

# Engenharia de Software II

Autoria de código e Truck Factor

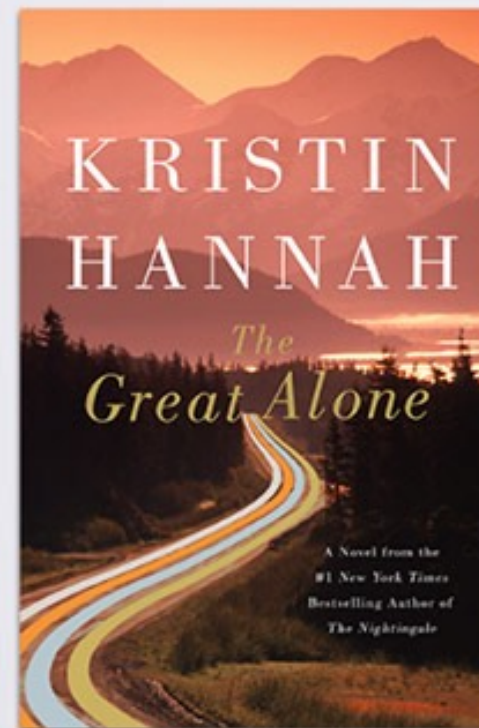
Prof. André Hora  
DCC/UFMG  
2019.1

**Autoria** é documentada na maioria dos trabalhos intelectuais

# KRISTIN HANNAH

[Home](#)[About](#) ▾[Books](#)[News](#)[Book Clubs](#)[Tour Dates](#)[Photo Gallery](#)[Newsletter](#)[Contact](#)

***#1 New York Times Bestselling  
Author***



ON SALE FEB 6: PRE-ORDER NOW



ON SALE IN PAPERBACK












# GAROTA DE IPANEMA

Bossa Nova

Tom Jobim & Vinícius de Moraes



# Assessing Code Authorship: The Case of the Linux Kernel

Guilherme Avelino<sup>1,2</sup> , Leonardo Passos<sup>3</sup> , Andre Hora<sup>1</sup> ,  
and Marco Tulio Valente<sup>1</sup>  

