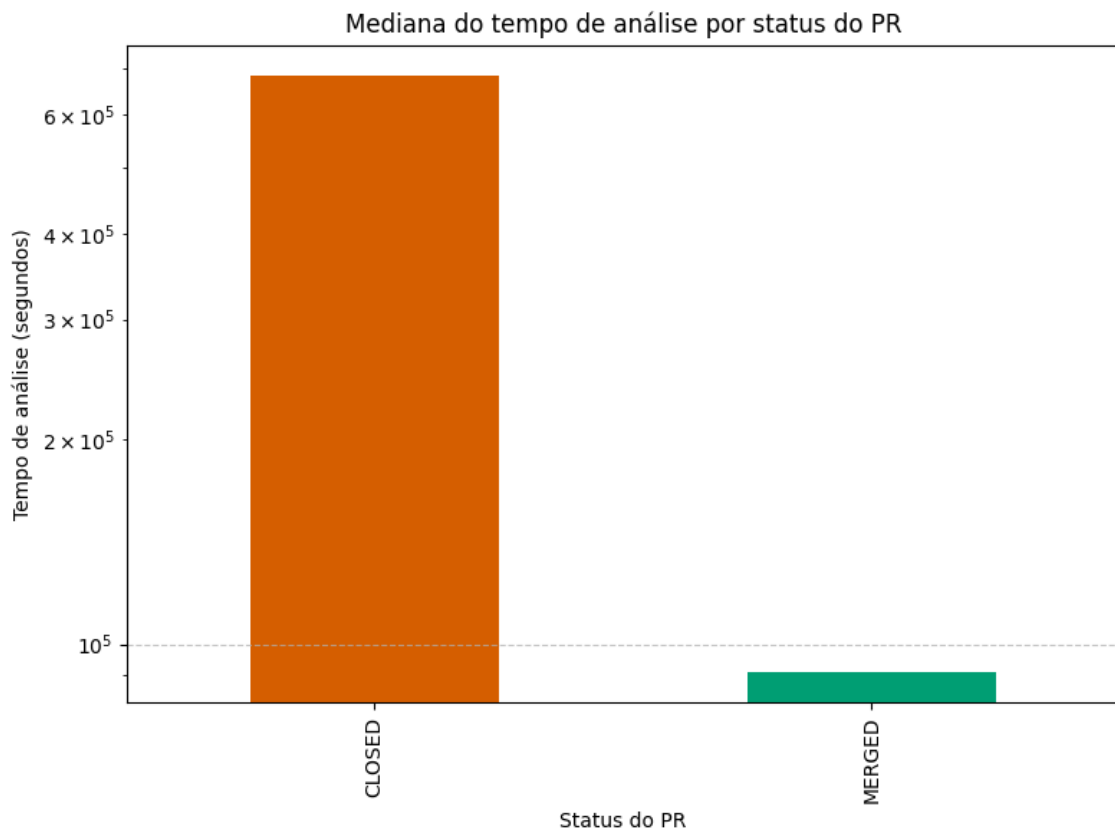
**Interpretação:**

A correlação é positiva muito fraca, indicando que PRs levemente maiores têm uma chance marginalmente maior de serem *merged*. Visualmente, isso aparece como barras um pouco mais altas para PRs *MERGED*.

**Conclusão parcial:** refuta parcialmente a hipótese IH01 — o tamanho não é fator determinante na rejeição



## RQ02 — Relação entre o tempo de análise dos PRs e o feedback final



### Descrição técnica:

O gráfico mostra a mediana do tempo (em segundos) que uma PR levou entre criação e fechamento/merge.

