

SME0121 – Processos Estocásticos

Thomas Peron

thomas.peron@usp.br

tkdmperon.github.io

  @thomas_peron

Sala 3-250 B

Objetivos do curso

Introduzir aos conceitos e métodos de modelagem estocástica.

Ilustrar aplicações de processos estocásticos – *especialmente no contexto de CC.*

(Jupiter Web, 2023)

Critérios de avaliação

23/04/2024: Exemplos, **plantão de dúvidas** e revisão

25/04/2024: Prova 1 (P1)

11/06/2024: Exemplos, **plantão de dúvidas** e revisão

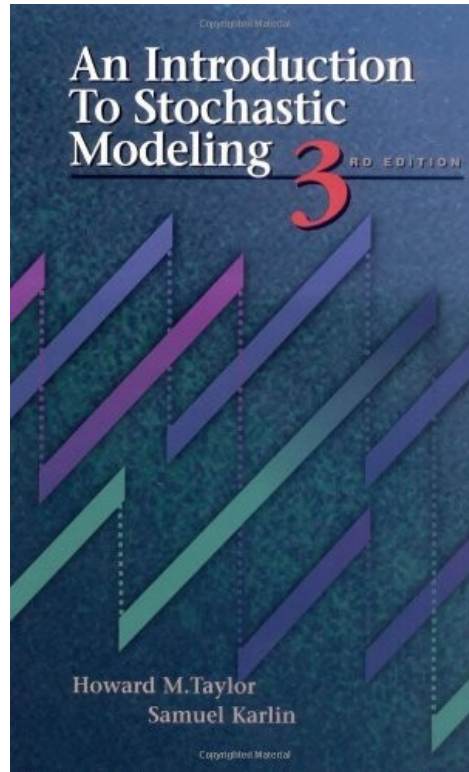
13/06/2024: Prova 2 (P2)

$$\text{Média final} = 0.8 \left[\frac{P1 + P2}{2} \right] + 0.2 \times \text{Média listas}$$

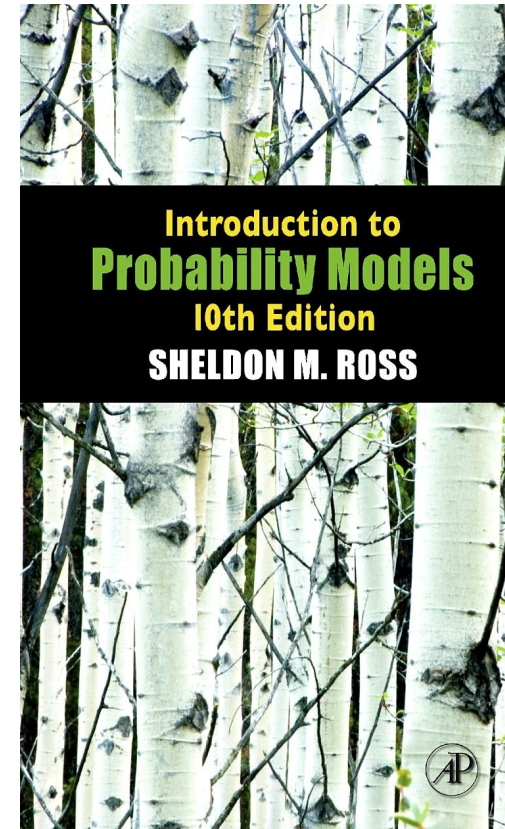
Critérios para a Recuperação: consulte o JúpiterWeb.

Não haverá prova substitutiva.

Bibliografia principal



Taylor & Karlin.
An Introduction to Stochastic Modeling
3rd edition



Sheldon M. Ross,
Introduction to Probability Models,
10th edition.



Sobre as listas de exercícios e estudos

Listas semanais ou quinzenais baseadas no que foi apresentado em aula.

Verifiquem o e-mail e o site da disciplina periodicamente!

Haverá exercícios teóricos e práticos.

Mantenham o ritmo constante na resolução dos exercícios.

Dúvidas: enviem e-mail com a sigla da disciplina (SME0121) no assunto.

Monitor: Vincenzo Bonasorte (vicenzop@usp.br)

Horário – monitoria: Quarta-feira 18h-19h.

Regras para as aulas

USP exige ao menos 70% de presença nas aulas.

Questionamentos sobre o assunto da aula podem ser feitos a qualquer momento.

