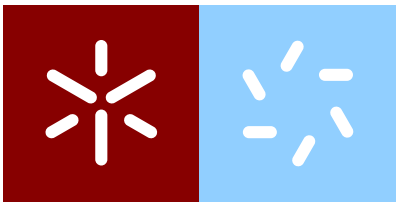
Universidade do Minho
Escola de Ciências

Célia Catarina Silva Ferreira

Análise do erro de estimação em Filtros de Bloom Lineares

Janeiro de 2017

Análise do erro de estimação em Filtros de Bloom Lineares
Célia Catarina Silva Ferreira
UMinho | 2017



Universidade do Minho
Escola de Ciências
Departamento de Matemática e Aplicações

Célia Catarina Silva Ferreira

Análise do erro de estimação em Filtros de Bloom Lineares

Dissertação de Mestrado
Mestrado em Estatística

Trabalho realizado sob a orientação de
Professora Doutora Raquel Menezes
Professor Doutor Carlos Baquero

Janeiro 2017

Declaração

Nome: Célia Catarina Silva Ferreira

Endereço eletrónico: celiacf64@gmail.com

Título da dissertação:
Análise do erro de estimação em Filtros de Bloom

Orientadores:
Professora Doutora Raquel Menezes
Professor Doutor Carlos Baquero

Ano de conclusão: 2017
Mestrado em Estatística

Universidade do Minho, ____ de _____ de 2017.

A autora: _____

"Deus quer, o homem sonha, a obra nasce."

Fernando Pessoa

AGRADECIMENTOS

A realização deste trabalho só foi possível graças a várias pessoas, a quem gostaria de deixar o meu sincero agradecimento por todo o apoio, confiança e colaboração ao longo deste período.

Aos meus orientadores, Professora Doutora Raquel Menezes e Professor Doutor Carlos Baquero, pela forma interessada e disponível com que acompanharam o desenvolvimento de todo o trabalho e sobretudo pelo conhecimento transmitido.

À minha família, de forma particular aos meus pais, a quem devo tudo aquilo que sou e que tenho.

Ao António Coutinho, pela inestimável ajuda que me prestou e pela imensa disponibilidade que demonstrou durante o meu percurso.

A todos os meus amigos, pela companhia, apoio e bons conselhos, em particular, à Patrícia Carvalho, ao Luís Marques, ao João Magalhães, e à Charlotte Bradley.

A todos o meu muito obrigado!

RESUMO

O registo preciso de grandes volumes de dados requer uma, proporcionalmente, grande quantidade de memória. Uma forma de reduzir esta necessidade passa por fazer um registo probabilístico com recurso à técnica de Filtros de Bloom. Esta técnica permite detetar, com uma determinada probabilidade de erro por falsos positivos, a pertença de um elemento a um conjunto. Pretende-se, nos Filtros de Bloom Lineares, generalizar esta técnica para associar um valor numérico a cada elemento e permitir a consulta desse valor. Torna-se assim possível a sua aplicação a situações onde se pretende qualificar numericamente os valores registados, como por exemplo na atribuição de um grau de confiança numérico a uma observação registada.

Neste projeto é feito um estudo analítico do erro esperado na consulta, em função da distribuição dos valores inseridos, nomeadamente para as distribuições: Uniforme, Exponencial ou Normal. Este estudo envolve a aplicação da teoria de valores extremos, usando a função generalizada de valores extremos e a função densidade de mínimos de máximos deduzida.

Com a ajuda do *software* estatístico R, efetuaram-se estudos de simulação do funcionamento dos Filtros de Bloom Lineares. Comparando o resultado dessas simulações face ao estudo analítico baseado na teoria de valores extremos, concluiu-se com ótimos resultados que o erro esperado é reduzido, para enchimentos convencionais do filtro, e que há um bom ajuste entre as funções teóricas e os resultados experimentais.

Palavras-chave: Filtros de Bloom Lineares, Teoria dos Valores Extremos, mínimos, máximos, mínimos de máximos, grau de confiança, falsos positivos.

ABSTRACT

The precise recording of large volumes of data requires a proportionally big amount of memory. Memory usage can be reduced by using Bloom Filters as a probabilistic representation of the data to be stored. This technique allows detecting, with a given probability for false positives, if an element belongs, or not, to a set. In an extension of the technique, Linear Bloom Filters, set membership is generalized in order to associate a numerical value to each element and allow the query to retrieve that value. This permits the application to settings where one intends to qualify numerically the registered values, for example in the attribution of a numeric quality degree to a registered observation.

In this project the analytic study of the query's expected error is done, depending on the distribution of the inserted values, for the Uniform, Exponential and Normal distributions. This study applies the extreme values theory, using the generalized function of extreme values and the derived density function of maxima minima.

With the help of R statistical software, several simulation studies of the operation of Bloom Linear Filters were made. By comparing the result of the simulations with the analytic study based on the extreme values theory, it was possible to conclude with good confidence that the expected error is small, for conventional fillings of the filter, and that there exists a good adjustment between the theoretical functions and the experimental results.

Keywords: Linear Bloom Filters, Extremes Value Theory, minima, maxima, maxima minima, quality degree, false positive.

CONTEÚDO

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE ABREVIAÇÕES

FB - Filtro de Bloom

FBs - Filtros de Bloom

FBL - Filtro de Bloom Linear

FBLs - Filtros de Bloom Lineares

EVT - *Extreme Value Theory* (Teoria de valores extremos)

iid - independentes e identicamente distribuídas

GEV - *Generalized Extreme Value* (Valores extremos generalizados)

GEVD - *Generalized Extreme Value Distribuition* (Distribuição de valores extremos generalizados)

EQM - Erro Quadrático Médio

MEA - Média dos Erros Absolutos

1

INTRODUÇÃO

Os Filtros de Bloom (FBs) são estruturas de dados probabilísticas que têm como principal vantagem a eficiência em termos de espaço no armazenamento de dados. Apresentados por Burton Bloom em 1970, inicialmente, os FBs foram utilizados em aplicações ligadas a bases de dados. Na área de redes de computadores, os FBs não criaram grande impacto até que em 1995, devido à popularização da Internet, começaram a ser usados em diversas aplicações como redes peer-to-peer, medição de dados e roteamento de pacotes.

O Filtro de Bloom (FB) tradicional é uma estrutura de dados que permite o armazenamento de informação de forma probabilística. A eficiência desta estrutura baseia-se principalmente na gestão do espaço de armazenamento de dados (Broder and Mitzenmacher [2004]).

Os FBs permitem a realização de duas operações, sendo a primeira armazenar os elementos no filtro e a segunda verificar a existência de determinado elemento no filtro. Estas operações denominam-se, respetivamente, de inserção (*insert*) e consulta (*query*).

O FB pode ser representado por um simples vetor, onde cada posição é iniciada com o valor zero, sendo que só podem ser tomados os valores $\{0, 1\}$. Na inserção dos elementos, aplica-se uma função *Hash* a cada elemento, que atribui um certo número de posições onde se substitui o

valor da posição atribuída por 1. Na segunda operação, para consultar se determinado elemento está no filtro, aplica-se também a função Hash, que atribui novamente posições. Se em todas essas posições o valor observado for 1, há uma forte probabilidade do elemento pertencer ao filtro; no caso de numa das posições o valor ser 0, tem-se a certeza de que o elemento não está presente.

Com o objetivo de melhorar a ultilização dos FBs, foram estudadas maneiras de manipular as propriedades destes. Assim, ao longo do tempo, surgiram diversas variações, tais como os *Counting Bloom Filters*, os *Bloomier Filters*, os *Scalable Bloom Filters* e mais recentemente os Filtros de Bloom Lineares (FBLs), sendo sobre estes últimos que recai o estudo apresentado.

Os *Counting Bloom Filters*, apresentados em Fan et al. [2000], para além das funcionalidades dos FBs, também permitem apagar elementos, utilizando um conjunto de marcadores que permitem controlar o número de elementos inseridos em cada posição.

Os *Bloomier Filters* , propostos em Chazelle et al. [2004], extendem o Filtro de Bloom para lidar com situações em que cada elemento do conjunto está associado a um valor atribuído. O *Bloomier Filter* fornece um valor apropriado através de uma função para qualquer elemento do conjunto e retorna um valor correspondente a 'indefinido' para os elementos não pertencentes ao conjunto.

Os *Scalable Bloom Filters*, apresentados em Almeida et al. [2007] permitem ajustar a dimensão do filtro à medida que os elementos são inseridos, adaptando dinamicamente a qualidade do filtro, sendo que é desconhecido o número de elementos a serem inseridos.

Por fim, surgem os *Filtros de Bloom Lineares* (FBLs), apresentados em Lima et al. [2015]. São semelhantes aos FBs, sendo que, neste caso, para

além de se inserir e consultar elementos, agregam-se características a cada elemento na inserção.

O assunto da presente dissertação surge da intenção de apresentar um estudo analítico dos erros de estimação nos FBLs com o intuito de complementar o trabalho que tem vindo a ser desenvolvido no âmbito da tese de doutoramento em engenharia de informática, parcialmente apresentado em Lima et al. [2015].

Assim, o presente trabalho terá como principais objetivos o estudo de conceitos básicos da teoria de valores extremos, passando pelo estudo da distribuição de valores extremos para dados independentes. Desenvolver-se-à o estudo de simulações, recorrendo a bibliotecas disponíveis no *software* estatístico *R*, onde se aplicarão os conhecimentos adquiridos ao cálculo de valores extremos e se fará a comparação dos resultados obtidos, adotando para isso diferentes distribuições.

Estrutura da tese

Este estudo contará com cinco capítulos principais para além da introdução.

O Capítulo 2, intitulado *Enquadramento teórico*, divide-se em duas secções, que abordam individualmente a temática dos FBs e da teoria dos valores extremos. Na primeira secção, apresenta-se a estrutura do FB e o seu funcionamento, abordando-se as operações inserção e consulta, e introduz-se a ideia de falso positivo. É apresentada a função *Hash* e finaliza-se com um exemplo, que engloba todo o conteúdo abordado até então, e com o apuramento das limitações do método, o que permite fazer uma ponte para o Capítulo 3. Na segunda secção é introduzida a teoria dos valores extremos, abordando-se o principal teorema

que incide nos extremos máximos, o Teorema de Fisher-Tippett, sendo apresentada a função de distribuição GEV (Generalized Extreme Value). Como consequência imediata do teorema anterior e da necessidade de um enquadramento mais preciso, surge o Teorema dos tipos extremais para mínimos.

O Capítulo 3, nomeado *Filtros de Bloom Lineares*, apresenta os FBLs, referindo as alterações em relação aos FBs. Definem-se os métodos de inserção e consulta apresentando ao fim de cada um deles um exemplo.

No Capítulo 4, chamado *Resultados teóricos*, trata-se do estudo da teoria dos valores extremos direcionado ao tema em estudo, os FBLs. Começa-se pela dedução das funções distribuições dos mínimos e dos máximos, cuja combinação culmina na função distribuição dos mínimos dos máximos. Posteriormente, abordam-se as três distribuições que serão alvo do estudo, apresentando as respetivas funções densidade de probabilidade e distribuição e representações gráficas.

O Capítulo 5, com o título *Simulações*, é o ponto central desta dissertação. Inicia-se com a predefinição dos parâmetros usados nas simulações e debruça-se, de seguida, na apresentação das simulações dos métodos associados aos FBLs, já referidas anteriormente. Em cada um dos métodos, descrevem-se as etapas do procedimento. A organização destes subcapítulos consiste na apresentação, para cada uma das distribuições, do código elaborado com recurso ao *software* estatístico *R* (versão 3.3.2), acompanhado por uma representação gráfica de comparação das funções de densidade teórica, de estimação tipo-núcleo, e GEV estimada e complementado por uma tabela demonstrativa de algumas estatísticas descritivas dos parâmetros GEV. O capítulo é concluído por uma análise comparativa das diferentes distribuições, bem como uma análise da distorção de valores inseridos e consultados.

No Capítulo 6, *Conclusões*, faz-se um balanço geral do estudo. Tiram-se as principais conclusões, confrontando os resultados com os objetivos propostos na introdução.

2

ENQUADRAMENTO TEÓRICO

2.1 FILTROS DE BLOOM

A estrutura dos FBs tradicionais é muito interessante, mas tem algumas limitações, sendo a mais relevante o facto de simplesmente permitir inserir e, posteriormente, consultar a sua existência. A verdadeira existência do elemento no filtro pode ser posta em causa devido à possibilidade da existência dos chamados falsos positivos. Os falsos positivos surgem pelo facto do filtro funcionar de uma forma probabilística, o que possibilita, na fase de consulta, a obtenção de uma resposta positiva quanto à existência de certo elemento, ainda que na realidade ele não exista.

