UNIVERSIDADE DE LISBOA
FACULDADE DE CIÊNCIAS
DEPARTAMENTO DE ESTATÍSTICA E INVESTIGAÇÃO OPERACIONAL

Métodos para Estimação dos Parâmetros da Distribuição de Pareto Generalizada: novas contribuições

André Simões Sintra

Mestrado em Estatística e Investigação Operacional
Especialização em Estatística e Investigação Operacional

Dissertação orientada por:
Prof. Doutora Patrícia Cortés de Zea Bermudez

2017

Agradecimentos

À Professora Doutora Patrícia de Zea Bermudez, por toda a orientação prestada no desenvolvimento desta dissertação, quer na partilha de ideias, quer na disponibilidade prestada. O meu especial agradecimento por toda a dedicação e paciência ao longo desta jornada que, por vezes, pareceu não ter fim.

À Professora Doutora Maria da Conceição da Fonseca, pelas constantes palavras de apoio e motivação no decorrer deste projeto. Um particular agradecimento pela pronta disponibilidade na revisão deste trabalho.

A todos os professores do Departamento de Estatística e Investigação Operacional que, ao longo de todo o meu percurso académico, permitiram a aquisição dos mais vastos conhecimentos nestas áreas, que culminaram na realização deste trabalho.

Aos meus colegas do Mestrado em Estatística e Investigação Operacional, pela partilha de bons momentos e pelo companheirismo demonstrado. Um agradecimento especial à Ana Sofia Carvalho, Ângela Santos, Bruno Oliveira, Filipa David, Isabel Candoso e Mariana Monteiro por terem estado sempre presentes ao longo desta importante etapa.

Aos meus colegas e amigos da Faculdade de Ciências da Universidade de Lisboa, por terem sempre demonstrado um carinho especial em todas os momentos do meu percurso académico. À minha Família Académica (Raquel Carmona, Hardica Cangi, Tatiana Baptista, Raquel Bernardino, Daniel Santos, Ana Carolina Guerreiro, Francisco Picado) e ao Grupo dos 9 (Cláudia Paradela, Diogo Costa, Joana Vasconcellos Dias, Joana Tiago, Mariana Pereira, Nuno Pinheiro, Vasco Alves) pela amizade incondicional e todos os momentos partilhados.

Aos meus familiares, por toda a paciência e carinho demonstrados nas melhores e piores alturas desta longa viagem a nível pessoal e profissional. Sem eles, não teria sido possível atingir o patamar em que me encontro neste momento.

Muito mais pessoas contribuíram para o bom rumo que me trouxe onde me encontro atualmente. A todos eles, um bem haja.

Resumo

A Teoria de Valores Extremos (TVE) surge naturalmente na presença de observações muito elevadas ou muito reduzidas. A TVE tem sido usada em diversas áreas, tais como seguros, hidrologia e ambiente. A distribuição generalizada de valores extremos e a distribuição de Pareto generalizada (GPD) destacam-se como sendo as distribuições usadas na modelação de observações extremas. A GPD foi introduzida na literatura por Pickands (1975) como a distribuição limite da amostra de excessos ou excedências acima de um limiar suficientemente elevado.

A estimação dos parâmetros da GPD é um assunto abordado frequentemente na literatura. Métodos clássicos como a Máxima Verosimilhança, Momentos ou Momentos Ponderados de Probabilidade apresentam certas limitações, pelo que têm vindo a ser desenvolvidos outros métodos de estimação ao longo dos tempos, tais como o Método dos Percentis Elementares, proposto por Castillo & Hadi (1997), e os estimadores *empirical bayes* desenvolvidos por Zhang & Stephens (2009). Certos estimadores apresentam boas propriedades quando a GPD tem cauda pesada, enquanto que outros são preferíveis quando a distribuição tem cauda leve. Este facto sugere a possibilidade de desenvolver procedimentos que permitam escolher, de forma adaptativa, os melhores estimadores para os parâmetros da GPD, consoante o peso de cauda da distribuição subjacente.

Com este trabalho, pretendeu-se mostrar que a combinação de métodos de estimação pode resultar na obtenção de procedimentos mais eficazes. Para este efeito, recorreu-se a algoritmos de otimização não linear, técnicas de ajustamento polinomial e metodologias de classificação, como o perceptrão. A avaliação destes procedimentos foi feita com recurso a um estudo de simulação, que produziu resultados muito promissores em termos de viés e raiz quadrada do erro quadrático médio. Os procedimentos foram aplicados a dados reais das áreas do desporto (triplo salto) e da atividade seguradora (danos corporais associados ao ramo automóvel).

Palavras-Chave: Distribuição de Pareto Generalizada, Teoria de Valores Extremos, Métodos para Estimação de Parâmetros, Métodos de Decisão.

Abstract

Extreme value theory (EVT) comes naturally whenever we are faced with very large or very small observations. EVT has been applied to several areas such as insurance, hydrology and environment. The generalized extreme value distribution and the generalized Pareto distribution (GPD) stand out as the distributions used for modeling extreme observations. The GPD was introduced in the literature by Pickands (1975) as the limit distribution of the excesses or exceedances above a sufficiently high threshold.

The estimation of the parameters of the GPD is a topic that is frequently addressed in the literature. Classical methods like maximum likelihood, moments or probability weighted moments have several limitations and consequently other estimation methods have been developed such as the Elemental Percentile Method, proposed by Castillo & Hadi (1997) and the empirical bayes estimators developed by Zhang & Stephens (2009). Some estimators have good properties when the GPD is heavy-tailed, whereas others perform better for light tails. These features suggest the possibility of developing procedures that enable the choice, in an adaptive manner, of the best estimation method for the parameters of the GPD according to the tail weight of the underlying distribution.

The purpose of this thesis is to show that the combination of estimation methods may result in better procedures. For that purpose, non-linear optimization algorithms were used, as well as polynomial fits and classification techniques, such as the perceptron. The performance of these techniques was assessed by means of a simulations study, which produced very good results both in terms of bias and root mean squared error. The procedures were applied to sports and insurance datasets.

Keywords: Generalised Pareto Distribution, Extreme Value Theory, Parameter Estimation Methods, Decision Methods.

Índice

Lista de Figuras

Lista de Tabelas

Lista de Algoritmos

"O caminho faz-se caminhando."
Hélder Sintra

"Nada nos demove da nossa rota."
Inês Candeias

Capítulo 1

Introdução

A Teoria de Valores Extremos (TVE) é o ramo da estatística que aborda questões da vida real que requerem a estimação de acontecimentos raros e acerca dos quais os dados recolhidos são escassos ou mesmo inexistentes. Esta escassez ou inexistência de informação deve-se ao tipo de acontecimentos estudados neste ramo, que passam pela estimação de probabilidades ou quantis com valores demasiado pequenos ou elevados, respetivamente, para os quais habitualmente não há dados suficientes para fornecer estimativas fiáveis. Esta área pode ser explorada em diversas obras e publicações relevantes (e.g., Beirlant *et al.* [5], Coles [12], Embrechts *et al.* [15], Gomes *et al.* [21], entre outros).

No contexto da TVE, a Distribuição de Pareto Generalizada (GPD) foi apresentada pela primeira vez por Pickands, em 1975, como uma distribuição para modelação dos excessos de uma amostra (ou excedências) acima de um nível suficientemente elevado. Esta distribuição é bastante versátil, uma vez que permite a modelação de dados observados de populações cujas distribuições subjacentes possuem pesos de cauda diversos. A sua adaptabilidade é propiciada pelos valores tomados pelo parâmetro de forma, que tem uma importância particular no âmbito da TVE. A utilização da GPD tem sido feita extensivamente em diversos campos científicos, sendo mais referidas na literatura aplicações nas áreas dos Seguros, Fiabilidade, Finanças, Meteorologia e Ambiente. No Capítulo 2 será feita uma apresentação detalhada desta distribuição do ponto de vista probabilístico e será mostrado o impacto do parâmetro de forma tanto nas expressões das estatísticas descritivas, como no formato dos gráficos distribucionais respetivos.

A estimação dos parâmetros da GPD é um assunto abordado frequentemente devido a aspetos relacionados com as limitações dos estimadores clássicos referidos na literatura. Um deles prende-se com a existência de estimativas dos parâmetros apenas para determinados valores do parâmetro de forma da GPD. Apesar de ser possível calcular o seu valor, as propriedades teóricas associadas podem ser desvantajosas no processo de inferência se o verdadeiro valor do parâmetro de forma da distribuição GPD subjacente à amostra tomar determinados valores. Outro aspeto está associado à consistência das estimativas obtidas relativamente à amostra utilizada no processo de estimação. No caso em que a distribuição GPD subjacente à amostra tem caudas leves, o seu suporte é limitado superiormente por um valor finito calculado como função dos parâmetros da distribuição. Em particular, se for obtida uma amostra a partir de uma GPD nestas condições, todos os seus valores serão inferiores ao limite superior do suporte. No entanto, se esta mesma amostra for utilizada para estimar os parâmetros da distribuição subjacente, as estimativas obtidas podem não ser consistentes com os dados, isto é, quando forem utilizadas para calcular o limite superior do suporte, não é certo que todas as observações da amostra sejam inferiores

a este valor. No Capítulo 3 serão apresentados os métodos de estimação mais utilizados no âmbito da estimação dos parâmetros da GPD e serão analisadas as limitações das estimativas obtidas por cada uma das técnicas.

O Método de Máxima Verosimilhança é um dos métodos mais populares para estimação paramétrica devido às propriedades assintóticas dos seus estimadores. Relativamente à GPD, esta técnica produz boas estimativas quando a amostra observada provém de uma população cuja distribuição subjacente tem caudas pesadas. Por outro lado, tal como outras técnicas, este método também apresenta limitações quando é utilizado para fazer estimação a partir de amostras provenientes de uma população cuja distribuição subjacente tem caudas leves. No Capítulo 4 será apresentada uma técnica para obtenção de estimativas de Máxima Verosimilhança consistentes com os dados através de metodologias de otimização não linear, bem como métodos *empirical bayes* alternativos que conseguem produzir estimativas mais robustas do que as estimativas de Máxima Verosimilhança.

Apesar das limitações referidas anteriormente, é possível que as estimativas obtidas por alguns destes métodos possuam boas propriedades estatísticas dentro das suas condições de validade e que estudos que as utilizem permitam retirar conclusões confiáveis. Para além disto, algumas destas técnicas podem ser consideradas complementares no sentido em que as limitações de umas podem ser contornadas recorrendo a técnicas alternativas com melhor desempenho face às mesmas condições. Assim, é possível concluir que a combinação de métodos para estimação paramétrica resulta na obtenção de algoritmos com menos limitações e que permitem obter estimativas mais robustas. Com base neste conceito, serão apresentados nos Capítulos 5 e 6 dois métodos híbridos que combinam técnicas de estimação robustas de modo a considerar amostras observadas de populações cuja distribuição subjacente pode ter qualquer peso de cauda.

Para que seja possível comparar o desempenho de cada uma das técnicas de estimação apresentadas através de simulação, serão apresentadas no Capítulo 7 medidas de desempenho para avaliar a qualidade das estimativas obtidas e o comportamento dos métodos híbridos ao nível da escolha da técnica clássica escolhida para aplicação. Para complementar esta análise com valores reais, serão apresentados no Capítulo 8 os resultados da aplicação dos métodos a dois casos de estudo com comportamentos de cauda distintos, de forma a avaliar os métodos abordados.

Capítulo 2

Distribuição de Pareto Generalizada (GPD)

Considera-se que uma variável aleatória X segue uma distribuição GPD, com parâmetros de forma, localização e escala designados por k ($k \in \mathbb{R}$), μ ($\mu \in \mathbb{R}$) e σ ($\sigma > 0$), respetivamente, se a sua função de distribuição for dada pela expressão

$$F(x \mid k, \mu, \sigma) = \begin{cases} 1 - \left(1 + k\frac{x-\mu}{\sigma}\right)^{-\frac{1}{k}}, & k \neq 0, \\ 1 - \exp\left(-\frac{x-\mu}{\sigma}\right), & k = 0. \end{cases} \tag{2.1}$$

Uma particularidade desta distribuição prende-se com a variação do seu suporte consoante o valor do parâmetro de forma. Desta forma, se $k \geq 0$, o suporte da GPD é dado por

$$\mathscr{D}_X = \{x \in \mathbb{R} \mid x > \mu\}, \tag{2.2}$$

enquanto que, se $k < 0$, o suporte da GPD passa a ser

$$\mathscr{D}_X = \left\{x \in \mathbb{R} \mid \mu < x < \mu - \frac{\sigma}{k}\right\}. \tag{2.3}$$

Repare-se que, apenas neste último caso, o suporte da distribuição tem limite superior finito, definido por $\mu - \frac{\sigma}{k}$. A função densidade de probabilidade da GPD é dada por

$$f(x \mid k, \mu, \sigma) = \begin{cases} \frac{1}{\sigma}\left(1 + k\frac{x-\mu}{\sigma}\right)^{-\frac{1}{k}-1}, & k \neq 0, \\ \frac{1}{\sigma}\exp\left(-\frac{x-\mu}{\sigma}\right), & k = 0, \end{cases} \tag{2.4}$$

enquanto que a correspondente função quantil de ordem p, com $p \in (0,1)$, é representada por

$$Q(p \mid k, \mu, \sigma) = \begin{cases} \mu + \frac{\sigma}{k}\left[(1-p)^{-k} - 1\right], & k \neq 0, \\ \mu - \sigma\log(1-p), & k = 0. \end{cases} \tag{2.5}$$

A forma geral da mediana da GPD é obtida a partir da expressão (2.5) fazendo $p = 0.5$, resultando assim

a expressão

$$Q(0.5 \mid k,\mu,\sigma) = \begin{cases} \mu + \frac{\sigma}{k}\left(2^k - 1\right), & k \neq 0, \\ \mu + \sigma \log 2, & k = 0. \end{cases} \tag{2.6}$$

A GPD é uma distribuição bastante versátil, dado que muitas distribuições de probabilidade são casos particulares desta, consoante o valor que os seus parâmetros tomam. As principais distribuições nestas condições encontram-se indicadas abaixo:

- Se $k = 0$ e $\mu = 0$, reduz-se à distribuição Exponencial de valor médio σ;
- Se $k > 0$ e $\mu = \sigma/k$, coincide com a distribuição Pareto (tipo I) com parâmetro de forma $1/k$ e parâmetro de escala σ/k;
- Se $k = -1$ e $\mu = 0$, reduz-se à distribuição Uniforme definida em $(0, \sigma)$.

Para realçar a versatilidade da distribuição, apresentam-se na Figura 2.1 os gráficos da sua função densidade de probabilidade para vários valores de k, fixando $\mu = 0$ e $\sigma = 1$. Como consequência da variação do suporte em função do sinal de k, é possível observar valores muito elevados quando $k > 0$ (situações de caudas pesadas), enquanto que para $k < 0$ são obtidos valores mais moderados (situações de caudas leves ou curtas).

Conforme se pode verificar na literatura, a maioria dos métodos para estimação dos parâmetros da GPD incide apenas sobre os parâmetros de forma e escala. Nesta tese, será explorada apenas a versão biparamétrica da distribuição, denotada por GPD(k, σ). As funções que lhe estão associadas são obtidas fazendo $\mu = 0$ nas expressões (2.1) a (2.6). De Zea Bermudez & Kotz [48] referem no seu artigo algumas expressões de momentos desta distribuição. O n-ésimo momento da GPD(k, σ), calculado em torno de zero, é dado pela expressão

$$E(X^n) = n! \frac{\sigma^n \Gamma\left(\frac{1}{k} - n\right)}{k^{n+1}\Gamma\left(1 + \frac{1}{k}\right)}, \qquad k < \tfrac{1}{n}, \quad n \in \mathbb{N}, \tag{2.7}$$

onde $\Gamma(.)$ representa a função Gama. As expressões para o valor médio, a variância, a assimetria e a curtose da GPD são obtidas a partir da expressão (2.7) e são dadas, respetivamente, por

$$E(X) = \frac{\sigma}{1-k}, \quad k < 1, \tag{2.8}$$

$$Var(X) = \frac{\sigma^2}{(1-k)^2(1-2k)}, \quad k < \tfrac{1}{2}, \tag{2.9}$$

$$Skew(X) = \frac{2(1+k)\sqrt{1-2k}}{1-3k}, \quad k < \tfrac{1}{3}, \tag{2.10}$$

$$Kurt(X) = \frac{3(1-2k)(3+k+2k^2)}{(1-3k)(1-4k)} - 3, \quad k < \tfrac{1}{4}. \tag{2.11}$$

A GPD possui propriedades muito importantes que levam à sua frequente aplicação na área da Teoria de Valores Extremos, sendo uma delas a sua estabilidade em operações que envolvam excessos acima de

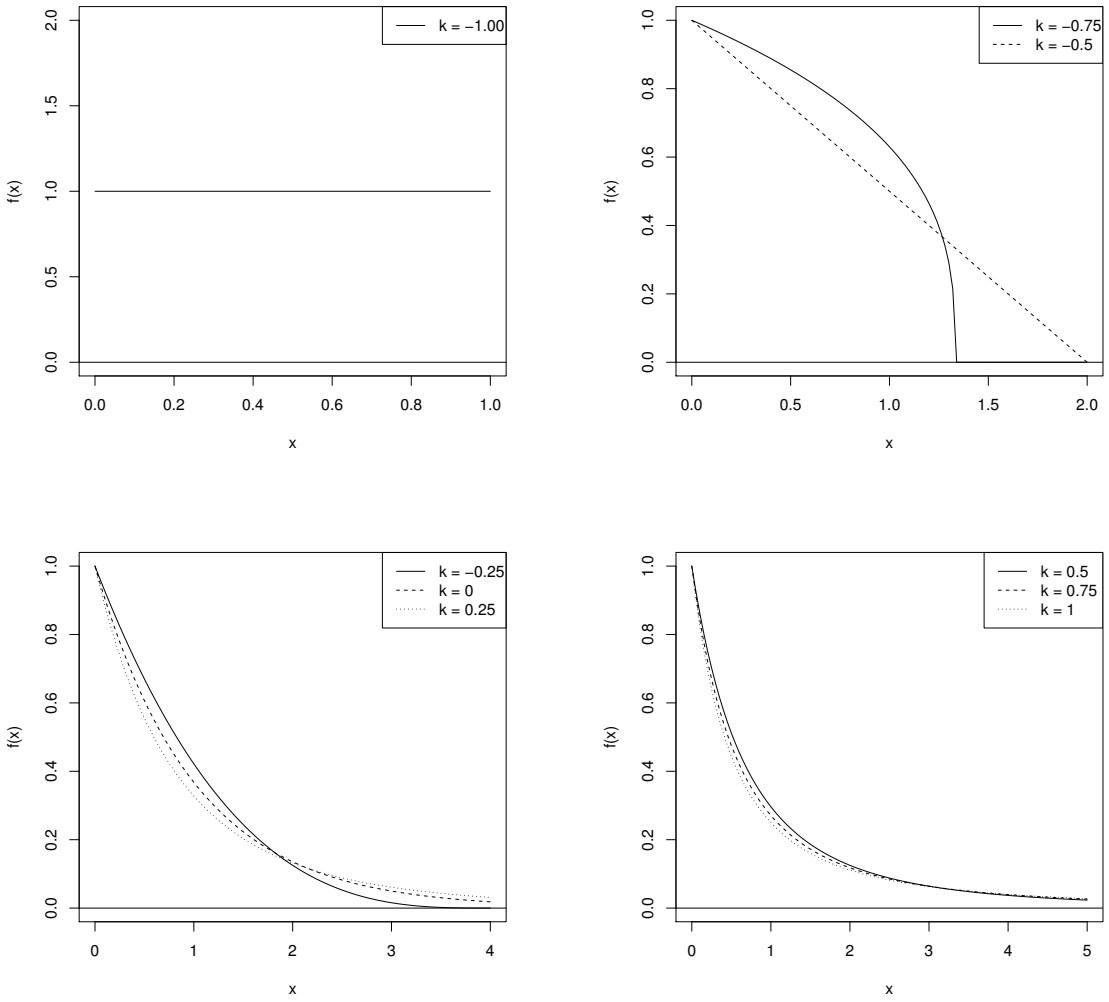

Figura 2.1: Função densidade de probabilidade da GPD para vários valores de k, fixando $\mu = 0$ e $\sigma = 1$

um determinado nível u $(u > 0)$. Mais concretamente, se X for uma variável aleatória com distribuição GPD(k, σ), então a variável aleatória $Y = X - u \mid X > u$ seguirá uma distribuição GPD$(k, \sigma + ku)$. Esta característica mostra que operações envolvendo excessos acima de um determinado nível não alteram o valor do parâmetro de forma da distribuição, sendo modificado apenas o valor do parâmetro de escala.

A escolha do nível u acima do qual é apropriado ajustar a GPD é uma tarefa minuciosa e tem sido abordada largamente na literatura [4,14,34,50]. A escolha de níveis mais baixos leva a que os estimadores para os parâmetros sejam mais enviesados, enquanto que a escolha de níveis demasiado elevados faz com que a variância dos mesmos aumente. O principal fator para este aumento é a dimensão reduzida da amostra de excessos que são considerados. Esta abordagem de ajustamento de um modelo paramétrico aos excessos de uma amostra, relativamente a um nível u fixado, é conhecida como Metodologia POT (*Peaks Over Threshold*) e é uma das técnicas de maior recurso na área de Valores Extremos.

De forma a auxiliar a escolha do nível u, é comum recorrer a representações gráficas que permitam obter informação acerca da qualidade do ajustamento face à escolha feita, sendo mais utilizados na prática os gráficos quantil-quantil (também conhecidos como QQ-Plot) e os gráficos das funções de excesso médio. No caso da GPD, a expressão da função de excesso médio é dada por

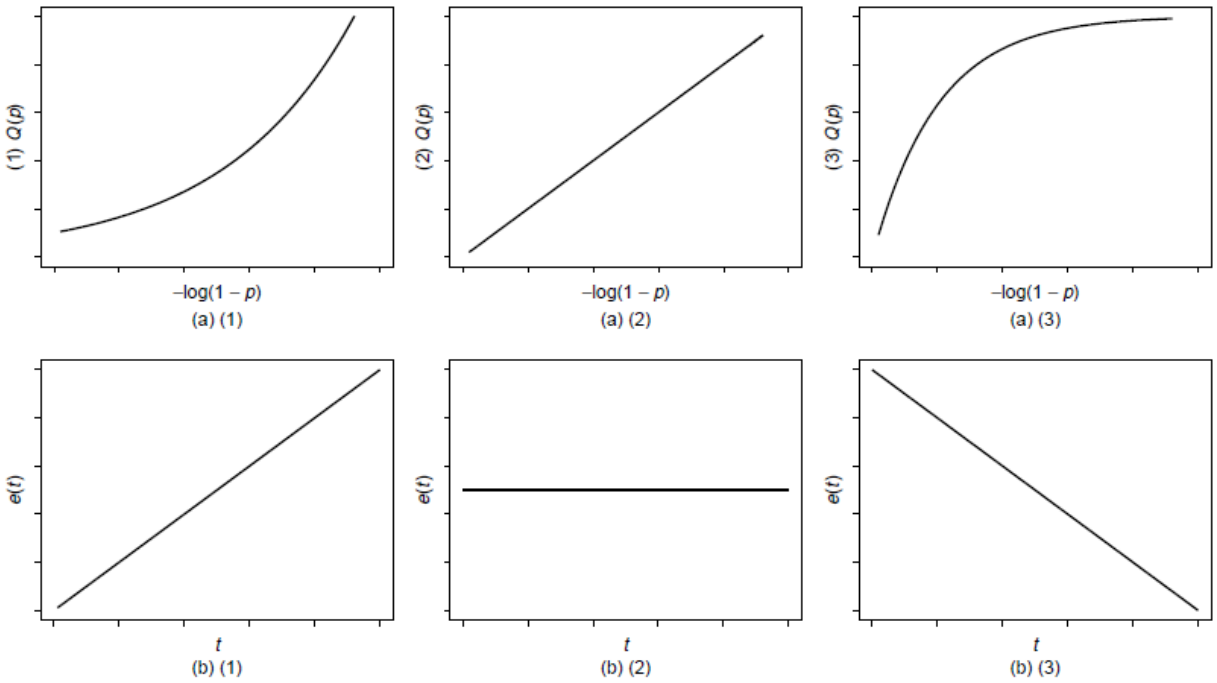

Figura 2.2: QQ-Plot Exponencial (a) e Gráfico de Excesso Médio (b) associados a distribuições de probabilidade com peso de cauda superior (1), igual (2) ou inferior (3) ao da distribuição Exponencial (Fonte: Beirlant *et al.* [5])

$$e(u) = E(X - u \mid X > u) = \frac{\sigma}{1-k} + u\frac{k}{1-k}, \quad k < 1, \quad u > 0, \quad \sigma + ku > 0. \tag{2.12}$$

Como consequência da linearidade da sua expressão, esta função é utilizada frequentemente para avaliar o peso da cauda da distribuição GPD subjacente aos excessos acima de um determinado nível u. O gráfico dos excessos médios deve apresentar um formato linear, com declive $\frac{k}{1-k}$ e ordenada na origem $\frac{\sigma}{1-k}$. O gráfico de $e(u)$, a par com o QQ-Plot, também permite obter informações acerca do peso de cauda da distribuição GPD subjacente à amostra dos excessos. Na Figura 2.2 são apresentados os gráficos dos QQ-Plots Exponenciais e da função de excesso médio conforme o peso de cauda considerado. Este tema será explorado com maior detalhe no Capítulo 5.

Na prática, como não se conhecem à partida as estimativas dos parâmetros da distribuição GPD subjacente à amostra dos excessos, é comum confrontar graficamente os excessos médios calculados para o correspondente nível u, que é estimado a partir da r-ésima estatística ordinal superior da amostra original, representada por $x_{(n-r):n}$. Desta forma, se a amostra considerada for $\mathbf{x} = (x_1, \ldots, x_n)$, a função de excesso médio empírica associada é dada pela expressão

$$e_n(u) = \frac{\sum_{i=1}^{n}(x_i - u)I(x_i > u)}{\sum_{i=1}^{n} I(x_i > u)}, \tag{2.13}$$

onde I representa a função indicatriz. Tal como foi deduzido através da expressão (2.12), o valor adequado para u corresponderá ao valor da estatística ordinal superior a partir da qual o gráfico de $e_n(u)$ apresentar uma forma linear.

Capítulo 3

Métodos Clássicos de Estimação dos Parâmetros da GPD

3.1 Método de Máxima Verosimilhança (ML)

O Método de Máxima Verosimilhança é considerado o método de eleição para estimação de parâmetros devido às boas propriedades que os estimadores resultantes possuem, tais como a consistência, normalidade assintótica e a invariância funcional.

Se $\mathbf{x} = (x_1, x_2, \ldots, x_n)$ for uma amostra observada de uma população X com distribuição GPD(k, σ), a função de verosimilhança que lhe está associada é dada por

$$L(k, \sigma \mid \mathbf{x}) = \prod_{i=1}^{n} f(x_i \mid k, \sigma) = \begin{cases} \sigma^{-n} \left[\prod_{i=1}^{n} \left(1 + k\frac{x_i}{\sigma}\right) \right]^{-\frac{1}{k}-1}, & k \neq 0, \\ \sigma^{-n} \exp\left(-\sum_{i=1}^{n} \frac{x_i}{\sigma} \right), & k = 0. \end{cases} \tag{3.1}$$

Este método consiste em determinar os valores de (k, σ) que maximizam a expressão (3.1). No entanto, é mais comum utilizar o seu logaritmo para facilitar a manipulação das expressões envolvidas. Esta nova expressão é conhecida como função de log-verosimilhança e a sua expressão é dada por

$$\log L(k, \sigma \mid \mathbf{x}) = \begin{cases} -n \log \sigma - \left(\frac{1}{k} + 1\right) \sum_{i=1}^{n} \log\left(1 + k\frac{x_i}{\sigma}\right), & k \neq 0, \\ -n \log \sigma - \sum_{i=1}^{n} \frac{x_i}{\sigma}, & k = 0. \end{cases} \tag{3.2}$$

Esta transformação não afeta o valor das estimativas obtidas porque o maximizante da expressão (3.1) é o mesmo da expressão (3.2). Isto deve-se ao facto da função logarítmica ser crescente e, ao compor com a função de verosimilhança, não haver alteração da monotonia original. Também apresenta a vantagem de converter os produtos em somas, o que facilita o manuseamento da expressão.

Na prática, considera-se apenas o ramo $k \neq 0$ da expressão (3.2) para determinar estas estimativas, uma vez que o ramo $k = 0$ trata-se de um caso limite da GPD. Repare-se que ao considerar que $k < -1$, vem que $\log L \to +\infty$ quando $\frac{\sigma}{k} \to -x_{n:n}$, onde $x_{n:n}$ representa o máximo da amostra observada. Então,

para que haja um máximo finito para a função de log-verosimilhança, é necessário impor a restrição $k \geq -1$, sendo assim possível concluir que esta função está definida apenas no conjunto

$$\mathscr{D}_L = \left\{-1 \leq k < 0, \frac{\sigma}{k} < -x_{n:n}\right\} \cup \{k > 0, \sigma > 0\}. \tag{3.3}$$

O maximizante da função (3.2) é obtido através da resolução do sistema

$$\begin{cases} \frac{\partial \log L}{\partial k} = \frac{1}{k^2} \sum_{i=1}^{n} \log\left(1 + k\frac{x_i}{\sigma}\right) - \left(\frac{1}{k} + 1\right) \sum_{i=1}^{n} \frac{x_i}{\sigma + k x_i} = 0, \\ \\ \frac{\partial \log L}{\partial \sigma} = -\frac{n}{\sigma} + \frac{k}{\sigma}\left(\frac{1}{k} + 1\right) \sum_{i=1}^{n} \frac{x_i}{\sigma + k x_i} = 0. \end{cases} \tag{3.4}$$

Apesar da existência de diversos métodos numéricos para a determinação de estimativas de máxima verosimilhança, como o célebre Método de Newton-Raphson para resolução do sistema anterior, um dos métodos de utilização mais conveniente é da autoria de Grimshaw [24] e consiste na redução da pesquisa de soluções num espaço bidimensional para um espaço unidimensional, através da utilização de uma transformação apropriada.

