

MINERAÇÃO DE REPOSITÓRIOS PARA AVALIAR A INFLUÊNCIA DAS MUDANÇAS DE CÓDIGO AO LONGO DO TEMPO

Leonardo Scarmato Jorge de Paula

Orientador: Prof. Dr. Higor Amario de Souza

CIÊNCIA DA COMPUTAÇÃO

CRONOGRAMA

■ INTRODUÇÃO

■ FUNDAMENTAÇÃO TEÓRICA

■ METODOLOGIA

■ RESULTADOS E ANÁLISES

Introdução

CONTEXTO E MOTIVAÇÃO

- Importância da Ciência de Dados
- Relevância da mineração de repositórios
- Adaptação de Frameworks e Bibliotecas
- Facilitação da tomada de decisão

JUSTIFICATIVA

1. Crescente demanda por análise robusta de dados
2. Frameworks e bibliotecas – avanços tecnológicos
3. Aprimorar processos de desenvolvimento
4. Evitar retrabalho
5. Manutenção direcionada
6. Contribuições em projetos Open-Source

OBJETIVOS

- Analisar alterações no código ao longo do tempo
- Repositórios Python
- Avaliar a relação entre modificações – commits – issues
- Insights que direcionem decisões de manutenção e atualização para os desenvolvedores

Fundamentação Teórica

- Frameworks e bibliotecas
- Controle de versão
- Git
- Mineração de repositórios



Fundamentação Teórica

- Python
- Pydriller
- Matplotlib
- Pandas
- Requests



METODOLOGIA

- Critérios de seleção de softwares
- Contribuição da comunidade
- Frequência de atualização
- Área de atuação
- Procedimento de coleta de dados (commits – issues)

Tabela 1 - Seleção de Software

Software	Área de atuação
TensorFlow	Machine Learning
Scrapy	Web Scraping
Pandas	Ciência de Dados
Flask	Desenvolvimento Web
Django	Desenvolvimento Web
FastAPI	Desenvolvimento Web
OpenCV	Processamento de Imagens
Pillow	Processamento de Imagens
Scikit-Learn	Machine Learning
SQLAlchemy	Banco de Dados

FERRAMENTAS

TensorFlow

- Ferramenta que usa machine learning e redes neurais

Flask

- Framework para desenv. web em Python

Scrapy

- Ferramenta de coleta de dados estruturados

Django

- Framework para desenv. web

Pandas

- Biblioteca de ciência de dados tabulares
- Análise e limpeza de grandes conjuntos

FastAPI

- Otimização de APIs

FERRAMENTAS

OpenCV

- Biblioteca para processamento de imagens

Scikit-Learn

- Essencial para machine learning
- Modelos de aprendizado supervisionado e ns

Pillow

- Biblioteca para manipulação de imagens

SQLAlchemy

- Ferramenta para interação com bancos de dados

Obtenção de Dados

- Ferramentas utilizadas:
 1. Pydriller: Extração de dados com histórico
 2. Requests: acesso à issues do github



Análise de Commits

- Pydriller e detalhamento de commits
 1. Identificar alterações em arquivos e métodos
 2. Dados do autor, data de criação e comentários
- Exemplo de código:

```
for commit in Repository("path/repo").trasverse_commits():
    print(f"Commit hash: {commit.hash}")
    print(f"Autor: {commit.author.name}")
```



Vantagens da abordagem

- Visão macro de evolução do código
- 1. 30, 60 e 120 dias
- 2. Identificação de padrões de mudança e manutenção
- 3. Estabilidade do código e evolução do projeto