Algumas regras para as provas

Material permitido durante as provas:

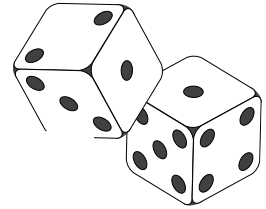
O óbvio: lápis, caneta, lapiseira, borracha, régua.

Calculadora **própria**.

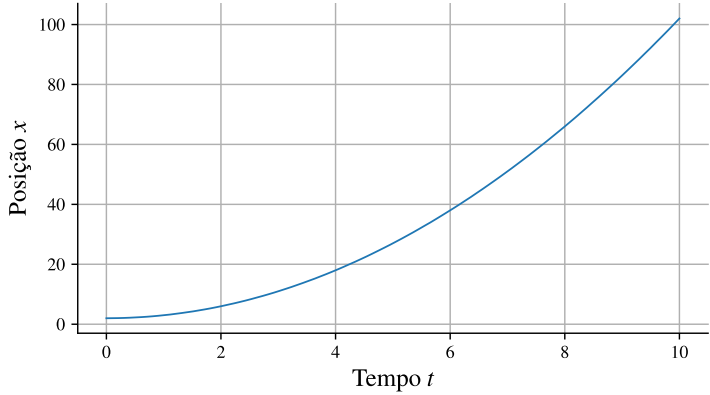
Resposta final à caneta.

Não é permitido o uso de celular.

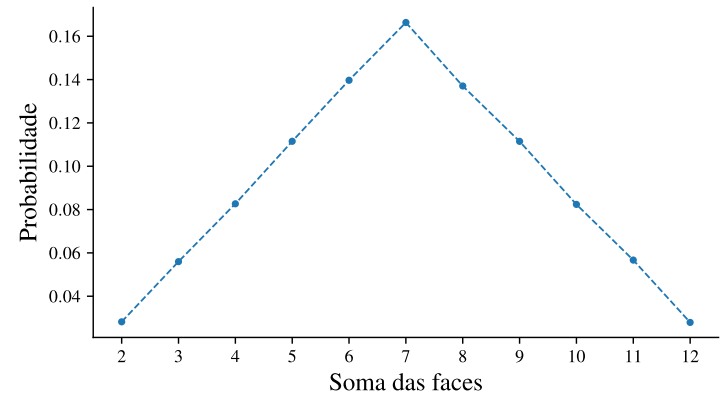
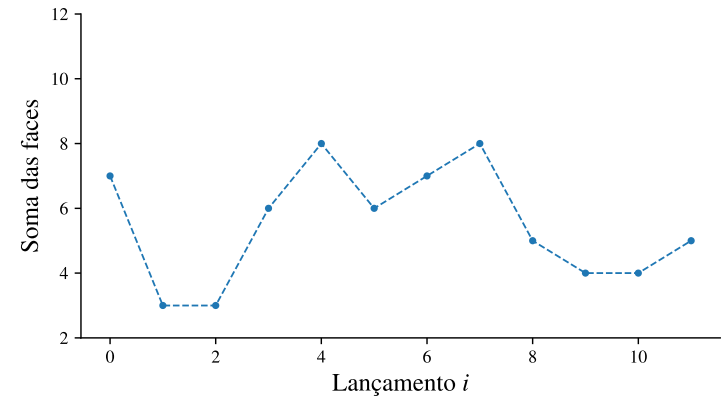
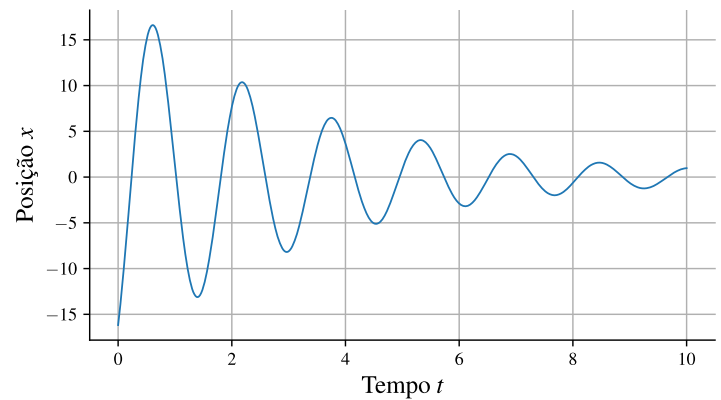
Experimentos determinísticos vs. estocásticos (probabilísticos)



$$\ddot{x} = \frac{F}{m}$$



$$\ddot{x} + \gamma \dot{x} + \omega_0^2 x = F(t)$$



Processos estocásticos

Def.: Família de variáveis aleatórias X_t , onde $t \in T$ é o *tempo*.