As propriedades assintóticas dos estimadores ML para os parâmetros da GPD, como a normalidade, consistência e eficiência, foram obtidas por Smith [41]. O autor mostra que apesar destas propriedades serem válidas apenas para $k > -0.5$, os estimadores de (k, σ) são assintoticamente normais com matriz de covariância dada por

$$\frac{1}{n} \begin{bmatrix} (1+k)^2 & \sigma(1+k) \\ \sigma(1+k) & 2\sigma^2(1+k) \end{bmatrix}. \tag{3.5}$$

Por vezes, a estimação dos parâmetros da GPD pelo MLE pode ser difícil, mesmo para $k > -1$. No seu artigo, Hosking & Wallis [26] referem que apesar deste método ser o mais eficiente para estimação dos parâmetros da GPD, os algoritmos utilizados para este efeito podem sofrer problemas de convergência mesmo para amostras grandes, e que são necessárias amostras de dimensão superior a 500 observações para o método ser mais eficiente que outros métodos tradicionais.

3.2 Método dos Momentos (MOM)

O método MOM tem sido extensivamente utilizado na estimação dos parâmetros da GPD e de outras distribuições univariadas. A sua popularidade está relacionada com a facilidade de cálculo das estimativas, uma vez que recorre somente à expressão dos momentos da distribuição de probabilidade envolvida neste processo.

Os estimadores MOM para os parâmetros da GPD(k, σ) obtêm-se facilmente através da utilização das expressões do valor médio e da variância indicadas pelas expressões (2.8) e (2.9), respetivamente. Cálculos diretos mostram que os estimadores MOM para k e σ são calculados através das expressões

$$\hat{k}_{MOM} = -\frac{1}{2}\left(\frac{\bar{X}^2}{S^2} - 1\right), \tag{3.6}$$

$$\hat{\sigma}_{MOM} = \frac{1}{2}\bar{X}\left(\frac{\bar{X}^2}{S^2}+1\right), \tag{3.7}$$

onde $\bar{X}$ e S^2 representam a média e variância amostrais, respetivamente. Note-se que estas estimativas só existem se $k < 0.5$, uma vez que a expressão (2.9) só está definida para esses valores do parâmetro de forma. Em particular, no caso em que $k < 0$, o limite superior do suporte da GPD é estimado por

$$-\frac{\hat{\sigma}}{\hat{k}} = \frac{\bar{X}\left(\bar{X}^2+S^2\right)}{\bar{X}^2-S^2}. \tag{3.8}$$

Hosking & Wallis [26] provam que os estimadores MOM para os parâmetros da GPD são assintoticamente normais para $k < 0.25$, com matriz de covariância dada por

$$\frac{1}{n}\frac{(1-k)^2}{(1-2k)(1-3k)(1-4k)}\begin{bmatrix} (1-2k)^2(1-k+6k^2) & \sigma(1-2k)(1-4k+12k^2) \\ \sigma(1-2k)(1-4k+12k^2) & 2\sigma^2(1-6k+12k^2) \end{bmatrix}. \tag{3.9}$$

Apesar da simplicidade de cálculo destes estimadores, as expressões envolvidas recorrem ao quadrado das observações, o que, no caso de caudas pesadas, pode aumentar os erros de amostragem. Também é preciso ter em conta que a amostra pode conter *outliers*, o que pode levar à distorção dos resultados. No entanto, se $k = 0$, as matrizes (3.5) e (3.9) são idênticas, pelo que os estimadores são assintoticamente 100% eficientes neste caso.

3.3 Método dos Momentos Ponderados de Probabilidade (PWM)

O método PWM foi introduzido na literatura por Greenwood *et al.* [23] e atualmente é utilizado com frequência em aplicações de caris hidrológico.

A função de distribuição de uma variável aleatória X pode ser caracterizada pelos momentos ponderados de probabilidade, que são definidos pela expressão

$$M_{p,r,s} = E\left[X^p\left(F(X)\right)^r\left(1-F(X)\right)^s\right], \quad p,r,s \in \mathbb{R}. \tag{3.10}$$

Segundo os autores, a utilização destes momentos é especialmente conveniente para distribuições que tenham uma função quantil simples de tratar, uma vez que é mais fácil exprimir os parâmetros de uma distribuição como funções destes momentos do que pelo procedimento que utiliza os momentos de ordem p ($p \in \mathbb{N}$) em torno da origem, $M_{p,0,0} = E(X^p)$. Para diversas distribuições é mais útil considerar um dos seguintes momentos:

$$\begin{cases} \alpha_s = M_{1,0,s} = E\left[X\left(1-F(X)\right)^s\right] \\ \beta_r = M_{1,r,0} = E\left[X\left(F(X)\right)^r\right] \end{cases}, \qquad r,s \in \mathbb{N}_0. \tag{3.11}$$

Nas expressões em (3.11), constata-se que os momentos vão depender diretamente das observações. As relações mais importantes entre $M_{p,r,0}$ e $M_{p,0,s}$ são as seguintes:

$$\begin{cases} M_{p,0,s} = \sum_{r=0}^{s} \binom{s}{r} (-1)^r M_{p,r,0} \\ M_{p,r,0} = \sum_{s=0}^{r} \binom{r}{s} (-1)^s M_{p,0,s} \end{cases}, \qquad r,s \in \mathbb{N}_0 \tag{3.12}$$

O número de momentos ponderados a utilizar coincide com o número de parâmetros da distribuição que precisam de ser estimados. Apesar da GPD ser uma distribuição que pode ser expressa de maneiras alternativas à sua função quantil, os estimadores PWM têm sido utilizados frequentemente para a estimação dos seus parâmetros devido à sua simplicidade computacional. Através dos momentos ponderados da GPD(k,σ), obtém-se a expressão

$$\alpha_s = E\left[X\left(1-F(X)\right)^s\right] = \frac{\sigma}{(s+1)(s+1-k)}, \quad k<1, \quad s \in \mathbb{N}_0. \tag{3.13}$$

Uma vez que se pretende estimar apenas 2 parâmetros, considera-se que $s \in \{0,1\}$ e as expressões que se obtêm para os estimadores PWM são dadas por

$$\hat{k}_{MOM} = 2 - \frac{\alpha_0}{\alpha_0 - 2\alpha_1}, \tag{3.14}$$

$$\hat{\sigma}_{MOM} = \frac{2\alpha_0\alpha_1}{\alpha_0 - 2\alpha_1}. \tag{3.15}$$

As quantidades α_0 e α_1 são substituídas pelas estimativas amostrais apropriadas denotadas por a_0 e a_1, respetivamente, cuja expressão geral é dada por

$$a_s = \frac{1}{n}\sum_{i=1}^{n} x_{i:n}(1-p_{i:n})^s, \quad s \in \mathbb{N}_0, \tag{3.16}$$

onde $x_{i:n}$ é o i-ésimo valor da amostra ordenada e $p_{i:n}$ representa a i-ésima *plotting position*. Repare-se que a expressão $1-p_{i:n}$ é uma estimativa da cauda da distribuição, $1-F$. Apesar das diversas expressões propostas para $p_{i:n}$, Landwehr *et al.* [30] recomendam a utilização da expressão

$$p_{i:n} = \frac{i+\gamma}{n+\beta}, \tag{3.17}$$

fazendo $\gamma = -0.35$ e $\beta = 0$. Combinando as expressões (3.14) a (3.16), é possível obter uma estimativa para o limite superior da GPD (no caso em que $k<0$), cuja expressão é dada por

$$-\frac{\hat{\sigma}}{\hat{k}} = \frac{2a_0a_1}{4a_1 - a_0}. \tag{3.18}$$

No que toca a resultados assintóticos, Hosking & Wallis [26] referem que, para amostras de grande dimensão, os estimadores PWM para (k,σ) são assintoticamente normais com matriz de covariância

$$\frac{1}{n}\frac{1}{(1-2k)(3-2k)}\begin{bmatrix} (1-k)(2-k)^2(1-k+2k^2) & \sigma(2-k)(2-6k+7k^2-2k^3) \\ \sigma(2-k)(2-6k+7k^2-2k^3) & \sigma^2(7-18k+11k^2-2k^3) \end{bmatrix}, \tag{3.19}$$

válida para $k<0.5$. Apesar destas boas propriedades, tanto os estimadores PWM como os estimadores

MOM podem produzir estimativas não consistentes com os dados, o que leva a que a sua utilização deva ser feita com prudência.

3.4 Método dos Percentis Elementares (EPM)

O método EPM foi proposto por Castillo & Hadi [10] é das primeiras abordagens à estimação dos parâmetros da GPD que contorna as dificuldades apresentadas pelos métodos ML, MOM e PWM relativamente à validade dos estimadores. Este método é aplicável para qualquer valor de k e, a par da sua facilidade de utilização, tem a vantagem de produzir estimativas que são sempre consistentes com os dados. O método EPM utiliza uma versão reparametrizada da função de distribuição da GPD que consiste na substituição de $\frac{\sigma}{k}$ por δ, resultando na expressão

$$F(x \mid k, \delta) = 1 - \left(1 + \frac{x}{\delta}\right)^{-\frac{1}{k}}, \quad k \neq 0, \quad \delta k < 0. \tag{3.20}$$

O procedimento começa com a escolha de dois valores de uma amostra ordenada de dimensão n, $x_{i:n}$ e $x_{j:n}$, tais que $x_{i:n} < x_{j:n}$, que são substituídos na expressão (3.20) e equacionados com as correspondentes *plotting positions* de acordo com as operações

$$\begin{cases} F(x_{i:n} \mid k, \delta) = p_{i:n}, \\ F(x_{j:n} \mid k, \delta) = p_{j:n}. \end{cases} \tag{3.21}$$

Para estas quantidades, os autores sugerem a utilização da expressão (3.17) fazendo $\gamma = 0$ e $\beta = 1$. Através de operações simples, mostra-se que as equações em (3.21) podem ser reescritas como

$$\begin{cases} -\log\left(1 + \frac{x_{i:n}}{\delta}\right) = kC_i, \\ -\log\left(1 + \frac{x_{j:n}}{\delta}\right) = kC_j, \end{cases} \tag{3.22}$$

onde $i \neq j$ $(1, 2, \ldots, n)$ e as constantes C_i e C_j são calculadas através das expressões

$$\begin{cases} C_i = \log(1 - p_{i:n}), \\ C_j = \log(1 - p_{j:n}). \end{cases} \tag{3.23}$$

Resolvendo as equações de (3.22) em ordem a k e δ, obtêm-se as expressões

$$\begin{cases} C_j \log\left(1 + \frac{x_{i:n}}{\delta}\right) = C_i \log\left(1 + \frac{x_{j:n}}{\delta}\right), \\ x_{i:n}\left[1 - (1 - p_{j:n})^{-k}\right] = x_{j:n}\left[1 - (1 - p_{i:n})^{-k}\right]. \end{cases} \tag{3.24}$$

As soluções das equações em (3.24), calculadas através de um método de obtenção de raízes de funções (por exemplo, método da bissecção), são utilizadas para produzir as estimativas para (k, δ) associadas a duas determinadas observações $x_{i:n}$ e $x_{j:n}$ nas condições referidas atrás. A solução da primeira equação determina uma estimativa para δ, $\hat{\delta}(i, j)$, que é substituída numa das expressões de (3.22) para determinar uma estimativa para k, $\hat{k}(i, j)$. Assim, as estimativas para os parâmetros da GPD(k, σ) são obtidas como

função de duas estatísticas ordinais $x_{i:n}$ e $x_{j:n}$ através das expressões

$$\begin{cases} \hat{k}(i,j) = -\frac{\log\left(1+\frac{x_{i:n}}{\hat{\delta}(i,j)}\right)}{C_i}, \\ \\ \hat{\sigma}(i,j) = \hat{\delta}(i,j)\hat{k}(i,j). \end{cases} \tag{3.25}$$

Os autores propõem um algoritmo para determinar as estimativas EPM de (k,σ) que consiste na aplicação dos procedimentos mencionados atrás para todos os possíveis pares de estatísticas ordinais $x_{i:n}$ e $x_{j:n}$, tais que $x_{i:n} < x_{j:n}$, $i = 1,2,\ldots,n$. Uma vez calculados todos estes valores, as estimativas EPM finais para (k,σ) são obtidas através das expressões

$$\begin{cases} \hat{k}_{EPM} = \text{Mediana}\left\{\hat{k}(1,2),\hat{k}(1,3),\ldots,\hat{k}(n-1,n)\right\}, \\ \hat{\sigma}_{EPM} = \text{Mediana}\left\{\hat{\sigma}(1,2),\hat{\sigma}(1,3),\ldots,\hat{\sigma}(n-1,n)\right\}. \end{cases} \tag{3.26}$$

Contudo, é possível notar que o número de pares de estatísticas ordinais envolvidos pode ser elevado, em particular quando as amostras são de grande dimensão. Para ultrapassar esta dificuldade, os autores sugerem diversas soluções, mas a mais simples consiste em considerar apenas os pares $(x_{i:n},x_{n:n})$, $i = 1,2,\ldots,n-1$, o que corresponde a fixar $j = n$. A aplicação desta alternativa mantém a propriedade de consistência das estimativas iniciais de (k,σ) com os dados observados.

Para avaliar o desempenho do método EPM, Castillo & Hadi compararam-no com os métodos PWM e MOM através da utilização de amostras com dimensão moderada (entre 50 e 100 observações) geradas por simulação, considerando valores para k no intervalo $(-2,2)$. Estes resultados foram combinados com os que já tinham sido obtidos por Hosking & Wallis [26], o que levou à proposta de uma regra prática de utilização destes métodos, conforme apresentado abaixo:

- Se a dimensão da amostra for muito elevada $(n > 500)$ e $-0.5 < k < 0.5$, o método ML é o melhor método para estimação.
- Se a dimensão da amostra não for muito elevada e $0 < k < 0.5$, é recomendada a utilização do método PWM.
- Em qualquer outro caso, deve ser utilizado o método EPM.

Relativamente à última regra apresentada, esta contempla não só os casos em que o método ML deve ser utilizado, mas apresenta problemas de convergência, mas também quando as estimativas obtidas pelos métodos PWM e MOM não são consistentes com os dados.

Capítulo 4

Método de Máxima Verosimilhança Revisitado

Tal como foi referido no Capítulo 3, o Método de Máxima Verosimilhança é a técnica de eleição para estimação paramétrica devido às boas propriedades que os seus estimadores possuem. No entanto, a sua utilização deve ser feita com precaução, uma vez que fatores como a dimensão das amostras utilizadas no processo de estimação [26] ou a possível inconsistência das estimativas com os dados [24] condicionam a sua aplicação.

Neste capítulo serão apresentadas técnicas alternativas ao método ML, baseadas em metodologias *empirical bayes* e abordagens de otimização não linear, cujos estimadores resultantes possuem propriedades semelhantes ou superiores.

4.1 *Likelihood Moment Estimation* (LME) e melhoramentos

O método LME foi proposto por Zhang [52] como uma possível solução dos problemas apresentados pelos métodos tradicionais, tanto ao nível da eficiência assintótica dos estimadores, como ao nível da complexidade computacional. Esta abordagem utiliza uma versão reparametrizada da GPD, em que a função de distribuição associada é dada por

$$F(x \mid \gamma, \theta) = 1 - (1 - \theta x)^{\frac{1}{\gamma}}, \quad \gamma \neq 0, \quad \theta \neq 0, \tag{4.1}$$

que se obtém a partir da expressão (2.1) fazendo $\mu = 0$, $k = -\gamma$ e $\theta = -\frac{k}{\sigma}$.

À semelhança do método ML, a obtenção dos estimadores LME baseia-se no anulamento das derivadas parciais da função de log-verosimilhança correspondente, dada por

$$\log L(\gamma, \theta \mid \mathbf{X}) = n \log\left(\frac{\theta}{\gamma}\right) + \left(\frac{1}{\gamma} - 1\right) \sum_{i=1}^{n} \log(1 - \theta X_i), \tag{4.2}$$

mas distingue-se pela utilização da expressão dos momentos da GPD nas expressões resultantes, com vista a facilitar os cálculos e a evitar problemas de convergência que possam surgir. Assim, a partir das expressões obtidas por derivação,

$$\frac{1}{n} \sum_{i=1}^{n} (1 - \theta X_i)^{-1} - (1 - \gamma)^{-1} = 0, \qquad \gamma = -\frac{1}{n} \sum_{i=1}^{n} \log(1 - \theta X_i), \tag{4.3}$$

com $\theta < X_{n:n}^{-1}$, e da expressão dos momentos da GPD nesta parametrização,

$$E[(1-\theta X)^r] = (1+r\gamma)^{-1}, \quad 1+r\gamma > 0 \tag{4.4}$$

é possível modificar a primeira equação de (4.3) a partir da versão empírica da expressão (4.4) e generalizar a expressão para qualquer valor válido de r, tendo-se assim

$$\frac{1}{n}\sum_{i=1}^{n}(1-\theta X_i)^r - (1+r\gamma)^{-1} = 0, \quad 1+r\gamma > 0. \tag{4.5}$$

Substituindo r por $-\frac{r}{\gamma}$ na expressão (4.5), obtém-se a equação que permite o cálculo do estimador LME para θ, $\hat{\theta}_{LME}$,

$$\frac{1}{n}\sum_{i=1}^{n}(1-\theta X_i)^p - (1-r)^{-1} = 0, \quad \theta < X_{n:n}^{-1}, \tag{4.6}$$

com $p = rn/\sum_{i=1}^{n}\log(1-\theta X_i)$ e $r < 1$, e que tem solução única em $(-\infty, X_{n:n}^{-1})$ devido às propriedades da expressão (4.6), conforme apresentado por Zhang [52]. Por fim, o estimador LME para γ é obtido como função de $\hat{\theta}_{LME}$ através da expressão

$$\hat{\gamma}_{LME} = -\frac{1}{n}\sum_{i=1}^{n}\log(1-\hat{\theta}_{LME}X_i). \tag{4.7}$$

Regressando à parametrização inicial, os estimadores LME para (k,σ) são obtidos com base no par de estimadores $(\hat{\gamma}_{LME}, \hat{\theta}_{LME})$ através das expressões

$$\begin{cases} \hat{k}_{LME} = -\hat{\gamma}_{LME}, \\ \hat{\sigma}_{LME} = -\frac{\hat{k}_{LME}}{\hat{\theta}_{LME}}. \end{cases} \tag{4.8}$$

Relativamente a propriedades limite, os estimadores LME obtidos para (k,σ) são assintoticamente normais com matriz de covariância

$$\frac{1}{1-2r}\begin{bmatrix} (1-r)(1-r+2k+2k^2) & \sigma(1-2r+r^2+k+k^2) \\ \sigma(1-2r+r^2+k+k^2) & \sigma^2(2-4r+(r-k)^2+2k) \end{bmatrix}, \tag{4.9}$$

com $r < \frac{1}{2}$ e $k > -\frac{1}{2}$. No que toca à escolha de r, é recomendada a utilização de $r = -\frac{1}{2}$ porque, de acordo com Zhang [52], este valor de r conduz habitualmente a estimadores LME com elevada eficiência assintótica.

Posteriormente, Zhang & Stephens [54] propuseram um melhoramento ao método anterior, que consiste em atribuir uma distribuição *a priori* para θ, estimada empiricamente, assegurando assim que as estimativas resultantes para (k,σ) existam sempre.

Este procedimento é motivado pela definição de um novo estimador para θ,

$$\hat{\theta}_{NEW} = \frac{\int \theta p(\theta) L(\theta) d\theta}{\int p(\theta) L(\theta) d\theta}, \tag{4.10}$$

onde $L(\theta)$ representa a função de verosimilhança restrita (*profile likelihood*) para θ e $p(\theta)$ é uma den-

sidade *a priori* estimada a partir de uma amostra $\mathbf{X} = (X_1, \ldots, X_n)$. Uma vez que a expressão (4.10) não é de cálculo fácil na maioria das situações, é proposta uma versão numérica simplificada com a seguinte expressão,

$$\hat{\theta}_{NEW} = \sum_{j=1}^{m} w_j \theta_j, \tag{4.11}$$

onde

$$w_j = \frac{L(\theta_j)}{\sum_{t=1}^{m} L(\theta_t)}, \quad j = 1, \ldots, m, \quad m = 20 + [\sqrt{n}], \tag{4.12}$$

representando $[x]$ a parte inteira de x, e θ_j o quantil de ordem $\frac{j-0.5}{m}$ da distribuição *a priori*. Esta última deve ser escolhida de forma a que os estimadores obtidos sejam eficientes e o menos enviesados possível.

Os autores propuseram $p(\theta) = g(X_{n:n}^{-1} - \theta)$, onde $g(.)$ é uma função densidade de probabilidade associada a uma variável aleatória com suporte em $(0, +\infty)$, uma vez que $X_{n:n}^{-1} - \theta > 0$ devido à condição imposta para θ nas expressões apresentadas em (4.3). Através de estudos de simulação, verificou-se que uma boa escolha para $g(.)$ é a função densidade de probabilidade da GPD$(k = 0.5, \sigma = \frac{1}{6X^*})$, onde X^* representa o primeiro quartil da amostra. A função de distribuição de θ é dada por $1 - G(X_{n:n}^{-1} - \theta)$, sendo $G(y) = 1 - (1 + 3X^* y)^{-2}$, com $y > 0$. Consequentemente, o quantil de ordem $\frac{j-0.5}{m}$ da distribuição *a priori* para θ é obtido através da expressão

$$1 - G(X_{n:n}^{-1} - \theta_j) = \frac{j - 0.5}{m}. \tag{4.13}$$

Assim, obtém-se a seguinte expressão para θ_j,

$$\theta_j = X_{n:n}^{-1} + \frac{1}{3X^*} \left(1 - \sqrt{\frac{m}{j - 0.5}} \right), \tag{4.14}$$

e conclui-se que os novos estimadores para (k, σ) são obtidos através das expressões

$$\begin{cases} \hat{k}_{NEW} = \frac{1}{n} \sum_{i=1}^{n} \log(1 - \hat{\theta}_{NEW} X_i) \\ \hat{\sigma}_{NEW} = -\frac{\hat{k}_{NEW}}{\hat{\theta}_{NEW}} \end{cases} \tag{4.15}$$

Uma vez que θ_j e $\hat{\theta}_{NEW}$ tomam valores inferiores a $X_{n:n}^{-1}$, as estimativas produzidas são sempre válidas. Para além disto, os valores para a eficiência e enviesamento dos estimadores obtidos são muito melhores que qualquer outro método clássico. No entanto, relativamente a propriedades limite, os autores alegam ser difícil obter as expressões das variâncias e das eficiências assintóticas para $(\hat{k}_{NEW}, \hat{\sigma}_{NEW})$.

Apesar das boas características destes novos estimadores, Zhang [53] verificou que o seu desempenho é fraco para valores de $k > 1$ (situações de caudas muito pesadas) e propôs uma modificação à metodologia anterior para corrigir imprecisões ao nível da eficiência e do enviesamento que possam ocorrer nestes casos. Esta alteração baseia-se na escolha de uma densidade *a priori* para θ mais adaptativa, $p(\theta) = h(\frac{n-1}{n+1} X_{n:n}^{-1} - \theta)$, onde $h(.)$ representa a função densidade de probabilidade da GPD(k^*, σ^*). Através da modificação da expressão (4.14), obtém-se o estimador

$$\hat{\theta}^*_{NEW} = \sum_{j=1}^{m} w^*_j \theta^*_j, \tag{4.16}$$

onde

$$w^*_j = \frac{L(\theta^*_j)}{\sum_{t=1}^{m} L(\theta^*_t)}, \quad j = 1, \ldots, m, \quad m = 20 + [\sqrt{n}], \tag{4.17}$$

sendo neste caso quantil de ordem $\frac{j-0.5}{m}$ desta nova distribuição *a priori* para θ definido por

$$\theta^*_j = \frac{n-1}{n+1} X^{-1}_{n:n} + \frac{\sigma^*}{k^*} \left[1 - \left(\frac{j-0.5}{m} \right)^{-k^*} \right]. \tag{4.18}$$

Assim, os novos estimadores para (k, σ) são dados pelas expressões

$$\begin{cases} \hat{k}^*_{NEW} = \frac{1}{n} \sum_{i=1}^{n} \log(1 - \hat{\theta}^*_{NEW} X_i) \\ \hat{\sigma}^*_{NEW} = -\frac{\hat{k}^*_{NEW}}{\hat{\theta}^*_{NEW}} \end{cases} \tag{4.19}$$

De acordo com o autor, os estimadores $\hat{k}^*_{NEW}$ e $\hat{\sigma}^*_{NEW}$ tornam-se mais eficientes e adaptativos se os parâmetros considerados tomarem os valores $k^* = 1$ e $\sigma^* = (2\tilde{\sigma})^{-1}$, onde $\tilde{\sigma}$ representa a mediana de $(\hat{\sigma}_{0.3}, \hat{\sigma}_{0.4}, \ldots, \hat{\sigma}_{0.9})$ e os valores $\hat{\sigma}_p$ envolvidos são obtidos através de

$$\begin{cases} \hat{\sigma}_p = \hat{k}_p \hat{x}_{1-p} / (1 - p^{\hat{k}_p}) \\ \hat{k}_p = \log_p (\hat{x}_{1-p^2} / \hat{x}_{1-p} - 1) \end{cases}, \qquad p \in (0,1), \tag{4.20}$$

sabendo que $\hat{x}_\alpha$ representa o quantil amostral de ordem α, com $\alpha \in (0,1)$. É ainda possível mostrar que os estimadores iniciais em (4.20) são consistentes para os respetivos parâmetros da GPD(k, σ), conforme mostrado por Zhang [53].

4.2 Método Baseado em Programação Matemática

Apesar da estimação pelo Método de Máxima Verosimilhança ser uma metodologia extensivamente utilizada no âmbito da Estatística e de ser abordada na prática recorrendo a métodos numéricos com algumas limitações (como o Método de Newton-Raphson), também é possível considerá-lo como um problema de Programação Matemática. Esta afirmação é válida porque a determinação de estimativas de Máxima Verosimilhança é equivalente à obtenção da solução ótima de um problema de maximização, para o qual a função objetivo é a função de verosimilhança (ou log-verosimilhança).

A abordagem que será seguida baseia-se no tratamento da estimação dos parâmetros da GPD com recurso a metodologias de otimização mais robustas e flexíveis que as utilizadas habitualmente. É ainda possível garantir que as estimativas obtidas são consistentes com os dados, uma vez que é possível condicionar o espaço de pesquisa de soluções admissíveis através da consideração de restrições. No caso da GPD, a questão da consistência das estimativas com os dados significa que quando os dados

têm uma distribuição subjacente com caudas leves, a estimativa produzida para o limite superior do suporte deve ser estritamente superior ao máximo da amostra observada. No entanto, a abordagem que se segue continuará a considerar a limitação relativa à existência de estimativas de Máxima Verosimilhança quando $k > -1$.

4.2.1 Formulação em Programação Matemática

Tal como já foi referido no Capítulo 2, a distribuição GPD tem comportamentos distintos de acordo com o sinal do parâmetro de forma. Relembrando, se $k \geq 0$, não existe limite superior finito para o suporte; no entanto, se $k < 0$, o limite superior finito do suporte da GPD existe e é definido por $-\sigma/k$. No decorrer da apresentação desta abordagem, não será considerado o caso em que $k = 0$, uma vez que se trata de um caso limite da distribuição.

Devido ao duplo comportamento da distribuição, uma das formas de abordar o tema da estimação dos parâmetros da GPD passa por calcular o máximo da função de verosimilhança (ou log-verosimilhança) restrito a cada região distinta do espaço paramétrico e considerar apenas as estimativas de Máxima Verosimilhança correspondentes ao problema com maior valor da função objetivo.

Antes de deduzir formulações para os dois problemas, é útil considerar novamente a versão reparametrizada da GPD [10] que consiste na substituição de σ/k por θ, e cuja função densidade resulta na expressão

$$f(x \mid k, \theta) = \frac{1}{k\theta}\left(1 + \frac{x}{\theta}\right)^{-\frac{1}{k}-1}, \quad k \neq 0, \quad \theta k > 0. \tag{4.21}$$

Tomando a função densidade anterior e considerando que está disponível uma amostra $\mathbf{g} = (g_1, \ldots, g_n)$, a função de verosimilhança correspondente é dada pela expressão

$$L(k, \theta \mid \mathbf{g}) = (k\theta)^{-n}\left[\prod_{i=1}^{n}\left(1 + \frac{g_i}{\theta}\right)\right]^{-\frac{1}{k}-1} \tag{4.22}$$

e a função de log-verosimilhança associada é apresentada na forma

$$l(k, \theta \mid \mathbf{g}) = -n\log(k\theta) - \left(\frac{1}{k} + 1\right)\sum_{i=1}^{n}\log\left(1 + \frac{g_i}{\theta}\right). \tag{4.23}$$

Uma vez deduzidas as expressões para as funções de verosimilhança e log-verosimilhança na versão reparametrizada, procede-se agora à formulação dos problemas de Programação Matemática mencionados. Os problemas em questão focam-se na maximização de uma das funções de verosimilhança apresentadas e estão sujeitos a restrições específicas consoante o valor de k. No caso em que $k < 0$, é necessário garantir que o limite superior do suporte, que é finito, positivo e representado por $-\theta$ nesta nova parametrização, é maior que todas as observações de $\mathbf{g}$, em particular do máximo da amostra, $g_{n:n}$. Isto implica que $\theta < -g_{n:n}$ e que, juntamente com o valor considerado para k, seja satisfeita a desigualdade $\theta k > 0$. No caso em que $k > 0$, apenas é necessário garantir que se verifica $\theta k > 0$, o que implica que $\theta > 0$.

Agora que todas as restrições estão discriminadas e as candidatas a função objetivo estão apresentadas, resta formular os problemas disjuntos que vão ser otimizados. Nesta formulação, será utilizada a função de log-verosimilhança como função objetivo por razões práticas que serão abordadas mais adi-

ante. A formulação do problema de Programação Matemática para otimização da hipótese de caudas leves ($k < 0$) é dada por

$$\begin{array}{ll} \underset{k,\theta}{\text{Maximizar}} & l(k,\theta \mid \mathbf{g}) \\ \text{sujeito a} & -1 < k < 0 \\ & \theta < -g_{n:n} \end{array} \tag{4.24}$$

e a formulação do problema de Programação Matemática para otimização da hipótese de caudas pesadas ($k > 0$) é dada por

$$\begin{array}{ll} \underset{k,\theta}{\text{Maximizar}} & l(k,\theta \mid \mathbf{g}) \\ \text{sujeito a} & k > 0 \\ & \theta > 0 \end{array} \tag{4.25}$$

Resta agora designar um algoritmo de otimização não linear para resolver os problemas (4.24) e (4.25), de forma a determinar as soluções ótimas de cada um deles, que correspondem às estimativas dos parâmetros da GPD em cada um dos casos.