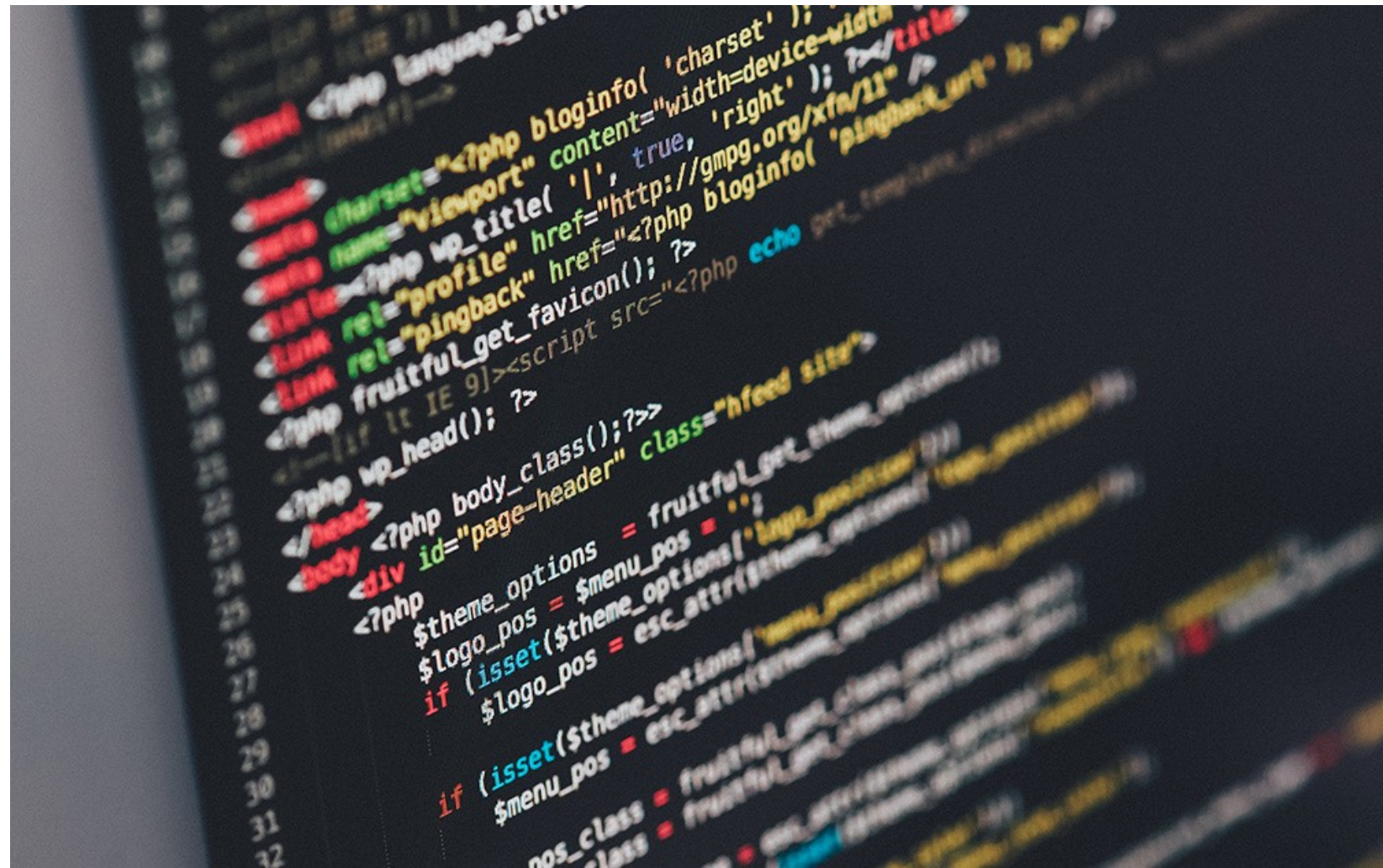
<sup>1</sup> Federal University of Minas Gerais (UFMG), Belo Horizonte, Brazil  
`{gaa,hora,mtov}@dcc.ufmg.br`

<sup>2</sup> Federal University of Piaui (UFPI), Teresina, Brazil

<sup>3</sup> University of Waterloo, Waterloo, Canada  
`lpassos@gsd.uwaterloo.ca`