Conjunto T pode ser discreto: $T = \{0, 1, 2, \dots\}$

$\{X_1, X_2, X_3, \dots\} = \{\text{cara, coroa, coroa, ...}\}$



$\{X_1, X_2, X_3, \dots\} = \{3, 2, 5, \dots\}$



Conjunto T pode ser contínuo: $T = [0, \infty)$

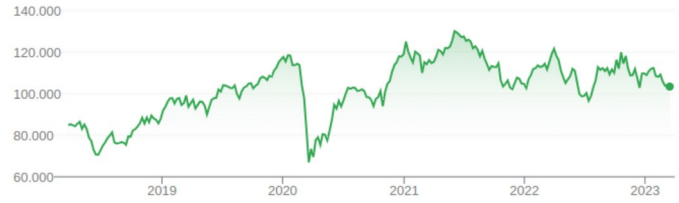
Resumo do mercado > Ibovespa

103.547,85

+18.661,36 (21,98%) ↑ últimos 5 anos

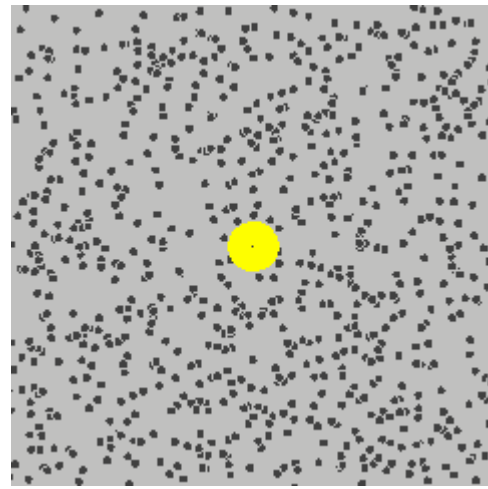
13 de mar., 15:30 BRT • Exoneração de responsabilidade

1 D | 5 D | 1 M | 6 M | YTD | 1 A | **5 A** | Máx

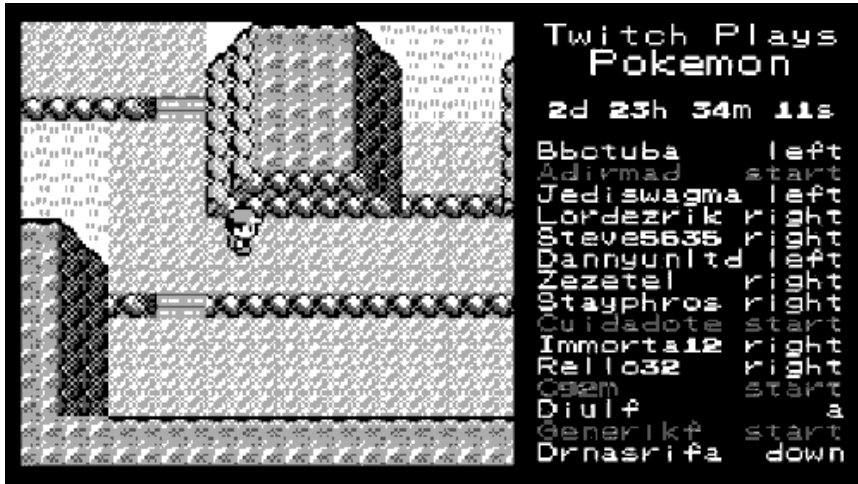


Abertura	103.607,98	Baixa	102.254,72	Alt 52 sem	121.628,22
Alta	103.906,78	Fech. ant.	103.618,20	Bai 52 sem	95.266,94

Movimento Browniano



Exemplos interessantes – Twitch plays Pokémon



Variável aleatória X_t : posição do Ash no tempo t .

$$X_{t+1} = X_t + \zeta_t$$

↘ direção resultante.
 $\zeta_t \in \{ \text{left}, \text{right}, \text{start}, \text{down} \}$

$$\begin{aligned} & \text{Prob}\{X_{t+1} = j | X_0 = i_0, \dots, X_{t-1} = i_{t-1}, X_t = i\} \\ &= \text{Prob}\{X_{t+1} = j | X_t = i\} \end{aligned}$$