4.2.2 Algoritmo de Programação Quadrática Sequencial (SQP)

Os métodos de Programação Quadrática Sequencial (SQP) constituem uma classe de técnicas *state of the art* para resolução de problemas de Programação Não Linear, baseadas na mimetização dos passos do Método de Newton-Raphson para a otimização de problemas com restrições. Estes métodos assentam na resolução iterativa de subproblemas quadráticos construídos a partir da Função Lagrangeana do problema inicial. Nesta secção é apresentada a versão mais popular do método SQP, desenvolvida separadamente por Wilson, Han e Powell [25, 36, 47], que utiliza aproximações quasi-Newton para estimar o valor da matriz hessiana da Função Lagrangeana em cada iteração.

Considere-se um problema de Programação Não Linear genérico para otimização, da forma

$$\begin{array}{ll} \underset{\mathbf{x}}{\text{Minimizar}} & f(\mathbf{x}) \\ \text{sujeito a} & g_i(\mathbf{x}) \leq 0, \quad i = 1,\ldots,m_g, \\ & h_j(\mathbf{x}) = 0, \quad j = 1,\ldots,m_h, \end{array} \tag{4.26}$$

onde $f : \mathbb{R}^n \to \mathbb{R}$, $g_i : \mathbb{R}^n \to \mathbb{R}$ e $h_j : \mathbb{R}^n \to \mathbb{R}$. Para além disto, assuma-se também que todas as funções f, g_i e h_j são continuamente diferenciáveis. Partindo do problema (4.26), é possível construir um subproblema quadrático cujas restrições consistem na linearização das restrições do problema inicial, considerando o seu desenvolvimento de Taylor de ordem 1 em torno de uma solução $\mathbf{x}_k$. Deste modo, o subproblema quadrático associado ao problema (4.26) é definido por

$$\begin{array}{ll} \underset{\mathbf{d}}{\text{Minimizar}} & (\mathbf{c}_k)^T\mathbf{d} + \frac{1}{2}\mathbf{d}^T\mathbf{B}_k\mathbf{d} \\ \text{sujeito a} & g_i(\mathbf{x}_k) + \nabla g_i(\mathbf{x}_k)^T\mathbf{d} \leq 0, \quad i = 1,\ldots,m_g, \\ & h_j(\mathbf{x}_k) + \nabla h_j(\mathbf{x}_k)^T\mathbf{d} = 0, \quad j = 1,\ldots,m_h, \end{array} \tag{4.27}$$

onde $\mathbf{d} = \mathbf{x} - \mathbf{x}_k$. O vetor $\mathbf{c}_k = (c_1^k,\ldots,c_n^k)^T$ e a matriz simétrica $\mathbf{B}_k$ serão definidos mais à frente.

Numa primeira abordagem, a escolha imediata para a função objetivo do subproblema (4.27) seria uma aproximação quadrática de f em torno de $\mathbf{x}_k$. No entanto, se o problema inicial for composto por

restrições não lineares, o subproblema quadrático pode tornar-se ilimitado [9], concluindo-se assim que esta escolha não é a mais correta. Para contornar esta desvantagem, considere-se a substituição da função objetivo do problema (4.26), f, pela Função Lagrangeana associada ao mesmo problema, representada pela expressão

$$\mathscr{L}(\mathbf{x},\mathbf{u},\mathbf{v}) = f(\mathbf{x}) + \sum_{i=1}^{m_g} u_i g_i(\mathbf{x}) + \sum_{j=1}^{m_h} v_j h_j(\mathbf{x}), \tag{4.28}$$

onde $\mathbf{u} = (u_1,\ldots,u_{m_g})^T$ e $\mathbf{v} = (v_1,\ldots,v_{m_h})^T$ são vetores de variáveis duais associadas às restrições g_i e h_j, respetivamente. Por definição, as componentes de $\mathbf{u}$ são não negativas e as componentes de $\mathbf{v}$ podem tomar qualquer valor real. A utilização de (4.28) como função objetivo do problema inicial permite contemplar a não-linearidade das suas restrições, mantendo simultaneamente a linearidade das restrições do subproblema quadrático. Adicionalmente, a solução ótima do problema inicial, $\mathbf{x}^*$, mantém-se inalterada face à mudança de função objetivo efetuada [9], agora considerando também o valor ótimo das variáveis duais, $\mathbf{u}^*$ e $\mathbf{v}^*$.

Desta forma, a aproximação quadrática da Função Lagrangeana, obtida através do desenvolvimento de Taylor de ordem 2 em torno de um ponto $\mathbf{w}_k = (\mathbf{x}_k,\mathbf{u}_k,\mathbf{v}_k)$, é dada por

$$\mathscr{L}(\mathbf{w}) \approx \mathscr{L}(\mathbf{w}_k) + \nabla_{\mathbf{x}}\mathscr{L}(\mathbf{w}_k)^T(\mathbf{w}-\mathbf{w}_k) + \frac{1}{2}(\mathbf{w}-\mathbf{w}_k)^T\nabla^2_{\mathbf{x}}\mathscr{L}(\mathbf{w}_k)(\mathbf{w}-\mathbf{w}_k), \tag{4.29}$$

onde $\nabla_{\mathbf{x}}\mathscr{L}(\mathbf{w}_k)$ e $\nabla^2_{\mathbf{x}}\mathscr{L}(\mathbf{w}_k)$ referem-se, respetivamente, aos valores do gradiente e da matriz hessiana da função (4.28) avaliados no ponto $\mathbf{w}_k$, e cujas derivadas parciais são calculadas apenas com respeito ao vetor de variáveis primais, $\mathbf{x}$. Tendo em conta a estrutura do subproblema (4.27), os valores $\mathbf{c}_k$ e $\mathbf{H}_k$ presentes na sua função objetivo são substituídos por $\nabla_{\mathbf{x}}\mathscr{L}(\mathbf{x}_k,\mathbf{u}_k,\mathbf{v}_k)$ e $\nabla^2_{\mathbf{x}}\mathscr{L}(\mathbf{x}_k,\mathbf{u}_k,\mathbf{v}_k)$, respetivamente. Contudo, a versão do subproblema quadrático que é mais considerada na literatura, e que será referida a partir deste ponto, é dada por

$$\begin{array}{ll} \underset{\mathbf{d}}{\text{Minimizar}} & \nabla f(\mathbf{x}_k)^T\mathbf{d} + \frac{1}{2}\mathbf{d}^T\mathbf{B}_k\mathbf{d} \\ \text{sujeito a} & g_i(\mathbf{x}_k) + \nabla g_i(\mathbf{x}_k)^T\mathbf{d} \leq 0, \quad i = 1,\ldots,m_g, \\ & h_j(\mathbf{x}_k) + \nabla h_j(\mathbf{x}_k)^T\mathbf{d} = 0, \quad j = 1,\ldots,m_h. \end{array} \tag{4.30}$$

Note-se que apesar desta formulação do subproblema quadrático considerar que $\mathbf{c}_k = \nabla f(\mathbf{x}_k)$, a eventual não-linearidade das restrições do problema (4.26) continua a ser contemplada, uma vez que a matriz $\mathbf{B}_k$ é construida a partir da expressão da sua Função Lagrangeana. Este último argumento também é válido para justificar que a matriz $\mathbf{B}_k$ escolhida é uma matriz simétrica, dado que a Função Lagrangeana é obtida através da soma de funções continuamente diferenciáveis.

Ainda relativamente à matriz $\mathbf{B}_k$, Han e Powell sugeriram a substituição do seu valor exato pela utilização de aproximações numéricas obtidas através de métodos quasi-Newton, para as quais demonstraram a convergência local do algoritmo [25, 36]. Para além disto, Powell também mostrou que se uma estimativa inicial para a matriz hessiana, $\mathbf{B}_0$, for definida positiva, então as estimativas obtidas em iterações seguintes através da utilização destes métodos também serão matrizes definidas positivas [36].

O subproblema quadrático é considerado a componente principal de qualquer método SQP porque permite a obtenção de novas direções de otimização do problema inicial. Em cada iteração k, as estimativas atuais para a solução ótima ($\mathbf{x}_k$) e para a matriz hessiana ($\mathbf{B}_k$) do problema inicial são utilizadas para

construir o subproblema (4.30), cuja solução ótima $\mathbf{d}_k$ corresponde a uma nova direção de otimização. Para além disto, também é exigido o cálculo do valor ótimo das variáveis duais do subproblema, $\mathbf{u}_k$ e $\mathbf{v}_k$, a utilizar posteriormente. A sua obtenção pode ser feita com recurso a um conjunto de condições de otimalidade conhecidas como Condições de Karush-Kuhn-Tucker, que permitem relacionar o valor das variáveis primais com o das variáveis duais [2, 35]. Em particular, se for conhecido o valor ótimo das variáveis primais, estas condições permitem determinar o valor ótimo das variáveis duais.

Uma vez determinada a nova direção de otimização do problema, é possível construir uma nova candidata a solução ótima a partir da expressão

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{d}_k, \tag{4.31}$$

onde α_k indica o tamanho do passo que deve ser considerado, de forma a produzir um decréscimo suficiente numa determinada função de mérito. Estas funções têm como objetivo contrabalançar a redução do valor da função objetivo com a não violação das restrições do problema inicial [35]. Por convenção, considera-se que o balanceamento é bom quando o valor da função de mérito é baixo. Com base neste conceito, Han e Powell introduziram a utilização de um processo de pesquisa linear [25, 36] apoiado na função de mérito definida por

$$\Psi(\mathbf{x}) = f(\mathbf{x}) + \sum_{i=1}^{m_g} r_i \max\{0, g_i(\mathbf{x})\} + \sum_{j=1}^{m_h} t_j |h_j(\mathbf{x})|, \tag{4.32}$$

com o objetivo de determinar o valor de α que minimiza a função

$$\phi(\alpha) = \Psi(\mathbf{x}_k + \alpha \mathbf{d}_k). \tag{4.33}$$

Os parâmetros de penalização r_i e t_j são calculados em cada iteração k de acordo com as expressões

$$r_i^k = \max\left\{ |u_i^k|, \frac{|u_i^k| + r_i^{k-1}}{2} \right\}, \quad i = 1, \ldots, m_g, \tag{4.34}$$

$$t_j^k = \max\left\{ |v_j^k|, \frac{|v_j^k| + t_j^{k-1}}{2} \right\}, \quad j = 1, \ldots, m_h, \tag{4.35}$$

onde u_i^k e v_j^k representam o valor ótimo de cada variável dual associada ao subproblema quadrático obtido nessa mesma iteração. Estas expressões permitem uma contribuição positiva por parte das restrições do subproblema que não foram satisfeitas como igualdade na iteração atual, mas que o foram recentemente [32]. Relativamente ao valor inicial dos parâmetros de penalização, este pode ser calculado utilizando as expressões

$$r_i^0 = \frac{\|\nabla f(\mathbf{x}_0)\|}{\|\nabla g_i(\mathbf{x}_0)\|}, \quad i = 1, \ldots, m_g, \tag{4.36}$$

$$t_j^0 = \frac{\|\nabla f(\mathbf{x}_0)\|}{\|\nabla h_j(\mathbf{x}_0)\|}, \quad j = 1, \ldots, m_h. \tag{4.37}$$

A utilização destas definições iniciais assegura contribuições mais significativas por parte das restrições cujo valor do gradiente calculado na solução inicial, $\mathbf{x}_0$, é mais reduzido [32].

Por fim, resta proceder à atualização da estimativa da matriz hessiana que será utilizada na construção

do subproblema quadrático da iteração $k+1$. Este processo apoia-se na fórmula utilizada pelo método BFGS para atualizar a matriz $\mathbf{B}_k$, que demonstrou a sua eficiência no âmbito da otimização de problemas sem restrições [35]. Esta fórmula é dada por

$$\mathbf{B}_{k+1} = \mathbf{B}_k + \frac{\mathbf{y}_k \mathbf{y}_k^T}{\mathbf{s}_k^T \mathbf{y}_k} - \frac{\mathbf{B}_k \mathbf{s}_k \mathbf{s}_k^T \mathbf{B}_k}{\mathbf{s}_k^T \mathbf{B}_k \mathbf{s}_k}, \tag{4.38}$$

onde os vetores $\mathbf{s}_k$ e $\mathbf{y}_k$ são definidos por

$$\mathbf{s}_k = \mathbf{x}_{k+1} - \mathbf{x}_k, \tag{4.39}$$

$$\mathbf{y}_k = \nabla_{\mathbf{x}} \mathscr{L}(\mathbf{x}_{k+1}, \mathbf{u}_k, \mathbf{v}_k) - \nabla_{\mathbf{x}} \mathscr{L}(\mathbf{x}_k, \mathbf{u}_k, \mathbf{v}_k). \tag{4.40}$$

A aplicação da fórmula (4.38) pode ser vista como um processo de atualização quasi-Newton, onde a função objetivo utilizada é a Função Lagrangeana do problema inicial. Contudo, esta expressão requer que o produto $\mathbf{s}_k^T \mathbf{y}_k$ tome um valor estritamente positivo para que a matriz $\mathbf{B}_{k+1}$ se mantenha definida positiva, o que pode não acontecer ao utilizar as expressões (4.39) e (4.40). Para tal, Powell sugeriu uma alteração mais eficiente que consiste na substituição do vetor $\mathbf{y}_k$ por um vetor $\mathbf{r}_k$, definido por

$$\mathbf{r}_k = \theta_k \mathbf{y}_k + (1 - \theta_k) \mathbf{B}_k \mathbf{s}_k, \tag{4.41}$$

onde o valor de θ_k é calculado através da expressão

$$\theta_k = \begin{cases} 1, & \mathbf{s}_k^T \mathbf{y}_k \geq 0.2 \mathbf{s}_k^T \mathbf{B}_k \mathbf{s}_k, \\ \frac{0.8 \mathbf{s}_k^T \mathbf{B}_k \mathbf{s}_k}{\mathbf{s}_k^T \mathbf{B}_k \mathbf{s}_k - \mathbf{s}_k^T \mathbf{y}_k}, & \mathbf{s}_k^T \mathbf{y}_k < 0.2 \mathbf{s}_k^T \mathbf{B}_k \mathbf{s}_k. \end{cases} \tag{4.42}$$

Qualquer valor θ_k calculado a partir da expressão (4.42) permite verificar a desigualdade

$$\mathbf{s}_k^T \mathbf{r}_k \geq 0.2 \mathbf{s}_k^T \mathbf{B}_k \mathbf{s}_k, \tag{4.43}$$

que é sempre minorada por um valor estritamente positivo. Isto justifica-se pelo facto da forma quadrática $\mathbf{s}_k^T \mathbf{B}_k \mathbf{s}_k$ considerar uma matriz definida positiva ($\mathbf{B}_k$) na sua definição e, consequentemente, obter valores estritamente positivos para quaisquer vetores $\mathbf{s}_k$ não nulos. Assim, a fórmula quasi-Newton adaptada ao método SQP para atualização da matriz $\mathbf{B}_k$ é dada por

$$\mathbf{B}_{k+1} = \mathbf{B}_k + \frac{\mathbf{r}_k \mathbf{r}_k^T}{\mathbf{s}_k^T \mathbf{r}_k} - \frac{\mathbf{B}_k \mathbf{s}_k \mathbf{s}_k^T \mathbf{B}_k}{\mathbf{s}_k^T \mathbf{B}_k \mathbf{s}_k}. \tag{4.44}$$

No que toca à obtenção da solução ótima do problema inicial, $\mathbf{x}^*$, esta pode ser estimada a partir da primeira candidata a solução ótima que está suficientemente próxima da solução exata, com o auxílio do gradiente da função objetivo, f, calculado nessa solução. Então, se a norma do gradiente calculado numa solução $\mathbf{x}_k$ for suficientemente pequena (isto é, inferior a uma precisão ε), essa solução é considerada ótima. Com base neste critério e nos passos anteriores, apresenta-se no Algoritmo 4.1 a versão do método SQP desenvolvida por Wilson, Han e Powell.

Algoritmo 4.1 Método SQP (Wilson, Han & Powell)

Input: $\mathbf{x}_0$, $\mathbf{B}_0$, ε

Inicialização de variáveis
Número de iterações ($k \leftarrow 0$)
Enquanto $\|\nabla f(\mathbf{x}_k)\| \geq \varepsilon$ **fazer**
    Obter $\mathbf{d}_k$, $\mathbf{u}_k$ e $\mathbf{v}_k$ a partir da minimização do subproblema quadrático (4.30)
    Obter α_k a partir da minimização da função de mérito (4.32)
    Obter $\mathbf{x}_{k+1}$ usando a expressão (4.31)
    Obter $\mathbf{s}_k$ usando a expressão (4.39)
    Obter $\mathbf{r}_k$ usando a expressão (4.41)
    Obter $\mathbf{B}_{k+1}$ usando a expressão (4.44)
    $k \leftarrow k+1$
Fim Enquanto

Output: $\mathbf{x}_k$

4.2.3 Método para estimação dos parâmetros da GPD

Enquanto algoritmo de otimização não linear, o método SQP pode ser utilizado para estimação dos parâmetros da GPD através da minimização dos problemas auxiliares (4.24) e (4.25), apresentados anteriormente. No entanto, estes problemas estão formulados para maximização e não para minimização, conforme exigido pelo algoritmo. Esta dificuldade pode ser contornada se se tiver em conta que a solução ótima de um problema de minimização é a mesma de um problema de maximização, no caso em que as funções objetivo são simétricas.

Para além disto, os problemas auxiliares à estimação não estão construídos de acordo com o formato do problema (4.26), uma vez que as desigualdades destes são estritas. Para manter simultaneamente as propriedades desejadas para estes problemas e as condições necessárias para a aplicação do método SQP, é utilizada para o efeito uma constante extra, δ, definida com um valor positivo suficientemente pequeno, para redefinição das restrições dos problemas. Desta forma, a formulação do problema de Programação Matemática para otimização da hipótese de caudas leves ($k < 0$) fica modificada para

$$\begin{aligned} \underset{k,\theta}{\text{Minimizar}} \quad & -l(k,\theta \mid \mathbf{g}) \\ \text{sujeito a} \quad & -1+\delta \leq k \leq -\delta \\ & \theta \leq -g_{n:n} - \delta \end{aligned} \tag{4.45}$$

e a formulação do problema de Programação Matemática para otimização da hipótese de caudas pesadas ($k > 0$) fica modificada para

$$\begin{aligned} \underset{k,\theta}{\text{Minimizar}} \quad & -l(k,\theta \mid \mathbf{g}) \\ \text{sujeito a} \quad & k \geq \delta \\ & \theta \geq \delta \end{aligned} \tag{4.46}$$

Resta agora reescrever as restrições dos problemas como desigualdades à esquerda, de forma a obter-se a formulação desejada. Assim, o problema de Programação Matemática final para otimização da hipótese de caudas leves ($k < 0$) é dada por

Algoritmo 4.2 Método de Máxima Verosimilhança Otimizado (MLO)

Input: $\mathbf{g} = (g_1, \ldots, g_n)$, $(k_{LT}^0, \theta_{LT}^0)$, $(k_{HT}^0, \theta_{HT}^0)$, ε, δ

1: Obter $(k_{LT}^*, \theta_{LT}^*)$ e z_{LT}^* através da otimização do problema (4.47), usando o Algoritmo 4.1
2: Obter $(k_{HT}^*, \theta_{HT}^*)$ e z_{HT}^* através da otimização do problema (4.48), usando o Algoritmo 4.1
3: **Se** $z_{HT}^* \leq z_{LT}^*$ **fazer**
4: $\quad (k_{MLO}, \sigma_{MLO}) \leftarrow (k_{HT}^*, k_{HT}^* \times \theta_{HT}^*)$
5: **Caso contrário**
6: $\quad (k_{MLO}, \sigma_{MLO}) \leftarrow (k_{LT}^*, k_{LT}^* \times \theta_{LT}^*)$
7: **Fim Se**

Output: (k_{MLO}, σ_{MLO})

$$\begin{aligned} \underset{k,\theta}{\text{Minimizar}} \quad & -l(k, \theta \mid \mathbf{g}) \\ \text{sujeito a} \quad & -k + \delta - 1 \leq 0 \\ & k + \delta \leq 0 \\ & \theta + \delta + g_{n:n} \leq 0 \end{aligned} \tag{4.47}$$

e a formulação do problema de Programação Matemática final para otimização da hipótese de caudas pesadas ($k > 0$) é dada por

$$\begin{aligned} \underset{k,\theta}{\text{Minimizar}} \quad & -l(k, \theta \mid \mathbf{g}) \\ \text{sujeito a} \quad & -k + \delta \leq 0 \\ & -\theta + \delta \leq 0 \end{aligned} \tag{4.48}$$

Antes de aplicar o método SQP, é necessário definir os valores iniciais que serão utilizados pelo algoritmo. No contexto da estimação dos parâmetros da GPD, a solução inicial $\mathbf{x}_0$ corresponde ao vetor de estimativas iniciais para os parâmetros relativamente a cada problema auxiliar. Para este efeito, consideram-se os vetores $(k_{LT}^0, \theta_{LT}^0)$ e $(k_{HT}^0, \theta_{HT}^0)$ como soluções iniciais para os problemas auxiliares (4.47) e (4.48), respetivamente. No que toca à estimativa inicial para a matriz hessiana, $\mathbf{B}_0$, esta deve ser uma matriz definida positiva, de ordem 2, uma vez que função objetivo do problema é composta por duas variáveis, correspondentes aos parâmetros da GPD a otimizar. De acordo com estes requisitos, a opção mais simples para o valor de $\mathbf{B}_0$ é a matriz identidade, de ordem 2. Por fim, o valor da precisão associado à solução ótima obtida, ε, é definido com um valor positivo suficientemente pequeno.

Uma vez otimizados os problemas auxiliares (4.47) e (4.48), interessa considerar os valores ótimos dos parâmetros da GPD e da função objetivo de cada problema. Sejam $(k_{LT}^*, \theta_{LT}^*)$ e z_{LT}^* os valores ótimos dos parâmetros e da função objetivo, respetivamente, relativamente ao problema (4.47), e sejam $(k_{HT}^*, \theta_{HT}^*)$ e z_{HT}^* os valores ótimos dos parâmetros e da função objetivo, respetivamente, relativamente ao problema (4.48). Uma vez que se pretende minimizar o simétrico da função de log-verosimilhança, este método escolhe as estimativas correspondentes ao par de valores ótimos para o qual se obtém o valor ótimo da função objetivo mais baixo. Uma vez que este método utiliza uma reparametrização da GPD, as estimativas finais (k_{MLO}, σ_{MLO}) são dadas por $(k, k \times \theta)$, independentemente do par de estimativas considerado.

Assim, considerando as estimativas iniciais apresentadas anteriormente, relativamente a cada um dos problemas, os parâmetros da distribuição GPD subjacente a uma amostra $\mathbf{g} = (g_1, \ldots, g_n)$ podem ser estimados de acordo com os passos indicados no Algoritmo 4.2.

Capítulo 5

Método Baseado em Ajustamento Polinomial

No contexto da distribuição GPD, duas das ferramentas mais utilizadas para avaliar o peso de cauda da distribuição subjacente a uma amostra $\mathbf{g} = (g_1, \ldots, g_n)$, comparativamente com a distribuição Exponencial, são o QQ-Plot Exponencial e o gráfico da função de excesso médio. Consoante a forma que estes gráficos apresentem, é possível determinar se a cauda da distribuição subjacente à amostra tem o mesmo peso que a distribuição Exponencial, ou se tem peso superior ou inferior à mesma.

Na Figura 2.2, apresentada no Capítulo 2, são mostrados os padrões que os gráficos referidos podem tomar e as conclusões relativas ao peso de cauda da distribuição subjacente à amostra. Considerando em primeiro lugar o QQ-Plot Exponencial, é possível verificar que o tipo de cauda é deduzido a partir da convexidade do gráfico obtido. Por outras palavras, é possível concluir que:

- Se o gráfico for convexo, então o peso de cauda é superior ao da Exponencial (cauda Pareto)
- Se o gráfico for linear, então o peso de cauda é igual ao da Exponencial (cauda Exponencial)
- Se o gráfico for côncavo, então o peso de cauda é inferior ao da Exponencial (cauda Beta)

No caso do gráfico da função de excesso médio, o peso de cauda da distribuição subjacente à amostra é proporcional ao declive da tendência linear do gráfico. Assim, é possível concluir que:

- Se o declive for positivo, então o peso de cauda é superior ao da Exponencial (cauda Pareto)
- Se o declive for nulo, então o peso de cauda é igual ao da Exponencial (cauda Exponencial)
- Se o declive for negativo, então o peso de cauda é inferior ao da Exponencial (cauda Beta)

Uma vez que os gráficos apresentados têm propriedades que permitem concluir acerca do peso de cauda da distribuição subjacente a uma amostra, é interessante combiná-las com alguns dos métodos de estimação apresentados anteriormente, de modo a, face a um conjunto de dados, propor uma forma de auxiliar na decisão do método que melhor se adequa à estimação dos parâmetros da GPD. Para esta abordagem, será considerado apenas o QQ-Plot Exponencial, uma vez que a sua estrutura gráfica é mais estável que a do gráfico da função de excesso médio. Neste último gráfico, a consideração de poucas estatísticas ordinais de topo, quando o limiar é muito elevado, faz com que os valores calculados para o excesso médio sejam mais variáveis e, consequentemente, afetem o ajustamento de modelos aos gráficos obtidos.

Recordando algumas propriedades dos métodos referidos, é sabido que o método EPM produz melhores estimativas que os métodos MOM e PWM quando $k < -0.4$ ou $k > 0.4$, e que as propriedades dos estimadores $(k^*_{NEW}, \sigma^*_{NEW})$ são melhores que as de qualquer outro estimador referido anteriormente, quando $k > -0.5$. Ao combinar estes dois métodos, é possível cobrir todo o espaço paramétrico e produzir boas estimativas para os parâmetros da GPD, independentemente da amostra em estudo.

5.1 Construção do QQ-Plot Exponencial

Enquanto ferramenta gráfica, os QQ-Plots são utilizados para comparar duas distribuições de probabilidade. Num contexto mais prático, é usual considerar amostras provenientes de cada distribuição e confrontá-las graficamente, com o objetivo comparar a sua origem probabilística. Esta ferramenta é de fácil interpretação, dado que o QQ-Plot resultante de duas amostras obtidas de uma mesma distribuição de probabilidade apresenta um padrão linear, que pode ser quantificado através da utilização de um coeficiente de correlação.

Uma vez que grande parte dos modelos estatísticos assenta na normalidade dos dados utilizados, o QQ-Plot Normal é uma ferramenta de primeiro recurso no que toca à avaliação da proveniência probabilística dos mesmos. No entanto, em Teoria de Valores Extremos, é mais útil considerar QQ-Plots Exponenciais porque o peso de cauda desta distribuição estabelece o intermédio dos tipos de pesos de cauda considerados nesta área de estudos.

Para avaliar o peso de cauda de uma amostra $\mathbf{g} = (g_1, \ldots, g_n)$ comparativamente com a distribuição Exponencial, começa-se por considerar uma amostra gerada desta distribuição a partir da sua função quantil, que não é mais do que a função inversa da função de distribuição. Assim, tendo em conta que a função de distribuição da Exponencial(λ) é dada por

$$F(x \mid \lambda) = 1 - \exp(-\lambda x), \quad x > 0, \quad \lambda > 0, \tag{5.1}$$

obtém-se através de operações simples a expressão da função quantil correspondente,

$$Q_\lambda(p) = -\frac{1}{\lambda}\log(1-p), \quad p \in (0,1). \tag{5.2}$$

Uma vez que a função quantil da Exponencial Padrão, $Q_1(p)$, se obtém de (5.2) quando $\lambda = 1$, é possível verificar a existência de uma relação linear entre as duas funções quantil, definida por

$$Q_\lambda(p) = \frac{1}{\lambda} Q_1(p). \tag{5.3}$$

Esta propriedade permite concluir que a construção de qualquer QQ-Plot Exponencial pode ser feita com base nos quantis obtidos da Exponencial Padrão. Desta forma, designando os quantis amostrais de $\mathbf{g}$ por $\hat{Q}_n(p)$, os pontos utilizados para construir o QQ-Plot Exponencial são da forma

$$(Q_1(p), \hat{Q}_n(p)). \tag{5.4}$$

No caso do gráfico apresentar uma forma linear, a estimativa do parâmetro λ pode ser obtida facilmente a partir do inverso do declive da reta ajustada ao QQ-Plot. É importante relembrar que esta reta deve ter ordenada na origem nula, uma vez que $Q_1(0) = 0$.

Algoritmo 5.1 Geração dos pontos do QQ-Plot Exponencial

Input: $\mathbf{g} = (g_1, \ldots, g_n)$

1: **Inicialização de variáveis**
2: Conjunto de pontos do QQ-Plot ($P \leftarrow \{\}$)
3: **Ciclo de geração**
4: **Para cada** $i = 1, \ldots, n$ **fazer**
5: Calcular p_i usando a expressão 5.6
6: Calcular $Q_1(p_i)$ usando a expressão 5.2
7: Obter a i-ésima estatística ordinal de $\mathbf{g}$, $g_{i:n}$
8: $P \leftarrow P \cup \{(Q_1(p_i), g_{i:n})\}$
9: **Fim Para cada**

Output: P

A obtenção dos pares referidos em (5.4) só é possível se for fornecido um valor de probabilidade, p, para ser substituído na expressão. A escolha imediata para os valores de p são os valores da forma

$$p_i = \frac{i}{n}, \quad i = 1, \ldots, n. \tag{5.5}$$

Contudo, quando $i = n$, vem que $p_i = 1$ e, por conseguinte, que $Q_1(p_i) \to +\infty$. Desta forma, o máximo da amostra, que é um valor finito, estaria a ser comparado com um valor infinito. [1] Assim, das várias formas mencionados na literatura [5] para os valores de p, será considerada a forma

$$p_i = \frac{i}{n+1}, \quad i = 1, \ldots, n. \tag{5.6}$$

Tendo em conta que $\hat{Q}_n(p_i)$ pode ser estimado pela i-ésima estatística ordinal da amostra, $g_{i:n}$, a expressão dos pontos utilizados para construir o QQ-Plot Exponencial é alterada para

$$(Q_1(p_i), g_{i:n}). \tag{5.7}$$

Assim, supondo que se dispõe de uma amostra $\mathbf{g} = (g_1, \ldots, g_n)$ relativamente à qual se pretende avaliar o peso de cauda da distribuição subjacente comparativamente com a distribuição Exponencial, apresentam-se no Algoritmo 5.1 as instruções necessárias para a obtenção do QQ-Plot Exponencial correspondente.