Resultados e Análises

- 10 Projetos
- 30, 60 e 120 dias
- Commits, Issues, alterações e interpretações

TensorFlow

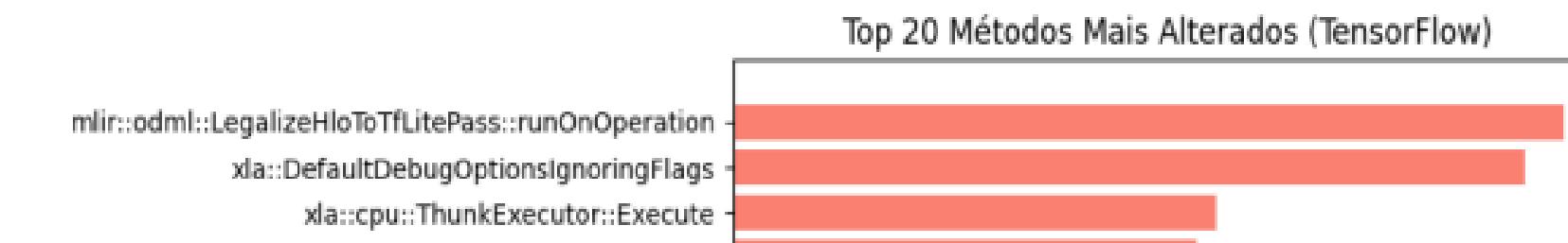
Tabela 2 - TensorFlow

Período	Commits	Issues	Arquivos Alterados	Métodos Alterados
30 dias	446	3598	781 (4.06%)	1752 (8.61%)
60 dias	450	3612	789 (4.08%)	1763(8.64%)
120dias	472	4039	813 (4.12%)	1792 (8.71%)

TensorFlow

- Aprimoramento contínuo em documentações
- Ajustes pontuais: estabilidade sem mudanças estruturais
- Otimização de performance (métodos de infraestrutura)

Figura 1- Top 20 Métodos Mais Alterados (TensorFlow)



Métodos repetidos (alterados mais de uma vez) em ordem decrescente:
stream_executor::gpu::GpuExecutor::CreateDeviceDescription: 10 alterações
stream_executor::gpu::GpuDriver::DestroyStream: 10 alterações
stream_executor::gpu::GpuDriver::LaunchKernel: 9 alterações

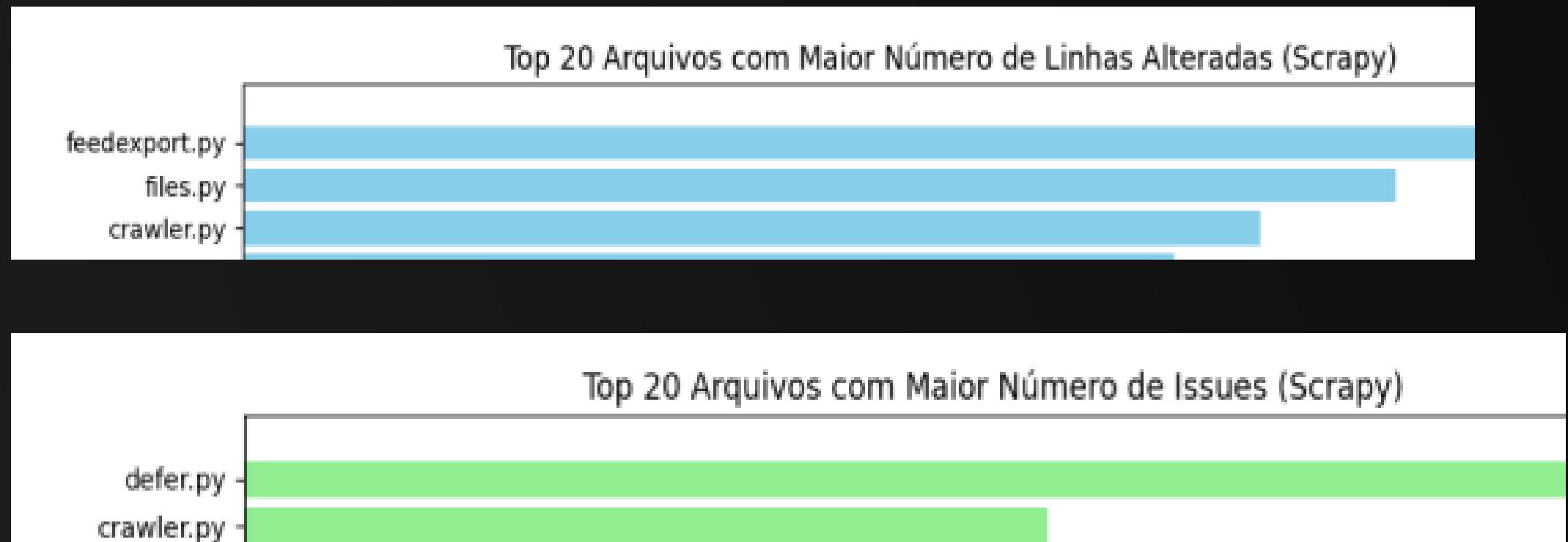
Scrapy

Tabela 3 - Scrapy

Período	Commits	Issues	Arquivos Alterados	Métodos Alterados
30 dias	24	2148	139 (24.47%)	416 (28.4%)
60 dias	37	2346	144 (25.35%)	419 (28.45%)
120 dias	66	2503	152(26.76%)	448 (29.17%)