Para todo o estudo será usada a seguinte notação

- m: dimensão do filtro;
- n: número de elementos inseridos;
- k: número de posições a serem atribuídas pela função Hash;
- p: probabilidade de determinada posição não ser 1;

A probabilidade de falsos positivos é dada por $f = p^k$.
A probabilidade de uma determinada posição ser diferente de 1, p, pode ser calculada da seguinte forma Bose et al. [2008]

$$p = 1 - \left(1 - \frac{1}{m}\right)^{nk} \approx 1 - e^{-\frac{n}{m}k} \tag{1}$$

O valor de k, que minimiza a probabilidade de falsos positivos, pode ser calculado usando a seguinte fórmula Bose et al. [2008]

$$k = \frac{m}{n} ln2 \tag{2}$$

Assim, por (1) e (2), tem-se que a probabilidade de falsos positivos, f, é dada por

$$\begin{aligned} f = p^k &\approx \left(1 - e^{-\frac{n}{m}k}\right)^k = \\ &= \left(1 - e^{-\frac{n}{m}\frac{m}{n}ln2}\right)^k = \\ &= \left(1 - e^{-ln2}\right)^k = \\ &= \left(1 - e^{ln\frac{1}{2}}\right)^k = \\ &= \left(1 - \frac{1}{2}\right)^k = \\ &= \left(\frac{1}{2}\right)^k \end{aligned} \tag{3}$$

Na prática, a equação (2) é aproximada a um valor inteiro, pois k representa o número de posições que a função Hash atribui ao elemento. A função de probabilidade de falsos positivos pode ser obtida em função

do tamanho do filtro e do número de elementos a inserir, da seguinte forma

$$f \approx \left(\frac{1}{2}\right)^{k} = \left(\frac{1}{2}\right)^{\frac{m}{n}ln2} \approx (0.6185)^{\frac{m}{n}} \tag{4}$$

2.1.1 *Função Hash*

A função Hash é um algoritmo que mapeia dados de comprimento variável para dados de comprimento fixo. As funções Hash são largamente utilizadas na busca de elementos em bases de dados, bem como na verificação da sua existência e no seu armazenamento. O funcionamento baseia-se na construção de índices.
Neste estudo, é aplicada a função Hash a um número n de elementos. Cada função Hash retribui um número k de posições relativas ao elemento a que foi aplicada, às quais serão indexados valores indicativos da presença do respetivo elemento no vetor. Para tal, a sequência de *bits* produzida pela função Hash, quando aplicada a um dado elemento, é manipulada no sentido de se obter k coordenadas para o vetor de dimensão m (na prática m deve ser potência de 2).

Esta função tem como principal propriedade ser unidirecional, isto é, a função Hash não é invertível. Na prática, ser unidirecional representa que não é possível recuperar o elemento a partir dos valores dados pela função. Isto ocorre pois, aplicando a função Hash a diversos elementos, as posições atribuídas podem colidir. Outra propriedade importante é que as funções têm de ser recorrentes, isto é, sempre que um mesmo elemento for avaliado, deve sempre retornar os mesmos valores. Rogaway

and Shrimpton [2004]

O seguinte exemplo pretende ilustrar a funcionalidade de um FB, assim como mostrar algumas das suas limitações.

Exemplo 2.1. *Considere-se um filtro de tamanho $m = 11$, o número de elementos a inserir $n = 3$ e sejam os elementos $\{a, b, c\}$. Tem-se $k = 3$, isto é, a função Hash aplicada a cada elemento a inserir devolve 3 posições. Aplicando a função Hash a cada elemento, obtêm-se as seguintes posições:*

- *Hash(a)={1,3,7}*
- *Hash(b)={3,7,10}*
- *Hash(c)={4,7,10}*

Figura 1.: Inserção de elementos no Filtro de Bloom

1	2	3	4	5	6	7	8	9	10	11
1	0	1	1	0	0	1	0	0	1	0

Na Figura 1 , observa-se o FB após a inserção dos elementos, sendo que nas posições preenchidas com o valor 0, nenhum elemento foi inserido, enquanto que o valor 1 indica que pelo menos um elemento foi inserido.

Supõe-se agora que se quer consultar se o elemento d existe. Aplicando a função Hash ao elemento d, obteve-se $Hash(d) = \{1, 4, 10\}$. Observando na Figura 1, as três posições mostram o valor 1, o que leva a crer que o elemento d existe. Na realidade, sabe-se que foram inseridos no FB unicamente os elementos $\{a, b, c\}$, portanto considera-se o elemento d um falso positivo.

Dada a densidade do filtro, caso sejam inseridos mais elementos, o filtro tende a saturar, o que levaria ao aumento da taxa de falsos positivos. No caso do filtro ficar totalmente saturado todas as posições seriam

preenchidas com o valor 1 e o FB acabaria por deixar de ter utilidade. Estudos anteriores, Rhea and Kubiatowicz [2002], exploram este efeito, de modo a ter-se um equilíbrio entre a eficiência na gestão do espaço de armazenamento e a precisão de respostas, concluindo que o número de posições com o valor 1 deve ser cerca de metade da dimensão do filtro.

2.2 TEORIA DOS VALORES EXTREMOS

Como será estudado no próximo capítulo, os FBLs funcionam de uma forma ligeiramente diferente dos FBs pelo que será necessário usar funções de máximos e de mínimos. Esta necessidade leva ao estudo da teoria de valores extremos.

A teoria dos valores extremos (do inglês, Extreme Value Theory (EVT)) é um ramo probabilista de suporte à Estatística que é usado nas situações em que os dados são inexistentes ou, se existem, são raros ou extremos. A EVT ajuda a descrever e quantificar o comportamento desses acontecimentos, procurando estimar uma distribuição limite para os extremos, mínimos ou máximos de uma amostra composta por variáveis aleatórias independentes e identicamente distribuídas (iid). Um dos teoremas mais importantes da EVT é o teorema de Fischer-Tippett, que irá incidir sobre os extremos máximos (ver por exemplo Reis and Thomas [2007]).

Teorema 2.2.1 (Teorema de Fisher-Tippett)**.**

Considere-se $M_n := max\left(X_1, X_2, ..., X_n\right)$ *. Sejam duas sucessões reais* $a_n > 0$ *e* b_n*, tais que*

$$\lim_{n\to\infty} P\left[\frac{M_n - b_n}{a_n} \leq x\right] = \lim_{n\to\infty} F^n\left(a_n x + b_n\right) = G\left(x\right)$$

para alguma função de densidade G não degenerada, admite-se que G é do mesmo tipo de uma das seguintes distribuições:

Tipo I:

$$G(x) = \Lambda(x) = exp(-exp(-x)), \; x \in \mathbb{R} \tag{5}$$

Tipo II:

$$G(x|\alpha) = \Phi_\alpha(x) = \begin{cases} 0, & x \leq 0 \\ exp(-x^{-\alpha}), & x < 0, \alpha > 0 \end{cases} \tag{6}$$

Tipo III:

$$G(x|\alpha) = \Psi_\alpha(x) = \begin{cases} exp(-(-x^\alpha)), & x < 0, \alpha > 0 \\ 1, & x \geq 0 \end{cases} \tag{7}$$

As funções (5), (6) e (7) são chamadas de função Gumbel, Fréchet e Weibull, respetivamente.

Aliando os três tipos apresentados anteriormente, este teorema pode ser resumido pela seguinte função

$$G(x|y) = G_\gamma(x) = \begin{cases} exp\left(\left(-(1+\gamma x)^{-\frac{1}{\gamma}}\right)\right), & 1+\gamma x > 0, \gamma \neq 0 \\ exp(-exp(-x)), & x \in \mathbb{R}, \gamma = 0 \end{cases} \tag{8}$$

onde γ representa o parâmetro de forma (*shape*). Ou ainda uma versão mais geral, *generalized extreme value distribuition* (GEVD), introduzindo os parâmetros de localização (*location*) ($\lambda \in \mathbb{R}$) e de escala (*scale*) ($\delta > 0$).

$$\begin{aligned} G_\gamma(x|\lambda,\delta) &= G_\gamma\left(\frac{x-\lambda}{\delta}\right) = \\ &= \begin{cases} exp\left(-\left(1+\gamma\left(\frac{x-\lambda}{\delta}\right)\right)^{-\frac{1}{\gamma}}\right), & 1+\gamma\left(\frac{x-\lambda}{\delta}\right) > 0, \gamma \neq 0 \\ exp\left(-exp\left(-\left(\frac{x-\lambda}{\delta}\right)\right)\right), & x \in \mathbb{R}, \gamma = 0 \end{cases} \end{aligned} \tag{9}$$

Na continuação do estudo, os parâmetros γ, λ e δ serão denominados por parâmetros *generalized extreme value* (GEV).

Através do sinal do parâmetro *shape*, γ, pode-se determinar qual a distribuição a ser tratada. Isto é, quando $\gamma < 0$, está-se perante a distribuição Weibull; quando $\gamma > 0$, assume-se a distribuição Fréchet; no caso de $\gamma = 0$, a GEVD é interpretada como o limite quando $\gamma \to 0$, o que corresponde a uma Gumbel.

Demonstram-se, de seguida, os três casos.

Começa-se pelo caso $\gamma > 0$. Na expressão (9), considere-se que $\lambda = 1$, $\delta = \gamma$ e $\gamma = \frac{1}{\alpha}$, com $\alpha > 0$

$$\begin{aligned} G_\gamma(x|1,\gamma) =& G_\gamma\left(\frac{x-1}{\gamma}\right) = \\ =& exp\left(-\left(1+\gamma\left(\frac{x-1}{\gamma}\right)\right)^{-\frac{1}{\gamma}}\right) = \\ =& exp\left(-(1+x-x)^{-\frac{1}{\gamma}}\right) = \\ =& exp\left(-x^{-\frac{1}{\gamma}}\right) = \\ =& exp\left(-x^{-\alpha}\right) = \\ =& \Phi_\alpha(x) \end{aligned}$$

□

Caso $\gamma < 0$. Demonstra-se analogamente ao caso anterior, tomando como valores $\lambda = 1, \delta = -\gamma$ e $\gamma = -\frac{1}{\alpha}$, com $\alpha > 0$

$$
\begin{aligned}
G_\gamma\left(x|-1,-\gamma\right) = & = G_\gamma\left(\frac{x+1}{-\gamma}\right) = \\
& = exp\left(-\left(1-\gamma\left(\frac{x+1}{\gamma}\right)\right)^{-\frac{1}{\gamma}}\right) = \\
& = exp\left(-\left(1-x-1\right)^{-\frac{1}{\gamma}}\right) = \\
& = exp\left(-\left(-x\right)^{-\frac{1}{\gamma}}\right) = \\
& = exp\left(-\left(-x\right)^{-\alpha}\right) = \\
& = \Psi_\alpha\left(x\right)
\end{aligned}
$$

□

Caso $\gamma = 0$. Calculam-se os limites laterais da expressão em (8), obtendo-se

$$
\begin{aligned}
\lim_{\gamma\to 0^+} G_\gamma(x) = & \lim_{\gamma\to 0^+} exp\left(-\left(1+\gamma x\right)^{-\frac{1}{\gamma}}\right) = \\
& = exp\left(-\lim_{\gamma\to 0^+}\left(1+\gamma x\right)^{-\frac{1}{\gamma}}\right) = \\
& = exp\left(-\lim_{\gamma\to 0^+}\left(1+\frac{x}{\frac{1}{\gamma}}\right)^{-\frac{1}{\gamma}}\right) = \\
& = exp\left(\left(-\lim_{\gamma\to 0^+}\left(1+\frac{x}{\frac{1}{\gamma}}\right)^{\frac{1}{\gamma}}\right)^{-1}\right) = \\
& = exp\left(-\left(exp\left(x\right)\right)^{-1}\right) = \\
& = exp\left(-exp\left(-x\right)\right)
\end{aligned}
$$

$$\begin{aligned}\lim_{\gamma\to 0^-} G_\gamma(x) &= \lim_{\gamma\to 0^-} exp\left(-\left(1+\gamma x\right)^{-\frac{1}{\gamma}}\right) = \\ &= exp\left(-\lim_{\gamma\to 0^-}\left(1+\gamma x\right)^{-\frac{1}{\gamma}}\right) = \\ &= exp\left(-\lim_{\gamma\to 0^-}\left(1+\frac{x}{\frac{1}{\gamma}}\right)^{-\frac{1}{\gamma}}\right) = \\ &= exp\left(-\lim_{\gamma\to 0^-}\left(1+\frac{-x}{-\frac{1}{\gamma}}\right)^{-\frac{1}{\gamma}}\right) = \\ &= exp\left(-exp\left(-x\right)\right)\end{aligned}$$

Portanto,

$$\lim_{\gamma\to 0^+} G_\gamma(x) = \lim_{\gamma\to 0^-} G_\gamma(x) = \lim_{\gamma\to 0} G_\gamma(x) = exp\left(-exp\left(-x\right)\right) = \Lambda\left(x\right)$$

□

Estudos anteriores revelam que a função distribuição Weibull define a distribuição de extremos da distribuição Uniforme, Beta, Weibull de máximos, etc. Assim como a distribuição de Gumbel representa a distribuição de extremos das distribuições Exponencial, Normal, Gama,... e a Fréchet define as distribuições Pareto, Fréchet e Cauchy.

Como consequência direta da EVT para máximos, e tendo em conta a seguinte dualidade

$$m_n := min\left(X_1, X_2, ..., X_n\right) = -max\left(-X_1, -X_2, ..., -X_n\right)$$

Assim, tem-se

$$\begin{aligned} P(m_n \leq x) &= P(-M_n \leq x) = \\ &= P(M_n \geq -x) = \\ &= 1 - P(M_n \leq -x) \end{aligned} \tag{10}$$

Reunidas as condições enunciadas no Teorema 2.2.1, tem-se que a equação (10) converge para uma das três distribuições apresentadas no Teorema 2.2.1.