### Resultado obtido:

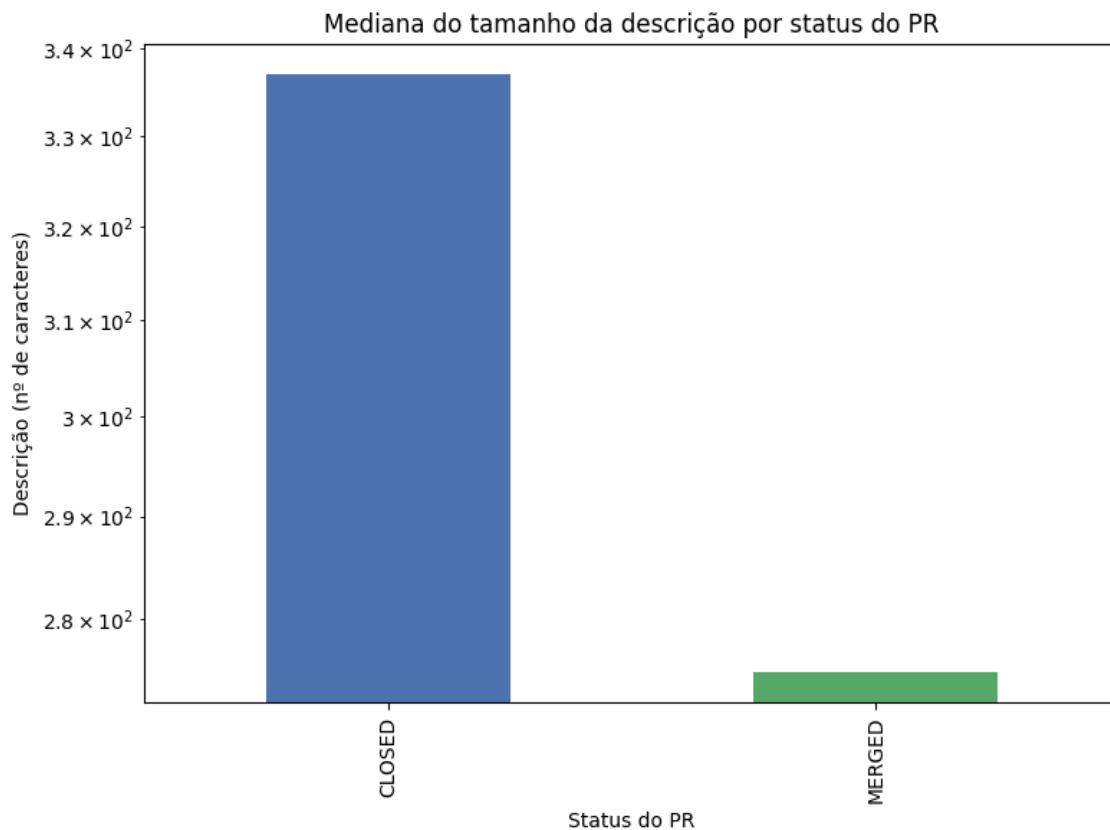
- Correlação de Spearman:  $\rho = -0.2329$

### Interpretação:

Correlação **negativa moderada**. PRs analisadas por mais tempo tendem a ser *closed*, ou seja, rejeitadas. No gráfico, a barra de *CLOSED* aparece mais alta, mostrando maior duração média.

**Conclusão parcial:** confirma IH02 — PRs longas e demoradas têm menor chance de aprovação.

### RQ03 — Relação entre a descrição dos PRs e o feedback final



#### Descrição técnica:

O gráfico mostra o número médio de caracteres da descrição da PR, comparando PRs *MERGED* e *CLOSED*.