Processo de Markov

Exemplos interessantes – Cadeias de Markov

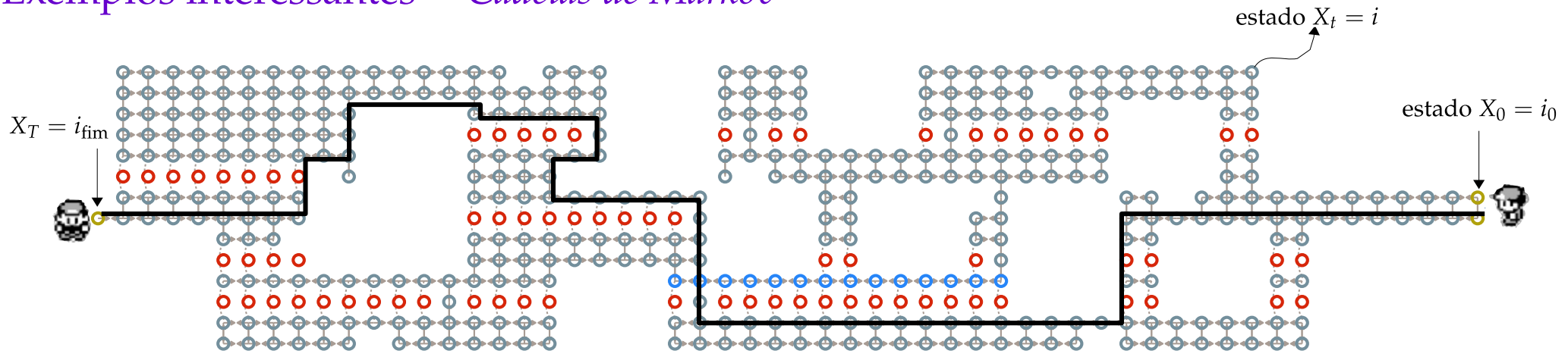
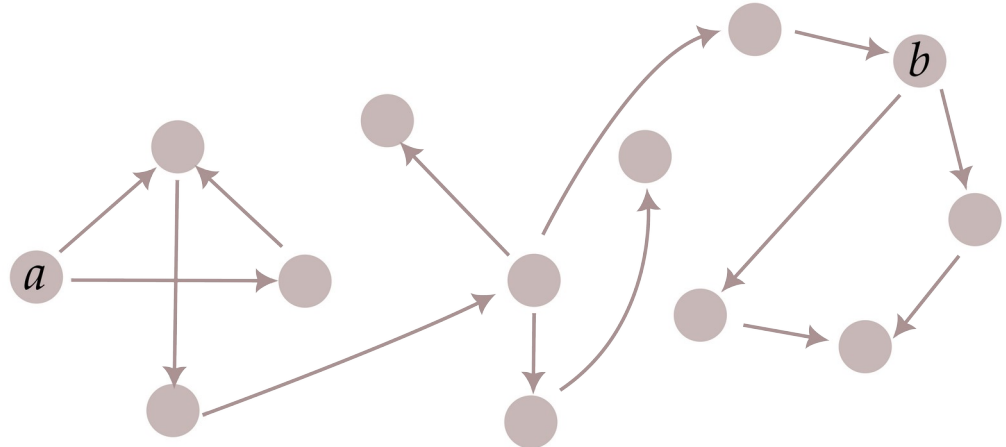


Figura: Aleta & Moreno (2019).

Quanto tempo levará para atingir o *estado absorvente* (fim do jogo)?



Exemplos interessantes – Page Rank

Vértices: páginas
Arestas: citações (links)

Figura criada com
graph-tool

Problema: Como ranquear páginas na web?

Caminhada aleatória!