5.2 Ajustamento polinomial dos dados

Uma vez construído o QQ-Plot Exponencial, é sensato ajustar um modelo à nuvem de pontos obtida para apoiar a tarefa de escolha do melhor método de estimação dos parâmetros da GPD consoante o tipo de cauda.

Observando os gráficos referentes a QQ-Plots Exponenciais da Figura 2.2, é possível notar que a forma dos mesmos varia entre uma forma parabólica e uma forma linear consoante o peso de cauda da distribuição subjacente à amostra em análise. Concretizando, quando a distribuição da amostra tem caudas pesadas ou leves, a forma do QQ-Plot é semelhante à de um troço de parábola com a concavidade voltada para cima ou para baixo, respetivamente. No caso em que a distribuição subjacente à amostra tem cauda exponencial, o QQ-Plot apresenta um formato retilíneo.

Algoritmo 5.2 Transformação dos pontos do QQ-Plot Exponencial

Input: $P = \{(Q_1(p_1), g_{1:n}), \ldots, (Q_1(p_n), g_{n:n})\}$

1: **Inicialização de variáveis**
2: Conjunto de pontos transformados ($Q \leftarrow \{\}$)
3: **Ciclo de transformação**
4: **Para cada** $(x,y) \in P$ **fazer**
5: $\quad x_1 \leftarrow x$
6: $\quad x_2 \leftarrow x^2$
7: $\quad Q \leftarrow Q \cup \{(x_1, x_2, y)\}$
8: **Fim Para cada**

Output: Q

Com base nos formatos possíveis para o gráfico, um dos modelos mais parcimoniosos e versáteis que pode ser ajustado à nuvem de pontos é o Modelo Polinomial de grau 2, dado pela expressão

$$y = \alpha_0 + \alpha_1 x + \alpha_2 x^2, \quad \alpha_j \in \mathbb{R}, \quad j = 0,1,2. \tag{5.8}$$

Este modelo é uma das melhores escolhas para a mimetização do formato variável do QQ-Plot Exponencial porque a sua forma também é facilmente adaptada consoante o valor que o coeficiente α_2 tome. Relembrando as propriedades do polinómio de grau 2, tem-se que:

- Se $\alpha_2 > 0$, então o gráfico do polinómio apresenta uma forma convexa
- Se $\alpha_2 = 0$, então o gráfico do polinómio apresenta uma forma linear
- Se $\alpha_2 < 0$, então o gráfico do polinómio apresenta uma forma côncava

Combinando estas propriedades com as conclusões tiradas inicialmente, relacionando a forma do QQ-Plot Exponencial e o peso de cauda da distribuição da amostra, é possível obter uma ferramenta simples para inferir acerca do peso de cauda a partir do valor de um coeficiente de um polinómio. Resta agora proceder à estimação dos coeficientes do polinómio pelo Método dos Mínimos Quadrados.

Observando a expressão (5.8), é possível notar que, apesar do modelo ser linear nos coeficientes, existem variáveis que não são lineares (por exemplo, x^2). Uma estratégia para estimação do valor dos coeficientes passa pela linearização destas variáveis, seguida do ajustamento de um modelo linear equivalente a (5.8), da forma

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \varepsilon, \quad \beta_j \in \mathbb{R}, \quad j = 0,1,2, \tag{5.9}$$

para o qual se conhecem expressões para estimação dos seus parâmetros [16]. Na expressão apresentada, o valor ε representa o erro de ajustamento do modelo.

Para proceder à linearização das variáveis não lineares, comece-se por considerar o conjunto de pontos, P, obtido pelo Algoritmo 5.1. Uma vez que P é composto por pontos da forma (x,y) e o modelo (5.8) efetua operações não lineares sobre x, a abordagem mais prática passa pela criação de um novo conjunto de pontos, Q, composto por elementos da forma (x_1, x_2, y) satisfazendo $x_1 = x$ e $x_2 = x^2$. O Algoritmo 5.2 ilustra o processo de criação deste novo conjunto de pontos. Desta forma, passam a ser consideradas duas variáveis independentes lineares (x_1 e x_2) para explicar a variável de resposta (y).

Supondo que o conjunto Q é composto por n pontos, é possível ajustar o modelo (5.9) a cada um deles, obtendo-se um conjunto de n equações da forma

$$y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \varepsilon_i, \quad i = 1, \ldots, n. \tag{5.10}$$

No entanto, a utilização de índices adicionais pode tornar-se pouco clara e inconveniente. Por esta razão, é útil condensar as n equações obtidas em expressões matriciais que, neste caso, são dadas por

$$\mathbf{y} = X\beta + \varepsilon, \tag{5.11}$$

onde $\mathbf{y} = (y_1, \ldots, y_n)^T$, $\beta = (\beta_0, \beta_1, \beta_2)^T$, $\varepsilon = (\varepsilon_1, \ldots, \varepsilon_n)^T$ e

$$X = \begin{pmatrix} 1 & x_{11} & x_{12} \\ 1 & x_{21} & x_{22} \\ \vdots & \ddots & \vdots \\ 1 & x_{n1} & x_{n2} \end{pmatrix}. \tag{5.12}$$

A matriz X é designada por matriz dos dados e é construida com base nos n valores observados de cada variável independente, guardados em cada ponto do conjunto Q. A construção do vetor $\mathbf{y}$ também recorre a este conjunto de pontos, especificamente aos n valores observados para a variável de resposta.

Uma vez modelado o problema e identificadas as suas componentes, resta estimar o vetor de parâmetros β, afim de ser possível fazer inferência usando o modelo linear estimado. Uma das técnicas mais utilizadas para este efeito é o Método dos Mínimos Quadrados, que consiste na estimação dos valores dos parâmetros que minimizam a soma de quadrados dos erros de ajustamento do modelo aos dados. Matricialmente, consiste em estimar o vetor β que minimiza o valor da expressão

$$\varepsilon^T \varepsilon = (\mathbf{y} - X\beta)^T (\mathbf{y} - X\beta). \tag{5.13}$$

Através de operações matriciais de simplificação e diferenciação [16], é possível concluir que as estimativas de β, $\hat{\beta}$, que minimizam a expressão anterior são obtidas a partir da expressão

$$\hat{\beta} = (X^T X)^{-1} X^T \mathbf{y}. \tag{5.14}$$

Note-se que esta expressão só pode ser aplicada no caso em que $(X^T X)$ é invertível.

5.3 Método para estimação dos parâmetros da GPD

Após estimar os parâmetros de (5.11), já é possível modelar o comportamento do QQ-Plot Exponencial da amostra $\mathbf{g} = (g_1, \ldots, g_n)$ através de um Modelo Polinomial de grau 2 e, assim, estabelecer regras para construção de uma técnica robusta no que toca à escolha dos métodos a aplicar para estimar os parâmetros da GPD.

Uma vez que o modelo (5.8) foi substituído pelo modelo (5.11) devido à facilidade de estimação dos parâmetros deste último, as relações referidas anteriormente entre o valor do coeficiente α_2 e a forma do polinómio definido são passadas para o valor do coeficiente β_2 do novo modelo. Se estas relações forem complementadas com as conclusões iniciais acerca do peso de cauda da distribuição de uma amostra, em função do forma do QQ-Plot Exponencial, é possível definir as seguintes regras para inferência:

- Se $\beta_2 > 0$, então o peso de cauda é superior ao da Exponencial (cauda Pareto)

Algoritmo 5.3 Método de Ajustamento Polinomial para Estimação (MAPE)

Input: $\mathbf{g} = (g_1, \ldots, g_n)$

1: Obter o conjunto de pontos P usando o Algoritmo 5.1
2: Obter o conjunto de pontos Q usando o Algoritmo 5.2
3: Obter a matriz dos dados X, a partir dos pontos de Q, usando a expressão (5.12)
4: Obter o vetor $\mathbf{y}$, a partir dos pontos de Q
5: Calcular $\hat{\beta}$ usando a expressão (5.14)
6: **Se** $\hat{\beta}_2 \geq 0$ **então**
7: $\quad (k_{MAPE}, \sigma_{MAPE}) \leftarrow (k^*_{NEW}, \sigma^*_{NEW})$
8: **Caso contrário**
9: $\quad (k_{MAPE}, \sigma_{MAPE}) \leftarrow (k_{EPM}, \sigma_{EPM})$
10: **Fim Se**

Output: $(k_{MAPE}, \sigma_{MAPE})$

- Se $\beta_2 = 0$, então o peso de cauda é igual ao da Exponencial (cauda Exponencial)
- Se $\beta_2 < 0$, então o peso de cauda é inferior ao da Exponencial (cauda Beta)

Com base em estudos já realizados [10,53], é possível verificar que o Método EPM produz melhores estimativas quando $k < 0$ e as estimativas $(k^*_{NEW}, \sigma^*_{NEW})$ são mais robustas quando $k \geq 0$. Desta forma, é sensato considerar a utilização do método EPM quando o peso de cauda da distribuição de $\mathbf{g}$ é inferior ao da Exponencial, e, no caso em que o peso de cauda de $\mathbf{g}$ é superior ao da Exponencial, o recurso às estimativas $(k^*_{NEW}, \sigma^*_{NEW})$ é o mais correto.

Uma vez que não é conhecido o valor exato de β_2, mas é possível tirar conclusões com base na sua estimativa, $\hat{\beta}_2$, é razoável relacionar os valores desta estimativa com os métodos apresentados para estimação dos parâmetros da GPD. Assim, se $\hat{\beta}_2 \geq 0$, devem ser utilizadas as estimativas $(k^*_{NEW}, \sigma^*_{NEW})$, ao passo que, se $\hat{\beta}_2 < 0$, deve ser utilizado o Método EPM. Assim, apresentam-se no Algoritmo 5.3 todos os passos necessários para estimar os parâmetros da distribuição GPD, supondo que está à disposição uma amostra $\mathbf{g} = (g_1, \ldots, g_n)$.

Numa nota final, repare-se que após a estimação dos parâmetros do modelo (5.11), o ideal seria testar a sua significância, isto é, avaliar se o valor estimado para os parâmetros era estatisticamente diferente de 0. Em particular, seria interessante efetuar este teste sobre β_2 porque, no caso de não se rejeitar a hipótese nula do seu verdadeiro valor ser 0, seria possível concluir de imediato que o modelo GPD ajustado teria caudas Exponenciais e, por isso, não seria errado considerar que o valor do seu parâmetro de forma tomaria o valor 0. No entanto, os testes utilizados para este efeito no contexto da Regressão Linear Múltipla não podem ser utilizados, uma vez que se exige o pressuposto de normalidade do vetor $\mathbf{y}$ e este é composto por valores gerados da distribuição GPD. Lembre-se que cada um dos valores $y_1, \ldots, y_n$ corresponde, respetivamente, aos valores $g_{(1)}, \ldots, g_{(n)}$, de acordo com a transformação feita no Algoritmo 5.2.

Capítulo 6

Método Baseado em Classificação

Tal como já foi mencionado anteriormente, os métodos apresentados não conseguem produzir estimativas para os parâmetros da GPD ao longo de todo o espaço paramétrico. Uma abordagem possível para contornar esta dificuldade baseia-se na combinação de métodos que produzem boas estimativas em determinadas subregiões do espaço com uma metodologia de classificação que permita catalogar as amostras utilizadas para estimação de acordo com o peso de cauda da distribuição de probabilidade subjacente.

No que toca à escolha da metodologia de classificação, é proposta neste capítulo a utilização de um perceptrão para decidir qual dos dois métodos (EPM ou metodologia desenvolvida por Zhang & Stephens) deve ser utilizado. O treino do perceptrão permite captar as caraterísticas necessárias para a classificação correta de novas amostras e os seus algoritmos de treino são adaptativos, o que faz com que a estimação dos seus parâmetros seja feita de uma forma mais regrada.

6.1 Perceptrão: contextualização e modelo

O perceptrão é um dos exemplos mais antigos de regras de classificação assentes em cálculo computacional. Trata-se de uma regra discriminante cuja construção fundamenta-se diretamente na ”aprendizagem”da fronteira da região de decisão, pelo que tem sido uma ferramenta fundamental no campo da Inteligência Artificial, nomeadamente na área de Reconhecimento de Padrões [7].

O objetivo principal das regras discriminantes é afetar uma observação com p componentes, $\mathbf{x} = (x_1, \ldots, x_p)^T$, a uma de K classes, denotadas por C_k. As regras mais simples centram-se na utilização de regiões de decisão lineares (hiperplanos) para classificar $\mathbf{x}$ numa de duas classes, C_1 e C_2, recorrendo a combinações lineares da forma

$$y(\mathbf{x}) = w_0 + \mathbf{w}^T\mathbf{x} = w_0 + w_1x_1 + \cdots + w_px_p \tag{6.1}$$

onde $\mathbf{w} = (w_1, \ldots, w_p)^T$ é um vetor de pesos e w_0 é um valor de desvio (também chamado de limiar, quando se considera o seu simétrico). Desta forma, a observação $\mathbf{x}$ deve de ser atribuída à classe C_1 se $y(\mathbf{x}) \geq 0$ e a C_2 caso contrário, sendo a fronteira de decisão definida pela relação $y(\mathbf{x}) = 0$. Esta fronteira corresponde a um hiperplano $(p-1)$-dimensional contido no espaço p-dimensional das observações, uma vez que cada componente de $\mathbf{x}$ pode ser escrita como combinação linear das restantes $p-1$ componentes.

O perceptrão é um exemplo de regra discriminante linear para classificação em duas classes, no qual a observação $\mathbf{x}$ é substituída por uma transformação $\phi(\mathbf{x})$ que será utilizada para construir um modelo linear generalizado da forma

$$y(\mathbf{x}) = f(\mathbf{w}^T \phi(\mathbf{x})) \tag{6.2}$$

onde $f(.)$ é uma função não linear designada por função de ativação. Repare-se que a inversa desta função não é mais que uma função de ligação, muito utilizada na construção de modelos lineares desta natureza. Existem diversas funções de ativação que podem ser utilizadas (identidade, logística, tangente hiperbólica, ...), mas neste caso, a função $f(.)$ a considerar corresponde à função *step* da forma

$$f(a) = \begin{cases} +1, & a \geq 0 \\ -1, & a < 0, \end{cases} . \tag{6.3}$$

onde o valor 1 indica que a classificação será feita na classe C_1 e o valor -1 indica que a classificação será feita na classe C_2.

Tradicionalmente, a transformação $\phi(\mathbf{x})$ consiste apenas em acrescentar uma componente unitária ao início da observação, de modo a que também seja possível juntar o parâmetro de desvio (w_0) ao vetor dos pesos, resultando em vetores $\mathbf{w}$ e $\phi(\mathbf{x})$ da forma

$$\begin{cases} \mathbf{w} = (w_0, w_1, \ldots, w_p)^T \\ \phi(\mathbf{x}) = (1, x_1, \ldots, x_p)^T . \end{cases} \tag{6.4}$$

No entanto, se for necessário realizar m transformações sobre o vetor de observações (por exemplo, para combinação dos valores das componentes), é possível reescrever os vetores $\mathbf{w}$ e $\phi(\mathbf{x})$ mais genericamente na forma

$$\begin{cases} \mathbf{w} = (w_0, w_1, \ldots, w_m)^T \\ \phi(\mathbf{x}) = (1, \phi_1(\mathbf{x}), \ldots, \phi_m(\mathbf{x}))^T . \end{cases} \tag{6.5}$$

Por exemplo, suponha-se que é necessário "sujeitar"uma observação $\mathbf{x} \in \mathbb{R}^5$ a um processo de redução de dimensão (para $\mathbb{R}^3$, incluindo o parâmetro de desvio) antes de ser utilizada para efeitos de cálculo. Uma transformação válida para este efeito pode ser

$$\phi(\mathbf{x}) = (1, \phi_1(\mathbf{x}), \phi_2(\mathbf{x}))^T = (1, x_1 + x_2, 3x_3 + 2x_4^2 + x_5^3)^T . \tag{6.6}$$

6.2 Estimação do vetor de pesos e propriedades

Para ser possível classificar uma observação $\mathbf{x}$ numa das classes, é necessário proceder a uma estimação do vetor de pesos segundo um algoritmo que permita ao perceptrão "aprender"quais as condições que levam a classificar a observação numa determinada classe. Neste tipo de procedimentos, a "aprendizagem"é feita com recurso a conjuntos de observações que já foram classificadas corretamente e tem como objetivo determinar os valores de $\mathbf{w}$ que permitem a reprodução destas classificações. Estes conjuntos são conhecidos por conjuntos de treino e são da forma $\{(\mathbf{x}_1, t_1), \ldots, (\mathbf{x}_n, t_n)\}$, onde t_i indica se a observação $\mathbf{x}_i$ foi corretamente classificada em C_1 $(t_i = 1)$ ou em C_2 $(t_i = -1)$.

No entanto, nem sempre é possível reproduzir corretamente todas as classificações, pelo que é habi-

tual fazer esta estimação com recurso à minimização de uma função de erro. Apesar da escolha trivial para esta função ser o número de observações mal classificadas, os métodos de otimização baseados no gradiente da função objetivo não poderiam ser aplicados, uma vez que a função de erro é constante em relação a $\mathbf{w}$ e o gradiente resultante seria nulo [7]. Isto significava que qualquer vetor $\mathbf{w} \in \mathbb{R}^{p+1}$ seria minimizante da função.

Em alternativa, é utilizada uma função conhecida como Critério do Perceptrão e que é dada por

$$E_P(\mathbf{w}) = -\sum_{j \in M} \mathbf{w}^T \boldsymbol{\phi}(\mathbf{x}_j) t_j, \tag{6.7}$$

onde M representa o conjunto dos elementos de treino mal classificados. A sua dedução é feita com recurso à combinação das expressões (6.2) e (6.3), para obter regras de classificação em cada uma das classes, bem como às classificações corretas de cada observação $\mathbf{x}$, para obter uma expressão de validação única. Assim, sabendo que as observações classificadas em C_1 satisfazem $\mathbf{w}^T \boldsymbol{\phi}(\mathbf{x}) \geq 0$, que as classificadas em C_2 satisfazem $\mathbf{w}^T \boldsymbol{\phi}(\mathbf{x}) < 0$ e que os valores de t relacionam-se com a classificação correta de cada observação do conjunto de treino, é possível concluir que uma observação corretamente classificada, $\mathbf{x}_i$, satisfaz $\mathbf{w}^T \boldsymbol{\phi}(\mathbf{x}_i) t_i > 0$ e, consequentemente, que uma observação mal classificada, $\mathbf{x}_j$, satisfaz $\mathbf{w}^T \boldsymbol{\phi}(\mathbf{x}_j) t_j \leq 0$.

Como a contribuição de uma observação mal classificada para a função de erro é uma função linear de $\mathbf{w}$ e a contribuição de uma observação bem classificada é 0, podemos concluir que (6.7) é linear por troços. Desta forma, já é possível aplicar o algoritmo de gradiente descendente à função de erro e obter um procedimento iterativo para atualização do vetor dos pesos baseado na expressão

$$\mathbf{w}_{(\tau+1)} = \mathbf{w}_{(\tau)} - \eta \nabla E_P(\mathbf{w}) = \mathbf{w}_{(\tau)} + \eta \boldsymbol{\phi}(\mathbf{x}_j) t_j, \quad j \in M, \tag{6.8}$$

onde τ é um inteiro que indexa os passos do algoritmo e η representa a taxa de aprendizagem do perceptrão (valor definido no intervalo]0,1] e que pode ser fixo a 1, sem perda de generalidade). É ainda de notar que o conjunto das classificações mal classificadas é alterado sempre que o vetor dos pesos é atualizado.

Com base nestes pressupostos, é possível dar uma interpretação simples ao processo de aprendizagem do perceptrão. Considerando que um conjunto de treino S contém n observações p-dimensionais de um determinado fenómeno corretamente classificadas, a estimação dos pesos do perceptrão pode ser esquematizada conforme se apresenta no Algoritmo 6.1.

A avaliação da convergência do algoritmo apoia-se nas propriedades de separabilidade linear dos elementos do conjunto de treino. Para clarificar este conceito, são apresentados na Figura 6.1 alguns exemplos de conjuntos de observações bidimensionais com tipos de separabilidade distintos. Os conjuntos de elementos em C_1 e C_2 dizem-se linearmente separáveis se existir um hiperplano (pelo menos) que separe as observações de cada classe em regiões exclusivas. Assim, se o conjunto de treino for linearmente separável, será possível encontrar uma solução exata num número finito de passos [38]. No entanto, este número pode ser elevado e causar incerteza na distinção entre um problema não separável e um problema de convergência lenta, que pode ter como origem os valores de inicialização do vetor de pesos e da taxa de aprendizagem.

No caso em que o conjunto de treino não é linearmente separável, o Algoritmo 6.1 nunca vai terminar, pelo que a melhor hipótese será a obtenção de um vetor de pesos que permita classificar corretamente

Algoritmo 6.1 Estimação dos Pesos do Perceptrão

Input: $S = \{(\mathbf{x}_1, t_1), \ldots, (\mathbf{x}_n, t_n)\}, \eta \in]0,1]$

Inicialização de variáveis
Vetor de pesos do perceptrão ($\mathbf{w} \leftarrow \mathbf{0}_{p+1}$)
Ciclo de atualização
Enquanto $E_P(\mathbf{w}) \neq 0$ **fazer**
    Para cada $(\mathbf{x}, t) \in S$ **fazer**
        Calcular $y(\mathbf{x})$ usando a expressão (6.2)
        Se $y(\mathbf{x}) \neq t$ **então**
            Determinar novo vetor de pesos do perceptrão ($\mathbf{w} \leftarrow \mathbf{w} + \eta\phi(\mathbf{x})t$)
        Fim Se
    Fim Para cada
Fim Enquanto

Output: **w**

grande parte dos elementos do conjunto. Com vista a este objetivo, Gallant apresenta no seu artigo [20] uma lista de algoritmos de aprendizagem supervisionada baseados no *pocket algorithm*, que é uma alteração do Algoritmo 6.1 que torna a aprendizagem do perceptrão "melhor comportada"para conjuntos de treino não separáveis. A ideia base deste algoritmo passa por "guardar no bolso"um vetor de pesos extra que corresponde ao último vetor que classificou correta e consecutivamente mais observações. Se for encontrado um novo vetor com melhores características no decorrer do algoritmo, passará a ser esse o vetor guardado.

Assim, se for considerado um conjunto de treino S com n observações p-dimensionais de um determinado fenómeno e as suas correspondentes classificações, bem como o número máximo de iterações desejadas (N), o *pocket algorithm* pode ser esquematizado conforme apresentado no Algoritmo 6.2 [20]. É interessante notar que quando o conjunto de treino é linearmente separável e que o número máximo de iterações é suficientemente elevado, haverá uma iteração τ para a qual o vetor de pesos extra classifica corretamente todas as observações do conjunto de treino (ou seja, $tot_{\mathbf{w}} = n$). Nesta situação, o algoritmo é interrompido e é devolvido o vetor de pesos extra calculado.

O Algoritmo 6.2 deve ser aplicado quando o conjunto de treino a utilizar é composto por um número relativamente pequeno de observações, podendo elas ser repetidas ou mesmo contraditórias, isto é, a mesma observação pode ter duas classificações distintas. No entanto, se o conjunto de treino tiver uma

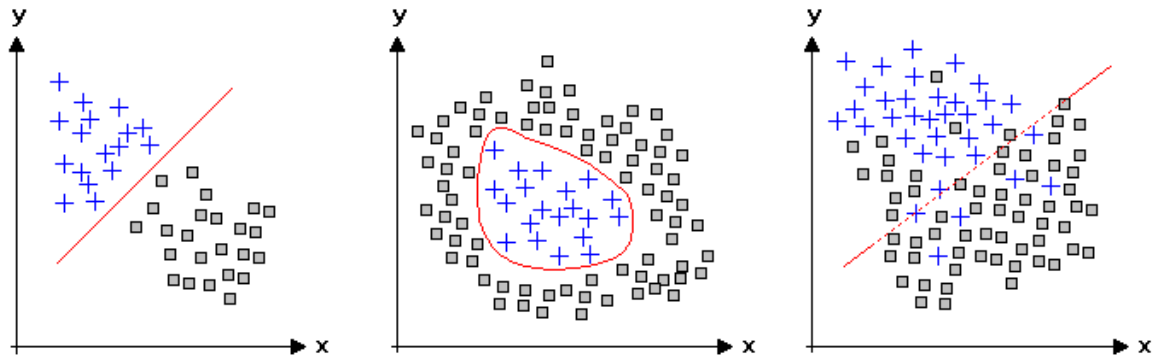

Figura 6.1: Exemplos de separabilidade de conjuntos de dados bidimensionais: separabilidade linear (à esquerda), separabilidade não linear (ao centro) e inseparabilidade (à direita) (Fonte: Statistics4U [42])

Algoritmo 6.2 Estimação dos Pesos do Perceptrão (*pocket algorithm*)

Input: $S = \{(\mathbf{x}_1, t_1), \ldots, (\mathbf{x}_n, t_n)\}, N > 0$

Inicialização de variáveis
Vetor de pesos do perceptrão ($\pi \leftarrow \mathbf{0}_{p+1}$) e pesos extra ($\mathbf{w} \leftarrow \mathbf{0}_{p+1}$)
Número de classificações corretas usando π ($run_\pi \leftarrow 0$) e $\mathbf{w}$ ($run_\mathbf{w} \leftarrow 0$)
Número total de classificações corretas usando π ($tot_\pi \leftarrow 0$) e $\mathbf{w}$ ($tot_\mathbf{w} \leftarrow 0$)
$\tau \leftarrow 1$
Ciclo de atualização
Enquanto $\tau \leq N$ e $tot_\mathbf{w} \neq n$ **fazer**
    Escolher aleatoriamente $(\mathbf{x}, t) \in S$
    Calcular $y(\mathbf{x}) = f(\pi^T \phi(\mathbf{x}))$
    Se $y(\mathbf{x}) = t$ **então**
        $run_\pi \leftarrow run_\pi + 1$
        Se $run_\pi > run_\mathbf{w}$ **então**
            Calcular o número total de classificações corretas usando π (tot_π)
            Se $tot_\pi > tot_\mathbf{w}$ **então**
                $\mathbf{w} \leftarrow \pi$
                $run_\mathbf{w} \leftarrow run_\pi$
                $tot_\mathbf{w} \leftarrow tot_\pi$
            Fim Se
        Fim Se
    Caso contrário
        Determinar novo vetor de pesos do perceptrão ($\pi \leftarrow \pi + \phi(\mathbf{x})t$)
        $run_\pi \leftarrow 0$
    Fim Se
    $\tau \leftarrow \tau + 1$
Fim Enquanto

Output: $\mathbf{w}$

dimensão relativamente elevada ou não estiver disponível na sua totalidade (por exemplo, devido a fornecimento parcelado dos dados, à observação dos mesmos apenas na altura da experiência ou por ser necessário simular este conjunto), o *pocket algorithm* pode ter uma execução demasiado demorada ou produzir estimativas de $\mathbf{w}$ pouco fiáveis. Para estes casos, é apresentado um algoritmo semelhante ao anterior que passa a focar-se apenas no número de classificações que foram feitas correta e consecutivamente através dos vetores de pesos mencionados.

Desta forma, supondo que não existe um conjunto de treino à disposição aquando do treino do perceptrão, mas que é possível obter N observações p-dimensionais de um determinado fenómeno por mecanismos alternativos, bem como as suas correspondentes classificações, o novo algoritmo pode ser esquematizado conforme apresentado no Algoritmo 6.3 [20].

6.3 Geração do conjunto de treino

Relativamente ao treino do perceptrão para efeitos de estimação paramétrica através de metodologias de classificação, é necessário gerar um conjunto de treino que seja o mais informativo possível e que permita ao perceptrão decidir que método de estimação deve ser utilizado para cada tipo de amostra. No entanto, é conveniente que as observações de treino não contenham informação redundante nem tenham uma dimensão demasiado elevada. Isto acontece quando são utilizados vetores informativos demasiado

Algoritmo 6.3 Estimação dos Pesos do Perceptrão (*pocket algorithm for* ∞ *training data*)

Input: $N > 0$

Inicialização de variáveis
Vetor de pesos do perceptrão ($\pi \leftarrow \mathbf{0}_{p+1}$) e pesos extra ($\mathbf{w} \leftarrow \mathbf{0}_{p+1}$)
Número de classificações corretas usando π ($run_\pi \leftarrow 0$) e $\mathbf{w}$ ($run_\mathbf{w} \leftarrow 0$)
$\tau \leftarrow 1$
Ciclo de atualização
Enquanto $\tau \leq N$ **fazer**
    Obter $(\mathbf{x}, t)$ através de mecanismo alternativo
    Calcular $y(\mathbf{x}) = f(\pi^T \phi(\mathbf{x}))$
    Se $y(\mathbf{x}) = t$ **então**
        $run_\pi \leftarrow run_\pi + 1$
        Se $run_\pi > run_\mathbf{w}$ **então**
            $\mathbf{w} \leftarrow \pi$
            $run_\mathbf{w} \leftarrow run_\pi$
        Fim Se
    Caso contrário
        Determinar novo vetor de pesos do perceptrão ($\pi \leftarrow \pi + \phi(\mathbf{x})t$)
        $run_\pi \leftarrow 0$
    Fim Se
    $\tau \leftarrow \tau + 1$
Fim Enquanto

Output: **w**

completos, mas que contêm informação pouco relevante para o processo de estimação. Por exemplo, no caso de ser considerada uma amostra proveniente de uma distribuição de probabilidade simétrica, é desnecessário considerar a média e a mediana da amostra, uma vez que o seu valor será semelhante devido à propriedades probabilísticas subjacentes.

Para ser possível obter estimativas o mais rigorosas possível para os parâmetros da GPD através desta metodologia, é necessário ter em conta não só a dimensão da amostra em estudo (que é um dos pontos mais limitadores da eficiência dos métodos existentes), mas também os valores das estimativas dos parâmetros obtidas por cada um dos métodos envolvidos, com especial foco nas estimativas do parâmetro de forma, que é o maior responsável pelas propriedades da distribuição GPD subjacente à amostra. Neste contexto, são utilizados apenas os estimadores EPM e $(k^*_{NEW}, \sigma^*_{NEW})$ devido às suas propriedades, que foram anteriormente mencionadas.