Scrapy

- Abordagem agressiva e dinâmica
- Correção de bugs e novas funcionalidades
 - Grande quantidade de issues
 - feedexport.py (exportação de dados)
- Arquivos mais alterados ~ Issues
- Crawl – navegar e coletar dados em sites



Django

Tabela 4 - Django

Período	Commits	Issues	Arquivos Alterados	Métodos Alterados
30 dias	52	N/A	87 (1.27%)	75 (5.45%)
60 dias	133	N/A	161 (2.34%)	168 (3.35%)
120 dias	280	N/A	244 (3.55%)	282 (4.18%)

Django

- Muitos arquivos de testes
- Integração entre diferentes idiomas
- Gerenciamento de eventos (handle, try–except)
ajustes críticos
- Aumento de arquivos e métodos
- Abordagem de melhorias a médio prazo

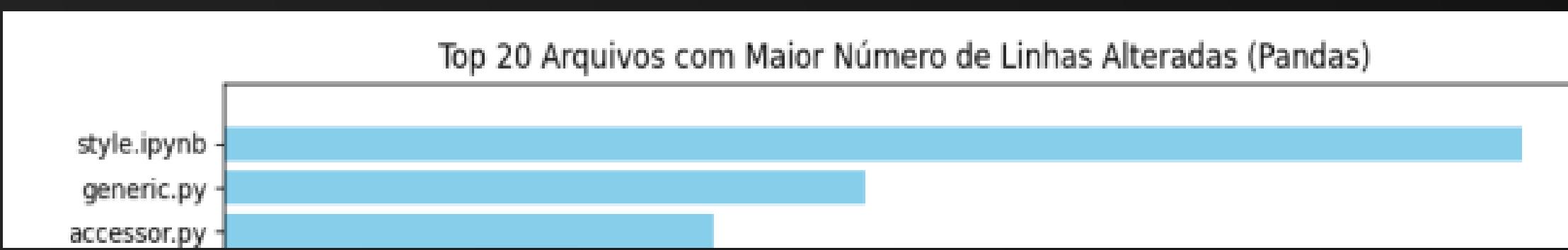
Pandas

Tabela 5 - Pandas

Período	Commits	Issues	Arquivos Alterados	Métodos Alterados
30 dias	82	3257	105 (3.98%)	137 (4.41%)
60 dias	203	3389	167 (6.33%)	317 (8.19%)
120 dias	467	3961	322 (12.20%)	609 (11.94%)

Pandas

- Foco em manipulação e visualização de dados
- Refinamento, otimização e estabilização
- Tratamento de Strings e melhorias de performance (map)
- Utiliza Jupyter para testar novas funcionalidades – gráficos