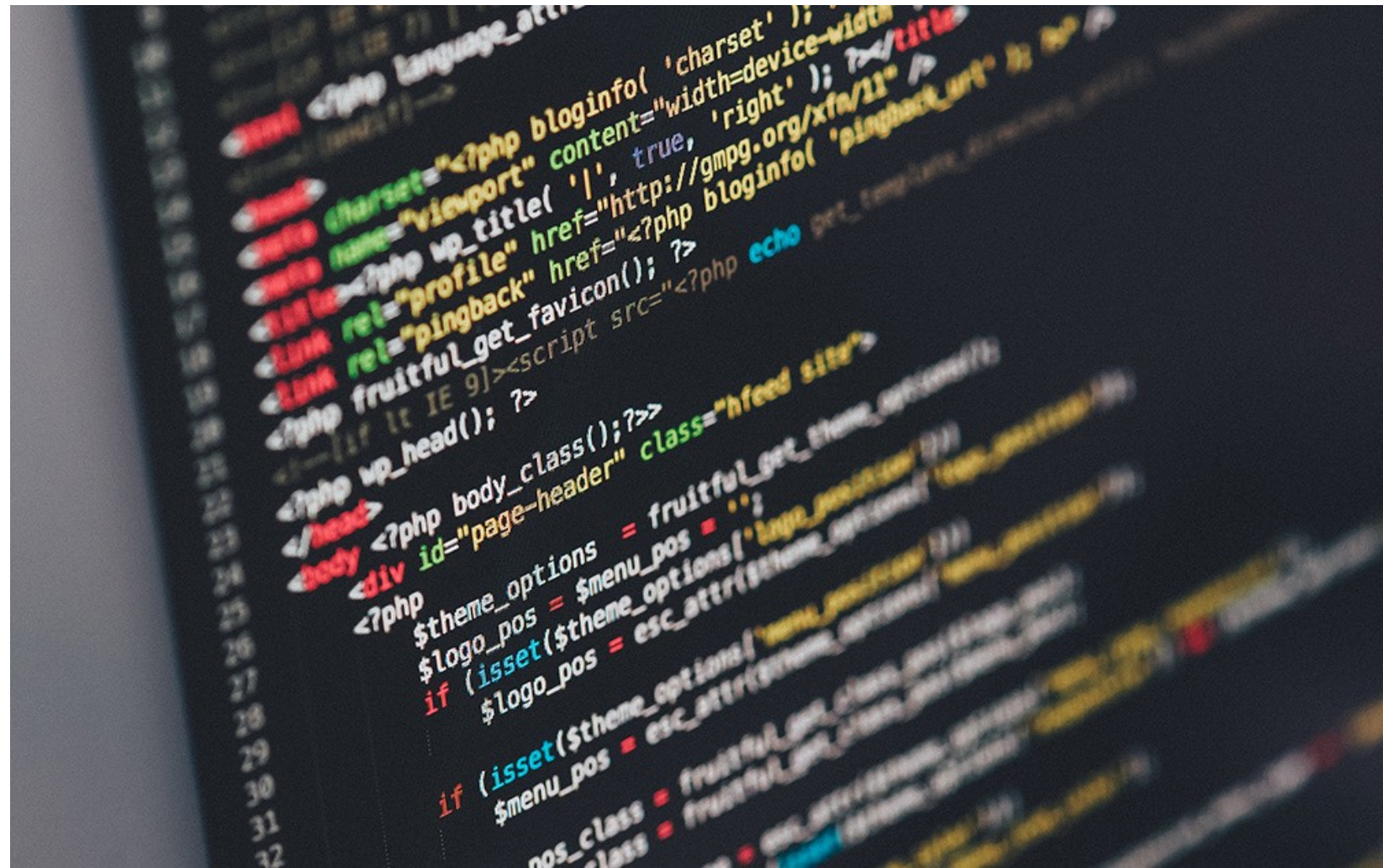
# Código fonte?





# Código fonte?

- Nomes dos autores não estão estampados no código
- Autoria de código pode evoluir ao longo do tempo






# Quem são os autores?





- Sistemas open source, por exemplo, possui milhares de contribuidores


# Linux


 **torvalds / linux**

 Watch ▾ 6,706

 Star 74,235

 Fork 26,004


 Code


 Pull requests 257

 Projects 0

 Insights

Linux kernel source tree

 826,603 commits

 1 branch

 602 releases

 ∞ contributors

 View license



# Autoria de Código

- Trabalho colaborativo é fundamental no desenvolvimento de software atualmente
- Em um ambiente colaborativo, **autoria de código** permite identificar membros chaves do projeto:
  - Quem deve dar manutenção no código
  - Quem deve corrigir um determinado bug
  - Quem tem maior conhecimento sobre parte do código
  - Qual a divisão de trabalho

# Autoria de Código

- No contexto de código, autoria está relacionada com quem realiza mudanças significativas em um determinado arquivo
- Partes envolvidas: ?
- Autoria de código: dinâmica pois muda a medida que o software evolui



# Autoria de Código

- No contexto de código, autoria está relacionada com quem realiza mudanças significativas em um determinado arquivo
- Partes envolvidas: o criador do arquivo e os devs que modificam o arquivo
- Autoria de código: dinâmica pois muda a medida que o software evolui

# Autoria de Código $\neq$ Autoria

- Diferente de autoria de código, autoria de livros e artigos científicos não mudam ao longo tempo
- Autoria de livros e artigos científicos: estática