A geração de cada observação do conjunto de treino centra-se na simulação de amostras de dimensão n da distribuição GPD(k,1), sendo que os parâmetros n e k também são simulados, mas a partir da distribuição Uniforme. No caso do parâmetro de forma, a geração é feita com recurso à distribuição Uniforme(-2.1,2.1) para possibilitar a obtenção de um valor dentro de um intervalo muito amplo de valores de k, semelhante ao considerado por Castillo & Hadi [10]. No que se refere à dimensão da amostra, a geração é feita a partir da distribuição Uniforme(10,200). Neste último caso, os valores dos parâmetros da Uniforme justificam-se com base nas aplicações usuais da distribuição GPD. Uma vez que a sua utilização é feita no âmbito do ajustamento de amostras de excessos de um determinado nível u relativamente elevado, é mais habitual que as dimensões obtidas sejam reduzidas, pelo que deverá ser dada maior importância a esta gama de dimensões de amostra.

Uma vez gerada a amostra, são obtidas as estimativas EPM e $(k^*_{NEW}, \sigma^*_{NEW})$ correspondentes que,

Algoritmo 6.4 Geração de elementos de treino para estimação dos parâmetros da GPD

Input: $-$

1: Gerar k da distribuição Uniforme$(-2.1, 2.1)$
2: Gerar n da distribuição Uniforme$(10, 200)$
3: Gerar amostra $\mathbf{g} = (g_1, \ldots, g_n)$ da distribuição GPD(k,1)
4: Obter estimativa k_{EPM} com base na amostra $\mathbf{g}$
5: Obter estimativa k^*_{NEW} com base na amostra $\mathbf{g}$
6: Construir observação de treino $\mathbf{x}$ usando a expressão (6.9)
7: Calcular classificação correta t para $\mathbf{x}$ usando a expressão (6.10)

Output: $(\mathbf{x}, t)$

juntamente com a dimensão n gerada, permitem a construção da observação de treino desejada, definida por

$$\mathbf{x} = (n, k_{EPM}, k^*_{NEW}). \tag{6.9}$$

No que toca à determinação da classificação correspondente, esta é feita combinando o valor de k gerado com as propriedades de cada um dos métodos envolvidos na composição desta metodologia. Uma vez que as estimativas obtidas pelo método EPM são preferidas às estimativas $(k^*_{NEW}, \sigma^*_{NEW})$ em casos de caudas leves, e o contrário em casos de caudas pesadas ou exponencial, é razoável afirmar que o método EPM é o mais adequado quando $k < 0$ e que as estimativas $(k^*_{NEW}, \sigma^*_{NEW})$ são as mais adequadas quando $k \geq 0$. Desta forma, ao considerar que a classe C_1 está associada às amostras que são bem ajustadas a uma GPD$(k^*_{NEW}, \sigma^*_{NEW})$ e que a classe C_2 está associada às amostras que são melhor ajustadas a uma GPD(k_{EPM}, σ_{EPM}), é possível adaptar a expressão (6.3) e estabelecer um critério de classificação definido por

$$t = \begin{cases} +1, & k \geq 0, \\ -1, & k < 0. \end{cases} \tag{6.10}$$

Assim, supondo que se pretende gerar uma observação de treino $\mathbf{x}$ e a sua classificação correta t, são apresentados no Algoritmo 6.4 os passos que devem ser executados para a sua obtenção.

6.4 Metodologia para estimação dos parâmetros da GPD

Perante os vários algoritmos apresentados para a estimação dos pesos de um perceptrão e o mecanismo para geração de elementos de treino do mesmo, resta agora estudar qual a melhor forma de os combinar para obter uma metodologia robusta para estimação dos parâmetros da GPD.

Em primeiro lugar, é importante relembrar a necessidade de se construir um conjunto de treino para estimar os parâmetros do perceptrão. Como é muito improvável encontrar conjuntos de treino relacionados com o objetivo do estudo apresentado, a opção mais prática passa por simular um conjunto de treino a partir do Algoritmo 6.4 apresentado anteriormente.

No que toca à escolha do algoritmo de estimação dos pesos do perceptrão, é de realçar a incerteza que se pode ter relativamente à separabilidade linear do conjunto de treino, independentemente de este ser simulado ou não. Por este motivo, a escolha do Algoritmo 6.1 para estimar os parâmetros do per-

Algoritmo 6.5 Metodologia de Classificação para Estimação (MCE)

Input: $\mathbf{g} = (g_1, \ldots, g_n)$

1: Obter $\mathbf{w}$ usando o Algoritmo 6.3, suportado pelo Algoritmo 6.4 (mecanismo alternativo)
2: Obter dimensão da amostra $\mathbf{g}$, n
3: Obter estimativas (k_{EPM}, σ_{EPM}) com base na amostra $\mathbf{g}$
4: Obter estimativas $(k^*_{NEW}, \sigma^*_{NEW})$ com base na amostra $\mathbf{g}$
5: Construir observação de treino $\mathbf{x}$ usando a expressão (6.9)
6: Calcular $y(\mathbf{x})$ usando a expressão (6.2)
7: **Se** $y(\mathbf{x}) \geq 0$ **então**
8: $\quad (k_{MCE}, \sigma_{MCE}) \leftarrow (k^*_{NEW}, \sigma^*_{NEW})$
9: **Caso contrário**
10: $\quad (k_{MCE}, \sigma_{MCE}) \leftarrow (k_{EPM}, \sigma_{EPM})$
11: **Fim Se**

Output: (k_{MCE}, σ_{MCE})

ceptrão não será a mais correta, deixando assim em aberto a possível utilização das técnicas baseadas no *pocket algorithm*. Contudo, recorde-se que no caso do conjunto de treino ser, de facto, linearmente separável, o Algoritmo 6.2 também é capaz de estimar corretamente o hiperplano que separa totalmente as observações das duas classes.

De forma a ser feita a escolha da versão mais correta do *pocket algorithm*, considerem-se algumas propriedades desejáveis para o conjunto de treino simulado. Como se sabe, é de todo o interesse que o processo de ”aprendizagem”do perceptrão seja o mais rigoroso possível, de modo a captar o comportamento do maior número de observações. Uma forma de se atingir este objetivo é através da utilização de um elevado número de observações de treino. No entanto, tal como foi referido no Capítulo 6.2, o Algoritmo 6.2 pode ter uma execução demasiado demorada se o conjunto de treino tiver uma dimensão relativamente grande. Assim, é possível deduzir que a utilização do Algoritmo 6.3 é a mais apropriada para o tipo de metodologia que se pretende criar.

A estrutura desta nova Metodologia de Classificação para Estimação (MCE) pode ser observada com detalhe no Algoritmo 6.5. Tendo em conta os argumentos utilizados, esta metodologia para estimação dos parâmetros da GPD combina a versão do *pocket algorithm for* ∞ *data* com o mecanismo utilizado para obtenção de elementos de treino descrito pelo Algoritmo 6.4. De modo a obter as estimativas (k_{MCE}, σ_{MCE}) de uma amostra observada $\mathbf{g} = (g_1, \ldots, g_n)$, esta é reduzida a uma observação $\mathbf{x}$ semelhante à da expressão (6.9), que é operada juntamente com o vetor de pesos $\mathbf{w}$ resultante do Algoritmo 6.3. Se o resultado dessa operação for não negativo, as estimativas (k_{MCE}, σ_{MCE}) serão iguais às estimativas $(k^*_{NEW}, \sigma^*_{NEW})$. Caso contrário, serão iguais às estimativas obtidas pelo método EPM.

Capítulo 7

Avaliação dos Métodos por Simulação

Neste capítulo serão apresentados os resultados de um estudo de simulação realizado de forma a avaliar e comparar entre si os métodos de estimação dos parâmetros da GPD apresentados ao longo deste trabalho. Antes de avançar, recordem-se os identificadores utilizados para cada método:

MLO Método de Máxima Verosimilhança Otimizado

EPM Método dos Percentis Elementares

ZS Método Quasi-Bayesiano desenvolvido por Zhang & Stephens

MAPE Método de Ajustamento Polinomial para Estimação

MCE Método de Classificação para Estimação

Para a simulação das amostras, recorreu-se à utilização de sementes (*seeds*) para possibilitar a replicação posterior dos resultados obtidos de forma aleatória. Esta técnica é útil para testar os métodos separadamente, mas utilizando a mesma amostra de teste. A semente utilizada para gerar as amostras depende não só da dimensão da amostra gerada aleatoriamente a partir da distribuição GPD(k,σ), como também do valor do parâmetro de forma considerado. A obtenção deste valor é feita a partir da concatenação das seguintes quantidades, pela ordem apresentada abaixo:

1. Dimensão da amostra gerada (n);
2. Valor absoluto do parâmetro de forma multiplicado por 100 $(|k| \times 100)$;
3. Valor binário indicando se o valor do parâmetro de forma é não negativo, ou não $(k \geq 0)$.

A implementação do método MCE também utiliza *seeds* no processo de geração de observações para treino do perceptrão, o que permite saber exatamente quais as observações de treino utilizadas neste processo. Para este caso, a *seed* utilizada é obtida através da soma entre o número total de observações de treino planeadas para este processo e a cardinalidade da observação de treino que está a ser gerada na iteração atual. Para efeitos de simulação e análise de casos de estudo, considera-se um conjunto de treino composto por 100 000 observações. A escolha deste número de observações relaciona-se com a necessidade de considerar um conjunto de treino suficientemente vasto, de modo a obter estimativas dos parâmetros do perceptrão que permitam uma classificação o mais fiável possível.

No âmbito da estimação paramétrica, é de interesse estudar o comportamento dos estimadores quer ao nível do desvio ao valor exato, quer ao nível da dispersão dos valores obtidos. Para este efeito,

é habitual considerar o viés dos estimadores e medidas que permitam avaliar a dispersão dos valores estimados. O viés de um estimador $\hat{\theta}$ de um parâmetro θ é obtido através da expressão

$$\text{Viés}(\hat{\theta}) = \text{E}_\theta(\hat{\theta} - \theta). \tag{7.1}$$

Nesta análise, será tomada como medida de dispersão a raiz quadrada do erro quadrático médio (RMSE), definido pela expressão

$$\text{RMSE}(\hat{\theta}) = \sqrt{\text{MSE}(\hat{\theta})} = \sqrt{\text{E}_\theta\left[(\hat{\theta} - \theta)^2\right]}. \tag{7.2}$$

Supondo que são geradas N amostras a partir da GPD(k, σ) e que é feito o ajustamento da distribuição GPD a cada uma delas, as estimativas do viés e do RMSE relativamente a um conjunto de estimativas $\hat{\theta}_1, \ldots, \hat{\theta}_N$ do parâmetro de interesse θ (com valor fixado) podem ser calculadas através das expressões

$$\text{Viés}(\hat{\theta}_1, \ldots, \hat{\theta}_N) = \frac{1}{N}\sum_{i=1}^{N}(\hat{\theta}_i - \theta), \tag{7.3}$$

$$\text{RMSE}(\hat{\theta}_1, \ldots, \hat{\theta}_N) = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(\hat{\theta}_i - \theta)^2}. \tag{7.4}$$

Neste estudo de simulação, todos os casos de análise serão feitos com base em 10 000 amostras simuladas da distribuição GPD$(k, 1)$, para diversos valores de k no intervalo $[-2, 2]$. Para além disto, consideram-se apenas amostras simuladas com dimensão 15, 25, 50, 100 e 200, dado que muitos estudos já existentes também utilizam amostras com dimensões semelhantes [48].

Na Tabela 7.1 é apresentado o valor do viés associado às estimativas obtidas para vários valores de $k < 0$ (caudas leves). Para esta gama de valores do parâmetro de forma, é possível verificar que o método EPM apresenta valores menos enviesados que as restantes técnicas, uma vez que o viés associado a cada estimativa obtida por este método é mais baixo que os restantes, independentemente da dimensão da amostra considerada. Para além disto, também é possível confirmar através destes resultados que as estimativas obtidas pelos métodos MLO e ZS não produzem boas estimativas para valores de k nesta zona do espaço paramétrico. No entanto, quando $k > -0.5$, os resultados obtidos pelo método ZS começam a aproximar-se dos valores obtidos pelo método EPM.

Relativamente aos valores do RMSE registados na Tabela 7.2, é possível verificar que a variabilidade das estimativas obtidas para os parâmetros diminui à medida que a dimensão das amostras aumenta, independentemente do método de estimação utilizado. Numa perspetiva mais alargada, é possível notar que os valores registados para o RMSE tendem a ser inferiores à medida que os valores do parâmetro de forma se aproximam de zero.

Considere-se agora a Tabela 7.3, contendo o valor do viés associado às estimativas obtidas para valores de $k \geq 0$ (caudas exponenciais e pesadas). Para estes valores do parâmetro de forma, o método ZS produz as estimativas mais precisas, com ainda melhores valores do que o método MLO, independentemente da dimensão da amostra considerada. No entanto, os valores obtidos permitem concluir que o método EPM pode não ser a melhor escolha quando a distribuição subjacente aos dados tem caudas pesadas.

No que toca aos valores do RMSE registados na Tabela 7.4, observa-se que a variabilidade das estimativas obtidas para os parâmetros da GPD aumenta à medida que se consideram valores mais elevados

para o parâmetro de forma. No entanto, continua-se a verificar que, para cada valor tomado para k, a variabilidade diminui quando a dimensão das amostras aumenta.

A consulta das Tabelas 7.1 a 7.4 também permite verificar que os métodos MAPE e MCE conseguem captar o desempenho da melhor metodologia de estimação para cada peso de cauda. Isto significa que, quando $k < 0$ (caudas leves), estes métodos têm um comportamento semelhante ao do método EPM. No caso em que $k \geq 0$ (caudas exponenciais ou pesadas), o seu comportamento é semelhante ao do método ZS.

	k_{MLO}	k_{EPM}	k_{ZS}	k_{MAPE}	k_{MCE}	σ_{MLO}	σ_{EPM}	σ_{ZS}	σ_{MAPE}	σ_{MCE}
n	(k,σ)=(-2,1)									
15	1.0004	-0.0569	1.0397	-0.0569	-0.0569	-0.5038	0.0299	-0.4205	0.0299	0.0299
25	1.0001	-0.0336	0.8286	-0.0336	-0.0336	-0.5013	0.0171	-0.3499	0.0171	0.0171
50	1.0001	-0.0164	0.6016	-0.0164	-0.0164	-0.5003	0.0081	-0.2662	0.0081	0.0081
100	1.0001	-0.0083	0.4416	-0.0083	-0.0083	-0.5000	0.0042	-0.2018	0.0042	0.0042
200	1.0001	-0.0028	0.3352	-0.0028	-0.0028	-0.5000	0.0014	-0.1566	0.0014	0.0014
n	(k,σ)=(-1.5,1)									
15	0.5023	-0.0383	0.7025	-0.0383	-0.0383	-0.3479	0.0303	-0.3479	0.0303	0.0303
25	0.5008	-0.0326	0.5273	-0.0326	-0.0326	-0.3402	0.0229	-0.2744	0.0229	0.0229
50	0.5001	-0.0129	0.3546	-0.0129	-0.0129	-0.3357	0.0087	-0.1959	0.0087	0.0087
100	0.5001	-0.0060	0.2400	-0.0060	-0.0060	-0.3342	0.0040	-0.1384	0.0040	0.0040
200	0.5001	-0.0024	0.1699	-0.0024	-0.0024	-0.3336	0.0016	-0.1008	0.0016	0.0016
n	(k,σ)=(-1,1)									
15	0.0259	-0.0086	0.4100	-0.0085	-0.0087	-0.0821	0.0273	-0.2567	0.0270	0.0273
25	0.0220	-0.0140	0.2824	-0.0140	-0.0140	-0.0578	0.0207	-0.1880	0.0207	0.0207
50	0.0199	-0.0015	0.1702	-0.0015	-0.0015	-0.0381	0.0046	-0.1223	0.0046	0.0046
100	0.0162	-0.0034	0.0977	-0.0034	-0.0034	-0.0256	0.0042	-0.0737	0.0042	0.0042
200	0.0138	-0.0016	0.0601	-0.0016	-0.0016	-0.0185	0.0019	-0.0470	0.0019	0.0019
n	(k,σ)=(-0.75,1)									
15	-0.1639	-0.0011	0.2814	-0.0011	-0.0016	0.0922	0.0350	-0.2001	0.0336	0.0350
25	-0.1436	-0.0022	0.1831	-0.0022	-0.0022	0.1068	0.0202	-0.1392	0.0202	0.0203
50	-0.1011	0.0025	0.0989	0.0025	0.0025	0.0881	0.0048	-0.0823	0.0048	0.0048
100	-0.0589	0.0009	0.0503	0.0009	0.0009	0.0544	0.0031	-0.0437	0.0031	0.0031
200	-0.0319	0.0001	0.0257	0.0001	0.0001	0.0298	0.0021	-0.0231	0.0021	0.0021
n	(k,σ)=(-0.5,1)									
15	-0.2620	0.0253	0.1742	0.0205	0.0196	0.2330	0.0293	-0.1421	0.0246	0.0300
25	-0.1818	0.0179	0.1039	0.0172	0.0176	0.1768	0.0142	-0.0908	0.0134	0.0144
50	-0.0907	0.0127	0.0483	0.0127	0.0127	0.0871	0.0035	-0.0463	0.0035	0.0035
100	-0.0462	0.0091	0.0213	0.0091	0.0091	0.0432	-0.0007	-0.0219	-0.0007	-0.0007
200	-0.0250	0.0062	0.0091	0.0062	0.0062	0.0231	-0.0017	-0.0100	-0.0017	-0.0017
n	(k,σ)=(-0.25,1)									
15	-0.2722	0.0514	0.0875	0.0208	0.0213	0.2964	0.0354	-0.0750	0.0256	0.0373
25	-0.1548	0.0425	0.0491	0.0276	0.0321	0.1634	0.0083	-0.0473	0.0028	0.0093
50	-0.0749	0.0290	0.0182	0.0244	0.0266	0.0753	-0.0005	-0.0188	-0.0022	0.0001
100	-0.0342	0.0242	0.0122	0.0238	0.0241	0.0331	-0.0077	-0.0129	-0.0078	-0.0076
200	-0.0190	0.0162	0.0055	0.0162	0.0162	0.0175	-0.0061	-0.0069	-0.0061	-0.0061

Tabela 7.1: Viés das estimativas de (k,σ) para $k < 0$ (caudas leves)

	k_{MLO}	k_{EPM}	k_{ZS}	k_{MAPE}	k_{MCE}	σ_{MLO}	σ_{EPM}	σ_{ZS}	σ_{MAPE}	σ_{MCE}
n	(k,σ)=(-2,1)									
15	1.0004	0.6727	1.0575	0.6727	0.6727	0.5038	0.3287	0.4327	0.3287	0.3287
25	1.0001	0.4952	0.8467	0.4952	0.4952	0.5014	0.2450	0.3615	0.2450	0.2450
50	1.0001	0.3412	0.6193	0.3412	0.3412	0.5003	0.1701	0.2767	0.1701	0.1701
100	1.0001	0.2372	0.4575	0.2372	0.2372	0.5000	0.1185	0.2108	0.1185	0.1185
200	1.0001	0.1665	0.3483	0.1665	0.1665	0.5000	0.0833	0.1637	0.0833	0.0833
n	(k,σ)=(-1.5,1)									
15	0.5031	0.5425	0.7279	0.5425	0.5425	0.3489	0.3395	0.3711	0.3395	0.3395
25	0.5010	0.3891	0.5524	0.3891	0.3891	0.3404	0.2507	0.2964	0.2507	0.2507
50	0.5001	0.2627	0.3794	0.2627	0.2627	0.3357	0.1725	0.2162	0.1725	0.1725
100	0.5001	0.1805	0.2623	0.1805	0.1805	0.3342	0.1196	0.1557	0.1196	0.1196
200	0.5001	0.1248	0.1882	0.1248	0.1247	0.3336	0.0830	0.1146	0.0830	0.0830
n	(k,σ)=(-1,1)									
15	0.1078	0.4256	0.4510	0.4256	0.4254	0.1291	0.3506	0.3070	0.3510	0.3506
25	0.0812	0.2980	0.3224	0.2980	0.2980	0.0960	0.2610	0.2376	0.2610	0.2610
50	0.0611	0.1940	0.2082	0.1940	0.1940	0.0687	0.1794	0.1685	0.1794	0.1794
100	0.0441	0.1269	0.1316	0.1269	0.1269	0.0484	0.1214	0.1138	0.1214	0.1214
200	0.0330	0.0862	0.0872	0.0862	0.0862	0.0353	0.0842	0.0785	0.0842	0.0842
n	(k,σ)=(-0.75,1)									
15	0.2469	0.3959	0.3428	0.3949	0.3948	0.2222	0.3740	0.2835	0.3753	0.3739
25	0.2204	0.2681	0.2414	0.2681	0.2681	0.2163	0.2747	0.2210	0.2748	0.2746
50	0.1719	0.1643	0.1506	0.1643	0.1643	0.1848	0.1809	0.1542	0.1809	0.1809
100	0.1164	0.1067	0.0955	0.1067	0.1067	0.1334	0.1252	0.1072	0.1252	0.1252
200	0.0733	0.0708	0.0626	0.0708	0.0708	0.0872	0.0865	0.0746	0.0865	0.0865
n	(k,σ)=(-0.5,1)									
15	0.3902	0.3694	0.2720	0.3596	0.3594	0.4293	0.3857	0.2777	0.3877	0.3846
25	0.2990	0.2441	0.1922	0.2425	0.2433	0.3554	0.2789	0.2187	0.2797	0.2787
50	0.1787	0.1510	0.1269	0.1510	0.1510	0.2223	0.1895	0.1590	0.1895	0.1895
100	0.1040	0.0955	0.0845	0.0955	0.0955	0.1367	0.1297	0.1147	0.1297	0.1297
200	0.0645	0.0621	0.0567	0.0621	0.0621	0.0896	0.0905	0.0816	0.0905	0.0905
n	(k,σ)=(-0.25,1)									
15	0.4569	0.3775	0.2534	0.3390	0.3402	0.5830	0.4156	0.3018	0.4123	0.4133
25	0.3030	0.2511	0.1925	0.2319	0.2361	0.3939	0.2966	0.2432	0.2964	0.2953
50	0.1741	0.1594	0.1352	0.1527	0.1552	0.2301	0.2007	0.1803	0.2012	0.1998
100	0.1030	0.1035	0.0922	0.1027	0.1032	0.1439	0.1382	0.1299	0.1383	0.1379
200	0.0660	0.0693	0.0616	0.0693	0.0693	0.0963	0.0962	0.0913	0.0962	0.0962

Tabela 7.2: RMSE das estimativas de (k,σ) para $k < 0$ (caudas leves)

	k_{MLO}	k_{EPM}	k_{ZS}	k_{MAPE}	k_{MCE}	σ_{MLO}	σ_{EPM}	σ_{ZS}	σ_{MAPE}	σ_{MCE}
n	(k,σ)=(0,1)									
15	-0.2443	0.0886	0.0263	0.0016	0.0021	0.2884	0.0281	-0.0228	0.0282	0.0390
25	-0.1276	0.0753	0.0174	0.0134	0.0175	0.1444	0.0019	-0.0139	0.0026	0.0089
50	-0.0586	0.0555	0.0086	0.0115	0.0156	0.0629	-0.0094	-0.0062	-0.0066	-0.0025
100	-0.0298	0.0407	0.0048	0.0069	0.0086	0.0313	-0.0107	-0.0035	-0.0064	-0.0031
200	-0.0145	0.0327	0.0031	0.0046	0.0039	0.0144	-0.0117	-0.0030	-0.0061	-0.0030
n	(k,σ)=(0.25,1)									
15	-0.2204	0.1262	-0.0126	-0.0244	-0.0226	0.2841	0.0292	0.0252	0.0513	0.0564
25	-0.1150	0.1066	-0.0043	-0.0071	-0.0035	0.1356	-0.0034	0.0113	0.0159	0.0183
50	-0.0519	0.0842	-0.0002	-0.0019	0.0005	0.0648	-0.0105	0.0106	0.0099	0.0111
100	-0.0244	0.0663	0.0019	0.0004	0.0017	0.0287	-0.0187	0.0019	0.0017	0.0020
200	-0.0118	0.0552	0.0012	0.0006	0.0012	0.0141	-0.0195	0.0010	0.0009	0.0010
n	(k,σ)=(0.5,1)									
15	-0.1954	0.1777	-0.0330	-0.0380	-0.0370	0.2672	0.0212	0.0585	0.0703	0.0730
25	-0.1075	0.1457	-0.0185	-0.0208	-0.0183	0.1460	0.0030	0.0436	0.0444	0.0454
50	-0.0441	0.1200	-0.0021	-0.0035	-0.0021	0.0553	-0.0266	0.0112	0.0113	0.0112
100	-0.0197	0.1009	0.0018	0.0014	0.0018	0.0258	-0.0315	0.0039	0.0038	0.0038
200	-0.0096	0.0793	0.0009	0.0009	0.0009	0.0127	-0.0282	0.0021	0.0021	0.0021
n	(k,σ)=(0.75,1)									
15	-0.1880	0.2203	-0.0517	-0.0556	-0.0535	0.2915	0.0373	0.1052	0.1113	0.1122
25	-0.0928	0.1966	-0.0172	-0.0198	-0.0171	0.1392	-0.0099	0.0526	0.0537	0.0533
50	-0.0440	0.1548	-0.0074	-0.0080	-0.0074	0.0573	-0.0349	0.0188	0.0189	0.0188
100	-0.0165	0.1323	0.0026	0.0026	0.0026	0.0247	-0.0403	0.0052	0.0052	0.0052
200	-0.0067	0.1125	0.0027	0.0027	0.0027	0.0129	-0.0378	0.0034	0.0034	0.0034
n	(k,σ)=(1,1)									
15	-0.1672	0.2877	-0.0494	-0.0525	-0.0504	0.2881	0.0377	0.1283	0.1324	0.1323
25	-0.0939	0.2352	-0.0257	-0.0278	-0.0258	0.1505	-0.0030	0.0720	0.0728	0.0723
50	-0.0468	0.1864	-0.0125	-0.0127	-0.0125	0.0658	-0.0345	0.0296	0.0297	0.0296
100	-0.0196	0.1630	-0.0011	-0.0012	-0.0011	0.0295	-0.0458	0.0108	0.0108	0.0108
200	-0.0099	0.1408	-0.0007	-0.0007	-0.0007	0.0159	-0.0441	0.0066	0.0066	0.0066
n	(k,σ)=(1.5,1)									
15	-0.1658	0.3983	-0.0612	-0.0639	-0.0615	0.3362	0.0757	0.1972	0.2009	0.1990
25	-0.0997	0.3288	-0.0350	-0.0356	-0.0350	0.1686	-0.0053	0.0961	0.0966	0.0961
50	-0.0376	0.2805	-0.0029	-0.0030	-0.0029	0.0698	-0.0498	0.0339	0.0340	0.0338
100	-0.0214	0.2356	-0.0013	-0.0013	-0.0013	0.0372	-0.0586	0.0168	0.0168	0.0168
200	-0.0113	0.1943	-0.0012	-0.0012	-0.0012	0.0185	-0.0578	0.0083	0.0083	0.0083
n	(k,σ)=(2,1)									
15	-0.1747	0.5142	-0.0605	-0.0621	-0.0606	0.3899	0.1171	0.2479	0.2525	0.2500
25	-0.0920	0.4456	-0.0192	-0.0195	-0.0192	0.1894	0.0052	0.1106	0.1110	0.1106
50	-0.0418	0.3618	0.0001	0.0001	0.0001	0.0839	-0.0521	0.0411	0.0411	0.0411
100	-0.0235	0.3000	0.0011	0.0011	0.0011	0.0406	-0.0690	0.0157	0.0157	0.0157
200	-0.0117	0.2463	0.0005	0.0005	0.0005	0.0186	-0.0689	0.0063	0.0063	0.0063

Tabela 7.3: Viés das estimativas de (k,σ) para $k \geq 0$ (caudas pesadas e exponencial)

	k_{MLO}	k_{EPM}	k_{ZS}	k_{MAPE}	k_{MCE}	σ_{MLO}	σ_{EPM}	σ_{ZS}	σ_{MAPE}	σ_{MCE}
n	(k,σ)=(0,1)									
15	0.4795	0.4147	0.2826	0.3338	0.3339	0.6500	0.4388	0.3476	0.4246	0.4311
25	0.3130	0.2958	0.2294	0.2425	0.2427	0.4180	0.3255	0.2887	0.3165	0.3201
50	0.1806	0.1927	0.1568	0.1542	0.1542	0.2405	0.2166	0.2057	0.2096	0.2114
100	0.1154	0.1357	0.1067	0.1046	0.1044	0.1566	0.1517	0.1448	0.1456	0.1463
200	0.0772	0.0998	0.0734	0.0725	0.0729	0.1048	0.1055	0.1002	0.1000	0.1005
n	(k,σ)=(0.25,1)									
15	0.5077	0.4831	0.3443	0.3679	0.3688	0.7177	0.4819	0.4226	0.4674	0.4725
25	0.3304	0.3581	0.2690	0.2716	0.2706	0.4295	0.3448	0.3240	0.3335	0.3353
50	0.2043	0.2539	0.1870	0.1871	0.1859	0.2613	0.2421	0.2326	0.2327	0.2338
100	0.1340	0.1898	0.1282	0.1301	0.1286	0.1688	0.1683	0.1597	0.1595	0.1598
200	0.0921	0.1504	0.0900	0.0912	0.0901	0.1158	0.1218	0.1126	0.1125	0.1126
n	(k,σ)=(0.5,1)									
15	0.5310	0.5771	0.4074	0.4179	0.4185	0.7286	0.5130	0.4784	0.5052	0.5066
25	0.3587	0.4443	0.3138	0.3150	0.3141	0.4648	0.3885	0.3764	0.3783	0.3800
50	0.2308	0.3335	0.2193	0.2212	0.2192	0.2717	0.2625	0.2496	0.2496	0.2496
100	0.1552	0.2609	0.1516	0.1524	0.1517	0.1856	0.1935	0.1782	0.1781	0.1782
200	0.1070	0.2067	0.1059	0.1059	0.1059	0.1242	0.1405	0.1219	0.1219	0.1219
n	(k,σ)=(0.75,1)									
15	0.5714	0.6773	0.4711	0.4760	0.4761	0.8143	0.5734	0.5679	0.5795	0.5811
25	0.3948	0.5480	0.3647	0.3667	0.3645	0.4775	0.4172	0.4070	0.4085	0.4087
50	0.2605	0.4136	0.2523	0.2535	0.2524	0.2895	0.2889	0.2707	0.2711	0.2706
100	0.1794	0.3375	0.1775	0.1775	0.1775	0.1957	0.2170	0.1897	0.1897	0.1897
200	0.1271	0.2818	0.1267	0.1267	0.1267	0.1368	0.1646	0.1347	0.1348	0.1347
n	(k,σ)=(1,1)									
15	0.6149	0.8144	0.5488	0.5523	0.5514	0.8576	0.6492	0.6438	0.6552	0.6544
25	0.4413	0.6531	0.4196	0.4219	0.4197	0.5198	0.4700	0.4557	0.4569	0.4562
50	0.2954	0.5007	0.2897	0.2902	0.2897	0.3179	0.3278	0.2995	0.2995	0.2995
100	0.2041	0.4162	0.2030	0.2031	0.2030	0.2080	0.2391	0.2023	0.2023	0.2023
200	0.1403	0.3478	0.1400	0.1400	0.1400	0.1463	0.1885	0.1442	0.1442	0.1442
n	(k,σ)=(1.5,1)									
15	0.7189	1.0819	0.6918	0.6943	0.6928	1.0078	0.8233	0.8176	0.8335	0.8238
25	0.5269	0.8778	0.5194	0.5206	0.5194	0.5856	0.5622	0.5324	0.5345	0.5324
50	0.3586	0.6952	0.3588	0.3591	0.3588	0.3566	0.3943	0.3405	0.3409	0.3404
100	0.2506	0.5778	0.2512	0.2512	0.2512	0.2343	0.2988	0.2281	0.2281	0.2281
200	0.1778	0.4858	0.1780	0.1780	0.1780	0.1646	0.2368	0.1624	0.1624	0.1624
n	(k,σ)=(2,1)									
15	0.8247	1.3644	0.8330	0.8351	0.8335	1.2694	1.1784	1.0463	1.1176	1.0918
25	0.6234	1.1290	0.6326	0.6333	0.6326	0.6592	0.6702	0.6049	0.6060	0.6049
50	0.4295	0.8927	0.4354	0.4355	0.4354	0.3964	0.4671	0.3787	0.3787	0.3787
100	0.3014	0.7450	0.3041	0.3041	0.3041	0.2581	0.3528	0.2508	0.2508	0.2508
200	0.2131	0.6194	0.2140	0.2140	0.2140	0.1804	0.2852	0.1779	0.1779	0.1779

Tabela 7.4: RMSE das estimativas de (k,σ) para $k \geq 0$ (caudas pesadas e exponencial)

Capítulo 8

Casos de Estudo

Neste capítulo são estudados dois conjuntos de dados reais aos quais se pretende fazer o ajustamento da GPD, sendo os parâmetros da distribuição estimados através dos métodos apresentados ao longo deste trabalho. Em seguida, pretende-se avaliar a qualidade do ajustamento do modelo GPD e comparar com os resultados empíricos. Este estudo envolve os seguintes tópicos:

- Análise gráfica preliminar dos dados;
- Estimação dos parâmetros para cada valor do nível u;
- Comparação das correlações entre os quantis empíricos e os quantis estimados;
- Testes de Ajustamento (*Goodness-of-fit tests*);
- Comparação dos níveis e períodos de retorno obtidos através dos diferentes modelos GPD estimados, com os correspondes valores empíricos.