#### Resultado obtido:

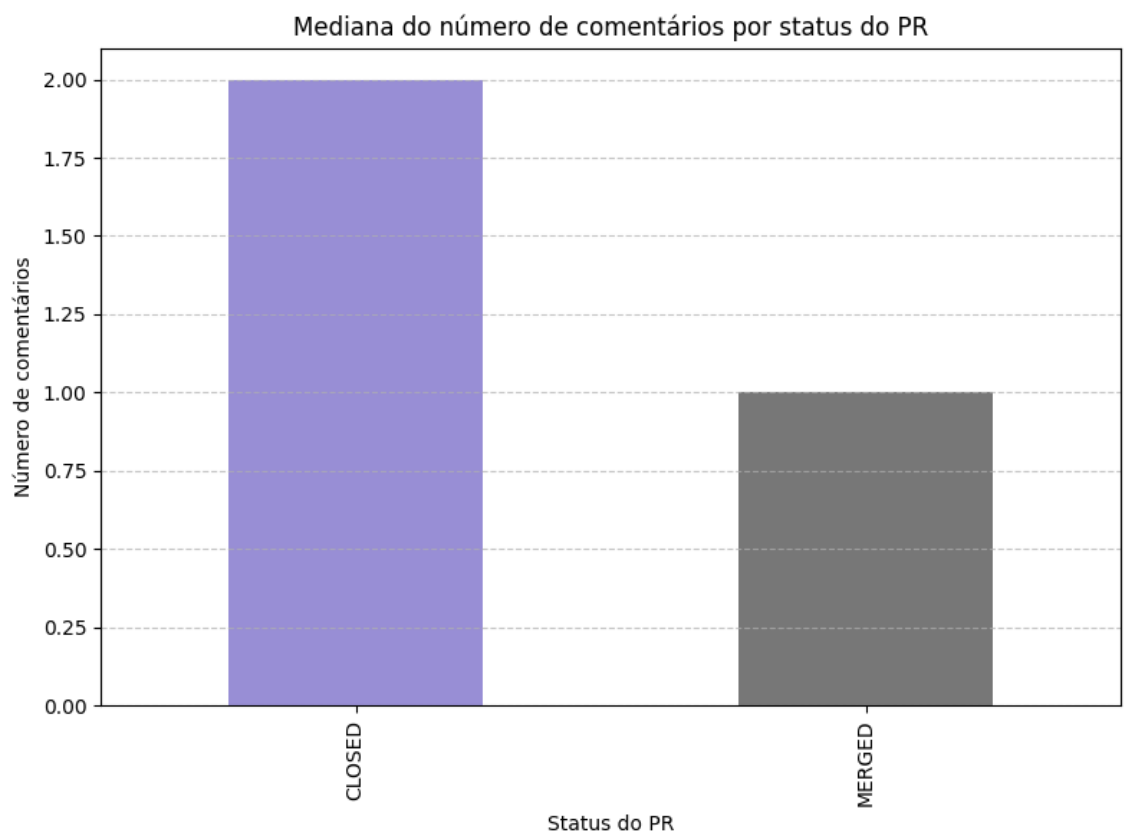
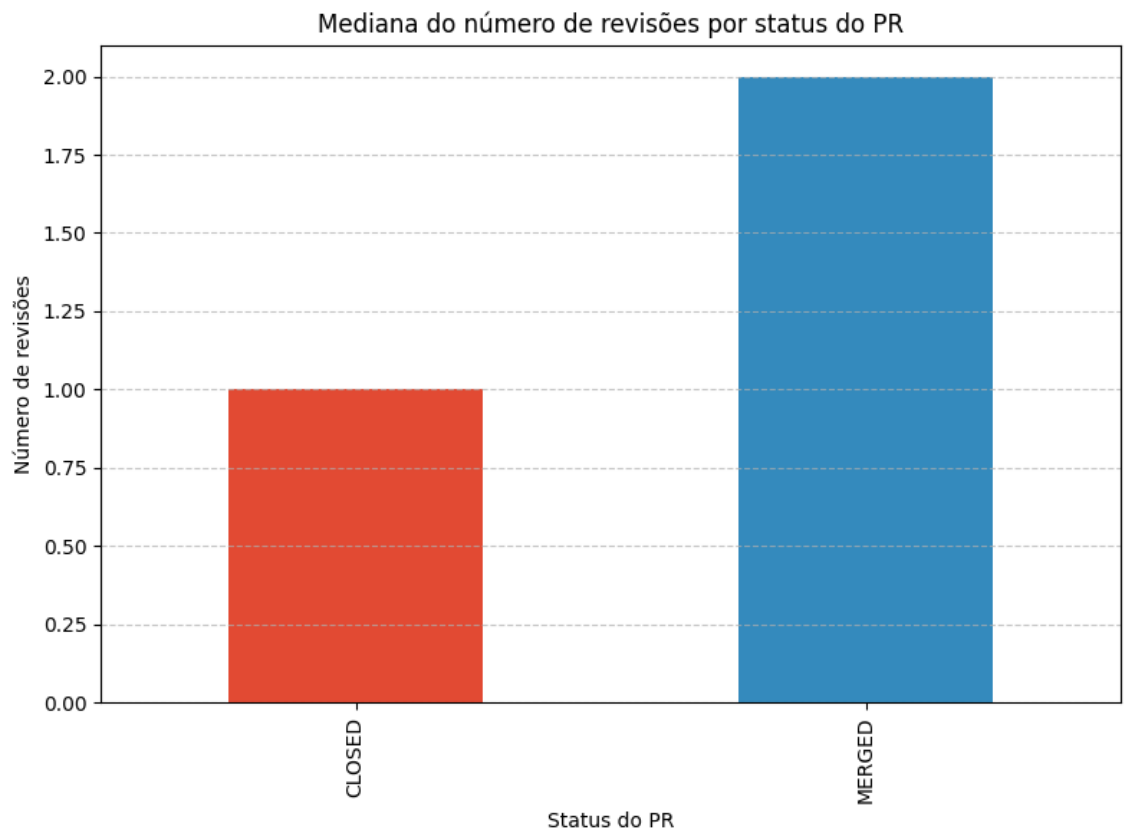
- Correlação de Spearman:  $\rho = -0.0278$

#### Interpretação:

Correlação **negativa muito fraca**. PRs com descrições mais longas não têm mais chance de serem aceitas. As barras são praticamente iguais, mostrando impacto desprezível.