# Como detectar autoria em código?

Fatores importantes

- Dado um um arquivo  $f$  e um desenvolvedor  $d$
- C: Criador do arquivo  $f$
- MD: Número de mudanças realizadas em  $f$  pelo desenvolvedor alvo  $d$
- MO: Número de mudanças realizadas em  $f$  por outros desenvolvedores

# Fórmula simples para calcular autoria

- Dado um um arquivo  $f$  e um desenvolvedor  $d$
- $\text{Autoria}(d,f) = C + 0,5 * MD - 0,1 * MO$
- Se  $d$  for o criador de  $f$  então  $C = 1$

# Fórmula simples para calcular autoria

- Dado um um arquivo  $f$  e um desenvolvedor  $d$
- $\text{Autoria}(d,f) = C + 0,5 * MD - 0,1 * MO$
- Se  $d$  for o criador de  $f$  então  $C = 1$

**pesos podem  
variar**



# Fórmula simples para calcular autoria

- Dado um um arquivo  $f$  e um desenvolvedor  $d$
- $\text{Autoria}(d,f) = C + 0,5 * MD - 0,1 * MO$
- Se  $d$  for o criador de  $f$  então  $C = 1$

**pesos podem  
variar**

## Outras soluções...

$$DOA_A(d, f) = 3.293 + 1.098 * FA + 0.164 * DL - 0.321 * \ln(1 + AC)$$

# Exemplo

$$\text{Autoria}(d,f) = C + 0,5 * MD - 0,1 * MO$$

- Commit1: dev1, file1  $\longrightarrow$   $\text{autoria}(\text{dev1}, \text{file1}) = 1$
- Commit2: dev1, file1  $\longrightarrow$   $\text{autoria}(\text{dev1}, \text{file1}) = 1,5$
- Commit3: dev2, file1  $\longrightarrow$   $\text{autoria}(\text{dev1}, \text{file1}) = 1,4$   
 $\text{autoria}(\text{dev2}, \text{file1}) = 0,4$
- Commit4: dev1, file1  $\longrightarrow$   $\text{autoria}(\text{dev1}, \text{file1}) = 1,9$   
 $\text{autoria}(\text{dev2}, \text{file1}) = 0,3$
- Commit5: dev2, file1  $\longrightarrow$   $\text{autoria}(\text{dev1}, \text{file1}) = 1,8$   
 $\text{autoria}(\text{dev2}, \text{file1}) = 0,8$
- Commit6: dev3, file1  $\longrightarrow$   $\text{autoria}(\text{dev1}, \text{file1}) = 1,7$   
 $\text{autoria}(\text{dev2}, \text{file1}) = 0,7$   
 $\text{autoria}(\text{dev3}, \text{file1}) = 0,1$

# Exercício

$$\text{Autoria}(d,f) = C + 0,5 * MD - 0,1 * MO$$

- Commit1: dev1, file1 —>
- Commit2: dev2, file1 —>
- Commit3: dev1, file1 —>
- Commit4: dev1, file1 —>
- Commit5: dev3, file1 —>
- Commit6: dev4, file1 —>