p_j : Prob. de estar na página j

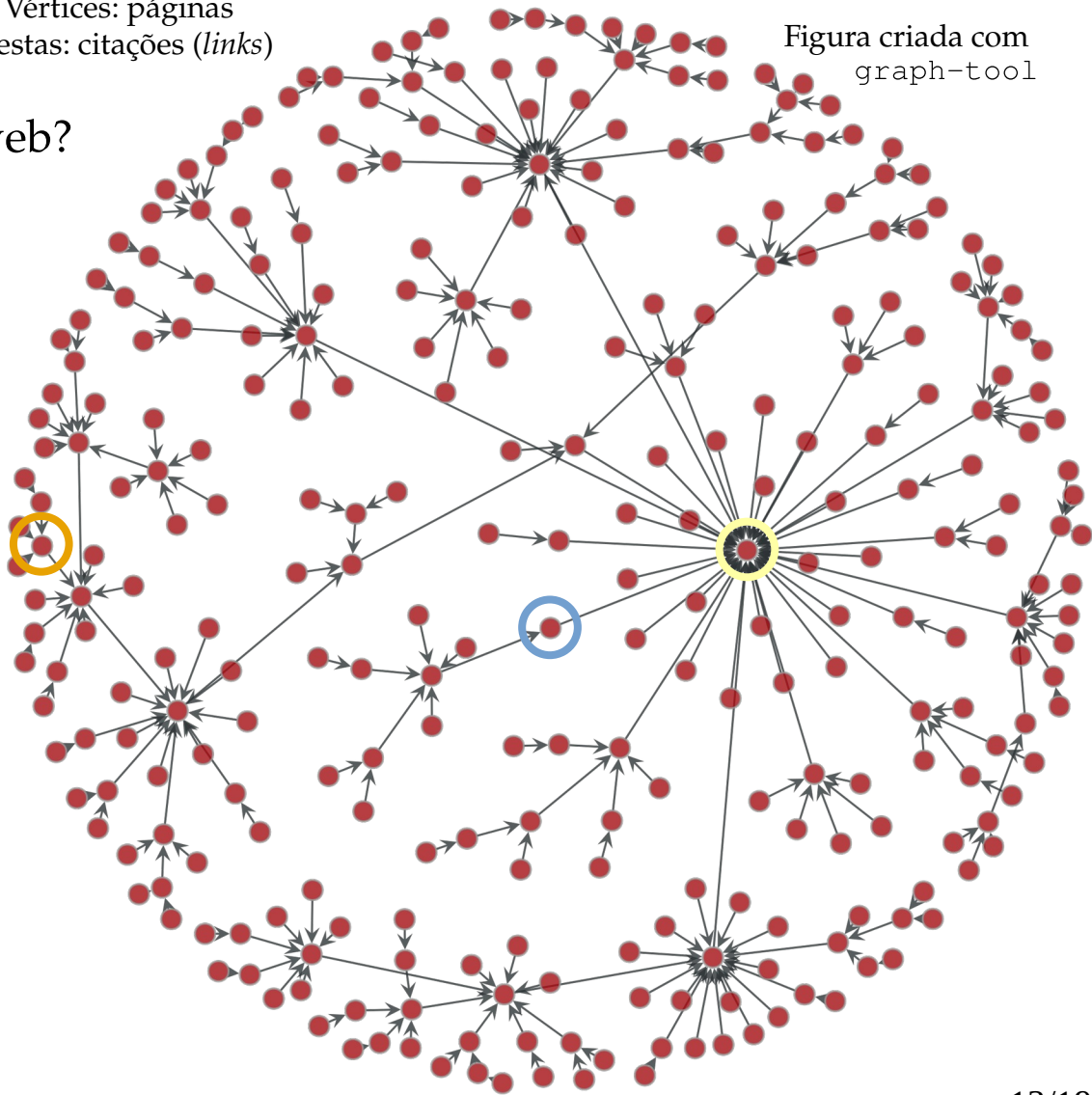
$$p_i = \sum_j \frac{A_{ij}}{k_{\text{out},j}} p_j \rightarrow \vec{p} = \mathbf{P}\vec{p}$$

A busca pode ficar presa!

Modificação:

$$p_i = \alpha \sum_j \frac{A_{ij}}{k_{\text{out},j}} p_j + (1 - \alpha) \frac{1}{n}$$