A análise gráfica preliminar dos dados baseia-se na utilização habitual de gráficos de barras, histogramas e *boxplots* para estudar a distribuição dos valores em estudo, bem como a sua dispersão. Ao nível do estudo de Valores Extremos, esta análise apoia-se na forma do QQ-Plot Exponencial e da representação gráfica da função de excesso médio, apresentados anteriormente.

O ajustamento dos modelos é avaliado através de testes não paramétricos apresentados por Choulakian & Stephens [11] para testar se uma amostra $\mathbf{x} = (x_1, \ldots, x_n)$ observada de uma população X pode ou não ser considerada como sendo proveniente de uma população GPD, com parâmetros estimados $\hat{k}$ e $\hat{\sigma}$, isto é, para realizar o teste

$$H_0 : X \sim \text{GPD}(\hat{k}, \hat{\sigma}) \quad \text{vs.} \quad H_1 : X \nsim \text{GPD}(\hat{k}, \hat{\sigma}).$$

A rejeição (ou não rejeição) de H_0 é feita com base nas estatísticas de Cramér-von Mises (W^2) e de Anderson-Darling (A^2), definidas respetivamente pelas expressões

$$W^2 = \frac{1}{12n} + \sum_{i=1}^{n} \left[\frac{z_{(i)} - (2i-1)}{2n} \right]^2, \quad z_{(i)} = F(x_{(i)} \mid \hat{k}, \hat{\sigma}), \tag{8.1}$$

$$A^2 = -n - \frac{1}{n} \sum_{i=1}^{n} (2i-1) \log \left[z_{(i)} \left(1 - z_{(n+1-i)} \right) \right], \quad z_{(i)} = F(x_{(i)} \mid \hat{k}, \hat{\sigma}), \tag{8.2}$$

bem como nos correspondentes valores-p (*p-values*), que são calculados com recurso a rotinas implementadas na biblioteca `goftest` do *software* R. Em particular, a estatística A^2 é calculada de acordo com

as metodologias apresentadas por Marsaglia & Marsaglia [31]. É mais vantajoso dar uma atenção particular aos valores desta última estatística, dado que a sua expressão atribui maior peso a observações na cauda da distribuição. Isto permite a deteção de *outliers* e, no caso da GPD, a validação das estimativas obtidas para os parâmetros, com particular interesse na estimativa do parâmetro de forma, uma vez que o comportamento da cauda da distribuição depende do seu valor.

No que toca aos níveis e aos períodos de retorno, estes serão utilizados para comparar em que medida os modelos ajustados estimam valores superiores ou inferiores aos valores observados. De forma sumária, o nível de retorno de m observações representa o valor que é excedido, em média, uma vez em cada m observações registadas, enquanto que o período de retorno de um valor x é o número de observações que devem ser observadas, em média, até que se volte a registar esse mesmo valor. Estas quantidades são calculadas de acordo com as expressões apresentadas por Coles [12]. Conjugando-as com a GPD, a expressão utilizada para calcular o nível de retorno de m observações, relativamente a um nível u fixado, é dada por

$$\widehat{x_m} = u + \frac{\hat{\sigma}}{\hat{k}}\left[(m\zeta_u)^{\hat{k}} - 1\right], \qquad \zeta_u = P(X > u), \tag{8.3}$$

onde m tem de ser suficientemente elevado, de forma a garantir que $x_m > u$. Para calcular o período de retorno de um valor x, relativamente a um nível u fixado, é utilizada a expressão

$$\widehat{R(x)} = \zeta_u\left[1 - F_u(x-u)\right], \qquad \zeta_u = P(X > u), \tag{8.4}$$

onde F_u representa a função de distribuição da GPD ajustada aos excessos acima do nível u. Supondo que existem n_u valores da amostra observada superiores ao nível u fixado, o valor de ζ_u pode ser estimado empiricamente por n_u/n.

8.1 Melhores Marcas Femininas em Triplo Salto

O Triplo Salto é uma modalidade do Atletismo que é realizada em pista (coberta ou descoberta) e faz parte da classe dos Saltos Horizontais, juntamente com o Salto em Comprimento. O seu nome resulta da decomposição do salto em três fases (*hop, step and jump*), que termina com uma queda numa caixa de areia.

8.1.1 Generalidades Históricas e Prática da Modalidade

De acordo com dados históricos [27], esta modalidade já era praticada no século XIX enquanto desporto exclusivamente masculino, até que começou a ter adesão feminina no início do século XX. Apesar dos registos de diversos recordes mundiais que tinham sido quebrados desde então, a International Association of Athletics Federations (IAAF) apenas começou a oficializar a lista de recordes em 1912, dando prioridade às marcas masculinas. A primeira a ser reconhecida foi a marca obtida pelo atleta irlandês Daniel Ahearn (15.52 metros), em 1911. Atualmente, o recorde mundial masculino pertence ao atleta britânico Jonathan Edwards (18.29 metros), que o conquistou em 1995, nos Campeonatos Mundiais de Atletismo.

Os recordes femininos só começaram a ser oficializados muito mais tarde, em 1990, apesar da prática competitiva regular da modalidade em países como os Estados Unidos da América, União das Repúblicas

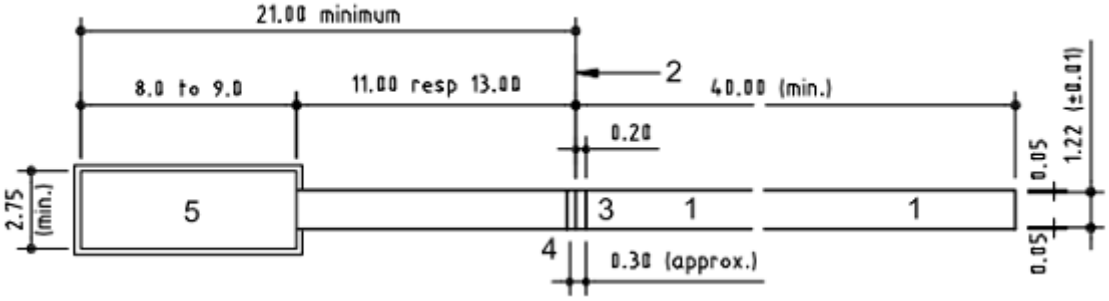

Figura 8.1: Zonas e medidas (em metros) de uma pista de Triplo Salto: 1) pista de balanço; 2) linha de chamada; 3) tábua de chamada; 4) tábua de ajustes; 5) zona de aterragem (Fonte: Government of Western Australia [22])

Socialistas Soviéticas e China. Nesse mesmo ano, foi registado o primeiro recorde feminino oficial, que foi atribuído à atleta chinesa Li Huirong (14.54 metros). O atual recorde mundial feminino está na posse da atleta ucraniana Inessa Kravets (15.50 metros), conquistado também em 1995, na mesma competição na qual Edwards conquistou o recorde referido anteriormente.

Nos últimos anos, o Triplo Salto português tem tido uma grande projeção em diversas competições europeias e mundiais, não só pela conquista de títulos, mas também pelo desempenho dos atletas da modalidade. Nelson Évora é um dos maiores representantes nacionais da modalidade por todo o mundo, tendo já marcado presença em diversas competições internacionais e conquistado diversas medalhas. Destes sucessos podem destacar-se as medalhas de ouro nos Jogos Olímpicos de Pequim (17.67 metros), em 2008, e nos Campeonatos Mundiais de Atletismo realizados em Osaka (17.74 metros), em 2007, onde estabeleceu a sua Melhor Marca Pessoal e Melhor Marca Nacional masculina, as quais mantém até ao momento. Em 2017, o atleta já renovou o título de campeão europeu conquistado nos Campeonatos Europeus de Pista Coberta (Belgrado, 17.20 metros) e sagrou-se campeão nacional em pista descoberta (Vagos, 16.78 metros).

Nas senhoras, Patrícia Mamona tem colocado a marca portuguesa em diversas competições internacionais e tem vindo a quebrar Recordes Nacionais de forma sucessiva. Neste último tópico, apenas em Campeonatos Europeus de Atletismo em pista descoberta, a atleta conquistou a medalha de prata em Helsínquia (14.52 metros), em 2012, e a medalha da ouro em Amesterdão (14.58 metros), em 2016. Neste mesmo ano, conseguiu ainda garantir o sexto lugar (14.65 metros) na final da modalidade nos Jogos Olímpicos realizados no Rio de Janeiro, juntamente com a conterrânea Susana Costa, que terminou a competição em nono lugar (14.12 metros). Em 2017, Mamona já conta com a vitória do Circuito Mundial de Pista Coberta (Dusseldorf, 14.11 metros), onde conquistou o *wild card* para os Campeonatos Mundiais de Pista Coberta de 2018, uma medalha de prata nos Campeonatos Europeus de Pista Coberta (Belgrado, 14.32 metros) e o título de campeã nacional em pista descoberta (Vagos, 14.40 metros).

Relativamente à prática da modalidade, apresenta-se na Figura 8.1 a estrutura de uma pista de Triplo Salto utilizada em competições internacionais, bem como as medidas de cada zona. Para os atletas masculinos, a distância entre a linha de partida e o fim da zona de aterragem deve ser de 21 metros, no mínimo, o que implica que o comprimento da zona de salto (limitada pela linha de partida e o início da zona de aterragem) não deve ser inferior a 13 metros. Para as atletas femininas, o comprimento da zona de salto não deve ser inferior a 11 metros. Nas restantes competições, estas distâncias são ajustadas consoante o nível das mesmas.

No final de cada prova, a distância percorrida (isto é, a marca) é medida entre a linha de chamada e o ponto de contacto mais recuado que o atleta teve com a caixa de areia. Apesar de válida, esta marca não será considerada para recorde se for conseguida com vento a favor do atleta com velocidade superior a 2 metros por segundo. Por outro lado, a marca também poderá ser anulada se o atleta tiver cometido alguma infração [28] durante a realização do salto.

8.1.2 Análise Exploratória dos Dados

O conjunto de dados em estudo refere-se às Melhores Marcas Pessoais de Triplo Salto, obtidas entre 2002 e 2016 em provas de pista descoberta, por atletas femininas sénior que foram incluídas em, pelo menos, uma *Top List* anual da IAAF no período considerado. Nestas condições, foram consideradas as Melhores Marcas Pessoais de 769 atletas femininas sénior, obtidas com base nas *Top Lists* anuais disponíveis no *site* da IAAF, para cada ano de interesse.

Comece-se por analisar os dados agrupados por cada ano de interesse. As representações gráficas da Figura 8.2 permitem comparar o número de atletas selecionadas para cada *Top List* anual, bem como a dispersão das Melhores Marcas Pessoais respetivas em cada ano. Numa primeira análise, é possível notar que o número de atletas consideradas anualmente é, em geral, superior a 150 e que todas as Melhores Marcas Pessoais são superiores ou iguais a 13.2 metros. Estes valores estão relacionados com os critérios utilizados para a inclusão das atletas nas *Top Lists* anuais. Uma vez que se exige que estas listas considerem perto de 150 atletas por ano, a regra aplicada consiste em recolher as Melhores Marcas Pessoais que, no ano de interesse, não sejam inferiores a 13.2 ou 13.3 metros, dependendo se já foi obtido o número de atletas pretendido ou não.

Focando agora a análise sobre os *boxplots* anuais, verifica-se que a variabilidade das amostras de Melhores Marcas Pessoais em cada ano é semelhante, à exceção de 2010 e 2011. Para 2010, observa-se uma menor variação dos valores das Melhores Marcas Pessoais, enquanto que para 2011, a variação dos valores é superior à dos restantes anos. Também é possível notar a assimetria positiva em todos os *boxplots*, indicando que as marcas consideradas em cada ano concentram-se mais em valores mais baixos. Este comportamento é natural, uma vez que a obtenção de marcas mais baixas é fisicamente menos exigente do que a obtenção de marcas mais elevadas.

A presença de *outliers* nos *boxplots* anuais também é um fator de interesse, uma vez que permite avaliar a existência de Melhores Marcas Pessoais que se sobressaíram das demais nesses anos e que, em particular, permite avaliar a relevância das Melhores Marcas anuais registadas. Repare-se que todos os *boxplots* apresentados contemplam a existência de *outliers* superiores, à exceção dos que estão associados a anos entre 2011 e 2013. Isto permite concluir que houve Melhores Marcas Pessoais que se destacaram das restantes (ou seja, correspondem a saltos mais longos que a maioria) na maioria dos anos, mas que entre 2011 e 2013, as Melhores Marcas registadas são igualmente relevantes e nenhuma delas merece um destaque particular.

De forma a completar as conclusões obtidas através da análise dos *boxplots* anuais, são apresentadas na Tabela 8.1 as Melhores Marcas anuais nos anos de interesse e a atleta que a alcançou. Com base nestes valores, verifica-se que as marcas máximas registadas para cada ano, à exceção dos anos entre 2011 e 2013, estão suficientemente distanciadas da barreira de *outliers* superior de cada ano para serem consideradas Melhores Marcas anuais relevantes. No entanto, é possível notar que a Melhor Marca de 2002 não está tão distanciada da barreira de *outliers* como os máximos anuais registados nos restantes

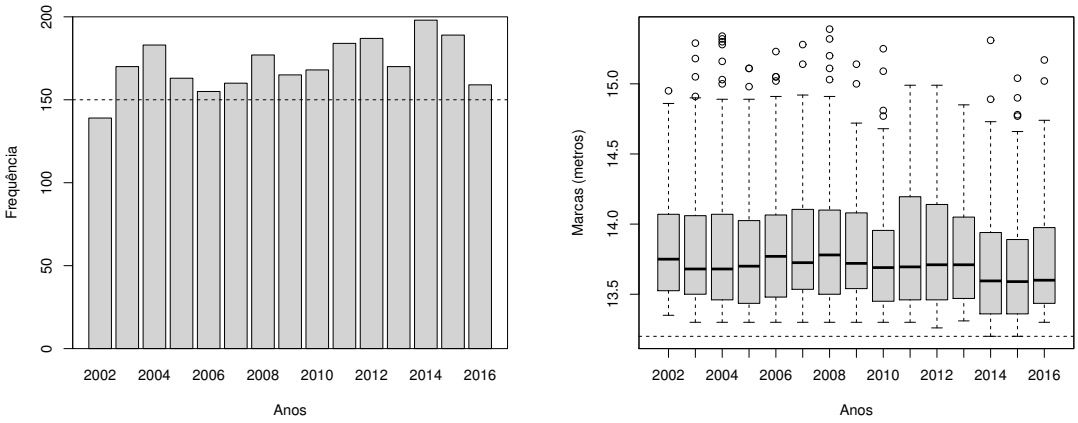

Figura 8.2: Frequências (à esquerda) e *boxplots* (à direita) anuais das Melhores Marcas Pessoais das atletas femininas de Triplo Salto consideradas

anos. Assim, é razoável concluir que a Melhor Marca de 2002 não é tão relevante como as restantes Melhores Marcas anuais.

Ano	Marca	Atleta
2002	14.95	Françoise MBANGO ETONE
2003	15.29	Yamilé ALDAMA
2004	15.34	Tatyana LEBEDEVA
2005	15.11	Tatyana LEBEDEVA
2005	15.11	Trecia SMITH
2006	15.23	Tatyana LEBEDEVA
2007	15.28	Yargeris SAVIGNE
2008	15.39	Françoise MBANGO ETONE
2009	15.14	Nadezhda ALEKHINA

Ano	Marca	Atleta
2010	15.25	Olga RYPAKOVA
2011	14.99	Caterine IBARGÜEN
2011	14.99	Yargeris SAVIGNE
2012	14.99	Olga SALADUKHA
2013	14.85	Caterine IBARGÜEN
2013	14.85	Olga SALADUKHA
2014	15.31	Caterine IBARGÜEN
2015	15.04	Ekaterina KONEVA
2016	15.17	Caterine IBARGÜEN

Tabela 8.1: Melhores Marcas anuais femininas entre 2002 e 2016

Considerem-se agora os dados agrupados por atleta, com o objetivo de estudar a amostra composta pelas Marcas Máximas de cada atleta feminina nas condições referidas inicialmente. Note-se que cada um destes valores máximos pode ser interpretado como o máximo de um bloco de observações, se pensarmos em cada atleta como o seu próprio bloco. As representações gráficas da Figura 8.3 permitem obter informações acerca da distribuição de valores extremos que está subjacente à amostra, uma vez que esta é composta por valores máximos. Apesar da alteração do critério de agrupamento dos dados, continua a observar-se a assimetria positiva dos dados, dada a grande concentração em torno de valores mais baixos, próximos da marca de 13.6 metros. Observando o histograma com maior detalhe, é ainda possível verificar que não existem muitas Marcas Máximas com valores significativamente elevados, o que indica que a distribuição de probabilidade subjacente aos dados não aparenta ter cauda pesada.

De forma a refinar as conclusões obtidas acerca do peso de cauda da distribuição subjacente aos dados, apresentam-se na Figura 8.4 o QQ-Plot Exponencial e o Gráfico de Excesso Médio, obtido como função da amostra ordenada. Uma vez que o QQ-Plot tem um formato ligeiramente côncavo e que o Gráfico de Excesso Médio apresenta um padrão decrescente, podemos concluir com segurança que a distribuição subjacente aos dados tem caudas mais leves que a distribuição Exponencial (conforme informação da Figura 2.2).

No que toca à escolha do valor do nível u a fixar, apresenta-se na Figura 8.5 o Gráfico de Excesso

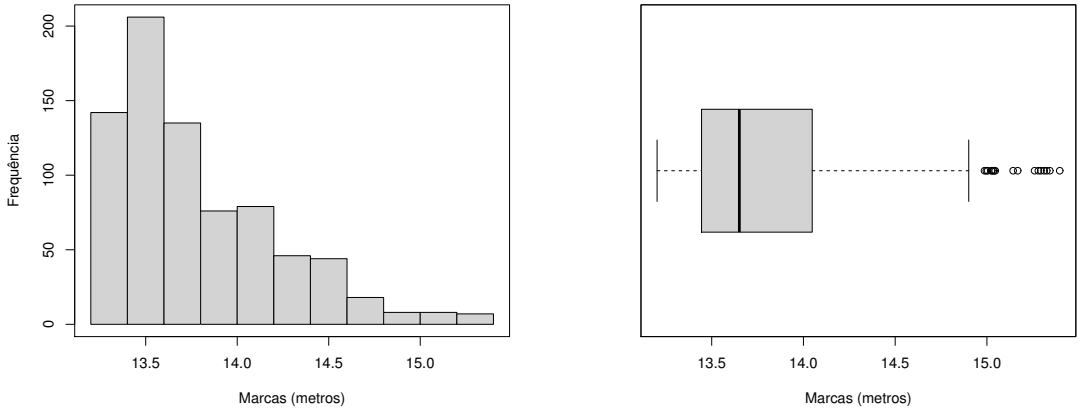

Figura 8.3: Histograma (à esquerda) e *boxplot* (à direita) das Melhores Marcas Pessoais das atletas femininas de Triplo Salto consideradas

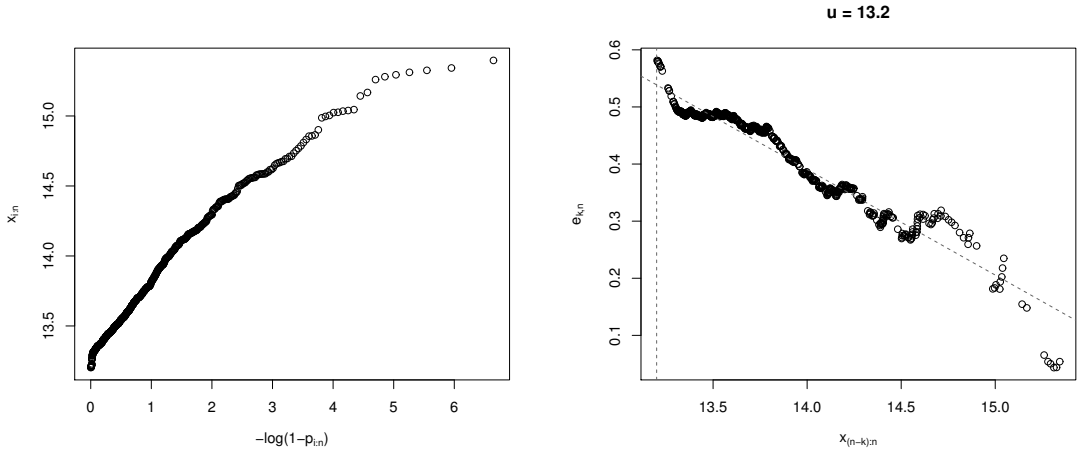

Figura 8.4: QQ-Plot Exponencial (à esquerda) e Gráfico de Excesso Médio (à direita) das Melhores Marcas Pessoais das atletas femininas de Triplo Salto consideradas

Médio da amostra de Melhores Marcas, juntamente com uma reta ajustada aos excessos médios calculados acima de cada nível u considerado. Por observação destes gráficos, é possível notar que se obtêm melhores ajustamentos lineares à medida que são considerados níveis mais elevados. Tal como é habitual, a análise gráfica não permite claramente identificar qual o melhor valor para o nível u. No entanto, $u = 14.7$ parece ser um possível candidato, embora o número de excessos seja bastante reduzido (29 observações).

8.1.3 Estimação Paramétrica e Inferência

Uma vez selecionados os níveis de interesse através da análise do Gráfico de Excesso Médio da amostra de Marcas Máximas, é possível proceder ao ajustamento da GPD sobre os excessos acima de cada valor u. Na Tabela 8.2 são apresentadas as estimativas de (k, σ) obtidas por cada metodologia mencionada, para cada nível u selecionado, bem como o número de excessos (m) desse nível.

Numa primeira análise dos resultados obtidos, é reforçada a conclusão acerca do peso de cauda da distribuição das Marcas Máximas (caudas mais leves que a distribuição Exponencial), uma vez que todas

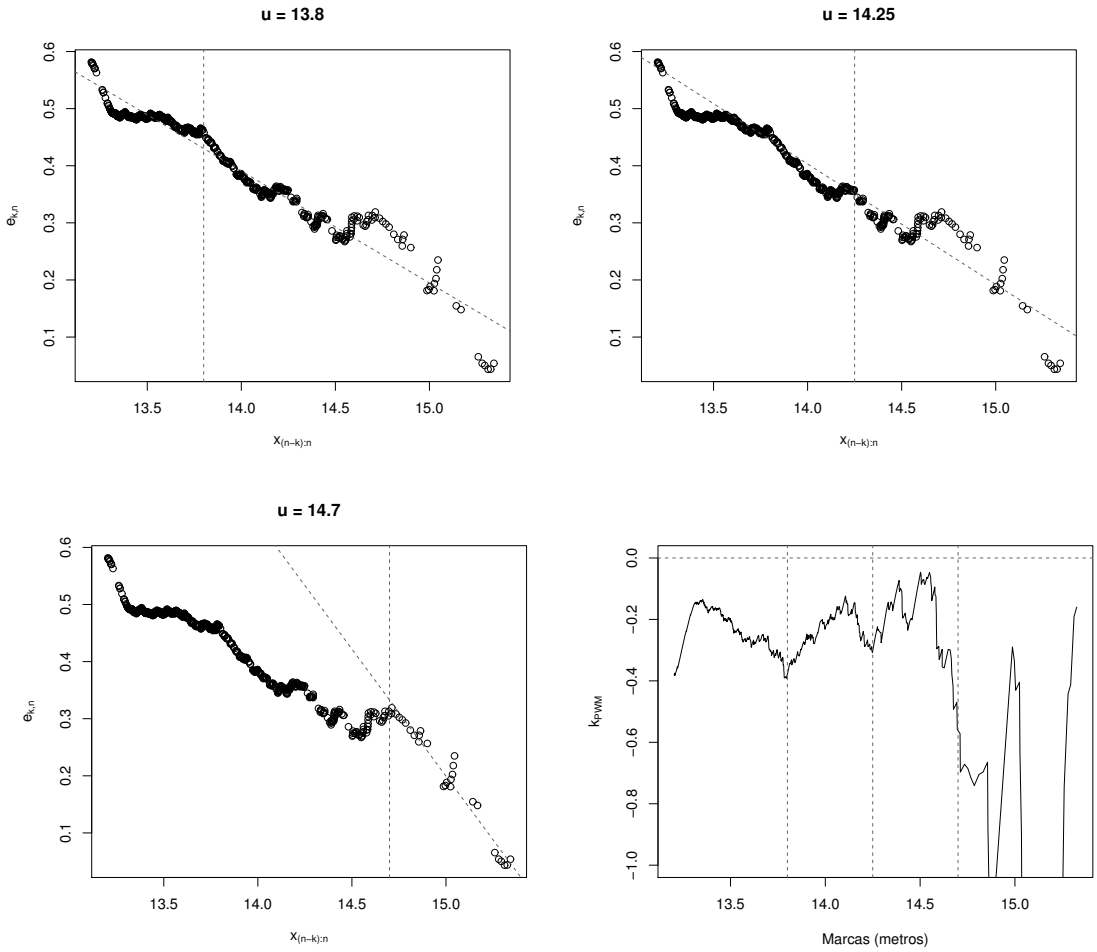

Figura 8.5: Gráficos de Excesso Médio para diversos valores do nível u e gráfico de estimativas PWM para k, como função do nível u considerado

as estimativas obtidas para k são negativas. Para além disto, também é possível notar que as estimativas obtidas pelos métodos MAPE e MCE são idênticas às obtidas pelo método EPM, o que significa que, em ambos os casos, a escolha do método a utilizar foi feita corretamente.

Por outro lado, para $u = 13.80$ e $u = 14.25$, os métodos MLO e ZS produziram estimativas para os parâmetros que sugerem caudas um pouco mais pesadas que as estimadas pelas metodologias anteriores, dado que o valor de k é superior. Para $u = 14.7$, as conclusões mantêm-se para as estimativas ZS, mas as estimativas MLO passam a sugerir caudas ainda mais leves que as estimadas pelo método EPM. No entanto, estas últimas não são confiáveis devido às fracas propriedades do Método de Máxima Verosimilhança quando a dimensão das amostras é reduzida.

u	m	k_{MLO}	k_{EPM}	k_{ZS}	k_{MAPE}	k_{MCE}	σ_{MLO}	σ_{EPM}	σ_{ZS}	σ_{MAPE}	σ_{MCE}
13.80	286	-0.3150	-0.3267	-0.3011	-0.3267	-0.3267	0.6003	0.6191	0.5921	0.6191	0.6191
14.25	113	-0.3259	-0.3338	-0.2896	-0.3338	-0.3338	0.4748	0.4820	0.4582	0.4820	0.4820
14.70	29	-0.9175	-0.5904	-0.5421	-0.5904	-0.5904	0.6418	0.4752	0.4653	0.4752	0.4752

Tabela 8.2: Estimativas de (k,σ) obtidas por cada método, para cada nível u selecionado

Uma vez calculadas as estimativas de (k, σ), é necessário saber se os modelos GPD resultantes se ajustam adequadamente aos dados. Para tal, são apresentados na Tabela 8.3, para cada nível u, os valores da correlação entre os quantis estimados pela GPD ajustada à amostra de excessos e os quantis da

amostra (Cor), bem como os *p-values* resultantes da aplicação dos Testes de Cramér-von Mises (CvM) e Anderson-Darling (AD). Através da análise destes valores, verifica-se que todos os modelos GPD ajustados são adequados aos dados em estudo, independentemente do nível u escolhido. Esta afirmação é sustentada pelos elevados valores obtidos para a correlação entre quantis estimados e quantis amostrais (todos elas próximas ou superiores a 0.99), e pelos valores elevados dos *p-values* obtidos, que permitem concluir claramente que não deve ser rejeitada a hipótese nula de que os modelos GPD indicados se ajustam às amostras de excessos obtidas para cada nível u. Repare-se novamente que, para $u = 14.7$, o modelo obtido pelo método MLO parece não ajustar-se tão bem aos dados, uma vez que o valores obtidos para a correlação e para os *p-values* são inferiores aos demais.