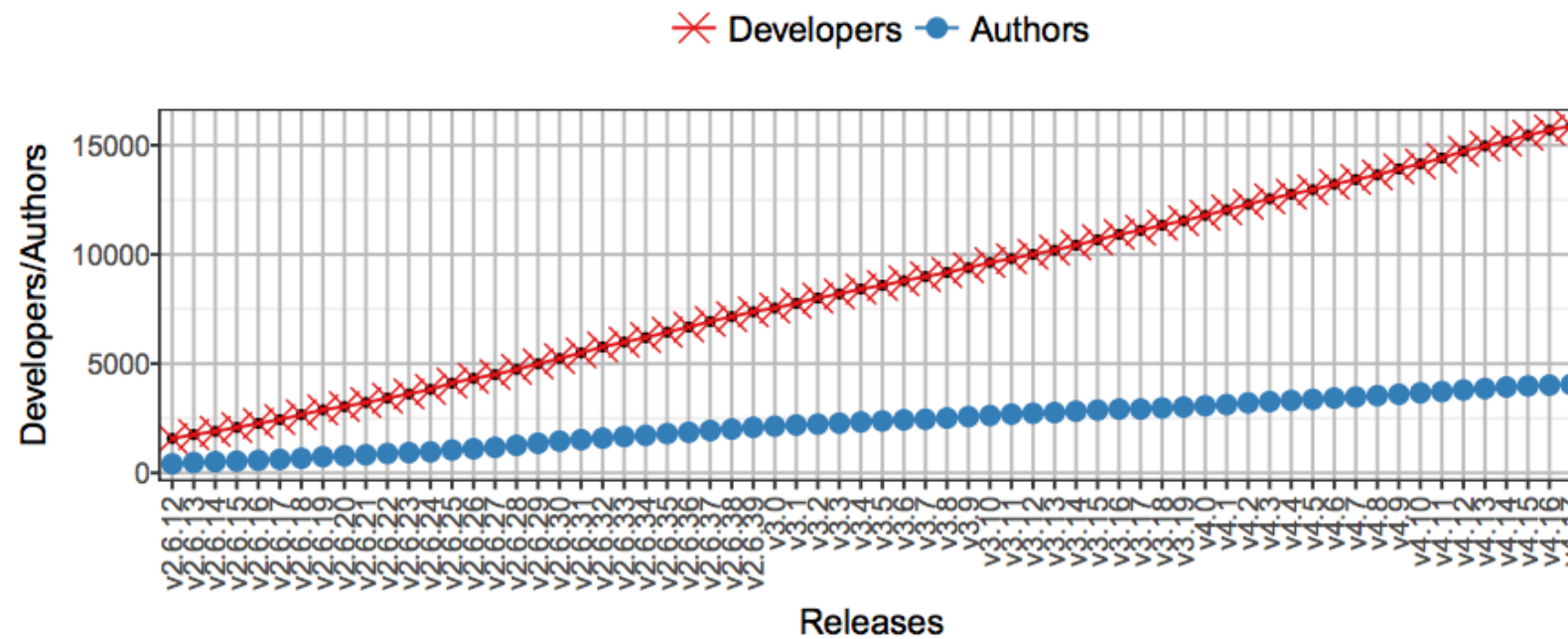


# Exercício

$$\text{Autoria}(d,f) = C + 0,5 * MD - 0,1 * MO$$

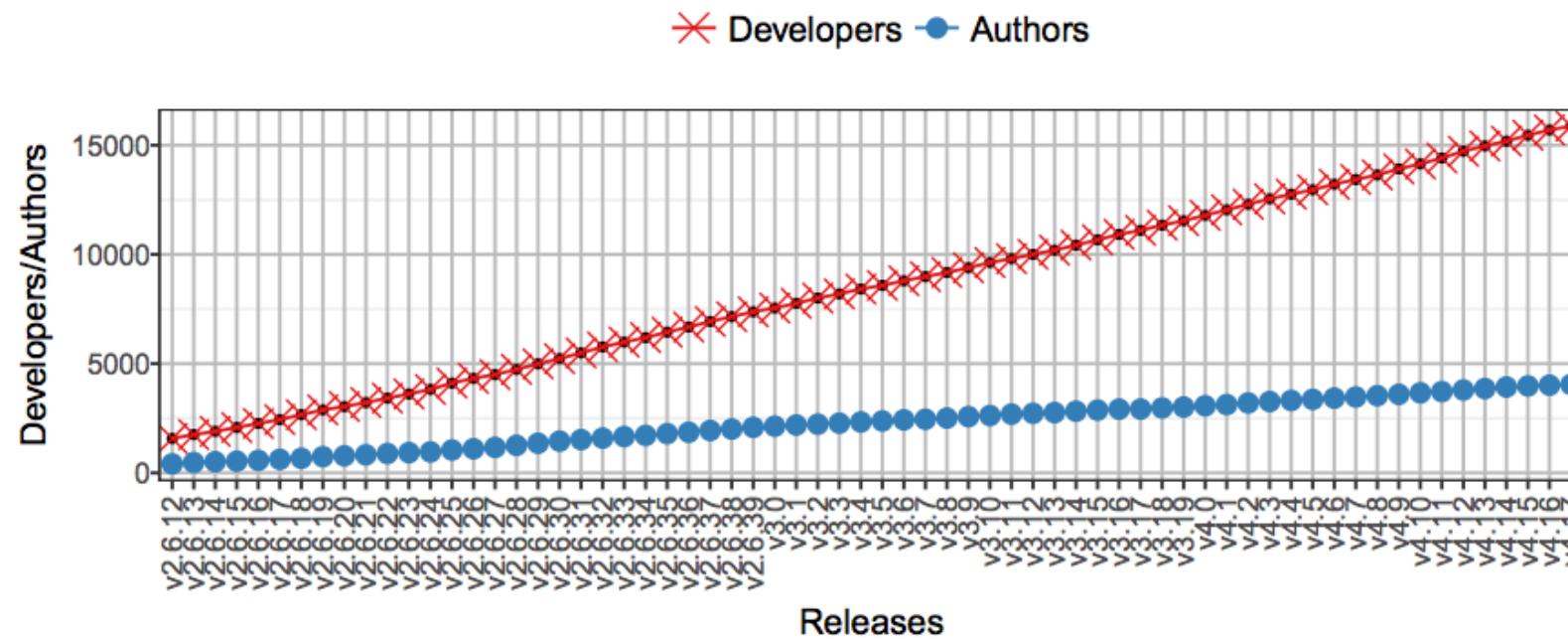
- Commit1: dev1, file1  $\longrightarrow$   $\text{autoria}(\text{dev1}, \text{file1}) = 1$
- Commit2: dev2, file1  $\longrightarrow$   $\text{autoria}(\text{dev1}, \text{file1}) = 0,9$   
 $\text{autoria}(\text{dev2}, \text{file1}) = 0,5$
- Commit3: dev1, file1  $\longrightarrow$   $\text{autoria}(\text{dev1}, \text{file1}) = 1,4$   
 $\text{autoria}(\text{dev2}, \text{file1}) = 0,4$
- Commit4: dev1, file1  $\longrightarrow$   $\text{autoria}(\text{dev1}, \text{file1}) = 1,9$   
 $\text{autoria}(\text{dev2}, \text{file1}) = 0,3$
- Commit5: dev3, file1  $\longrightarrow$   $\text{autoria}(\text{dev1}, \text{file1}) = 1,8$   
 $\text{autoria}(\text{dev2}, \text{file1}) = 0,2$   
 $\text{autoria}(\text{dev3}, \text{file1}) = 0,2$
- Commit6: dev4, file1  $\longrightarrow$   $\text{autoria}(\text{dev1}, \text{file1}) = 1,7$   
 $\text{autoria}(\text{dev2}, \text{file1}) = 0,1$   
 $\text{autoria}(\text{dev3}, \text{file1}) = 0,1$   
 $\text{autoria}(\text{dev4}, \text{file1}) = 0,1$