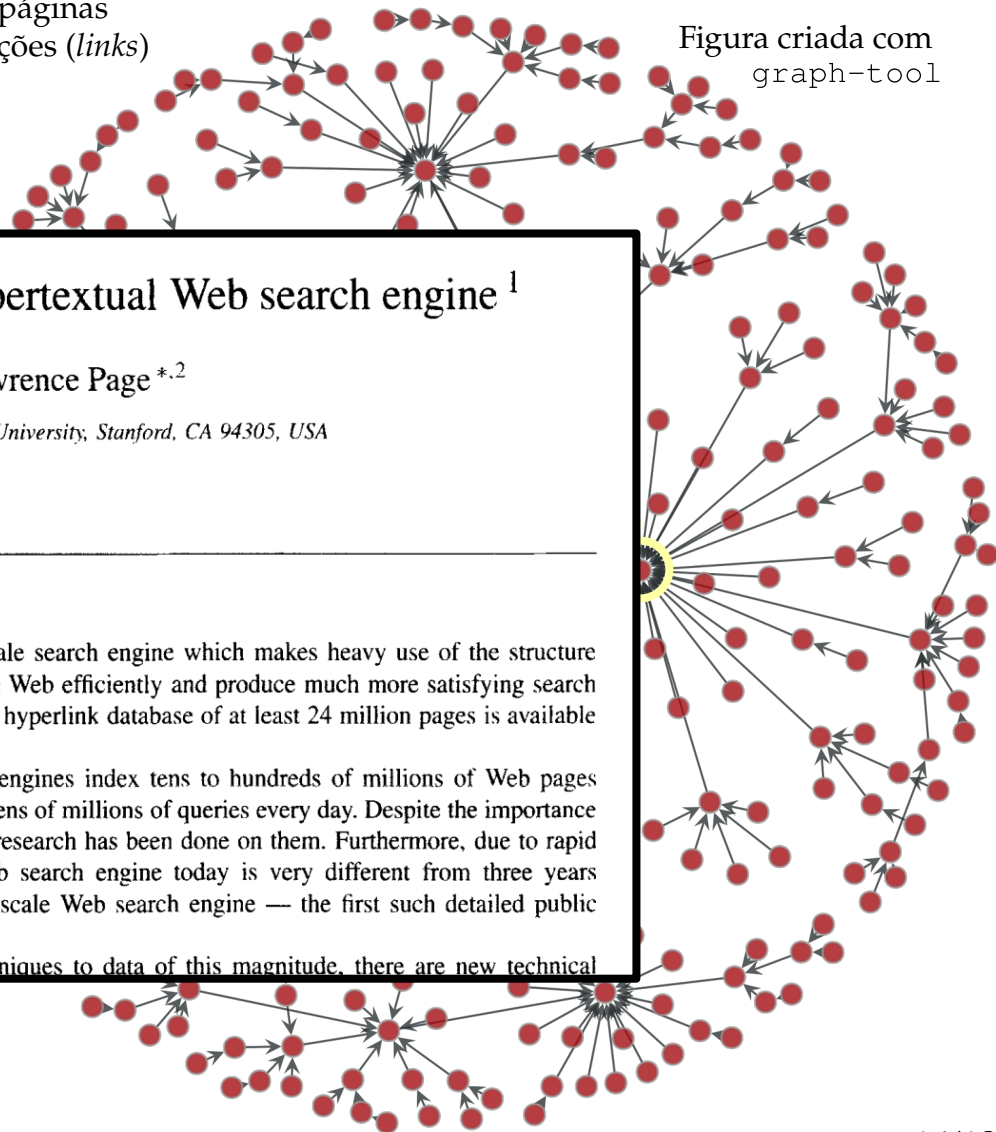
$\alpha < 1$, n : número de páginas.



Exemplos interessantes – Page Rank

Vértices: páginas
Arestas: citações (links)

Figura criada com
graph-tool



Problema: Como ranquear páginas na web?

Características:

p_j : Prob. de estar na página j

$$p_i = \sum_j \frac{A_{ij}}{k_{out,j}}$$

A busca é feita através de um algoritmo de busca em profundidade

Modificação:

$$p_i = \alpha \sum_j$$

$\alpha < 1$, n : número de páginas.

The anatomy of a large-scale hypertextual Web search engine ¹

Sergey Brin ², Lawrence Page ^{*,2}

Computer Science Department, Stanford University, Stanford, CA 94305, USA

Abstract

In this paper, we present Google, a prototype of a large-scale search engine which makes heavy use of the structure present in hypertext. Google is designed to crawl and index the Web efficiently and produce much more satisfying search results than existing systems. The prototype with a full text and hyperlink database of at least 24 million pages is available at <http://google.stanford.edu/>

To engineer a search engine is a challenging task. Search engines index tens to hundreds of millions of Web pages involving a comparable number of distinct terms. They answer tens of millions of queries every day. Despite the importance of large-scale search engines on the Web, very little academic research has been done on them. Furthermore, due to rapid advance in technology and Web proliferation, creating a Web search engine today is very different from three years ago. This paper provides an in-depth description of our large-scale Web search engine — the first such detailed public description we know of to date.