Portanto, evoca-se o teorema dos tipos extremais para mínimos.

Teorema 2.2.2 (Teorema dos tipos extremais para mínimos)**.** *Sejam duas sucessões reais $a_n^* > 0$ e b_n^*, tais que*

$$\lim_{n\to\infty} P\left[\frac{M\_b_n^*}{a_n^*} \leq x\right] = \lim_{n\to\infty} F^n(a_n^* x + b_n^*) = G^*(x)$$

para alguma função distribuição G^* não degenerada, então G^* é do mesmo tipo de uma das seguintes distribuições

Tipo I:

$$G^*(x) = \Lambda^*(x) = 1 - \Lambda(-x) = 1 - exp(-exp(x)),\ x \in \mathbb{R} \tag{11}$$

Tipo II:

$$\begin{aligned} & G^*(x|\alpha) = \Phi_\alpha^*(x) = 1 - \Phi_\alpha(-x) \\ \Leftrightarrow\ & G^*(x|\alpha) = \begin{cases} 1 - exp(x^{-\alpha}), & x < 0,\ \alpha > 0 \\ 1, & x \geq 0 \end{cases} \end{aligned} \tag{12}$$

Tipo III:

$$G^*(x|\alpha) = \Psi^*_\alpha(x) = 1 - \Psi_\alpha(-x)$$
$$\Leftrightarrow G^*(x|\alpha) = \begin{cases} 0, & x \leq 0 \\ 1 - exp(-x^\alpha), & x < 0, \alpha > 0 \end{cases} \quad (13)$$

As funções (11), (12) e (13) são denominadas, respetivamente, de função Gumbel de mínimos, Fréchet de mínimo e Weibull de mínimo.

Tal como para o caso do Teorema para os extremos máximos, também se pode reduzir os três tipos numa função generalizada de mínimos.

$$G^*(x|y) = G^*_\gamma(x) = 1 - G_\gamma(-x)$$
$$\Leftrightarrow G^*(x|y) = \begin{cases} 1 - exp\left(-(1-\gamma x)^{-\frac{1}{\gamma}})\right), & 1 - \gamma x > 0, \gamma \neq 0 \\ 1 - exp(-exp(x)), & x \in \mathbb{R}, \gamma = 0 \end{cases}, \quad (14)$$

onde γ representa o parâmetro *shape*.

Mais uma vez, através do sinal do parâmetro *shape*, γ, determina-se a qual tipo de distribuição pertence. Isto é, se $\gamma < 0$, está-se perante a distribuição Weibull de mínimos; quando $\gamma > 0$, assume-se a distribuição Fréchet de mínimos; no caso de $\gamma = 0$, está-se perante a distribuição Gumbel de mínimos.

Neste trabalho, aplicar-se-á esta teoria no capítulo das simulações. Com a ajuda do *software* R, em particular usando a função *fgev* presente na biblioteca *evd*, obtêm-se as estimativas para os parâmetros GEV que permitirão construir a função GEV estimada.

3

FILTROS DE BLOOM LINEARES

Face às limitações do FB, explora-se a ideia do Filtro de Bloom Linear (FBL), considerado como uma evolução do FB, que permite resolver algumas limitações ou melhorá-las.

O artigo Lima et al. [2013] mostra uma aplicação positiva dos FBLs, que é a optimização do algoritmo de encaminhamento em redes multi-hop.

O FBL é uma estrutura de dados semelhante a um FB que, para além de armazenar elementos, permite armazenar junto a cada elemento uma grandeza totalmente ordenável que poderá representar uma certa característica do elemento, c_{elem_i}. Durante o estudo, essa grandeza será tratada como grau de confiança. Enquanto o FB tradicional se inicia a zeros e toma o valor 1 no caso de inserção de um elemento, no FBL insere-se um grau de confiança, $c_{elem_i} \in]0, 1]$, que indica uma característica do elemento a ser inserido. A cada elemento a ser inserido, $elem_i$, associa-se então um grau de confiança, obtendo-se assim o par $(elem_i, c_{elem_i})$. No caso particular em que todos os elementos inseridos estejam associados a um grau de confiança de 1, o FBL toma uma aparência igual à de um FB tradicional, apenas com 0 e 1.

Embora os métodos de inserção e de consulta também sejam contemplados no FBL, o seu funcionamento difere em relação ao FB tradicional.

3.1 MÉTODO DE INSERÇÃO

A diferença no método de inserção em relação do FB, surge devido à variação do grau de confiança a introduzir no filtro. Assim, é necessário ter-se em atenção a possibilidade da existência de dois valores associados à mesma posição do filtro. Sendo que o objetivo continua a ser guardar uma grande quantidade de elementos com o seu respetivo grau de confiança, inserem-se os elementos nas posições definidas através da função Hash de igual forma e, na presença de sobreposição de elementos numa mesma posição, o filtro assume o grau de confiança mais elevado. O valor guardado é, então, obtido pela aplicação da função de máximo, que consiste na seleção do maior valor entre o valor presente numa determinada posição e o grau de confiança característico do elemento que está a ser inserido. Assim, o grau de confiança a introduzir na posição l, doravante denominado $bloom[l]$, pode tomar dois valores.

- Se o valor característico do elemento for menor do que o valor já presente, este último mantém-se;
- Se o valor característico do elemento for maior ou igual ao valor já presente, este último é substituído pelo valor característico do elemento.

Resumidamente o método de inserção, para cada posição l, aplica-se através da seguinte função

$$bloom[l] = max(c_{obs_l}, c_{elem_{i,l}})$$

considerando c_{obs_l} o valor observado na posição l e $c_{elem_{i,l}}$ o valor do grau de confiança do novo elemento correspondente à posição l.

Exemplo 3.1. *Retomando a base do exemplo do FB tradicional, considere-se um filtro de tamanho* $m = 11$*, o número de elementos a inserir* $n = 3$ *e sejam os elementos e respetivos graus de confiança o seguinte conjunto* $\{(a, 0.3), (b, 0.5), (c, 0.7)\}$.

Tem-se $k = 3$*, isto é, a função Hash aplicada a cada elemento a inserir devolve 3 posições. Aplicando a função Hash a cada elemento, obtêm-se as respetivas posições. O método de inserção dos elementos pode ser observado na Figura 2.*

Figura 2.: Inserção de elementos no Filtro de Bloom Linear

$\Downarrow\ Hash(a) = \{1, 3, 7\}$, com $c_{elem_a} = 0.3$

1	2	3	4	5	6	7	8	9	10	11
0.3	0	0.3	0	0	0	0.3	0	0	0	0

$\Downarrow\ Hash(b) = \{3, 7, 10\}$, com $c_{elem_b} = 0.5$

1	2	3	4	5	6	7	8	9	10	11
0.3	0	0.5	0	0	0	0.5	0	0	0.5	0

$\Downarrow\ Hash(c) = \{4, 7, 10\}$, com $c_{elem_c} = 0.7$

1	2	3	4	5	6	7	8	9	10	11
0.3	0	0.5	0.7	0	0	0.7	0	0	0.7	0

Note-se que esta situação também acontece nos FBs tradicionais. No entanto, como estes apenas manifestam a presença do elemento através do valor 1, um conflito de valores numa qualquer posição termina sempre com a apresentação do valor 1, visto que o máximo entre dois valores iguais é sempre o próprio valor.

3.2 MÉTODO DE CONSULTA

O método de consulta está, de certa forma, ligado ao método de inserção. Se inicialmente se queria inserir os elementos, usando a função dos máximos, agora o objetivo é, observando os valores de grau de confiança visíveis do Filtro, saber se determinado elemento foi inserido.

Assim, o método de consulta, assenta na aplicação da função dos mínimos. Isto é, a cada elemento a consultar, aplica-se a função Hash para se saber as posições a observar. Consoante o resultado do mínimo entre os valores visíveis em cada posição dada pela função Hash, e tendo em conta que os valores visíveis do filtro foram selecionados com base numa função de máximos, podem-se tirar as seguintes conclusões.

- Se o valor mínimo for 0, tem-se que o elemento consultado não existe no filtro;
- Se o valor mínimo for inferior ao grau de confiança do elemento consultado, o elemento não está presente no filtro;
- Se o valor mínimo for igual ao grau de confiança do elemento consultado, há forte probabilidade do elemento estar presente no filtro;
- Se o valor mínimo for superior ao grau de confiança do elemento consultado, há probabilidade do elemento existir no filtro.

Exemplo 3.2. *No exemplo seguinte, retoma-se o filtro anterior, com o objetivo de se observar duas situações distintas: consultar se o elemento a está presente no filtro, tendo consciência que foi realmente inserido anteriormente, e consultar se o elemento d foi inserido, sabendo que na realidade não foi.*

- ***Situação 1***

Inicialmente, aplica-se a função Hash ao elemento a para se saber quais são as posições atribuídas. Neste caso, a função Hash já tinha sido aplicada no método de inserção e obteve-se $Hash(a) = \{1, 3, 7\}$.

Figura 3.: Consulta do elemento *a* na situação 1

1	2	3	4	5	6	7	8	9	10	11
0.3	0	0.5	0.7	0	0	0.7	0	0	0.7	0

Observando a Figura 3, as posições definidas para o elemento a estão preenchidas com os valores $\{0.3, 0.5, 0.7\}$. *Aplicando a função dos mínimos, obtém-se* $min\,(0.3, 0.5, 0.7) = 0.3$. *Sabe-se que o grau de confiança atribuído ao elemento a é* 0.3, *que corresponde ao mínimo calculado, o que indica que o elemento a tem forte probabilidade de existir no Filtro de Bloom Linear.*

- ***Situação 2***

Neste caso, averigua-se se o elemento d está inserido no FBL. Para tal, e seguindo o raciocínio da situação anterior, aplica-se a função Hash ao elemento d, obtendo-se $Hash\,(d) = \{1, 4, 10\}$. *Consultam-se os graus de confiança nas posições obtidas, como ilustrado na Figura 4.*

Figura 4.: Consulta do elemento *d* na situação 2

1	2	3	4	5	6	7	8	9	10	11
0.3	0	0.5	0.7	0	0	0.7	0	0	0.7	0

Calcula-se o mínimo entre os graus de confiança observados nas posições atribuídas para o elemento pela função Hash. Tem-se então $min\,(0.3, 0.7, 0.7) = 0.3$.

Nesta situação, seja o grau de confiança do elemento d, c_{elem_d}, *só se pode questionar a existência do elemento no filtro caso esse grau de confiança seja menor ou igual ao mínimo calculado anteriormente* ($c_{elem_d} \leq 0.3$). *Caso contrário,*

sendo maior que 0.3*, tem-se a certeza de que o elemento não foi introduzido no Filtro de Bloom Linear.*

Por exemplo, supondo que o grau de confiança do elemento d seja 0.4*, a probabilidade do elemento pertencer ao filtro de Bloom seria nula, pois no momento de inserção, em cada posição, teria ficado visível o grau de confiança mais elevado.*

Nos FBLs também podem surgir falsos positivos, que neste caso são chamados de sobreestimação de pertença, resultante de um erro de estimação por excesso. Isto acontece caso a sobreposição seja completa, isto é, se em todas as posições atribuídas ao elemento se observar um grau de confiança superior ao característico desse elemento. Tendo em conta que nos FBLs os valores introduzidos são muito mais variados, a probabilidade de falsos positivos reduz em relação ao caso do FB tradicional.

4

RESULTADOS TEÓRICOS

4.1 TEORIA DOS VALORES EXTREMOS

Partindo da premissa de que os valores extremos, mínimos e máximos, estão inteiramente associados aos FBLs, o seguimento desta análise passa pelo estudo da Teoria dos Valores Extremos.

Considere-se $X_1, X_2, ..., X_n$, um conjunto de variáveis iid, caracterizadas por uma função densidade de probabilidade $f(x)$ e por uma função de distribuição $F(x)$. Ordenando o conjunto de variáveis por ordem crescente, as estatísticas ordinais podem ser denotadas por $(X_{(1)}, X_{(2)}, ..., X_{(n)})$, com $X_{(1)} < X_{(2)} < ... < X_{(n)}$. Assim, a primeira e a última estatística de ordem, $X_{(1)}$ e $X_{(n)}$ correspondem ao mínimo e ao máximo, respetivamente.

Seja então $X_{(n)} = max\,(X_1, X_2, ..., X_n)$.

A função densidade de probabilidade de $X_{(n)}$ é dada por

$$\begin{aligned} G_n(u) =& P\left(X_{(n)} \leq u\right) \\ =& P\left(X_1 \leq u, X_2 \leq u, ..., X_n \leq u\right) \\ =& P\left(X_1 \leq u\right) P\left(X_2 \leq u\right) ... P\left(X_n \leq u\right) \\ =& \prod_{i=1}^{n} P\left(X_i \leq u\right) \\ =& \left[P\left(X_i \leq u\right)\right]^n \\ =& \left[F(u)\right]^n \end{aligned} \tag{15}$$

Pela função (15), calcula-se a função densidade de probabilidade de $X_{(n)}$

$$g_n(u) = \frac{d[F(u)]^n}{du} = nf(u)[F(u)]^{n-1} \tag{16}$$

Analogamente, pode-se determinar a função de mínimos

Seja $X_{(1)} = min\left(X_1, X_2, ..., X_n\right)$.