# Autores vs. Desenvolvedores (Linux)

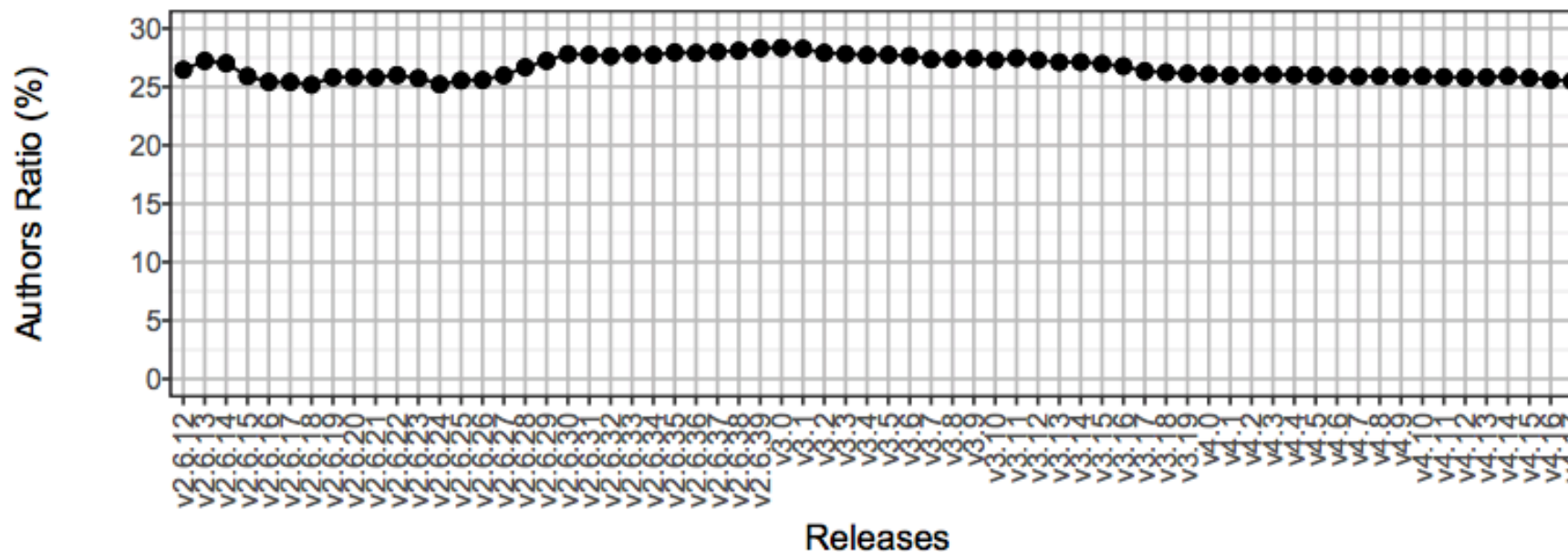


(a) Number of authors and developers over time

# Autores vs. Desenvolvedores (Linux)



(a) Number of authors and developers over time

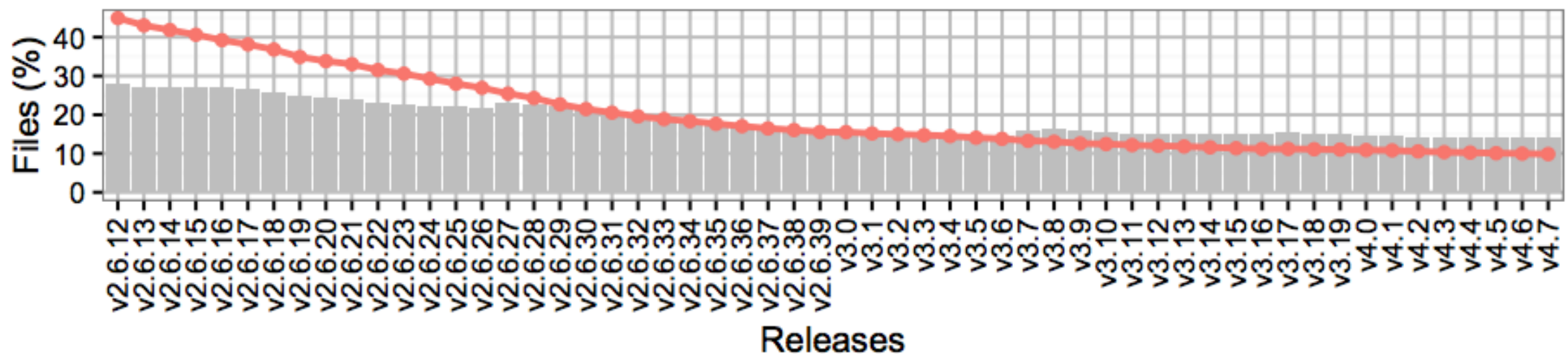


(b) Proportion of authors over time

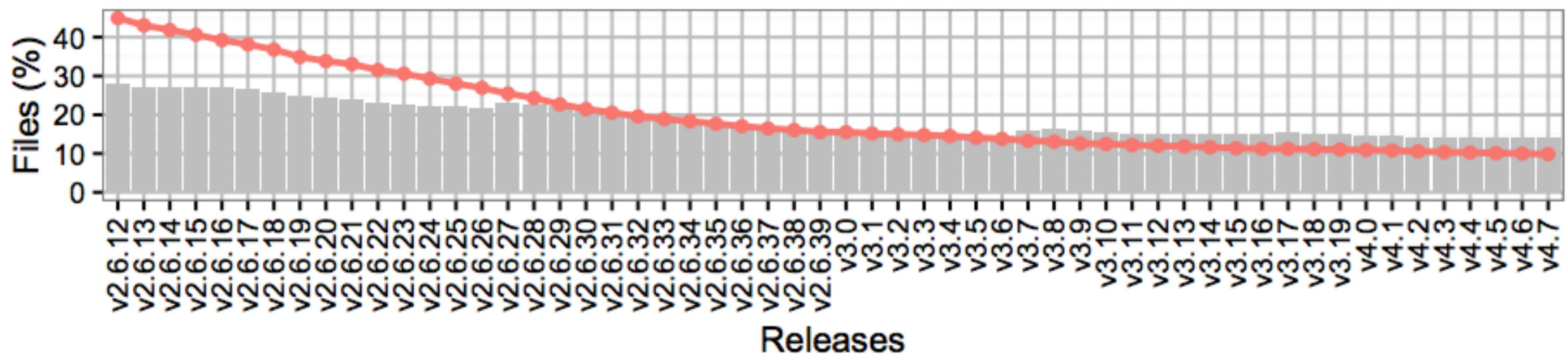
**Constante em  
25%**



# Arquivos dos top-10 autores



# Arquivos dos top-10 autores



**Linha vermelha: Linus Torvalds (45% —> 10%)**



# Truck Factor





“I just returned from a meeting in which the major objection to using Python was its dependence on Guido. They wanted to know if Python would survive if Guido disappeared. This is an important issue for businesses that may be considering the use of Python in a product.” — employee at the National Institute of Standards and Technology/USA, 1994



# Truck Factor

- “the number of people on your team that have to be hit by a truck (or quit) before the project is in serious trouble” [Williams & Kessler]
- Sistema com baixo Truck Factor apresenta forte dependência em membros específicos