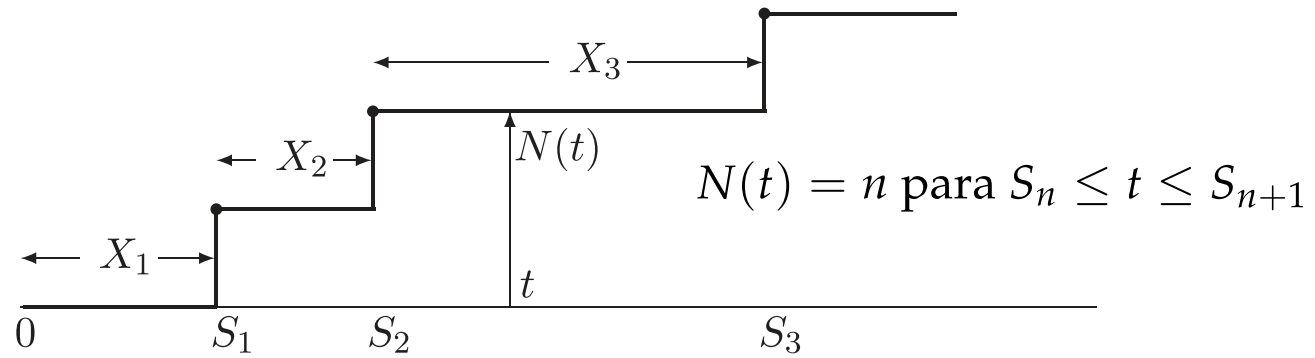
Apart from the problems of scaling traditional search techniques to data of this magnitude, there are new technical

Exemplos interessantes – Detecção de anomalias

	source_ip	datetime	username	success	failures	attempts	success_rate	failure_rate
0	1.138.149.116	2018-11-01 04:00:00	1	5	1	6	0.833333	0.166667
1	1.138.149.116	2018-11-05 18:00:00	1	1	0	1	1.000000	0.000000
2	1.138.149.116	2018-11-05 19:00:00	1	1	0	1	1.000000	0.000000
3	1.138.149.116	2018-11-06 03:00:00	1	2	0	2	1.000000	0.000000
4	1.138.149.116	2018-11-06 04:00:00	1	2	0	2	1.000000	0.000000

Como detectar ataques?

- Distribuição dos acessos.
- Acessos normais vs. maliciosos.
- Processos de Poisson.