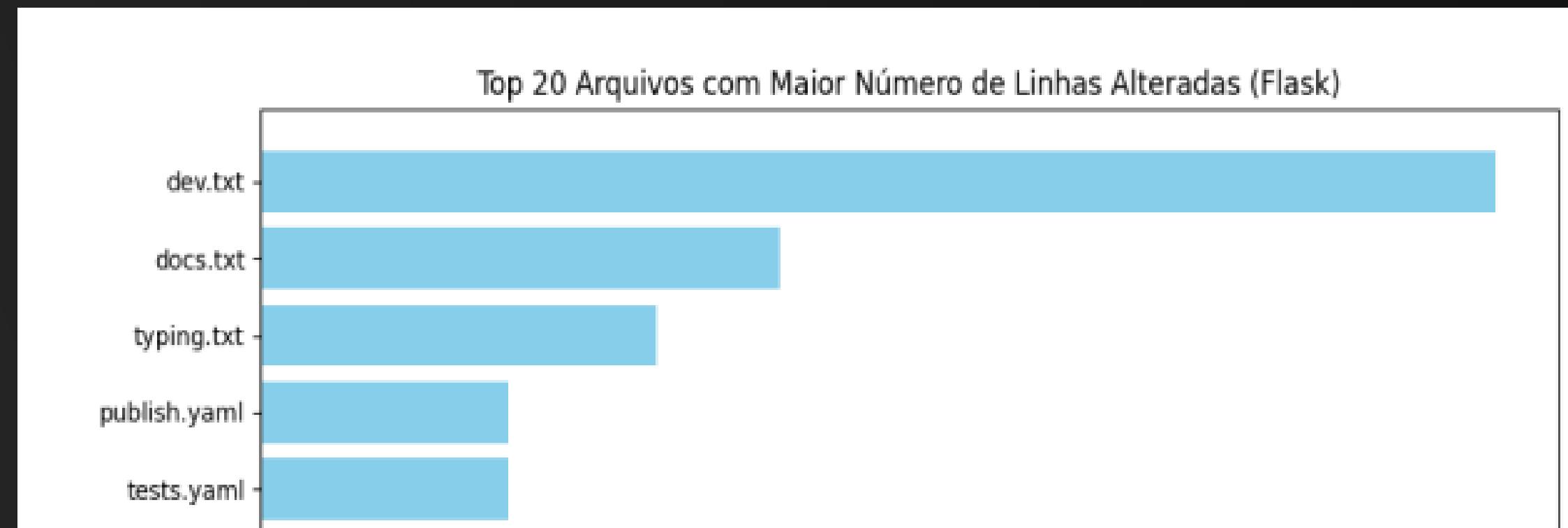
Flask

Tabela 6 - Flask

Período	Commits	Issues	Arquivos Alterados	Métodos Alterados
30 dias	4	71	11 (3.97%)	2 (8%)
60 dias	7	71	12 (4.33%)	2 (8%)
120 dias	32	360	20 (7.22%)	11 (10%)

Flask

- Foco em documentação
- Poucas mudanças em métodos: estabilidade
- Ajustes na lógica do aplicativo (app e testes)
- open_resource e stream indicam gestão de recursos, qualidade e eficiência

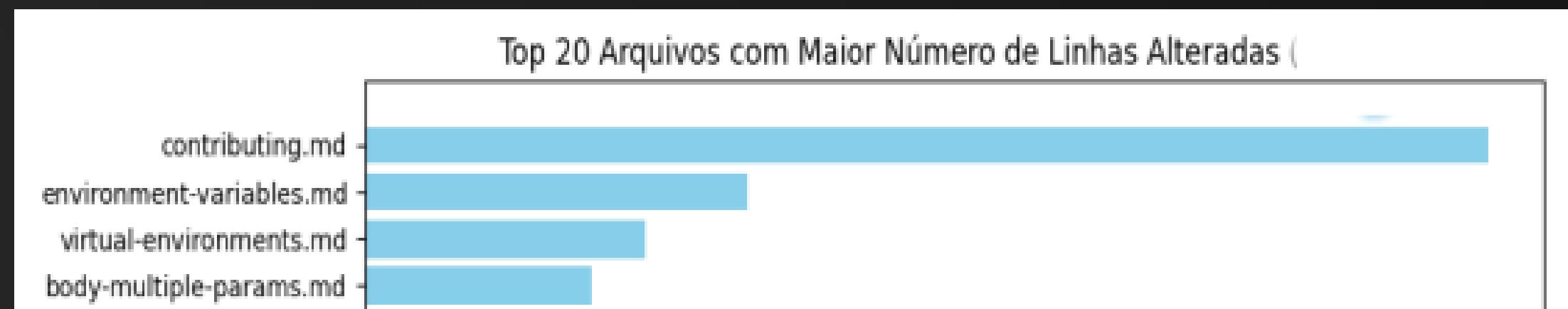


FastAPI

Período	Commits	Issues	Arquivos Alterados	Métodos Alterados
30 dias	376	1532	125 (5.06%)	3 (4.55%)
60 dias	475	1608	173 (7%)	39 (21.55%)
120 dias	801	1934	228 (9.23%)	85 (31.37%)

FastAPI

- Instabilidade inicial – variáveis de ambiente e escopos de projeto
- Esforços em estabilização
- Mudanças em virtualização



OpenCV

Tabela 8 – OpenCV

Período	Commits	Issues	Arquivos Alterados	Métodos Alterados
30 dias	48	1234	84 (1.11%)	191 (6.96%)
60 dias	95	1307	162 (2.13%)	479 (7.55%)
120 dias	246	1422	466 (6.13%)	979 (9.48%)