- Caso esses membros deixem o projeto, existe grande risco do sistema ser afetado e em último caso descontinuado

# Algoritmo para calcular TF

---

**Algorithm 1:** AVL Algorithm

---

**Input:** TA (list of top-authors of a system S)

**Output:** tf (truck factor), TFSet (TF developers)

```
1 begin
2   tf  $\leftarrow$  0;
3   TFSet  $\leftarrow$   $\emptyset$ ;
4   TA  $\leftarrow$  list of top-authors;
5   while TA  $\neq$   $\emptyset$  do
6     dev  $\leftarrow$  head(TA);
7     remove-author(dev);
8     tf++;
9     TFSet  $\leftarrow$  TFSet + dev;
10    if rate-abandoned-files()  $\geq$  0.5 then
11      | break;
12    end
13    TA  $\leftarrow$  TA - dev;
14  end
15  return tf, TFSet;
16 end
```

---

System	TF	
torvalds/linux	57	<div></div>
fzaninotto/Faker	23	<div></div>
android/platform_frameworks_base	19	<div></div>
moment/moment	19	<div></div>
php/php-src	18	<div></div>
odoo/odoo	14	<div></div>
fog/fog	12	<div></div>
git/git	12	<div></div>
webscalesql/webscalesql-5.6	11	<div></div>
v8/v8	11	<div></div>
Seldaek/monolog	11	<div></div>
saltstack/salt	11	<div></div>
JetBrains/intellij-community	9	<div></div>
rails/rails	9	<div></div>
puppetlabs/puppet	9	<div></div>