Tem-se que a função distribuição de $X_{(1)}$ é dada por

$$\begin{aligned} G_1(u) =& P\left(X_{(1)} \leq u\right) \\ =& 1 - P\left(X_{(1)} > u\right) \\ =& 1 - P\left(X_1 > u, X_2 > u, ..., X_n > u\right) \\ =& 1 - \left[P\left(X_1 > u\right) P\left(X_2 > u\right) ... P\left(X_n > u\right)\right] \\ =& 1 - \prod_{i=1}^{n} P\left(X_i > u\right) \\ =& 1 - \left[1 - P\left(X_i \leq u\right)\right]^n \\ =& 1 - \left[1 - F(u)\right]^n \end{aligned} \tag{17}$$

Pela função (17), calcula-se a função densidade de probabilidade de $X_{(1)}$

$$g_1(u) = \frac{d[1-(1-F(u))^n]}{du} = nf(u)[1-F(u)]^{n-1} \tag{18}$$

Partindo das duas funções anteriormente referidas, deduz-se a função dos mínimos dos máximos, que também será indispensável no decorrer deste estudo.

Por (17), tem-se que $G_1(u) = 1-[1-F(u)]^n$. No caso concreto da operação de mínimos de máximos, tem-se $F(u) = G_n(u)$, logo $G_1(u) = 1-[1-G_n(u)]^n$. Obtém-se então a função distribuição dos mínimos dos máximos

$$G_1^*(u) = 1-[1-F(u)^n]^n \tag{19}$$

Derivando a função obtida, (19), tem-se que a função densidade dos mínimos dos máximos se define da seguinte forma

$$\begin{aligned} g_1^*(u) =& \frac{d[1-[1-F(u)^n]^n]}{du} = \\ =& -n(1-F(u)^n)^{n-1}(-1)nF(u)^{n-1}f(u) = \\ =& n^2 f(u)F(u)^{n-1}[1-F(u)^n]^{n-1} \end{aligned} \tag{20}$$

com $f(u)$ e $F(u)$, identificando as funções de densidade e de distribuição originalmente adotadas para gerar os graus de confiança, respetivamente.

4.2 GERAÇÃO PROBABILÍSTICA DE GRAUS DE CONFIANÇA

Após a dedução da função dos máximos $G_n(.)$ e da função dos mínimos dos máximos $G_1^*(.)$, são estudados os casos específicos de três distribuições $F(.)$, que poderão ser adotadas para a definição de graus

de confiança. As distribuições escolhidas são as distribuições Uniforme, Exponencial e Normal.

4.2.1 *Distribuição Uniforme*

Seja X uma variável aleatória que segue uma distribuição Uniforme no intervalo real $[a, b]$, a função densidade de probabilidade é dada por

$$f(x) = \begin{cases} \frac{1}{b-a} & \text{se } a \leq x \leq b \\ 0 & \text{caso contrário} \end{cases} \tag{21}$$

e a sua função distribuição é definida por

$$F(x) = \begin{cases} 0 & \text{se } x < a \\ \frac{x-a}{b-a} & \text{se } a \leq x \leq b \\ 1 & \text{caso contrário} \end{cases} \tag{22}$$

Ao longo desta tese, os parâmetros a e b tomam os valores 0 e 1, respetivamente.

Representa-se graficamente a função densidade da distribuição Uniforme(0,1) na Figura 5.

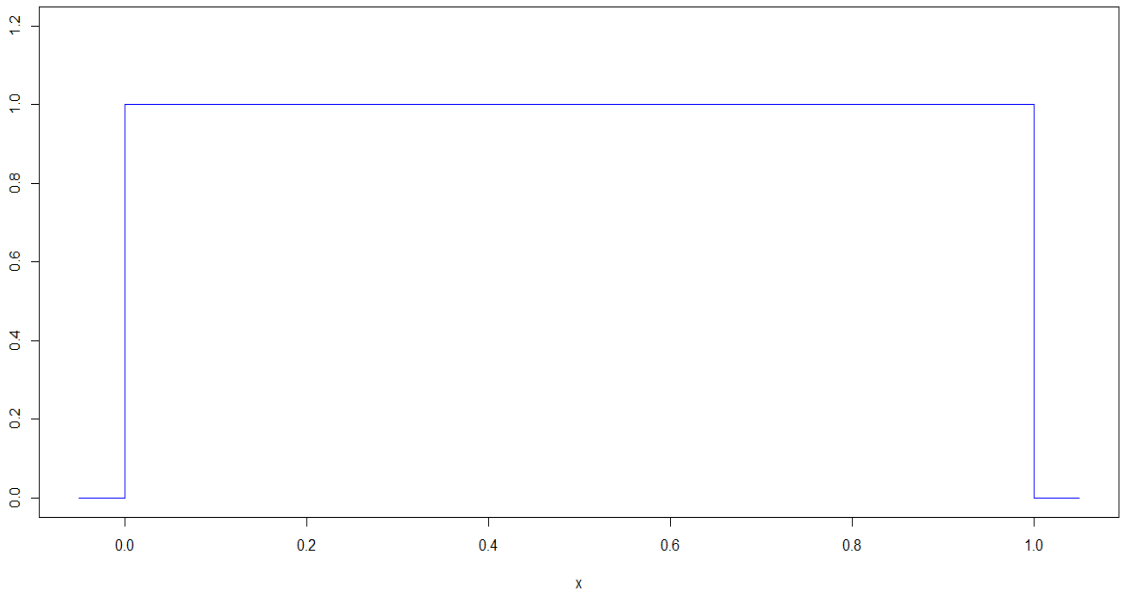


Figura 5.: Representação gráfica da função densidade da distribuição Uniforme(0,1)

4.2.2 *Distribuição Exponencial*

Seja X uma variável aleatória que segue uma distribuição Exponencial caracterizada pelo parâmetro λ , a função densidade de probabilidade é dada por:

$$f(x|\lambda) = \begin{cases} \lambda e^{-\lambda x} & \text{se } x \geq 0 \\ 0 & \text{caso contrário} \end{cases} \tag{23}$$

onde $E[X] = \frac{1}{\lambda}$, e a sua função distribuição é definida por

$$F(x|\lambda) = \lambda e^{-\lambda x} \text{ se } x \geq 0 \tag{24}$$

No decorrer desta dissertação, o parâmetro λ adota o valor 1.

Representa-se graficamente a função densidade da distribuição Exponencial(1) na Figura 6.

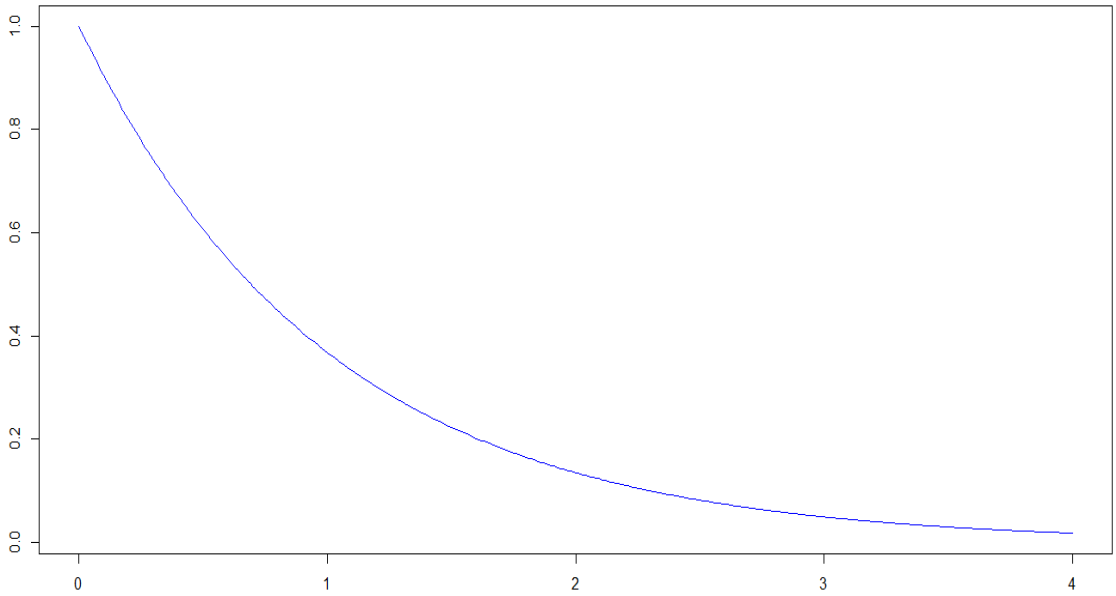


Figura 6.: Representação gráfica da distribuição Exponencial (1)

4.2.3 *Distribuição Normal*

Seja X uma variável aleatória que segue uma distribuição Normal com média μ e desvio padrão σ, a função densidade de probabilidade é dada por:

$$f(x|\mu,\sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \tag{25}$$

Não é possível explicitar a sua função distribuição devido ao facto da função densidade não ser de fácil tratamento matemático.

Para efeito deste estudo, os parâmetros μ e σ assumem, respetivamente, os valores 0 e 1.

Representa-se graficamente a função densidade da distribuição Normal(0,1) na Figura 7.

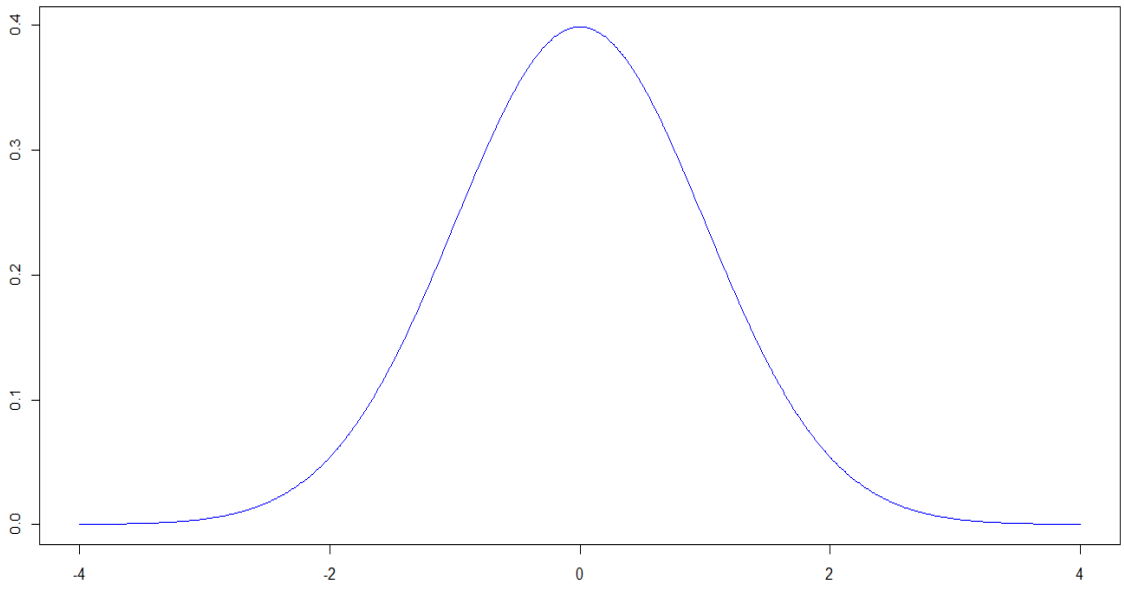


Figura 7.: Representação gráfica da distribuição Normal(0,1)

5

SIMULAÇÕES

Neste capítulo, apresentam-se os estudos de simulações envolvendo as três distribuições referidas no Capítulo 4, Uniforme(0,1), Exponencial(1) e Normal(0,1). Nas primeiras duas secções do presente capítulo, descreve-se o processo adotado para as simulações associadas ao método de inserção e de consulta. Simulações estas feitas com recurso ao *software R*. Após as simulações, e já nas terceira e quarta partes deste capítulo, faz-se uma comparação das distribuições estimadas *versus* teóricas, bem como uma análise de distorção dos valores inseridos e consultados.

Para todas as simulações, considera-se um filtro de tamanho $m = 2^{20}$, com $k = 7$ posições definidas pela função Hash e, pela fórmula (2), tem-se $n = 103831$ elementos a inserir.

Cada simulação foi repetida um número elevado de vezes afim de se obter uma banda de confiança a $(1-\alpha)\%$ para as estimativas não paramétricas da função densidade, assim como intervalos de confiança a $(1-\alpha)\%$ para as estimativas dos parâmetros GEV, para um determinado nível de confiança α. Tendo em conta o peso computacional de cada simulação, optou-se pelo total de 100 simulações.

5.1 SIMULAÇÃO DO MÉTODO DE INSERÇÃO

Começa-se por criar um vetor, *bloom*, em que cada posição fica preenchida com o valor -99, admitindo-se este como sendo o FBL inicial vazio. Opta-se por preencher com o valor -99 em vez de 0 para evitar conflitos de valores gerados aleatoriamente para o grau de confiança de cada elemento, c_{elem_i}.

Depois do filtro ser criado, procede-se à inserção de cada elemento, $elem_i$, e à criação aleatória do valor do grau de confiança, c_{elem_i}, seguindo uma determinada distribuição associada a cada elemento. Após a comparação do valor gerado com o valor observado em cada uma das k posições, guarda-se o valor máximo entre os dois. Recorre-se a esse procedimento para cada elemento a ser inserido, o que significa o processo é realizado n vezes, até que se obtém, por fim, o filtro preenchido.