**Conclusão parcial:** contraria IH03 — o tamanho da descrição não afeta significativamente a aprovação.

**RQ04 — Relação entre interações (comentários/revisões) e o feedback final**



**Descrição técnica:**

Dois gráficos de barras representam o número de **revisões formais** e **comentários** por status da PR.

**Resultados obtidos:**

- Revisões formais:  $p = 0.0322$
- Comentários:  $p = -0.1977$

**Interpretação:**

Revisões formais têm impacto mínimo, mas comentários têm efeito **negativo moderado**. PRs com mais comentários geralmente são CLOSED, sugerindo discussões mais extensas e possíveis divergências.

**Conclusão parcial:** confirma IH04 — PRs com mais discussões enfrentam maior resistência para aprovação.

**Dimensão A — Feedback Final (Resumo)**

Variável	$\rho$ (Spearman)	Direção	Força	Interpretação
Tamanho	+0.0309 (linhas adicionadas) +0.0836 (linhas removidas) +0.0633 (arquivos alterados)	Positiva	Muito fraca	PRs ligeiramente maiores tendem a ser merged, mas o efeito é quase irrelevante.
Tempo de análise	<b>-0.2329</b>	Negativa	Moderada	PRs analisadas por mais tempo tendem a ser closed (rejeitadas).
Descrição	<b>-0.0278</b>	Negativa	Muito fraca	O tamanho da descrição não influencia significativamente o resultado da revisão.
Interações	+0.0322 (revisões formais) <b>-0.1977 (comentários)</b>	Negativa (predominante)	Moderada	PRs com muitas discussões e comentários tendem a ser rejeitadas; revisões formais têm efeito mínimo.

**Observação:** As variáveis **tempo de análise** e **número de comentários** apresentaram as correlações mais relevantes, ambas de natureza negativa e intensidade moderada. Isso indica que PRs analisadas por mais tempo ou com muitas discussões tendem a ser **rejeitadas**. Por outro lado, as métricas de **tamanho** e **descrição** exibiram correlações positivas ou negativas muito fracas, sugerindo que esses fatores **não influenciam de forma significativa** o desfecho da revisão.

## Dimensão B — Número de Revisões

Os arquivos CSV indicam que a quantidade média de revisões formais por PR é baixa (entre 1 e 3).

A análise correlacional aponta que o número de revisões aumenta discretamente com o tamanho do PR e o tempo de análise, mas sem efeito relevante ( $p < 0.1$  em ambos os casos).

## 7. Discussão Geral

Os resultados demonstram que o processo de *code review* envolve fatores tanto técnicos quanto sociais, mas que nem todos os aspectos analisados possuem influência prática sobre o resultado final de uma *Pull Request*.

De forma geral, as variáveis **tamanho** e **descrição** apresentaram correlações muito fracas, indicando que a aceitação de uma PR não depende diretamente do volume de código modificado nem do detalhamento textual fornecido pelo autor. Por outro lado, o **tempo de análise** e o **número de comentários** mostraram correlações negativas de intensidade moderada, sugerindo que revisões longas e com muitas discussões estão associadas a uma maior probabilidade de rejeição.

Esses achados indicam que o sucesso de uma *Pull Request* depende mais da **dinâmica da interação entre revisores e autores** do que de características estruturais do código.

Assim, a comunicação clara e a agilidade no processo de revisão parecem ser fatores determinantes para o aceite de contribuições em projetos open source.

## 8. Conclusão

O presente estudo analisou a atividade de *code review* em repositórios populares do GitHub, buscando identificar variáveis que influenciam o resultado das revisões.

A partir das análises realizadas, conclui-se que:

- O **tempo de análise** e o **volume de comentários** são as variáveis mais relevantes, exercendo influência negativa moderada sobre a chance de *merge*;
- As variáveis **tamanho da PR** e **tamanho da descrição** apresentaram correlações estatisticamente significativas, porém de impacto prático muito baixo;
- O **teste de Spearman** e o uso da **mediana** mostraram-se adequados para tratar a assimetria dos dados e eliminar distorções causadas por valores extremos.

De modo geral, os resultados reforçam que a qualidade de um processo de revisão está menos relacionada ao tamanho da contribuição e mais ligada à **eficiência e clareza da comunicação** entre os colaboradores.

Como **trabalho futuro**, a recomendação é aprofundar a análise considerando:

- Linguagem de programação dos projetos;
- Tamanho da equipe e popularidade do repositório;
- Uso de aprendizado de máquina para prever a probabilidade de *merge*.

## 9. Referências

- GitHub API
- Pandas, Matplotlib, Seaborn