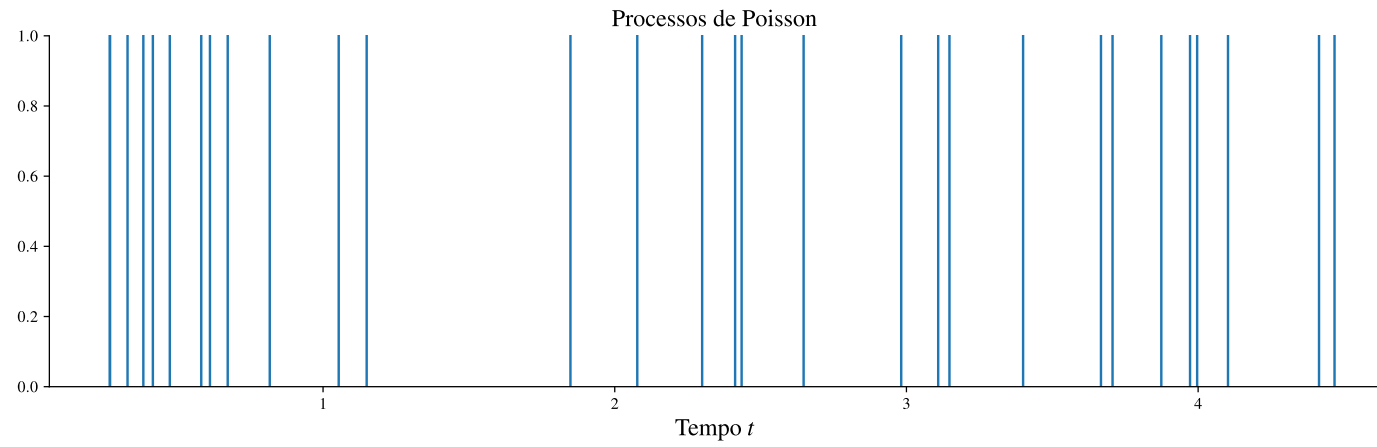


Exemplos interessantes – *Detecção de anomalias*

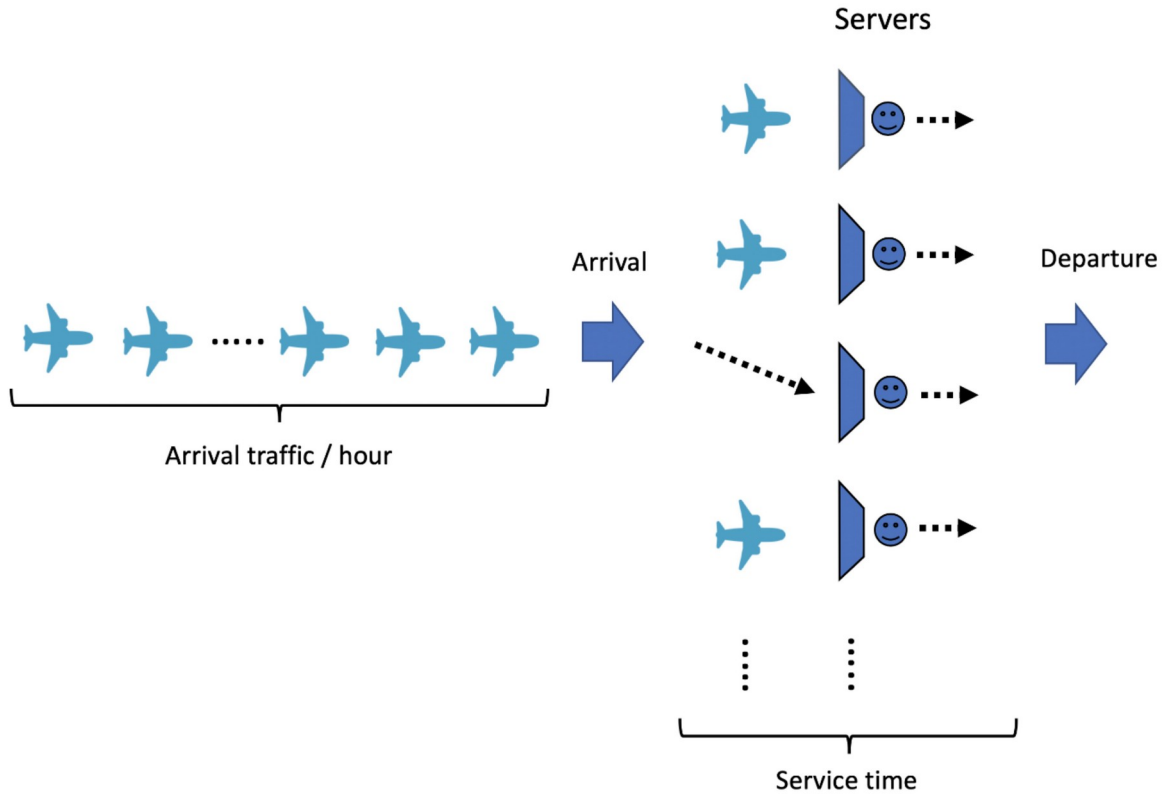
	source_ip	datetime	username	success	failures	attempts	success_rate	failure_rate
0	1.138.149.116	2018-11-01 04:00:00	1	5	1	6	0.833333	0.166667
1	1.138.149.116	2018-11-05 18:00:00	1	1	0	1	1.000000	0.000000
2	1.138.149.116	2018-11-05 19:00:00	1	1	0	1	1.000000	0.000000
3	1.138.149.116	2018-11-06 03:00:00	1	2	0	2	1.000000	0.000000
4	1.138.149.116	2018-11-06 04:00:00	1	2	0	2	1.000000	0.000000

Como detectar ataques?

- Distribuição dos acessos.
- Acessos normais vs. maliciosos.
- Processos de Poisson.



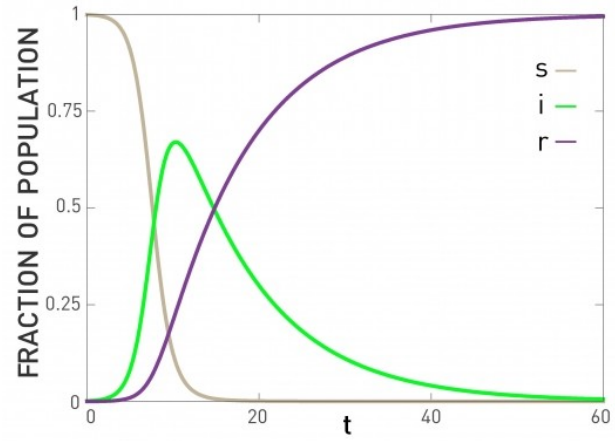
Exemplos interessantes – Filas



Propagação de epidemias



$$\frac{dS}{dt} = -\beta IS$$
$$\frac{dI}{dt} = -\mu I + \beta IS$$
$$\frac{dR}{dt} = \mu I$$



Programa resumido

1. Breve revisão: Teoria de probabilidades.
2. Cadeias de Markov em tempo discreto.
3. Processos de Poisson.
4. Cadeias de Markov em tempo contínuo.
5. Filas.
6. Simulação estocástica (Monte Carlo, Markov Chain Monte Carlo, ...) .