Para cada FBL gerado, estima-se a densidade para os valores inseridos por uma abordagem não-paramétrica, recorrendo a uma função tipo-núcleo Gaussiana. Adicionalmente, estimam-se os três parâmetros da função densidade generalizada dos valores extremos (GEV). À custa do total dos 100 FBLs gerados, torna-se possível obter uma banda de confiança, por exemplo a 95%, para a estimação tipo-núcleo, assim como um intervalo de confiança a 95% para cada um dos parâmetros estimados da GEV. Nas secções seguintes, para cada uma das 3 distribuições escolhidas, procede-se com a representação gráfica das várias densidades estimadas, e a respetiva função de densidade teórica estudada no Capítulo 4.

Afim de se poder utilizar posteriormente as informações geradas da operação de inserção, guardam-se numa matriz, $matriz_{est}$, as 100 estimativas não paramétricas da densidade, calculadas para 400 pontos re-

sultantes da discretização do domínio desta função. De forma análoga, as 100 estimativas dos 3 parâmetros GEV são guardados numa matriz denominada $matriz_{gev}$.

A título meramente ilustrativo, as 100 primeiras entradas de cada FBL aparecem no apêndice A.

Apresentam-se de seguida os procedimentos adotados, em ambiente R, para o preenchimento dos FBLs e para o armazenamento de toda a informação posteriormente necessária para se prosseguir com os estudos de simulação propostos.

5.1.1 *Uniforme(0,1)*

Começa-se pelo caso da distribuição $Uniforme(0,1)$. Observe-se no código seguinte que a única dependência da distribuição é aquando da geração dos valores aleatórios para o grau de confiança, sendo que neste caso os valores são gerados seguindo uma distribuição Uniforme no intervalo $[0,1]$.

```
rm(list=ls())
library(evd)
r=100
f=1
m=2^20
k=7
n=round(m/k*log(2),0)
matriz_est=matrix(rep(0,400*r),ncol=400,nrow=r)
matriz_gev=matrix(rep(0,3*r),ncol=3,nrow=r)
matriz_est1=matrix(rep(0,400*r),ncol=400,nrow=r)
matriz_gev1=matrix(rep(0,3*r),ncol=3,nrow=r)
matriz_bloom=matrix(ncol=m,nrow=r)
matriz_minimum=matrix(ncol=(n*f),nrow=r)
```

```
matriz_elem_i=matrix(ncol=k,nrow=(n*f))
for(b in 1:r){
 bloom=rep(-99,m)                          #criacao do filtro (vazio)
 for (i in 1:(n*f)){                  #processo de criacao do elemento
  elem_i=sample.int(m,k,replace = T)     #criacao das 7 posicoes
  matriz_elem_i[i,]=elem_i
  c_elem_i=round(runif(1,0,1),3)  #criacao do grau de confianca
  for (j in 1:k){                    #processo de insercao
    bloom[elem_i[j]] = max(bloom[elem_i[j]] , c_elem_i)
  }
 }
bloom_NZ = bloom[bloom != -99]                    #vetor de maximos
matriz_bloom[b,]=bloom                          #matriz com 100 filtros
aux=density(bloom_NZ, n=400, from=0, to=1)
matriz_est[b,]=aux$y                          #matriz pontos de densidade
p=fgev(bloom_NZ)                  #funcao para obter os parametros GEV
parametros=p$estimate                                 # parametros GEV
matriz_gev[b,]=parametros                      #matriz parametros GEV
}
```

Pela análise da Figura 8, conclui-se que tanto a função densidade teórica como a função estimada pela abordagem GEV parecem aproximar-se bem à função de densidade estimada tipo-núcleo, sendo que a teórica aparenta ser melhor.

Na Tabela 1, apresentam-se algumas estatísticas descritivas para as estimativas dos parâmetros GEV. Os valores relativos ao parâmetro *shape* são todos negativos, tal como seria de esperar pela teoria de valores extremos (ver Capítulo 2) que apoia a tese da distribuição dos máximos de uma Uniforme ser a distribuição Weibull.

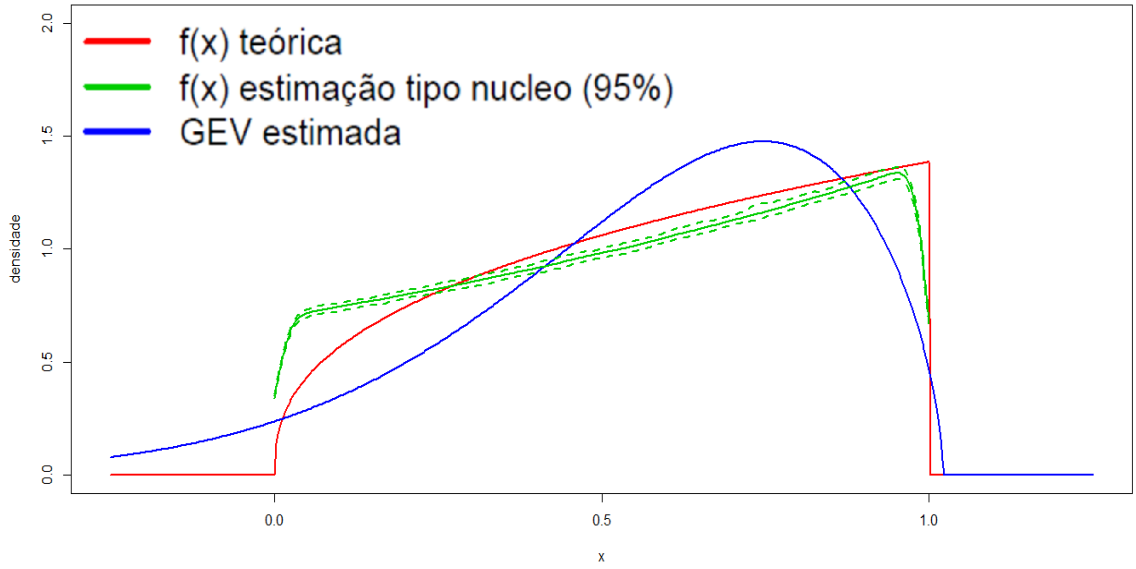


Figura 8.: Gráfico das densidades dos máximos da Uniforme(0,1): f(x) teórica; f(x) estimada tipo-núcleo, com linha a cheio para a média das estimativas e linhas a tracejado para os limites da banda de confiança; e GEVD obtida para as medianas das estimativas dos parâmetros GEV

Tabela 1.: Estatísticas descritivas para as estimativas dos parâmetros GEV da distribuição dos máximos da Uniforme(0,1)

	location (λ)	scale (δ)	shape (γ)
min	0.5036	0.3222	-0.6404
25%	0.5059	0.3235	-0.6327
50%	0.5067	0.3241	-0.6311
75%	0.5075	0.3246	-0.6287
max	0.5106	0.3257	-0.6216
média	0.5067	0.3241	-0.6309
desvio padrão	0.0013	0.0008	0.0035

5.1.2 *Exponencial(1)*

No caso da distribuição Exponencial, o código R responsável pelo preenchimento dos FBLs é semelhante ao caso anterior, exceto na linha 20, relativa à geração dos valores atribuídos como graus de confiança, que

neste caso seguem uma distribuição exponencial com valor esperado igual a 1.

```
rm(list=ls())
library(evd)
r=100
f=1
m=2^20
k=7
n=round(m/k*log(2),0)
matriz_est=matrix(rep(0,400*r),ncol=400,nrow=r)
matriz_gev=matrix(rep(0,3*r),ncol=3,nrow=r)
matriz_est1=matrix(rep(0,400*r),ncol=400,nrow=r)
matriz_gev1=matrix(rep(0,3*r),ncol=3,nrow=r)
matriz_bloom=matrix(ncol=m,nrow=r)
matriz_minimum=matrix(ncol=(n*f),nrow=r)
matriz_elem_i=matrix(ncol=k,nrow=(n*f))
for(b in 1:r){
 bloom=rep(-99,m)                          #criacao do filtro (vazio)
 for (i in 1:(n*f)){                  #processo de criacao do elemento
  elem_i=sample.int(m,k,replace = T)     #criacao das 7 posicoes
  matriz_elem_i[i,]=elem_i
  c_elem_i=round(rexp(1,1),3)       #criacao do grau de confianca
  for (j in 1:k){                              #processo de insercao
    bloom[elem_i[j]] = max(bloom[elem_i[j]] , c_elem_i)
 }}
bloom_NZ = bloom[bloom != -99]                    #vetor de maximos
matriz_bloom[b,]=bloom                          #matriz com 100 filtros
aux=density(bloom_NZ, n=400, from=0)
matriz_est[b,]=aux$y                       #matriz pontos de densidade
p=fgev(bloom_NZ)                 #funcao para obter os parametros GEV
parametros=p$estimate                                 #parametros GEV
matriz_gev[b,]=parametros}                     #matriz parametros GEV
```

Conclui-se pela análise da Figura 9 que as funções estimadas são muito semelhantes entre si. Comparando estas funções com a distribuição Exponencial(1) apresentada na Figura 6, constata-se que a forma se assemelha à distribuição estimada dos máximos da Exponencial(1).

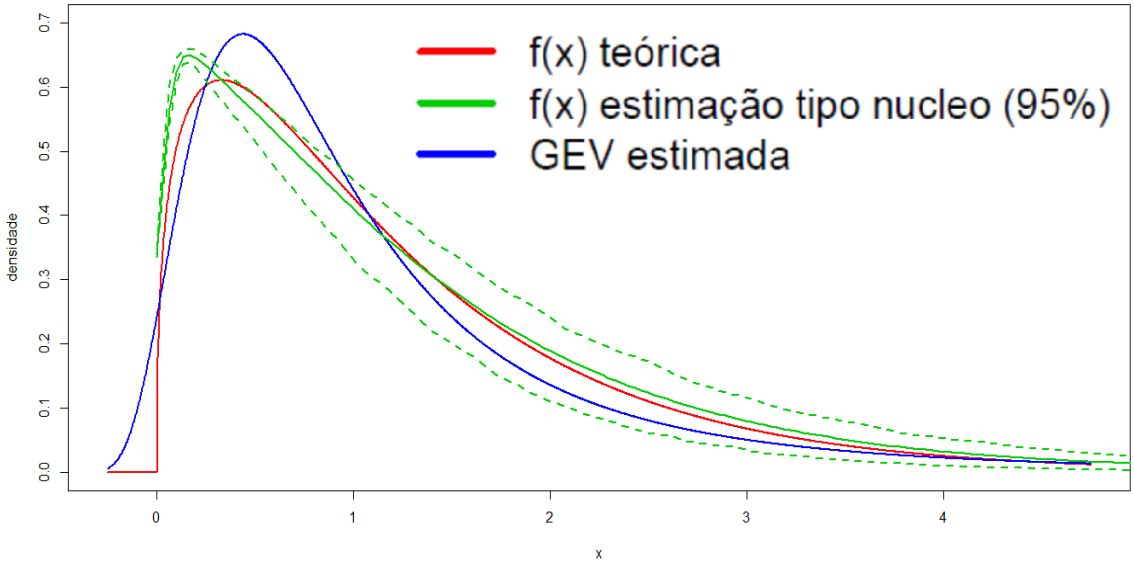


Figura 9.: Gráfico das densidades dos máximos da Exponencial(1): f(x) teórica; f(x) estimada tipo-núcleo, com linha a cheio para a média das estimativas e linhas a tracejado para os limites da banda de confiança; e GEVD obtida para as medianas das estimativas dos parâmetros GEV

As principais estatísticas descritivas calculadas para os parâmetros GEV obtidos pelos máximos da Exponencial apresentam-se na Tabela 2. Relembra-se que, segundo a teoria de valores extremos apresentada previamente, a distribuição dos máximos de uma exponencial segue a distribuição Gumbel, que por norma é caracterizada pela tendência do valor do parâmetro *shape* para 0. Neste caso concreto, ao contrário do expectável, todos os valores obtidos para a *shape* são positivos.