	$u = 13.8$			$u = 14.25$			$u = 14.7$		
	Cor	CvM	AD	Cor	CvM	AD	Cor	CvM	AD
MLO	0.9957	0.2894	0.2363	0.9895	0.3774	0.3459	0.9888	0.5307	0.5893
EPM	0.9953	0.2776	0.2325	0.9892	0.3483	0.3338	0.9900	0.8703	0.9003
ZS	0.9962	0.2608	0.2139	0.9910	0.3986	0.3563	0.9889	0.8929	0.9226
MAPE	0.9953	0.2776	0.2325	0.9892	0.3483	0.3338	0.9900	0.8703	0.9003
MCE	0.9953	0.2776	0.2325	0.9892	0.3483	0.3338	0.9900	0.8703	0.9003

Tabela 8.3: Medidas de validação da adequabilidade dos modelos GPD ajustados, para cada nível u selecionado

Sabendo que todos os modelos GPD estimados se adequam aos excessos de cada nível u, é de interesse comparar alguns dos quantis extremais da amostra de Marcas Máximas com os quantis extremais estimados com base nos modelos anteriores. Na Tabela 8.4 apresentam-se, para cada nível u, as estimativas dos níveis de retorno de m observações obtidas através de cada metodologia, considerando $m \in \{100, 200, 1000\}$. Relembre-se que estimar o nível de retorno de m observações é equivalente a estimar o quantil de probabilidade $1 - 1/m$.

É possível verificar que os quantis estimados tomam valores semelhantes para qualquer nível u e para qualquer valor m considerado, à exceção dos que são obtidos pelo modelo GPD estimado pelo método MLO. Para $u = 13.80$ e $u = 14.25$, as estimativas obtidas são inferiores às amostrais, o que significa que as estimativas das Marcas Máximas que são ultrapassadas, em média, por 1 em cada m atletas femininas não são tão exigentes quanto as observadas a partir da amostra. Para $u = 14.70$, os valores estimados são superiores aos quantis amostrais, permitindo assim concluir que as estimativas das Marcas Máximas que são ultrapassadas, em média, por 1 em cada m atletas femininas são mais exigentes que as observadas a partir da amostra.

	$u = 13.8$			$u = 14.25$			$u = 14.7$		
	$Q_{0.99}$	$Q_{0.995}$	$Q_{0.999}$	$Q_{0.99}$	$Q_{0.995}$	$Q_{0.999}$	$Q_{0.99}$	$Q_{0.995}$	$Q_{0.999}$
MLO	15.0957	15.2153	15.4104	15.1002	15.2229	15.4205	15.1925	15.2899	15.3745
EPM	15.1136	15.2314	15.4211	15.1052	15.2269	15.4211	15.1373	15.2608	15.4105
ZS	15.1045	15.2292	15.4356	15.1057	15.2379	15.4593	15.1404	15.2713	15.4383
MAPE	15.1136	15.2314	15.4211	15.1052	15.2269	15.4211	15.1373	15.2608	15.4105
MCE	15.1136	15.2314	15.4211	15.1052	15.2269	15.4211	15.1373	15.2608	15.4105
Amostra	15.1509	15.2969	–	15.1509	15.2969	–	15.1509	15.2969	–

Tabela 8.4: Estimativas dos níveis de retorno de m observações, para cada nível u selecionado

Outra maneira de avaliar a qualidade do ajustamento dos modelos GPD estimados é através da estimação do período de retorno de um determinado evento. Neste caso, é interessante estimar quantas atletas femininas, nas condições mencionadas inicialmente, deverão ser registadas até que se volte a obter uma Marca Máxima com um determinado valor. No Atletismo, como em diversas outras modali-

dades, dois dos títulos mais referidos são os Recordes Olímpico (OR) e Mundial (WR). No âmbito do Triplo Salto feminino, estes títulos pertencem atualmente a Françoise Mbango Etone (Pequim, 2008), com 15.38 metros, e Inessa Kravetz (Gotemburgo, 1995), com 15.50 metros, respetivamente. Assim, apresentam-se na Tabela 8.5 as estimativas dos períodos de retorno do OR e WR, a partir dos modelos GPD ajustados para cada nível u.

Antes de analisar os resultados obtidos, é relevante mencionar que a marca respeitante ao OR foi obtida em 2008, que é um dos anos considerados na recolha dos dados em estudo. Portanto, sabendo que estão a ser consideradas 769 atletas distintas, podemos deduzir que o OR é alcançado 1 vez por cada 769 atletas registadas nas *Top Lists* anuais. Assim, observando os valores da Tabela 8.5, é possível verificar que, para todos os níveis u considerados, os métodos EPM, MAPE e MCE produziram estimativas mais precisas que os restantes métodos para o período de retorno do OR. Note-se ainda que as estimativas obtidas pelo método ZS são superiores ao período de retorno observado (ou seja, estima-se maior probabilidade de ocorrência do OR), enquanto que as estimativas obtidas pelo método MLO são inferiores (ou seja, estima-se menor probabilidade de ocorrência do OR).

Comparando agora as estimativas obtidas para o período de retorno do WR, é possível verificar novamente uma subestimação do método ZS face aos métodos EPM, MAPE e MCE, bem como uma sobrestimação do método MLO quando comparado com os métodos de referência para este caso. No entanto, repare-se que as estimativas obtidas quando $u = 14.70$ são bastante superiores às obtidas para os outros níveis u.

	$u = 13.8$		$u = 14.25$		$u = 14.7$	
	OR	WR	OR	WR	OR	WR
MLO	733	3151	666	2709	1314	∞
EPM	652	2827	657	2780	622	147688
ZS	597	2052	514	1490	481	3787
MAPE	652	2827	657	2780	622	147688
MCE	652	2827	657	2780	622	147688

Tabela 8.5: Estimativas do período de retorno dos Recordes Olímpico (OR) e Mundial (WR), para cada nível u selecionado

8.2 Montante de Sinistros Automóvel

8.2.1 Generalidades Históricas e Metodologias

A atividade seguradora no ramo Automóvel começou a ser praticada no Reino Unido, em 1898, onde já se emitiam apólices cobrindo a responsabilidade contra terceiros e nas quais também se podiam incluir os riscos de incêndio de veículos mediante um sobreprémio adicional. Mais tarde, estas apólices foram atualizadas de modo a cobrir danos acidentais causados por colisão, e, em 1901, passaram a ser segurados ainda os danos causados por não colisão e os riscos associados a incêndio e roubo [17].

Após alguns anos sem grandes desenvolvimentos no esquema de negócio, as seguradoras ganharam algum dinamismo através da aplicação de técnicas de *marketing*. A oferta de esquemas alternativos de cobertura tem vindo a ser feita como consequência dos aumentos do prémio de responsabilidade civil, mas também tem exposto o público a critérios seletivos para aceitação, como a tendência para recusar contratos singulares em detrimento de pacotes de seguros. Por outro lado, as seguradoras oferecem atualmente coberturas contra todos os riscos de uma forma muito flexível, ao contrário do que se sucedia no passado. Para além de alterações ao nível do esquema de negócio, também têm vindo a ser feitas

atualizações ao nível da diferenciação dos condutores e segurados. Neste sentido, há prémios especiais para condutores com uma conduta exemplar ou para grupos sociais reconhecidos como mais prudentes na condução automóvel.

No que toca a metodologias aplicadas nos setores financeiros, é usual exprimir os montantes em estudo relativamente a um determinado ano de referência. Tal como nas áreas da Economia e Finanças, é habitual atualizar o efeito das taxas de inflação aplicado sobre os montantes em análise, de forma a serem considerados os seus valores reais no lugar dos seus valores nominais.

A atualização do efeito das taxas de inflação aplicadas sobre um determinado bem é feita com recurso a índices anuais, que podem ser calculados relativamente a um determinado ano de interesse [44]. Este ano de interesse é fixado de acordo com o ponto temporal para o qual se pretende calcular o valor de um determinado bem que foi adquirido num ponto temporal anterior. Por definição, assume-se que o valor do índice anual para o ano de interesse é 100. Os restantes índices anuais são calculados recursivamente a partir do índice (I_{i+1}) e da taxa de inflação (t_{i+1}) do ano seguinte, dada como proporção, através da expressão

$$I_i = \frac{I_{i+1}}{1+t_{i+1}}, \quad i = 1,\ldots,N-1, \tag{8.5}$$

onde N representa o número de anos considerados. Uma vez calculados estes valores, é possível obter a taxa de inflação total entre dois anos i e j recorrendo à expressão

$$T_{ij} = \frac{I_j}{I_i} - 1, \quad 1 \leq i \leq j \leq N. \tag{8.6}$$

Por fim, é possível calcular o valor real de um montante registado no ano i (M_i), se este tivesse sido contabilizado no ano j (M_j), através da expressão

$$M_j = (1+T_{ij}) \times M_i, \quad 1 \leq i \leq j \leq N. \tag{8.7}$$

Através da interpretação da expressão (8.7), é possível concluir que, no ano j, seria necessário um montante M_j para adquirir o mesmo bem que, no ano i, tinha sido adquirido por um valor M_i.

8.2.2 Análise Exploratória dos Dados

O conjunto de dados em estudo refere-se a sinistros de danos corporais do ramo Automóvel ocorridos entre 2001 e 2011, com montantes superiores a 10 000 Euros. Este conjunto é composto por 9720 registos relativos a eventos reais, cujos valores foram ajustados a preços de 2011. Para além disto, os dados também serão tratados em escala logarítmica devido à magnitude dos seus valores. Na prática, é usual aplicar esta transformação quando os valores considerados são muito elevados [33, 49].

Começando por analisar os dados agrupados por ano de ocorrência do sinistro, é possível observar através das representações gráficas da Figura 8.6 que o número de registos anuais com montantes superiores a 10 000 Euros varia entre os 800 e 1000. Os anos em que foram registadas menos ocorrências variam entre 2004 e 2009, enquanto que o ano com maior número de sinistros elevados foi 2002. No que toca às indemnizações pagas, pode observar-se que houve um ligeiro decréscimo anual dos montantes medianos pagos ao longo do período de análise. Porém, é possível verificar que os valores dos montantes medianos variam em torno da quantia de 22 000 Euros, aproximadamente. Relativamente à análise de

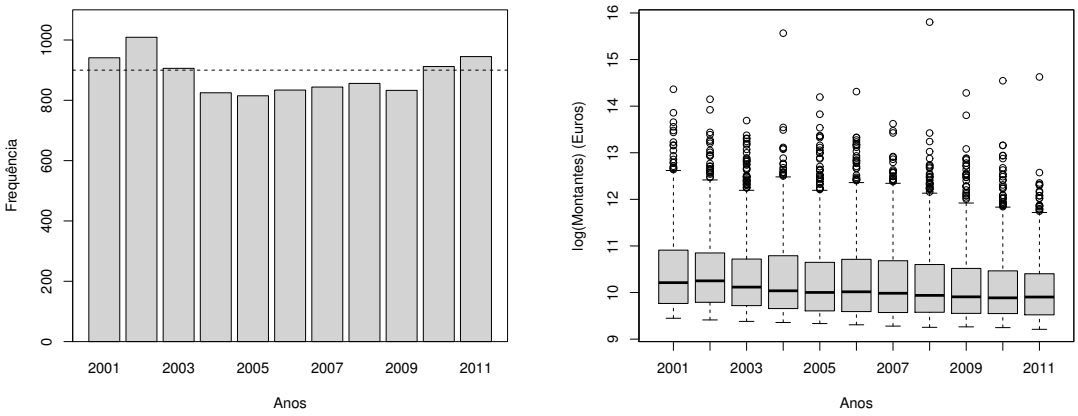

Figura 8.6: Frequências (à esquerda) e *boxplots* (à direita) anuais dos montantes logaritmizados

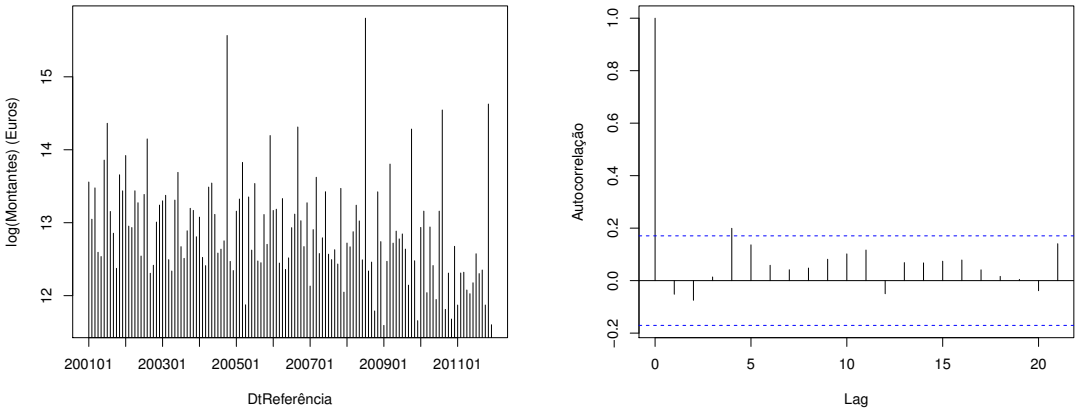

Figura 8.7: Valores máximos mensais dos montantes logaritmizados (à esquerda) e respetivo gráfico de autocorrelação (à direita)

outliers, verifica-se que todos os anos foi registado um número significativo de sinistros com montantes discrepantes dos restantes, tendo sido registados os sinistros mais gravosos em 2004 (5 759 682 Euros) e 2008 (7 301 741 Euros).

Considerando agora os dados agrupados numa base mensal, apresentam-se na Figura 8.7 os montantes máximos registados em cada mês de análise. Numa primeira observação do gráfico à esquerda, é possível colocar em hipótese a existência de sazonalidade dos dados ao nível mensal, devido ao seu aparente formato sinusoidal. No entanto, através do gráfico de autocorrelação associado, conclui-se que este pressuposto não se verifica, uma vez que os seus valores não são estatisticamente significativos. Com efeito, o valor de cada *lag* encontra-se dentro dos limites do intervalo de confiança (representados pelas linhas tracejadas), o que indica que o valor da autocorrelação para cada *lag* não é estatisticamente diferente de 0.

Assuma-se agora que a amostra dos montantes registados não está agrupada segundo qualquer tipo de critério. Os gráficos da Figura 8.8 sugerem que a distribuição subjacente à amostra dos montantes logaritmizados tem caudas pesadas devido à existência de valores demasiado discrepantes dos valores centrais da amostra.

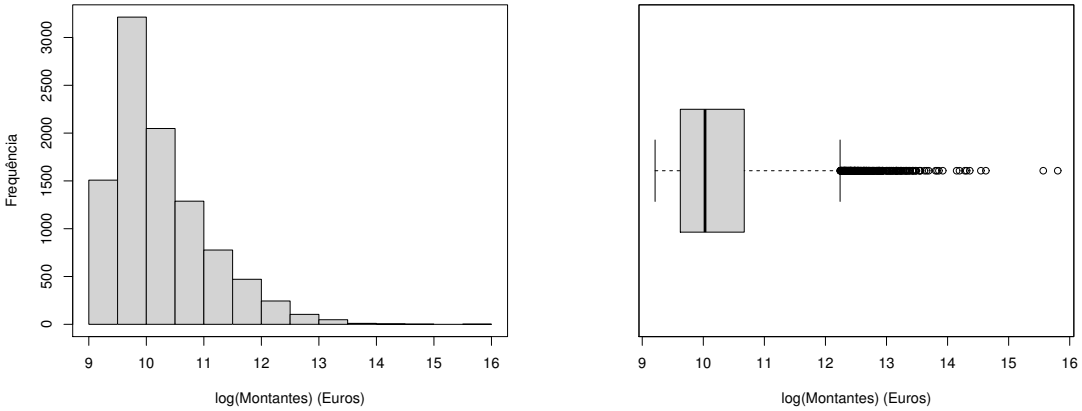

Figura 8.8: Histograma (à esquerda) e *boxplot* (à direita) dos montantes logaritmizados

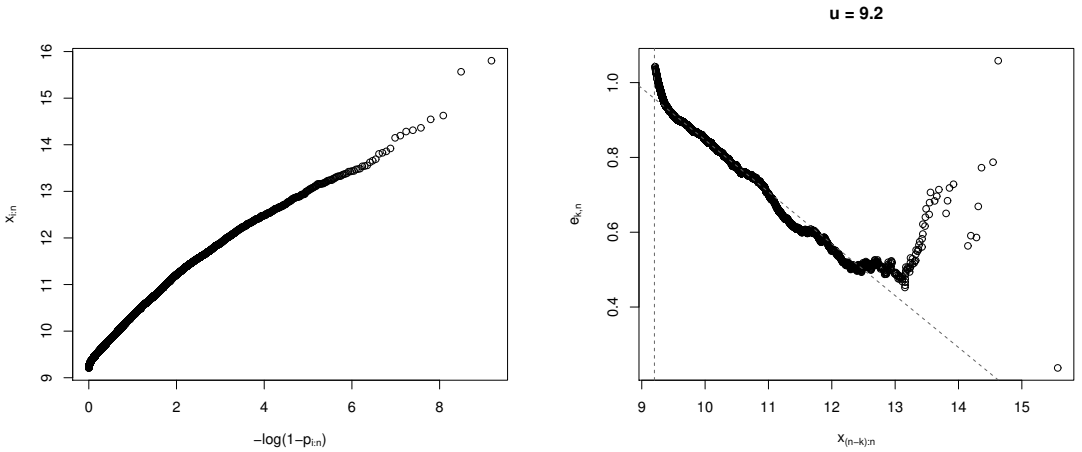

Figura 8.9: QQ-Plot Exponencial (à esquerda) e Gráfico de Excesso Médio (à direita) dos montantes logaritmizados

Para complementar as conclusões obtidas, são apresentadas na Figura 8.9 as representações do QQ-Plot Exponencial e do Gráfico de Excesso Médio obtidas a partir da amostra dos montantes logaritmizados. Através do primeiro gráfico, é possível notar um padrão aproximadamente linear, o que sugere que a distribuição subjacente aos dados originais tem caudas pesadas. Com efeito, quando os quantis da distribuição Exponencial são confrontados num QQ-Plot com valores amostrais logaritmizados, o gráfico resultante é um QQ-Plot Pareto. Esta conclusão pode ser complementada através da análise do segundo gráfico, no qual é possível notar que o padrão do mesmo, para níveis com valor superior a 13.2, é linear, com declive positivo. Isto permite reforçar a conclusão de que a cauda da distribuição subjacente aos dados é mais pesada que a da distribuição Exponencial.

De acordo com as metodologias práticas utilizadas para analisar este tipo de dados, é usual considerar para o nível u valores localizados entre os quantis de probabilidade 0.98 e 0.99. No caso da amostra em estudo, estas estatísticas correspondem aos valores 251 555 e 359 501.90 que, em escala logarítmica, se traduzem em valores próximos de 12.43 e 12.79, respetivamente. É possível verificar a partir da Figura 8.10 que estes níveis não são os mais corretos para se considerar neste estudo, uma vez que se encontram numa zona de estabilidade do Gráfico de Excesso Médio, o que leva crer que o peso de cauda da

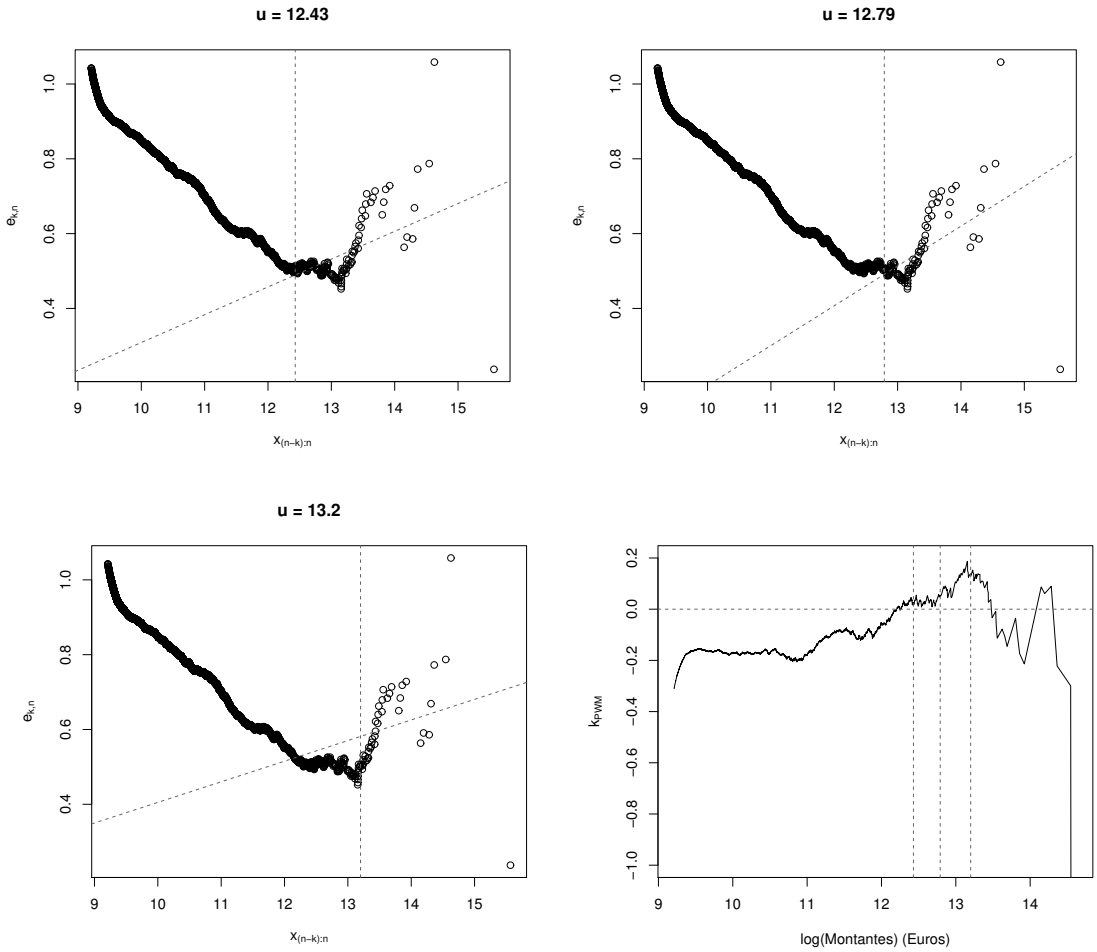

Figura 8.10: Gráficos de Excesso Médio para diversos valores do nível u e gráfico de estimativas PWM para k, como função do nível u considerado

distribuição subjacente aos dados logaritmizados é igual ao da distribuição Exponencial. Por outro lado, ao considerar o nível $u = 13.2$, são contemplados apenas os excessos médios que produzem o padrão linear descrito atrás, permitindo assim continuar a tirar conclusões consistentes com as já retiradas relativamente ao peso de cauda. Para além disto, é ainda possível verificar pelo gráfico de estimativas PWM para k que o nível $u = 13.2$ permite obter um valor estimado coerente com os argumentos apresentados, enquanto que as estimativas produzidas para $u = 12.43$ e $u = 12.79$ induzem a ideia de Exponencialidade das caudas, tal como referido atrás.

8.2.3 Estimação Paramétrica e Inferência

Após a discussão dos impactos causados pela escolha dos níveis u referidos, proceda-se ao ajustamento da GPD sobre os excessos acima de cada um desses valores. Na Tabela 8.6 são mostradas as estimativas de (k, σ) obtidas por cada método mencionado e para cada nível u considerado, bem como o número de excessos (m) desse nível.

Num primeira análise dos resultados obtidos, é validada a conclusão acerca da escolha errada dos níveis $u = 12.43$ e $u = 12.79$, uma vez que todas as estimativas obtidas para k não são suficientemente elevadas para modelar o peso de cauda da distribuição subjacente aos dados. Para além disto, também é

possível verificar que as estimativas obtidas pelos métodos MAPE e MCE são idênticas às obtidas pelo método ZS, o que significa que, em ambos os casos, a escolha do método a utilizar foi feita corretamente.

Focando a análise nas estimativas obtidas para $u = 13.2$, é possível verificar que o método ML produz a estimativa mais baixa para k, enquanto que o método EPM estima o mesmo parâmetro com o valor mais elevado. Os métodos ZS, MAPE e MCE estimam o parâmetro k com um valor inferior ao atribuído pelo método EPM, apesar das estimativas tomarem valores muito próximos.

u	m	k_{MLO}	k_{EPM}	k_{ZS}	k_{MAPE}	k_{MCE}	σ_{MLO}	σ_{EPM}	σ_{ZS}	σ_{MAPE}	σ_{MCE}
12.43	197	0.0263	0.0924	0.0330	0.0330	0.0330	0.4966	0.4948	0.4942	0.4942	0.4942
12.79	99	0.0535	0.1443	0.0813	0.0813	0.0813	0.4744	0.4609	0.4615	0.4615	0.4615
13.20	43	0.2124	0.2939	0.2746	0.2746	0.2746	0.3963	0.3749	0.3728	0.3728	0.3728

Tabela 8.6: Estimativas de (k,σ) obtidas por cada método, para cada nível u selecionado

Após o cálculo das estimativas de (k, σ), é necessário verificar se os modelos GPD resultantes se ajustam adequadamente aos dados. Para este efeito, são apresentados na Tabela 8.7, para cada nível u, os valores da correlação entre os quantis estimados e os quantis da amostra (Cor), bem como os *p-values* resultantes da aplicação dos Testes de Cramér-von Mises (CvM) e Anderson-Darling (AD). A análise destes valores permite concluir que todos os modelos GPD ajustados são adequados aos dados em estudo, independentemente do nível u escolhido. Esta afirmação é sustentada pelos elevados valores obtidos para a correlação entre quantis estimados e quantis amostrais (todos eles próximos ou superiores a 0.99), e pelos valores elevados dos *p-values* obtidos, que permitem concluir claramente que não deve ser rejeitada a hipótese nula de que os modelos ajustados são adequados para os excessos calculados. Em particular, no caso em que $u = 13.2$, o valor da correlação e dos *p-values* sobressaem ainda mais do que os valores homólogos para $u = 12.43$ e $u = 12.79$.

	$u = 12.43$			$u = 12.79$			$u = 13.20$		
	Cor	CvM	AD	Cor	CvM	AD	Cor	CvM	AD
MLO	0.9910	0.9501	0.9193	0.9861	0.7942	0.8558	0.9902	0.9799	0.9909
EPM	0.9942	0.8851	0.7624	0.9920	0.6727	0.7255	0.9904	0.9934	0.9961
ZS	0.9914	0.9405	0.9087	0.9883	0.7320	0.8125	0.9906	0.9936	0.9954
MAPE	0.9914	0.9405	0.9087	0.9883	0.7320	0.8125	0.9906	0.9936	0.9954
MCE	0.9914	0.9405	0.9087	0.9883	0.7320	0.8125	0.9906	0.9936	0.9954

Tabela 8.7: Medidas de validação da adequabilidade dos modelos GPD ajustados, para cada nível u selecionado

Uma vez que todos os modelos GPD estimados se adequam aos excessos de cada nível u, é de interesse comparar alguns dos quantis extremais da amostra de montantes logaritmizados com os quantis extremais estimados com base nos modelos anteriores, de modo a averiguar a qualidade do ajustamento. Na Tabela 8.8 apresentam-se, para cada nível u, as estimativas dos níveis de retorno de m observações obtidas através de cada metodologia, considerando $m \in \{100, 200, 1000\}$. Novamente, relembre-se que estimar o nível de retorno de m observações é equivalente a estimar o quantil de probabilidade $1 - 1/m$.

É possível verificar que os quantis estimados tomam valores semelhantes para qualquer nível u e para qualquer valor m considerado, exceto no caso em que $u = 12.43$ e $u = 12.79$, onde as estimativas produzidas pelo método EPM diferem das restantes. Nos casos em que $m = 100$ e $m = 200$, as estimativas obtidas são inferiores às amostrais, o que significa que as estimativas dos montantes registados que são ultrapassados, em média, por 1 em cada m registos não são tão exigentes quanto os observados a partir da amostra. Para $u = 13.2$, não foi possível obter os níveis de retorno para $m = 100$ observações, pois o valor de m não era suficiente elevado para que a estimativa x_m fosse superior ao nível u fixado.

	$u = 12.43$			$u = 12.79$			$u = 13.20$		
	$Q_{0.99}$	$Q_{0.995}$	$Q_{0.999}$	$Q_{0.99}$	$Q_{0.995}$	$Q_{0.999}$	$Q_{0.99}$	$Q_{0.995}$	$Q_{0.999}$
MLO	12.7841	13.1379	13.9848	12.7987	13.1340	13.9622	–	–	13.8930
EPM	12.7912	13.1692	14.1463	12.7985	13.1353	14.0606	–	–	13.8992
ZS	12.7832	13.1378	13.9931	12.7985	13.1281	13.9689	–	–	13.8847
MAPE	12.7832	13.1378	13.9931	12.7985	13.1281	13.9689	–	–	13.8847
MCE	12.7832	13.1378	13.9931	12.7985	13.1281	13.9689	–	–	13.8847
Amostra	12.7925	13.1587	–	12.7925	13.1587	–	12.7925	13.1587	–

Tabela 8.8: Estimativas dos níveis de retorno de m observações, para cada nível u selecionado

Por fim, considere-se a avaliação da qualidade do ajustamento dos modelos GPD estimados através da análise dos períodos de retorno estimados. Na área de Seguros, um dos temas que tem de ser considerado é o número de sinistros que deverá ser registado até que se observe um novo sinistro com um determinado valor (habitualmente, elevado). Neste estudo são consideradas as estimativas dos períodos de retorno dos quantis de probabilidade 0.9995 (14.32) e 0.9999 (15.57), bem como do valor máximo da amostra (15.8), conforme se pode observar na Tabela 8.9. Na amostra não logaritmizada, estes valores correspondem, respetivamente, a 1654637, 5798205 e 7301741. Usando o período de retorno do montante logaritmizado máximo como referência, deduz-se pela definição que esse valor ocorre uma vez em cada 9720 sinistros registados. Observando os valores da tabela, é possível verificar que, apesar do Método EPM não ser a primeira escolha nos casos em que a distribuição subjacente à amostra tem caudas pesadas, as suas estimativas permitem estimar um valor do período de retorno mais próximo do observado do que os restantes métodos.