OpenCV

- Software de interface gráfica: Sen, cos, exp
- Instabilidade inicial na análise
- Muitos arquivos alterados no maior período

Pillow

Período	Commits	Issues	Arquivos Alterados	Métodos Alterados
30 dias	89	458	30 (1.71%)	35 (7.10%)
60 dias	283	534	88 (5.01%)	97 (10.31%)
120 dias	638	642	243 (13.85%)	637 (31.32%)

Pillow

- Problemas de compatibilidade
- Muitos testes em um plugin específico (EpsImage) – gráficos encapsulados de alta qualidade
- Grande quantidade de métodos alterados – Novas funcionalidades

Scikit-Learn

Período	Commits	Issues	Arquivos Alterados	Métodos Alterados
30 dias	126	314	286 (16.00%)	131 (8.52%)
60 dias	211	347	347 (19.42%)	208 (12.51%)
120 dias	380	389	641 (35.87%)	436 (17.17%)

Scikit-Learn

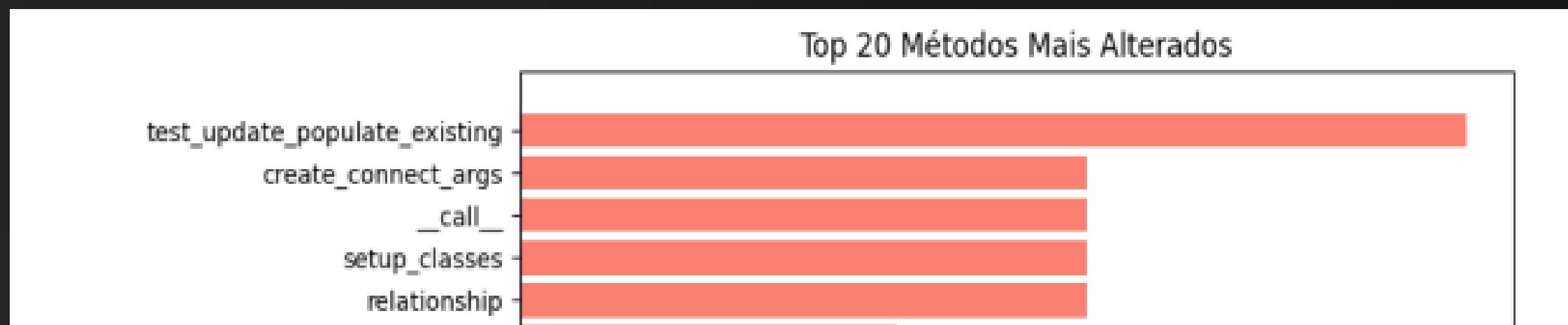
- Validação e processamento de modelos (pipeline e estimator checks)
- Alta atividade de Issues
- Melhorias específicas para otimizar fluxos e execução de modelos de machine learning
- LogisticRegression – DecisionTree

SQLAlchemy

Período	Commits	Issues	Arquivos Alterados	Métodos Alterados
30 dias	27	254	57 (5.74%)	50 (3.26%)
60 dias	73	306	81 (8.16%)	79 (2.91%)
120 dias	164	341	164 (16.52%)	201 (4.78%)

SQLAlchemy

- Melhorias na conexão com a DB
- Maioria dos issues relacionados à compatibilidade
- Após 60 dias: Sincronização e mapeamento de colunas
- Desempenho e robustez



Conclusão

CONTEXTO E MOTIVAÇÃO

- Compreender a evolução dos projetos
- Motivações da equipe
- Adaptação ao mercado x Correções

Conclusão

Ciência de dados e Mineração de repositórios

- Essencial para extrair informações
- Identificar padrões – tomada de decisão estratégica
- Compreender mudanças e interações da comunidade
- Insights valiosos sobre o trabalho dos desenvolvedores

Conclusão

Facilitação da Tomada de Decisão

- Ignorar partes do código
- Direcionar mudanças pra métodos ou arquivos específicos
- Evitar retrabalho
- Adaptação ao mercado (Flask, Pandas, Scikit-Learn)

Perspectiva futura

- Estudos longitudinais
- Análise de diferentes áreas (Segurança e automação)
- Aplicação em projetos menores – startups
- Visualização gráfica 3D



Referências



OBRIGADO