Tabela 2.: Estatísticas descritivas para as estimativas dos parâmetros GEV da distribuição dos máximos da Exponencial(1)

	location (λ)	scale (δ)	shape (γ)
min	0.5989	0.5665	0.3544
25%	0.6032	0.5707	0.3599
50%	0.6046	0.5721	0.3623
75%	0.6063	0.5734	0.3653
max	0.6125	0.5783	0.3698
média	0.6048	0.5721	0.3623
desvio padrão	0.0026	0.0020	0.0034

5.1.3 *Normal(0,1)*

O terceiro caso estudado consiste na distribuição dos máximos da Normal(0,1). Assim, a mudança no processo de preenchimento dos FBLs consiste na alteração da função distribuição escolhida para a geração dos graus de confiança, que agora se apresenta como a distribuição Normal, com média nula e desvio padrão igual a 1 (ver linha 20).

```
rm(list=ls())
library(evd)
r=100
f=1
m=2^20
k=7
n=round(m/k*log(2),0)
matriz_est=matrix(rep(0,400*r),ncol=400,nrow=r)
matriz_gev=matrix(rep(0,3*r),ncol=3,nrow=r)
matriz_est1=matrix(rep(0,400*r),ncol=400,nrow=r)
matriz_gev1=matrix(rep(0,3*r),ncol=3,nrow=r)
matriz_bloom=matrix(ncol=m,nrow=r)
matriz_minimum=matrix(ncol=(n*f),nrow=r)
matriz_elem_i=matrix(ncol=k,nrow=(n*f))
```

```
for(b in 1:r){
 bloom=rep(-99,m)                          #criacao do filtro (vazio)
 for (i in 1:(n*f)){                  #processo de criacao do elemento
  elem_i=sample.int(m,k,replace = T)     #criacao das 7 posicoes
  matriz_elem_i[i,]=elem_i
  c_elem_i=round(rnorm(1,0,1),3)  #criacao do grau de confianca
  for (j in 1:k){                              #processo de insercao
    bloom[elem_i[j]] = max(bloom[elem_i[j]] , c_elem_i)}
 }
bloom_NZ = bloom[bloom != -99]                    #vetor de maximos
matriz_bloom[b,]=bloom                        #matriz com 100 filtros
aux=density(bloom_NZ, n=400)
matriz_est[b,]=aux$y                       #matriz pontos de densidade
p=fgev(bloom_NZ)              #funcao para obter os parametros GEV
parametros=p$estimate                              # parametros GEV
matriz_gev[b,]=parametros                     #matriz parametros GEV
}
```

A Figura 10 mostra que tanto a função densidade teórica como a estimada GEV estão a acompanhar a função estimada pela simulação. Pode-se observar também que a forma da distribuição dos máximos da Normal(0,1) segue com bastante precisão a distribuição Normal(0,1) original apresentada na Figura 7.

Quanto às estimativas dos parâmetros GEV dos máximos da Normal(0,1), observa-se na Tabela 3 que o parâmetro de forma toma valores negativos. De acordo com a teoria de valores extremos, seria espectável que a distribuição dos máximos da Normal seguisse uma distribuição Gumbel, onde o parâmetro *shape* tenderia para 0.

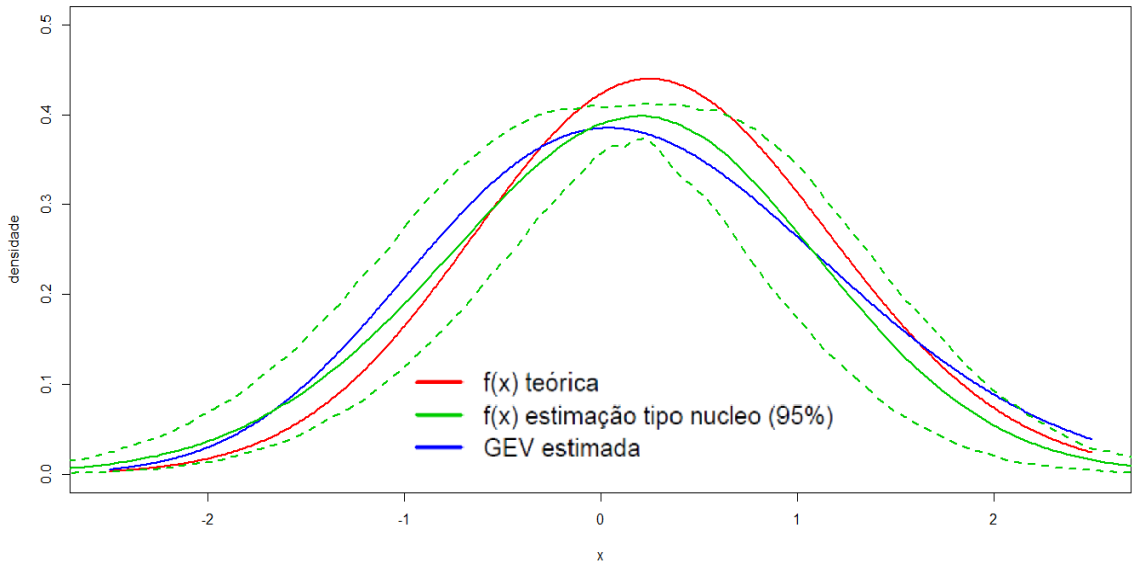


Figura 10.: Gráfico das densidades dos máximos da Normal(0,1): f(x) teórica; f(x) estimada tipo-núcleo, com linha a cheio para a média das estimativas e linhas a tracejado para os limites da banda de confiança; e GEVD obtida para as medianas das estimativas dos parâmetros GEV

Tabela 3.: Estatísticas descritivas para as estimativas dos parâmetros GEV da distribuição dos máximos da Normal(0,1)

	location (λ)	scale (δ)	shape (γ)
min	-0.2449	0.8787	-0.2420
25%	-0.1993	0.9734	-0.2214
50%	-0.1900	0.9792	-0.2135
75%	-0.1714	0.9847	-0.2053
max	-0.1425	1.0188	-0.1721
média	-0.1870	0.9719	-0.2120
desvio padrão	0.0188	0.0302	0.0146

5.2 SIMULAÇÃO DO MÉTODO DE CONSULTA

Após obtermos o filtro preenchido por elementos, pode-se consultar ou testar se determinado elemento pertence ou não ao filtro. Inicia-se, assim, a simulação do método de consulta.

A simulação do método de consulta é a continuação lógica da simulação do método de inserção supra apresentada. Devido a este facto, é apresentado o código contínuo desde o início da simulação, sendo que a análise recai, nesta fase, sobre as linhas de código correspondentes à simulação do método de consulta (linhas $33 - 45$). Relembra-se que este método recorre à distribuição dos mínimos dos máximos de uma determinada distribuição.

Analogamente ao caso de inserção, o processo de consulta também é repetido 100 vezes, embora seja apenas necessário criar um vetor vazio que será preenchido à medida que se consulta um elemento. Assim, no final do processo de consulta, ter-se-á um vetor de tamanho n.

Como estudado no Capítulo 4, o método consiste em verificar se determinado elemento, $elem_i$, existe no Filtro que foi criado e guardado numa variável denominada *bloom* conforme descrito na secção anterior. Para tal, recuperam-se as k posições atribuídas pela função hash ao elemento i, guardadas na $matriz_{elem_i}$ na linha i, e consulta-se o mínimo dessas k posições. O resultado obtido é guardado no vetor criado denominado *minimum*. Todo este processo é repetido n vezes, obtendo-se o vetor preenchido com os valores dos mínimos dos máximos.

Depois de repetido o procedimento anterior para cada um dos 100 FBLs, estima-se a função de densidade para cada um deles, o que permite obter uma banda de confiança a 95% para estas estimativas. De forma análoga à secção anterior, pode-se então recorrer à representação

gráfica da estimação não-paramétrica da densidade, à custa da respetiva banda de confiança. Adicionalmente, acrescenta-se ao gráfico a função estimada GEV, obtida à custa das medianas das estimativas dos parâmetros GEV, e a função de densidade teórica dos mínimos dos máximos estudada anteriormente.

Tal como na primeira secção, também se guardaram numa matriz, $matriz_{est1}$, 400 estimativas da densidade calculadas em 400 pontos do seu domínio e noutra matriz, $matriz_{gev1}$, as estimativas dos três parâmetros GEV.

Note-se que o método de consulta é independente da distribuição usada.

```
rm(list=ls())
library(evd)
r=100
f=1
m=2^20
k=7
n=round(m/k*log(2),0)
matriz_est=matrix(rep(0,400*r),ncol=400,nrow=r)
matriz_gev=matrix(rep(0,3*r),ncol=3,nrow=r)
matriz_est1=matrix(rep(0,400*r),ncol=400,nrow=r)
matriz_gev1=matrix(rep(0,3*r),ncol=3,nrow=r)
matriz_bloom=matrix(ncol=m,nrow=r)
matriz_minimum=matrix(ncol=(n*f),nrow=r)
matriz_elem_i=matrix(ncol=k,nrow=(n*f))
for(b in 1:r){
# OPERACAO DE INSERCAO
bloom=rep(-99,m)
 for (i in 1:(n*f)){
```

```
19    elem_i=sample.int(m,k,replace = T)
      matriz_elem_i[i,]=elem_i
21    c_elem_i=round(rdist(1,.,.),3)
      for (j in 1:k){
23      bloom[elem_i[j]] = max(bloom[elem_i[j]] ,  c_elem_i)
      }
25  }
bloom_NZ = bloom[bloom != -99]
27 matriz_bloom[b,]=bloom
aux=density(bloom_NZ, n=400)
29 matriz_est[b,]=aux$y
p=fgev(bloom_NZ)
31 parametros=p$estimate
matriz_gev[b,]=parametros}
33 # OPERACAO DE CONSULTA
minimum=c()                              #criacao do filtro(vazio)
35  for (v in 1:n){          #processo de consulta de cada elemento
      minimum[v] = min(bloom[matriz_elem_i[v,]])
37  }
 minimum_NZ = minimum[minimum != -99]        #vetor dos minimos
39  matriz_minimum[b,]=minimum            #matriz com os 100 filtros
 aux1=density(minimum_NZ,n=400)                #400pontos densidade
41  matriz_est1[b,]=aux1$y               #matriz pontos de densidade
 p1=fgev(-minimum_NZ,std.err = F) #funcao para os parametros
     GEV
43  parametros1=p1$estimate                          #parametros GEV
 matriz_gev1[b,]=parametros1          #matriz dos parametros GEV
45 }
```

Apresentam-se agora os resultados obtidos para o método de consulta, isto é, os resultados obtidos para os mínimos dos máximos das distribuições Uniforme(0,1), Exponencial(1) e Normal(0,1).

5.2.1 *Uniforme(0,1)*

Observando a Figura 11, a função densidade teórica parece estar mais próxima da estimação não paramétrica da função de densidade. A função densidade estimada GEV toma uma forma um pouco distorcida mas também acompanha a função densidade estimada pela simulação. Note-se que a estimação não paramétrica da densidade dos mínimos dos máximos da Uniforme(0,1) é muito semelhante à função densidade original apresentada na Figura 5.

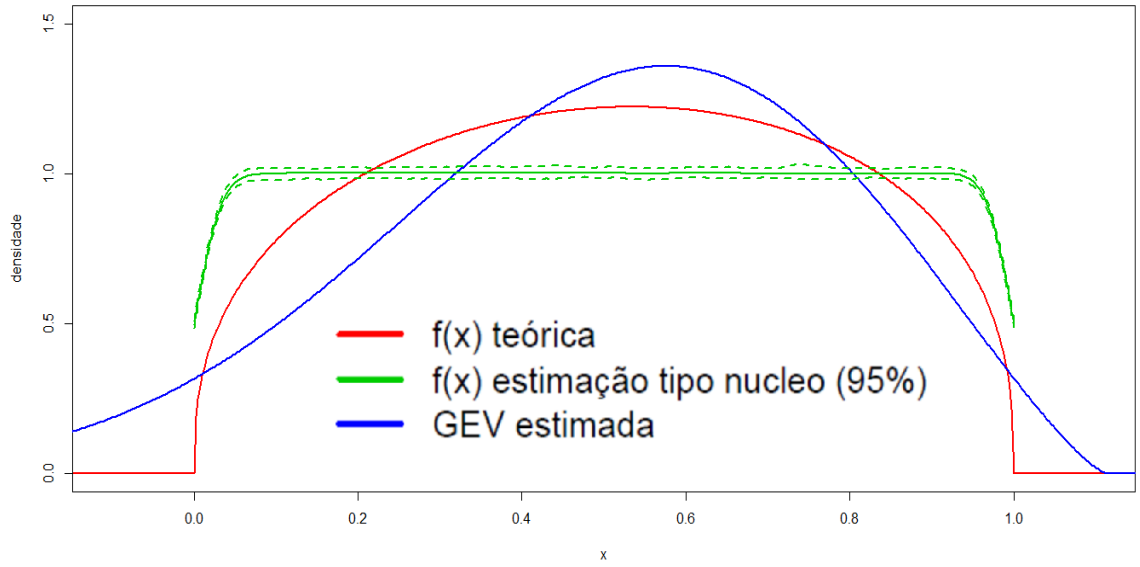


Figura 11.: Gráfico das densidades dos mínimos dos máximos da Uniforme(0,1): f(x) teórica; f(x) estimada tipo-núcleo, com linha a cheio para a média das estimativas e linhas a tracejado para os limites da banda de confiança; e GEVD obtida para as medianas das estimativas dos parâmetros GEV

A Tabela 4 mostra que os valores do parâmetro *shape* são todos negativos, e que rondam o valor -0.44. Este facto leva a crer que os dados simulados podem seguir uma distribuição de Weibull.

Tabela 4.: Estatísticas descritivas para as estimativas dos parâmetros GEV da distribuição dos mínimos dos máximos da Uniforme(0,1)

	location (λ)	scale (δ)	shape (γ)
min	0.4169	0.3018	−0.4500
25%	0.4200	0.3033	−0.4419
50%	0.4209	0.3037	−0.4399
75%	0.4217	0.3041	−0.4380
max	0.4245	0.3051	−0.4308
média	0.4208	0.3037	−0.4399
desvio padrão	0.0014	0.0007	0.0037

5.2.2 *Exponencial(1)*

No caso da Exponencial, observando a Figura 12, conclui-se que as três funções representadas são muito parecidas, o que dá indicação de boas aproximações. Como no caso dos máximos da Exponencial, a função distribuição dos mínimos dos máximos da Exponencial assemelha-se à função distribuição original da Exponencial, representada na Figura 6.

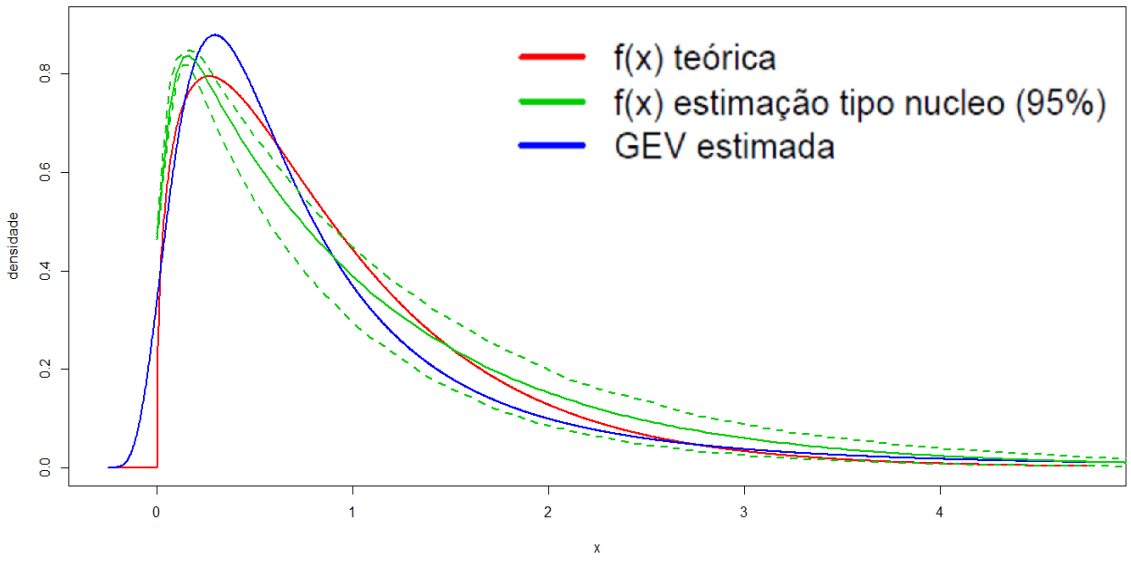


Figura 12.: Gráfico das densidades dos mínimos dos máximos da Exponencial(1): f(x) teórica; f(x) estimada tipo-núcleo, com linha a cheio para a média das estimativas e linhas a tracejado para os limites da banda de confiança; e GEVD obtida para as medianas das estimativas dos parâmetros GEV

Na Tabela 5, verifica-se que, tal como na simulação do método de inserção, os valores dos parâmetros GEV são todos positivos. Estes resultados indicam a possibilidade da distribuição dos mínimos dos máximos da Exponencial seguir a distribuição de Fréchet.

Tabela 5.: Estatísticas descritivas para as estimativas dos parâmetros GEV da distribuição dos mínimos dos máximos da Exponencial(1)

	location (λ)	scale (δ)	shape (γ)
min	0.4530	0.4580	0.4677
25%	0.4569	0.4612	0.4740
50%	0.4585	0.4622	0.4762
75%	0.4596	0.4631	0.4790
max	0.4643	0.4675	0.4841
média	0.4584	0.4622	0.4763
desvio padrão	0.0022	0.0017	0.0036

5.2.3 *Normal(0,1)*

Por último tem-se o caso dos mínimos dos máximos da Normal(0,1). Na Figura 13 constata-se uma boa aproximação da função distribuição GEV à função distribuição estimada tipo-núcleo, sendo que a função estimada GEV está sempre dentro dos limites a 95% de confiança. Em relação à função distribuição teórica, esta também se aproxima da estimação não paramétrica da função densidade, mas claramente a distribuição GEV estimada está melhor. Mais uma vez, as funções representadas têm uma forma semelhante à da distribuição original da Normal(0,1), representada na Figura 7.

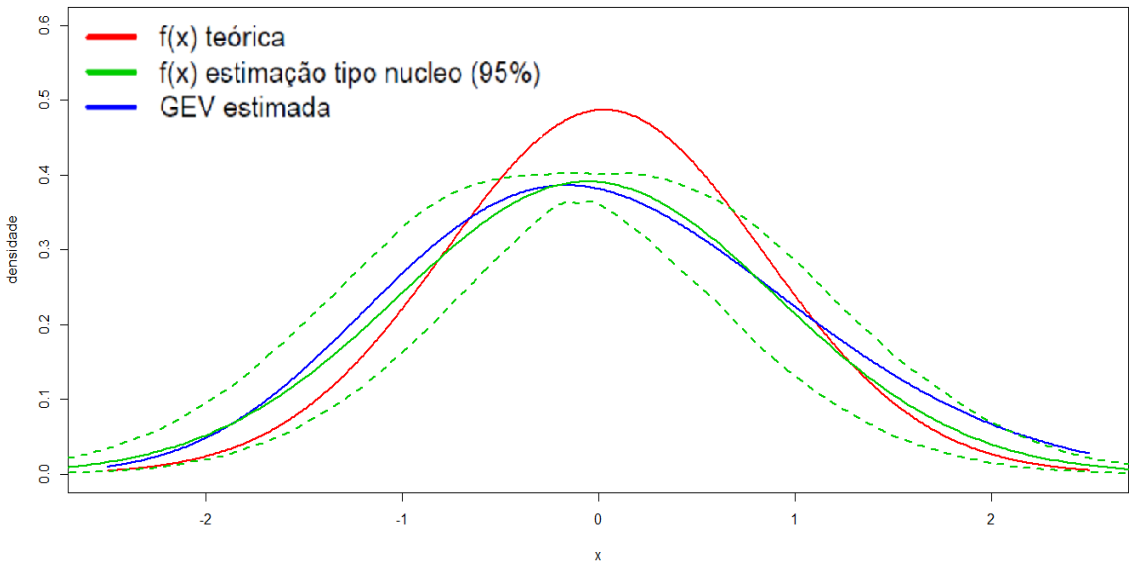


Figura 13.: Gráfico das densidades dos mínimos dos máximos da Normal(0,1): f(x) teórica; f(x) estimada tipo-núcleo, com linha a cheio para a média das estimativas e linhas a tracejado para os limites da banda de confiança; e GEVD obtida para as medianas das estimativas dos parâmetros GEV

Observando a Tabela 6, tem-se que os parâmetros de l*ocation* e de *shape* apresentam valores inteiramente negativos, enquanto que o parâmetro *scale* toma valores positivos. Comparando os valores de *shape* deste caso ao caso dos máximos da Normal, verifica-se que estes não apresentam uma diferença significativa.

Tabela 6.: Estatísticas descritivas para as estimativas dos parâmetros GEV da distribuição dos mínimos dos máximos da Normal(0,1)

	location (λ)	scale (δ)	shape (γ)
min	-0.4213	0.7877	-0.2247
25%	-0.3945	0.9635	-0.2102
50%	-0.3807	0.9742	-0.2031
75%	-0.3670	0.9790	-0.1895
max	-0.3282	1.0214	-0.1680
média	-0.3790	0.9594	-0.2008
desvio padrão	0.0223	0.0377	0.0132

5.3 COMPARAÇÃO DE DISTRIBUIÇÕES

Depois de se analisar graficamente as diferentes distribuições, optou-se por fazer uma análise comparativa das três distribuições para confirmar qual a função que melhor aproxima a estimação não paramétrica da função de densidade. Para tal, decidiu-se calcular o erro quadrático médio e a média dos erros absolutos.

O erro quadrático médio (EQM) define-se como sendo a média da diferença ao quadrado entre duas funções, $\hat{f}_1(x_i)$ e $\hat{f}_2(x_1)$, num determinado domínio

$$EQM = \frac{1}{400}\sum_{i=1}^{400}\left(\hat{f}_1(x_i) - \hat{f}_2(x_i)\right)^2 \tag{26}$$

onde x_i com $i = 1, ..., 400$ representa a discretização do domínio da função.

A média dos erros absolutos (MEA) consiste em calcular a média da diferença absoluta entre duas funções, $\hat{f}_1(x_i)$ e $\hat{f}_2(x_1)$, num determinado domínio.

$$MEA = \frac{1}{400}\sum_{i=1}^{400}\left|\hat{f}_1(x_i) - \hat{f}_2(x_i)\right| = \frac{1}{400}\sum_{i=1}^{400}|e_i| \tag{27}$$

Sendo o objetivo avaliar qual a função que melhor aproxima os dados, tem-se $\hat{f}_1(x_i)$ como sendo o conjunto de valores obtidos para a função densidade teórica (TEO) ou GEV, e $\hat{f}_2(x_i)$ como sendo o conjunto de valores obtidos pela estimação tipo-núcleo (SIM).

Note-se que, para o efeito de comparação de resultados, os valores obtidos pelos dois métodos são melhores quanto mais próximos estiverem do valor 0.

5.3.1 *Uniforme(0,1)*

As distribuições dos máximos e dos mínimos dos máximos da Uniforme$(0,1)$, representadas pelas Figuras 8 e 11, indicavam uma melhor aproximação dos dados simulados pela função densidade teórica. De um ponto de vista analítico, confirma-se que a função densidade teórica é a que melhor se ajusta.

Tabela 7.: Comparação de distribuições dos máximos da Uniforme(0,1)

	EQM	MEA
GEV vs SIM	0.0739	0.2412
TEO vs SIM	0.0161	0.0902

Nas Tabelas 7 e 8, apresentam-se os resultados dos EQM e das MEA da função densidade teórica e da função GEV, em relação à estimação não paramétrica da função densidade dos dados simulados para as distribuições dos máximos e dos mínimos dos máximos da Uniforme. Nos dois casos, tem-se que o erro é inferior para a função densidade teórica, o que está em concordância com as representações gráficas.

Tabela 8.: Comparação de distribuições dos mínimos dos máximos da Uniforme(0,1)

	EQM	MEA
GEV vs SIM	0.0967	0.2738
TEO vs SIM	0.0390	0.1720

5.3.2 *Exponencial(1)*

Quanto à Tabela 9, relativa às comparações de distribuições dos máximos da Exponencial(1), obtiveram-se resultados análogos ao caso da Uniforme$(0,1)$, observando que o EQM é duas vezes menor no caso da função densidade teórica. Concluí-se novamente que a função que melhor se adapta aos dados é a função densidade teórica.

Tabela 9.: Comparação de distribuições dos máximos da Exponencial(1)

	EQM	MEA
GEV vs SIM	0.0011	0.0128
TEO vs SIM	0.0005	0.0054

Relativamente à comparação de distribuições dos mínimos dos máximos da Exponencial(1), representada na Tabela 10, tem-se que, em relação ao cálculo do EQM, a função GEV se superioriza. No entanto, no cálculo da MEA, obteve-se um valor ligeiramente mais baixo para a função densidade teórica, o que leva a crer que as duas distribuição aproximam-se de igual forma. Note-se que a análise da Figura 12 parecia indicar que a função densidade teórica se aproximava melhor do que a GEV.

Tabela 10.: Comparação de distribuições dos mínimos dos máximos da Exponencial(1)

	EQM	MEA
GEV vs SIM	0.0011	0.0129
TEO vs SIM	0.0012	0.0119

5.3.3 *Normal(0,1)*

A Tabela 11 mostra claramente que a função densidade teórica se assemelha mais à estimação não paramétrica, calculada para os dados simulados, do que a função GEV. Recorde-se que esse era já o resultado previsto após a análise da Figura 10.

Tabela 11.: Comparação de distribuições dos máximos da Normal(0,1)

	EQM	MEA
GEV vs SIM	0.0005	0.0139
TEO vs SIM	0.0002	0.0095

Na Tabela 12, a função GEV evidencia-se como sendo a melhor aproximação para a distribuição dos mínimos dos máximos da Normal, o que já era expectável. O EQM e a MEA mostram resultados mais próximos de 0 para a função GEV.

Tabela 12.: Comparação de distribuições dos mínimos dos máximos da Normal(0,1)

	EQM	MEA
GEV vs SIM	0.0002	0.0100
TEO vs SIM	0.0010	0.0204

5.4 ANÁLISE DA DISTORÇÃO DOS VALORES

A análise da distorção dos valores consiste em avaliar a diferença entre os resultados dos valores obtidos na consulta de elementos e os verdadeiros valores inseridos. Calcula-se o vetor de distorções, isto é, a diferença

entre o vetor dos graus de confiança ($vetor_{c_{elem_i}}$), criado no método de inserção, e o vetor obtido no final do método de consulta (*minimum*). Essa diferença deverá ser sempre positiva, pois na consulta de determinado elemento, o valor ou equivale ao grau de confiança inicialmente inserido ou é superior. Para as três distribuições estudadas, analisa-se o número de elementos distorcidos, a taxa de elementos distorcidos, a distorção média (restrita aos elementos distorcidos) e a variação média geral para diversos fatores de preenchimento ($f = 1, 2, 4$ e 8). Recorda-se que a taxa de ocupação do FBL ideal é de 50%, o que equivale ao fator de preenchimento 1. Quando $f = 2$, $f = 4$ ou $f = 8$, duplica-se, quadruplica-se ou octuplica-se, respetivamente, o número de elementos a inserir.

No caso da distribuição Uniforme$(0, 1)$ observando a Tabela 13, tem-se que, em 103831 elementos, 153 elementos consultados não deram o resultado certo, ou seja, 0.147% de elementos são distorcidos e exibem uma distorção média de 0.130 sobre um intervalo de $[0, 1]$.

Tabela 13.: Análise de distorções na Uniforme(0,1)

	tamanho vetor, A	nº elementos distorcidos, B	taxa de elementos distorcidos, $\frac{B}{A} * 100$ (%)	distorção média em B	distorção média em A
f=1	103831	153	0.147	0.130	0.00019
f=2	207662	5795	2.791	0.165	0.00460
f=4	415324	86468	20.819	0.221	0.04601
f=8	830648	446016	53.695	0.308	0.16538

Note-se que a variação do fator de preenchimento influencia drasticamente os resultados, o que confirma que a taxa de ocupação ideal será de 50%.

Quanto à distribuição Exponencial(1), observa-se na Tabela 14, para o fator de preenchimento $f = 1$, que existem 131 elementos distorcidos e a variação média das distorções é de 0.164, sendo que o intervalo é $[0, 13.785]$, enquanto que para $f = 4$, o número de elementos distorcidos é de 86468 em 415324, o que resulta numa taxa de 20.819% de elementos distorcidos.

Tabela 14.: Análise de distorções na Exponencial(1)

	tamanho vetor, A	nº elementos distorcidos, B	taxa de elementos distorcidos, $\frac{B}{A} * 100$ (%)	distorção média em B	distorção média em A
f=1	103831	131	0.126	0.164	0.00021
f=2	207662	5884	2.833	0.243	0.00688
f=4	415324	86754	20.883	0.360	0.07518
f=8	830648	446222	53.720	0.621	0.33360

Na Tabela 15, continua a comprovar-se que o fator de preenchimento igual a 1 é o ideal. A taxa de elementos distorcidos é de 0.143% para $f = 1$, sendo que se duplica o número de elementos a inserir, a taxa de elementos distorcidos aumenta mais de 20 vezes.

Tabela 15.: Análise de distorções na Normal(0,1)

	tamanho vetor, A	nº elementos distorcidos, B	taxa de elementos distorcidos, $\frac{B}{A} * 100$ (%)	distorção média em B	distorção média em A
f=1	103831	148	0.143	0.607	0.00087
f=2	207662	5958	2.869	0.656	0.1882
f=4	415324	86552	20.839	0.764	0.15921
f=8	830648	445872	53.678	0.947	0.50833

Em todos os casos conclui-se que o fator de preenchimento $f = 1$ é o mais adequado. Observa-se que, para todas as distribuições analisadas, com $f = 8$, tem-se uma taxa de elementos distorcidos superior a

50%, isto é, na consulta de um elemento que foi inserido no filtro, a probabilidade do elemento não pertencer é superior a 50%.

6

CONCLUSÃO

Nesta tese faz-se um estudo estatístico da técnica dos FBLs, técnica essa que permite guardar, probabilisticamente, associações de elementos a valores numéricos. Com o objetivo de analisar os erros de estimação evidenciados aquando da consulta dos elementos nos FBLs, compararam-se duas funções de densidade. A função GEV cujos parâmetros dependem dos dados obtidos na simulação e, a função teórica deduzida para a densidade, que depende da distribuição subjacente aos valores numéricos inseridos, e também das características dimensionais do filtro. Adicionalmente, efetuou-se um estudo à análise da distorção induzida pelos erros.

Na comparação das funções relativas ao caso da Uniforme(0,1), a nível das representações gráficas, conclui-se que a função teórica deduzida para a densidade é claramente melhor do que a GEV, apesar da última ser calibrada pelos dados simulados. Quando se fez a comparação através do cálculo dos EQM, obtiveram-se os resultados 0.0739 e 0.0161, respetivamente, para a função GEV e teórica. Relativamente à MEA, obteve-se também um valor inferior para o caso da função densidade teórica, o que confirma a conclusão anterior. No que respeita à distribuição dos mínimos dos máximos da Uniforme(0,1), a situação repete-se.

Comparando as funções graficamente, a função densidade teórica parece acompanhar a função resultante da simulação, ligeiramente melhor do que a função GEV. Esta ideia foi consolidada na comparação dos EQM e MEA, pois nos dois cálculos obtiveram-se valores mais baixos no caso da função densidade teórica.

Quanto ao caso da distribuição Exponencial(1), pelas representações gráficas para as distribuições dos máximos e dos mínimos dos máximos da Exponencial(1), as duas funções são muito semelhantes. No entanto a função densidade teórica parece levemente melhor, conclusão corroborada pela comparação das MEA. No entanto, na comparação dos resultados dos EQM, verifica-se o contrário. Este conflito entre os resultados, MEA *versus* EQM, aponta para uma equivalência entre as duas aproximações.

Em relação à distribuição Normal(0,1), na comparação gráfica das distribuições dos máximos da Normal(0,1), é prematuro tirar conclusões, dado a similaridade das funções. No que respeita ao cálculo do EQM e da MEA, os resultados obtidos para a função densidade teórica evidenciaram-se pela positiva. Contrariamente às conclusões apresentadas até agora, no caso das distribuições dos mínimos dos máximos da Normal(0,1), a representação gráfica da função GEV parece melhor do que a teórica, o que é confirmado pelos resultados obtidos nos EQM e nas MEA.

Pela análise da distorção, comprovou-se que os FBLs são muito eficazes. Para as três distribuições estudadas, obteve-se, para um filtro com fator de preenchimento igual a 1, uma taxa de elementos distorcidos inferior a 0.15%. Isto é, em 103831 elementos inseridos no filtro, existem menos de 155 elementos distorcidos. Quanto ao factor de preenchimento, e seguindo os estudos mencionados no Capítulo 2 desta tese,

partiu-se do princípio de que o número de posições preenchidas deve ser cerca de 50% da dimensão do filtro, o que foi comprovado derivado ao facto da taxa de elementos distorcidos aumentar consideravelmente à medida que se aumentava o factor de preenchimento.

Tendo-se observado uma distorção pouco expressiva nos elementos guardados no FBL, quando considerado taxas de enchimento standard, pode-se concluir que esta técnica preserva em grande medida a fidelidade dos dados armazenados, bem como permite reduções expressivas na memória associada ao seu armazenamento.

BIBLIOGRAFIA

Paulo Sérgio Almeida, Carlos Baquero, Nuno Preguiça, and David Hutchison. *Scalable Bloom Filters*, volume 101. 2007. doi: http://dx.doi.org/10.1016/j.ipl.2006.10.007. URL http://www.sciencedirect.com/science/article/pii/S0020019006003127.

Prosenjit Bose, Hua Guo, Evangelos Kranakis, Anil Maheshwari, Pat Morin, Jason Morrison, Michiel Smid, and Yihui Tang. *On the false-positive rate of Bloom filters*, volume 108. 2008. doi: http://dx.doi.org/10.1016/j.ipl.2008.05.018. URL http://www.sciencedirect.com/science/article/pii/S0020019008001579.

Andrei Broder and Michael Mitzenmacher. *Network Applications of Bloom Filters: A Survey*, volume 1. 2004. doi: 10.1080/15427951.2004.10129096. URL http://dx.doi.org/10.1080/15427951.2004.10129096.

Bernard Chazelle, Joe Kilian, Ronitt Rubinfeld, and Ayellet Tal. *The Bloomier Filter: An Efficient Data Structure for Static Support Lookup Tables*. SODA '04. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2004. ISBN 0-89871-558-X. URL http://dl.acm.org/citation.cfm?id=982792.982797.

Li Fan, Pei Cao, Jussara Almeida, and Andrei Z. Broder. *Summary Cache: A Scalable Wide-area Web Cache Sharing Protocol*, volume 8. IEEE Press, Piscataway, NJ, USA, June 2000. doi: 10.1109/90.851975. URL http://dx.doi.org/10.1109/90.851975.

Rui Lima, Carlos Baquero, and Hugo Miranda. *Broadcast Cancellation in Search Mechanisms*. SAC '13. ACM, New York, NY, USA, 2013. ISBN 978-1-4503-1656-9. doi: 10.1145/2480362.2480467. URL `http://doi.acm.org/10.1145/2480362.2480467`.

Rui Lima, Carlos Baquero, and Hugo Miranda. *Adaptive Broadcast Cancellation Query Mechanism for Unstructured Networks*. Sept 2015. doi: 10.1109/NGMAST.2015.41.

Rolf-Dieter Reis and Michael Thomas. *Statistical Analysis of Extreme Values with Applications to Insurance, Finance, Hydrology and Other Fields*. Addison-Wesley, 2007.

Sean C. Rhea and John Kubiatowicz. *Probabilistic location and routing*, volume 3. 2002. doi: 10.1109/INFCOM.2002.1019375.

Phillip Rogaway and Thomas Shrimpton. *Cryptographic Hash-Function Basics: Definitions, Implications, and Separations for Preimage Resistance, Second-Preimage Resistance, and Collision Resistance*. Springer Berlin Heidelberg, Berlin, Heidelberg, 2004. ISBN 978-3-540-25937-4. doi: 10.1007/978-3-540-25937-4_24. URL `http://dx.doi.org/10.1007/978-3-540-25937-4_24`.

A

AMOSTRA DOS FILTROS DE BLOOM

A.1 UNIFORME(0,1)

```
>bloom[1:100]
  [1] -99.000   0.556 -99.000 -99.000 -99.000   0.389   0.360
      0.851 -99.000   0.173 -99.000   0.630   0.814
 [14] -99.000 -99.000   0.542   0.011   0.585   0.270 -99.000
   -99.000 -99.000 -99.000 -99.000 -99.000   0.522
 [27] -99.000 -99.000 -99.000   0.740   0.802   0.633   0.734
      0.554   0.224 -99.000   0.542   0.508 -99.000
 [40]   0.037 -99.000   0.607   0.940   0.106   0.864 -99.000
   -99.000   0.547   0.377   0.806 -99.000   0.556
 [53] -99.000   0.939 -99.000 -99.000 -99.000 -99.000   0.690
   -99.000   0.804   0.841 -99.000   0.943 -99.000
 [66] -99.000   0.117 -99.000   0.840 -99.000 -99.000 -99.000
      0.087   0.301 -99.000   0.228 -99.000 -99.000
 [79] -99.000 -99.000 -99.000 -99.000   0.283   0.453 -99.000
   -99.000 -99.000   0.644 -99.000   0.885 -99.000
 [92]   0.613 -99.000   0.808   0.337   0.410   0.977 -99.000
   -99.000   0.643
```

A.2 EXPONENCIAL(1)

```
> bloom[1:100]
  [1] -99.000   0.291 -99.000 -99.000 -99.000 -99.000 -99.000
      0.641 -99.000   1.467 -99.000   1.888   3.688
 [14] -99.000 -99.000   0.703   1.198   0.229   0.634   0.354
   -99.000 -99.000 -99.000 -99.000 -99.000   0.910
 [27] -99.000 -99.000 -99.000   2.633   0.604   0.798   0.565
     1.135   1.902 -99.000   0.069   1.951 -99.000
 [40] -99.000 -99.000 -99.000   0.490   0.720   2.131 -99.000
   -99.000   3.862   0.303   0.034 -99.000   0.053
 [53] -99.000   0.376 -99.000 -99.000   0.431 -99.000   0.409
   -99.000   0.622   4.353 -99.000   0.160   2.029
 [66] -99.000   4.018 -99.000   0.097 -99.000   0.395 -99.000
     2.507   2.217 -99.000 -99.000 -99.000 -99.000
 [79] -99.000 -99.000 -99.000 -99.000   1.854   0.515 -99.000
   -99.000 -99.000   0.101   3.860   1.905 -99.000
 [92]   2.179 -99.000 -99.000   3.186   1.332   1.822   1.209
   -99.000   3.209
```

A.3 NORMAL(0,1)

```
> bloom[1:100]
  [1] -99.000   2.179 -99.000 -99.000 -99.000   0.754  -1.700
      1.039 -99.000 -99.000 -99.000  -0.608  -1.223
 [14] -99.000 -99.000  -0.667   0.353 -99.000  -0.718   0.672
      0.985 -99.000 -99.000 -99.000   0.688   2.017
 [27]  -0.909 -99.000 -99.000  -0.042  -0.056   0.340   2.404
     -1.127   2.618 -99.000  -0.676   0.697 -99.000
 [40]  -0.506 -99.000  -0.219   2.162  -0.654   0.403 -99.000
     -99.000  -0.512  -0.001   0.474 -99.000  -0.893
 [53] -99.000 -99.000 -99.000 -99.000   0.006 -99.000   0.537
     -99.000 -99.000   0.997 -99.000  -0.338   0.582
 [66] -99.000   0.475 -99.000 -99.000 -99.000 -99.000 -99.000
      0.864  -0.021 -99.000   1.060 -99.000 -99.000
 [79] -99.000 -99.000 -99.000 -99.000   0.937   0.538 -99.000
     -99.000 -99.000  -1.175  -0.848   0.402 -99.000
 [92]   0.287  -2.507  -1.494   0.881 -99.000  -0.299 -99.000
     -99.000   0.716
```