	$u = 12.43$			$u = 12.79$			$u = 13.20$		
	$Q_{0.9995}$	$Q_{0.9999}$	Max	$Q_{0.9995}$	$Q_{0.9999}$	Max	$Q_{0.9995}$	$Q_{0.9999}$	Max
MLO	1853	17223	25593	1921	16151	23298	2065	10764	13812
EPM	1299	7312	9769	1477	7553	9819	1927	8074	9946
ZS	1807	15886	23307	1846	13283	18490	2018	8971	11161
MAPE	1807	15886	23307	1846	13283	18490	2018	8971	11161
MCE	1807	15886	23307	1846	13283	18490	2018	8971	11161

Tabela 8.9: Estimativas do período de retorno do montante logaritmizado máximo e dos respetivos quantis de probabilidade 0.9995 e 0.9999, para cada nível u selecionado

Capítulo 9

Conclusão

9.1 Sumário e Conclusões

Neste trabalho foi explorado o tema da estimação dos parâmetros da distribuição de Pareto Generalizada, com particular ênfase sobre as metodologias clássicas que são utilizadas para este efeito, de modo a analisar os pontos positivos e negativos de cada uma dessas técnicas. Também foram propostas novas abordagens a este tema, baseadas na otimização e combinação de técnicas de estimação já existentes. Neste sentido, recorreu-se ao ajustamento polinomial de representações gráficas usuais no estudo de Valores Extremos, bem como à utilização de regras de classificação para determinação do peso de cauda da distribuição subjacente a uma amostra.

A partir de estudos de simulação, foi possível concluir que as ferramentas utilizadas para otimização ou combinação das técnicas já existentes foi eficiente. Isto deve-se ao facto dos métodos híbridos propostos conseguirem decidir corretamente, na maioria dos testes, quais das técnicas iniciais a utilizar consoante o peso de cauda da distribuição subjacente. Também foi possível garantir que as estimativas obtidas são sempre consistentes com os dados, dado que as técnicas subjacentes partilham destas propriedades.

Os casos de estudo considerados permitiram confirmar as propriedades acima descritas num contexto de aplicação a dados reais. O estudo das Melhores Marcas Femininas em Triplo Salto confirmou a utilização de técnicas para estimação associadas ao caso de caudas leves, dado que se tratam de dados relativos a provas de esforço físco. No entanto, verificou-se no estudo dos Montantes de Sinistros Automóvel que, apesar de ser um fenómeno habitualmente caraterizado pela existência de caudas pesadas, nem sempre é sensato considerar técnicas de estimação coerentes com este facto.

9.2 Desenvolvimentos Futuros

Conforme verificado neste trabalho, a hibridização de metodologias pode tornar-se bastante eficiente quando as técnicas base já produzem bons resultados. Neste sentido, o objetivo principal para desenvolvimentos futuros passa pelo levantamento de metodologias utilizadas nas mais diversas áreas científicas (Biologia, Estatística, Investigação Operacional, *Machine Learning*, entre outras) e a sua combinação com as técnicas já existentes no âmbito da estimação dos parâmetros da GPD.

9.2.1 Meta-heurísticas

Dentro deste tema, um dos tópicos a abordar é a combinação das técnicas clássicas para estimação dos parâmetros da GPD com meta-heurísticas utilizadas frequentemente para resolução de problemas de otimização complexos. Duas das abordagens que se pretende considerar são o *Simulated Annealing* e os Algoritmos Genéticos.

Estes algoritmos baseiam-se na mimetização de fenómenos naturais (geológicos e biológicos, respetivamente) e replicação desses comportamentos enquanto algoritmos de otimização. Uma vez que se tratam de técnicas iterativas, é necessário fornecer estimativas iniciais para que estes algoritmos sejam executados. Estes valores podem ser obtidos através de algoritmos para estimação dos parâmetros da GPD que tenham um desempenho global menos eficiente só por si (como os métodos PWM e MOM), mas que permitam o cálculo de uma estimativa inicial com um valor satisfatório.

9.2.2 Metodologias Bayesianas

Outro dos tópicos a desenvolver passa pela revisão e melhoramento de metodologias bayesianas já existentes para este efeito, que utilizam Métodos de Monte Carlo via Cadeias de Markov [8, 51], bem como a exploração de outras técnicas desta área de estudo que possam ser adaptadas ao âmbito da estimação dos parâmetros da GPD [3, 6, 46].

Estas metodologias focam-se na quantificação da certeza ou incerteza acerca de um certo fenómeno através de probabilidades. Considera-se que os parâmetros do modelo subjacente ao fenómeno são variáveis aleatórias e que toda a inferência é feita a partir da distribuição de probabilidade destas. Antes de observar o fenómeno, as características distribucionais dos parâmetros do modelo, *a priori*, são definidas a partir de resultados previamente obtidos acerca do fenómeno, ou mesmo através de experiências anteriores. Após observar o fenómeno, a amostra resultante é utilizada para atualizar os parâmetros, o que permite a realização de inferência (designada por inferência *a posteriori*) com resultados mais fidedignos e coerentes com os dados disponíveis no momento. Assim, uma das vantagens deste tipo de metodologias é a capacidade dos algoritmos adaptarem-se continuamente a diversos conjuntos de dados observados.

No contexto da estimação dos parâmetros da GPD, é mais vantajoso utilizar distribuições de probabilidade conjuntas *a priori* para (k, σ), ou mesmo distribuições condicionais, uma vez que os dois parâmetros são dependentes entre si e o seu valor é condicionado pelo máximo da amostra considerada para estimação. Como candidatas, podem ser consideradas distribuições já utilizadas em outros estudos [51], ou até misturas destas [46], de modo a que não seja necessário efetuar a estimação dos parâmetros em duas fases, isto é, sob a suposição de que a distribuição subjacente aos dados tenha caudas leves ou pesadas. Para além disto, a distribuição do parâmetro de escala (σ) também pode ser condicionada ao valor do parâmetro de forma (k), no sentido de haver consistência entre as estimativas e os dados em casos de caudas leves.

Anexo A

Método de Máxima Verosimilhança (Código Matlab)

```
%%=========================================================================
%% - MÉTODO DE MÁXIMA VEROSIMILHANÇA -
%% Método centrado na utilização do Método de Programação Quadrática para
%% obtenção das estimativas de máxima verosimilhança.
%%=========================================================================
function out = ML(sample)

    %-------------------------------------------------------------------------
    % FUNÇÃO OBJECTIVO (MINIMIZAR)
    % Simétrico da função de log-verosimilhança
    %-------------------------------------------------------------------------
    f = @(x) gplike([x(1),x(1)*x(2)], sample);
    %-------------------------------------------------------------------------
    % RESTRIÇÕES DO PROBLEMA
    % Limites inferior e superior de k e theta
    %-------------------------------------------------------------------------
    delta = 1e-06;
    LTI = [-1+delta, -Inf]'; LTS = [-delta, -max(sample)-delta]';
    HTI = [delta, delta]'; HTS = [Inf, Inf]';
    %-------------------------------------------------------------------------
    % CONFIGURAÇÕES DE OPTIMIZAÇÃO
    % Escolha do algoritmo (Método SQP - Sequential Quadratic Programming)
    %-------------------------------------------------------------------------
    opt = optimoptions('fmincon','Algorithm','sqp','Display','off');
    %-------------------------------------------------------------------------
    % ESCOLHA DAS SOLUÇÕES INICIAIS
    %-------------------------------------------------------------------------
    LT0 = [-0.5, -max(sample)-1]';
    HT0 = [1, 1]';
```

```
    %-------------------------------------------------------------------------
    % OPTIMIZAÇÃO DOS PROBLEMAS
    %-------------------------------------------------------------------------
    [xLT,zLT] = fmincon(f,LT0,[],[],[],[],LTI,LTS,[],opt);
    [xHT,zHT] = fmincon(f,HT0,[],[],[],[],HTI,HTS,[],opt);
    %-------------------------------------------------------------------------
    % OBTENÇÃO DAS ESTIMATIVAS FINAIS
    %-------------------------------------------------------------------------
    if zHT <= zLT
        out = [xHT(1), xHT(1)*xHT(2)]';
    else
        out = [xLT(1), xLT(1)*xLT(2)]';
    end

end

%%=========================================================================
%% - MÉTODO DE MÁXIMA VEROSIMILHANÇA -
%% Análise de desempenho do método, através da avaliação do viés e da raiz
%% quadrada do erro quadrático médio das estimativas.
%%=========================================================================
function out = performML(n,k,rep)

rng(str2num([num2str(n),num2str(abs(k)*100),num2str(k>=0)]))

sigma = 1;
par = arrayfun(@(x) ML(gprnd(k,sigma,0,[1,n])), 1:rep, 'UniformOutput', 0);
par = cell2mat(par);
kML = par(1,:);
sML = par(2,:);

theta = [mean(kML), mean(sML)];
bias = [mean(kML)-k, mean(sML)-sigma];
rmse = sqrt([mean((kML-k).^2), mean((sML-sigma).^2)]);
out = [1, k, n, theta, bias, rmse];

end
```

Anexo B

Método dos Percentis Elementares (Código Matlab)

```
%%=========================================================================
%% - MÉTODO DOS PERCENTIS ELEMENTARES -
%% Método baseado na utilização de percentis da amostra para obtenção das
%% estimativas.
%%=========================================================================
function out = EPM(sample)

    y = sort(sample);
    n = length(y);
    lim = n-1;
    %-------------------------------------------------------------------------
    % Cálculo das estimativas de (k,sigma) para cada par distinto
    % x(i:n) < x(j:n), onde j = 1,2,...,n-1
    %-------------------------------------------------------------------------
    p = (1:n)/(n+1); C = log(1-p);
    d = C(n)*y(1:lim) - C(1:lim)*y(n);
    d0 = (y(1:lim)*y(n).*(C(n)-C(1:lim)))./d;
    ests = arrayfun(@(i) estsPairs(i,y,C,d0), 1:lim, 'UniformOutput', 0);
    %-------------------------------------------------------------------------
    % Cálculo das estimativas finais recorrendo à mediana das estimativas
    % obtidas para cada par distinto x(i:n) < x(j:n), onde j = 1,2,...,n-1
    %-------------------------------------------------------------------------
    ests = cell2mat(ests);
    k_EPM = ests(1,:);
    sigma_EPM = ests(2,:);
    out = [-median(k_EPM), median(sigma_EPM)]';

end
```

```
function out = estsPairs(i,y,C,d0)

    n = length(y);
    f = @(x,i) (C(i)*log(1-y(n)/x) - C(n)*log(1-y(i)/x))^2;
    findZero = @(L,U,i) fminbnd(@(x)f(x,i),L,U);

    if (d0(i)>0)
        raiz = findZero(y(n)-eps,d0(i),i);
    else
        raiz = findZero(d0(i),-eps,i);
    end

    if (y(i)~=raiz)
        k = log(1-y(i)/raiz)/C(i); sigma = k*raiz;
    else
        k=0; sigma=0;
    end

    out = [k,sigma]';

end

%%=========================================================================
%% - MÉTODO DOS PERCENTIS ELEMENTARES -
%% Análise de desempenho do método, através da avaliação do viés e da raiz
%% quadrada do erro quadrático médio das estimativas.
%%=========================================================================
function out = performEPM(n,k,rep)

rng(str2num([num2str(n),num2str(abs(k)*100),num2str(k>=0)]))

sigma = 1;
par = arrayfun(@(x) EPM(gprnd(k,sigma,0,[1,n])), 1:rep, 'UniformOutput', 0);
par = cell2mat(par);
kEPM = par(1,:);
sEPM = par(2,:);

theta = [mean(kEPM), mean(sEPM)];
bias = [mean(kEPM)-k, mean(sEPM)-sigma];
rmse = sqrt([mean((kEPM-k).^2), mean((sEPM-sigma).^2)]);
out = [2, k, n, theta, bias, rmse];

end
```

Anexo C

Likelihood Moment Estimators (Código Matlab)

```
%%=========================================================================
%% - LIKELIHOOD MOMENT ESTIMATORS -
%% Método centrado na utilização do Método de Programação Quadrática para
%% obtenção das estimativas de máxima verosimilhança.
%%=========================================================================
function out = ZS(sample)

    x = sort(sample); n = length(x);
    p = (3:9)/10; xp = x(round(n*(1-p)+0.5));
    m = 20+round(sqrt(n)); xq = x(round(n*(1-p.*p)+0.5));
    k = log((xq/xp)-1)./log(p); a = k.*xp./(1-(p.^k));
    v = -xp./log(p); a(k==0) = v(k==0);

    k = -1; sigma = 1/(2*median(a));
    b = (n-1)/(x(n)*(n+1)) - (sigma/k)*(1-(((1:m)-0.5)/m).^k);
    L = arrayfun(@(i) n*lx(b(i),x), 1:m);
    w = arrayfun(@(i) 1/sum(exp(L-L(i))), 1:m);
    b = sum(b.*w); k = -mean(log(1-b*x));
    out = [-k; k/b];

end

function out = lx(b, x)

    k = -mean(log(1-b*x));

    if b==0
        out = k-1-log(mean(x));
    else
```

```
        out = k-1+log(b/k);
    end

end

%%=========================================================================
%% - LIKELIHOOD MOMENT ESTIMATORS -
%% Análise de desempenho do método, através da avaliação do viés e da raiz
%% quadrada do erro quadrático médio das estimativas.
%%=========================================================================
function out = performZS(n,k,rep)

rng(str2num([num2str(n),num2str(abs(k)*100),num2str(k>=0)]))

sigma = 1;
par = arrayfun(@(x) ZS(gprnd(k,sigma,0,[1,n])), 1:rep, 'UniformOutput', 0);
par = cell2mat(par);
kZS = par(1,:);
sZS = par(2,:);

theta = [mean(kZS), mean(sZS)];
bias = [mean(kZS)-k, mean(sZS)-sigma];
rmse = sqrt([mean((kZS-k).^2), mean((sZS-sigma).^2)]);
out = [3, k, n, theta, bias, rmse];

end
```

Anexo D

Método de Ajustamento Polinomial para Estimação (Código Matlab)

```
%%=========================================================================
%% - MÉTODO DE AJUSTAMENTO POLINOMIAL PARA ESTIMAÇÃO -
%% Método baseado na forma do QQ-Plot Exponencial utilizado para análise
%% preliminar do peso da cauda da distribuição subjacente a uma amostra.
%%=========================================================================
function out = MAPE(sample)

    %-------------------------------------------------------------------
    % Ajustamento de polinómio de 2o grau ao QQ-Plot Exponencial
    %-------------------------------------------------------------------
    n = length(sample); x = expinv((1:n)/(n+1),1)';
    X = [x.^2 x ones(n,1)]; y = sort(sample)';
    b = inv(X'*X)*(X'*y);
    %-------------------------------------------------------------------
    % Decisão acerca do método a aplicar
    %-------------------------------------------------------------------
    if b(1) >= 0
        out = ZS(sample);
    else
        out = EPM(sample);
    end

end

%%=========================================================================
%% - MÉTODO DE AJUSTAMENTO POLINOMIAL PARA ESTIMAÇÃO -
%% Análise de desempenho do método, através da avaliação do viés e da raiz
%% quadrada do erro quadrático médio das estimativas.
%%=========================================================================
```

```
function out = performMAPE(n,k,rep)

rng(str2num([num2str(n),num2str(abs(k)*100),num2str(k>=0)]))

sigma = 1;
par = arrayfun(@(x) MAPE(gprnd(k,sigma,0,[1,n])), 1:rep, 'UniformOutput', 0);
par = cell2mat(par);
kMAPE = par(1,:);
sMAPE = par(2,:);

theta = [mean(kMAPE), mean(sMAPE)];
bias = [mean(kMAPE)-k, mean(sMAPE)-sigma];
rmse = sqrt([mean((kMAPE-k).^2), mean((sMAPE-sigma).^2)]);

out = [4, k, n, theta, bias, rmse];

end
```

Anexo E

Método de Classificação para Estimação (Código Matlab)

```
%%=========================================================================
%% - MÉTODO DE CLASSIFICAÇÃO PARA ESTIMAÇÃO -
%% Estimação dos parâmetros do perceptrão com o auxílio do Pocket Algorithm
%% for oo Training Data.
%%=========================================================================
function W = pocketAlgorithm(N)

    %-------------------------------------------------------------------------
    % Nome do ficheiro do conjunto de treino
    %-------------------------------------------------------------------------
    file = ['trainingSet_N', num2str(N), '.txt'];
    %-------------------------------------------------------------------------
    % No caso de já existir um ficheiro com o nome indicado, extrair
    % estimativas dos parâmetros
    %-------------------------------------------------------------------------
    if exist(file, 'file') == 2
        tmp = importdata(file);
        W = tmp(end, 1:(end-1));
        return
    end
    %-------------------------------------------------------------------------
    % No caso de ainda não existir um ficheiro com o nome indicado, criar
    % observações de treino e estimar parâmetros
    %-------------------------------------------------------------------------
    % Pesos (temporários) do perceptrão
    p = zeros(1,4);
    % Número de classificações correctas consecutivas utilizando p
    runp = 0;
    % Pesos (assumidos óptimos) do perceptrão
```

```
    W = p;
    % Número de classificações correctas consecutivas utilizando W
    runW = 0;
    % Algoritmo
    for i = 1:N
        % Obter observação de treino e registar no ficheiro
        [x,t] = getTrainObs(N,i);
        x = [1,x];
        dlmwrite(file, [x,t], '-append')
        % Atualização dos parâmetros do perceptrão
        if (p*x'>=0 && t==1) || (p*x'<0 && t==-1)
            runp = runp + 1;
            if runp > runW
                W = p;
                runW = runp;
            end
        else
            p = p + x*t;
            runW = 0;
        end
    end
    %-------------------------------------------------------------------------
    % Registar estimativas dos parâmetros no ficheiro
    %-------------------------------------------------------------------------
    dlmwrite(file, [W,runW], '-append')

end

function [x,t] = getTrainObs(N,i)

    rng(N+i)

    k = unifrnd(-2.1,2.1,1);
    n = datasample(10:200,1);
    sample = gprnd(k,1,0,[1,n]);
    ests = [EPM(sample);ZS(sample)];
    ests([2,4]) = [];

    x = [n, ests'];
    t = (k>=0) - (k<0);

end
```

```
%%=========================================================================
%% - MÉTODO DE CLASSIFICAÇÃO PARA ESTIMAÇÃO -
%% Determinação do método de estimação dos parâmetros da GPD a utilizar,
%% com recurso às estimativas dos parâmetros do perceptrão já calculadas
%%=========================================================================
function out = MCE(sample, w)

    %-------------------------------------------------------------------
    % CONSTRUÇÃO DA OBSERVAÇÃO A CLASSIFICAR
    %-------------------------------------------------------------------
    n = length(sample);
    kEPM = EPM(sample); kEPM1 = kEPM(1);
    kZS = ZS(sample); kZS1 = kZS(1);
    obs = [1, n, kEPM1, kZS1]';
    %-------------------------------------------------------------------
    % UTILIZAÇÃO DOS PARÂMETROS DO PERCEPTRÃO PARA CLASSIFICAR A OBSERVAÇÃO
    %-------------------------------------------------------------------
    if w*obs >= 0
        out = kZS;
    else
        out = kEPM;
    end

end

%%=========================================================================
%% - MÉTODO DE CLASSIFICAÇÃO PARA ESTIMAÇÃO -
%% Análise de desempenho do método, através da avaliação do viés e da raiz
%% quadrada do erro quadrático médio das estimativas.
%%=========================================================================
function out = performMCE(n,k,rep)

w = pocketAlgorithm(1e05);

rng(str2num([num2str(n),num2str(abs(k)*100),num2str(k>=0)]))

sigma = 1;
par = arrayfun(@(x) MCE(gprnd(k,sigma,0,[1,n]), w), 1:rep, 'UniformOutput', 0);
par = cell2mat(par);
kMCE = par(1,:);
sMCE = par(2,:);

theta = [mean(kMCE), mean(sMCE)];
```

```
bias = [mean(kMCE)-k, mean(sMCE)-sigma];
rmse = sqrt([mean((kMCE-k).^2), mean((sMCE-sigma).^2)]);

out = [5, k, n, theta, bias, rmse];

end
```

Anexo F

Avaliação dos Métodos por Simulação (Código Matlab)

```
function [] = performGPD(method, tailweight)

    if strcmp(tailweight,'LT') == 1
        kValues = [-2,-1.5,-1,-0.75,-0.5,-0.25,0];
    elseif strcmp(tailweight,'HT') == 1
        kValues = [0.25,0.5,0.75,1,1.5,2];
    else
        kValues = [];
    end

    for k = kValues
        for n = [15,25,50,100,200]
            if strcmp(method,'ML') == 1
                performML(n,k,10000)
            elseif strcmp(method,'EPM') == 1
                performEPM(n,k,10000)
            elseif strcmp(method,'ZS') == 1
                performZS(n,k,10000)
            elseif strcmp(method,'MAPE') == 1
                performMAPE(n,k,10000)
            elseif strcmp(method,'MCE') == 1
                performMCE(n,k,10000)
            end
        end
    end

end
```

Anexo G

Melhores Marcas Femininas em Triplo Salto (Código R)

```
#=============================================================================
#================  MELHORES MARCAS FEMININAS EM TRIPLO SALTO  ================
#=============================================================================
dados = read.table("IAAF_TripleJumpF_Input.txt", header=TRUE)

#=============================================================================
# (1) ANÁLISE PRELIMINAR ANUAL
#=============================================================================
#-----------------------------------------------------------------------------
# --> Nova estrutura para guardar os dados (inclui coluna para os anos)
#-----------------------------------------------------------------------------
dadosLista = data.frame()
for(j in 2:16)
{
    dadosAno = cbind(2000+j, dados[dados[,j]!=0, c(j,1)])
    colnames(dadosAno) = c("Ano", "Marca", "Atleta")
    dadosLista = rbind(dadosLista, dadosAno)
}
#-----------------------------------------------------------------------------
# --> Frequência anual de atletas na Top List (Gráfico de Barras)
#-----------------------------------------------------------------------------
barplot(table(dadosLista[,"Ano"]), ylim=c(0,200), col="lightgrey")
title(xlab="Anos", ylab="Frequência")
abline(h=150, lty=2)
box()
#-----------------------------------------------------------------------------
# --> Variabilidade anual das marcas das atletas na Top List (Boxplots)
#-----------------------------------------------------------------------------
boxplot(dadosLista[,"Marca"] ~ dadosLista[,"Ano"], col="lightgrey")
```

```
title(xlab="Anos", ylab="Marcas (metros)")
abline(h=13.2, lty=2)
box()
#-------------------------------------------------------------------------------
# --> Melhores marcas anuais
#-------------------------------------------------------------------------------
melhoresAnuais = data.frame()
for(j in 2:16)
{
    m = max(dados[,j])
    melhorAno = cbind(2000+j, dados[dados[,j]==m,c(j,1)])
    colnames(melhorAno) = c("Ano", "Marca", "Atleta")
    melhoresAnuais = rbind(melhoresAnuais, melhorAno)
}

#===============================================================================
# (2) ANÁLISE PRELIMINAR GLOBAL
#===============================================================================
#-------------------------------------------------------------------------------
# --> Determinação da melhor marca de cada atleta + "shake" dos valores
#-------------------------------------------------------------------------------
saltos = apply(dados[,-1],1,max)
set.seed(2016)
rnd = runif(length(saltos), 0.00001, 0.00999)
saltos = sort(saltos + rnd)
write(saltos, file="IAAF_TripleJumpF.txt", ncolumns=1)
#-------------------------------------------------------------------------------
# --> Frequência de marcas por classe (Histograma)
#-------------------------------------------------------------------------------
hist(saltos, freq=T, main="", xlab="", ylab="", col="lightgrey")
title(xlab="Marcas (metros)", ylab="Frequência")
box()
#-------------------------------------------------------------------------------
# --> Variabilidade global das marcas (Boxplot)
#-------------------------------------------------------------------------------
boxplot(saltos, horizontal=TRUE, main="", xlab="", col="lightgrey")
title(xlab="Marcas (metros)")
box()
#-------------------------------------------------------------------------------
# --> QQ-Plot Exponencial
#-------------------------------------------------------------------------------
n = length(saltos)
```

```
p = (1:n)/(n+1)
x = qexp(p, rate=1)
plot(x, saltos, xlab="", ylab="")
title(xlab=expression("-log(1-p"[i:n]*")"), ylab=expression("x"[i:n]))
#-------------------------------------------------------------------------------
# --> Gráfico de Excesso Médio (ME-Plot)
#-------------------------------------------------------------------------------
mePlot = function(x,u)
{
  n = length(x)
  k = 1:(n-1)
  me = (1/k)*cumsum(rev(x[k+1])) - x[n-k]
  me = rev(me)
  plot(x[k], me, main=paste("u =", u), xlab="", ylab="")
  title(xlab=expression("x"["(n-k):n"]), ylab=expression("e"["k,n"]))
  abline(v=u, lty=2, col="gray40")

  y = x[(sum(x<=u)+1):(n-1)]
  mey = me[(sum(x<=u)+1):(n-1)]
  abline(lm(mey~y), lty=2, col="gray40")
}
mePlot(saltos, 13.2)
mePlot(saltos, 13.8)
mePlot(saltos, 14.25)
mePlot(saltos, 14.7)
#-------------------------------------------------------------------------------
# --> Gráfico de estimativas (PWM) de k, para cada nível u definido
#-------------------------------------------------------------------------------
k = c()
n = length(saltos)
for(i in 1:(n-2))
{
    u = saltos[i]
    y = saltos[saltos>u] - u
    k = c(k, -0.5*((mean(y)^2)/var(y) - 1))
}
plot(saltos[1:(n-2)], k, type="l", xlab="", ylab="", ylim=c(-1,0))
title(xlab="Marcas (metros)", ylab=expression("k"["PWM"]))
abline(h=0, lty=2, col="gray40")
u = 13.80; abline(v=u, lty=2, col="gray40")
u = 14.25; abline(v=u, lty=2, col="gray40")
u = 14.70; abline(v=u, lty=2, col="gray40")
```

```
#=============================================================================
# (3) ESTIMAÇÃO DOS PARÂMETROS DA GPD
#=============================================================================
tab = read.table("IAAF_TripleJumpF_Estimates.txt", header=FALSE)
colnames(tab) = c("Metodo","u","m","k","sigma")

ML = tab[tab[,1]==1,]
EPM = tab[tab[,1]==2,]
ZS = tab[tab[,1]==3,]
MAPE = tab[tab[,1]==4,]
MCE = tab[tab[,1]==5,]

#=============================================================================
# (4) TESTES DE AJUSTAMENTO E INFERÊNCIA
#=============================================================================
install.packages("fExtremes"); library(fExtremes)
install.packages("goftest"); library(goftest)

#-----------------------------------------------------------------------------
# --> Medidas de ajustamento dos modelos GPD, para cada u
#-----------------------------------------------------------------------------
u = unique(tab[,2])
gof = matrix(0, nrow=5, ncol=3*length(u))
colnames(gof) = c("Cor1","CvM1","AD1","Cor2","CvM2","AD2","Cor3","CvM3","AD3")

for(i in 1:length(u))
{
    y = saltos[saltos > u[i]] - u[i]
    m = length(y)
    p = (1:m)/(m+1)
    j = i-1

    gof[1,3*j+1] = cor(qgpd(p, xi=ML[i,4], beta=ML[i,5]), y)
    gof[2,3*j+1] = cor(qgpd(p, xi=EPM[i,4], beta=EPM[i,5]), y)
    gof[3,3*j+1] = cor(qgpd(p, xi=ZS[i,4], beta=ZS[i,5]), y)
    gof[4,3*j+1] = cor(qgpd(p, xi=MAPE[i,4], beta=MAPE[i,5]), y)
    gof[5,3*j+1] = cor(qgpd(p, xi=MCE[i,4], beta=MCE[i,5]), y)

    gof[1,3*j+2] = cvm.test(y, "pgpd", xi=ML[i,4], beta=ML[i,5])$p.value
    gof[2,3*j+2] = cvm.test(y, "pgpd", xi=EPM[i,4], beta=EPM[i,5])$p.value
    gof[3,3*j+2] = cvm.test(y, "pgpd", xi=ZS[i,4], beta=ZS[i,5])$p.value
```

```
    gof[4,3*j+2] = cvm.test(y, "pgpd", xi=MAPE[i,4], beta=MAPE[i,5])$p.value
    gof[5,3*j+2] = cvm.test(y, "pgpd", xi=MCE[i,4], beta=MCE[i,5])$p.value

    gof[1,3*j+3] = ad.test(y, "pgpd", xi=ML[i,4], beta=ML[i,5])$p.value
    gof[2,3*j+3] = ad.test(y, "pgpd", xi=EPM[i,4], beta=EPM[i,5])$p.value
    gof[3,3*j+3] = ad.test(y, "pgpd", xi=ZS[i,4], beta=ZS[i,5])$p.value
    gof[4,3*j+3] = ad.test(y, "pgpd", xi=MAPE[i,4], beta=MAPE[i,5])$p.value
    gof[5,3*j+3] = ad.test(y, "pgpd", xi=MCE[i,4], beta=MCE[i,5])$p.value
}
#-------------------------------------------------------------------------------
# --> Nível de retorno dos modelos GPD, para cada u
#-------------------------------------------------------------------------------
u = unique(tab[,2])
m = c(100,200,1000)
rLevel = matrix(0, nrow=6, ncol=length(u)*length(m))
colnames(rLevel) = paste(rep(u,each=length(m)),"|",rep(m,times=length(u)),sep="")

for(i in 1:length(u))
  for(k in 1:length(m))
  {
    c = sum(saltos>u[i])/length(saltos)
    p = 1/(c*m[k])
    j = i-1

    if(p>1)
    {
      rLevel[,3*j+k] = -1
      rLevel[6,3*j+k] = quantile(saltos, probs=1-(1/m[k]))
    }
    else
    {
      rLevel[1,3*j+k] = qgpd(p, xi=ML[i,4], beta=ML[i,5], lower.tail=FALSE)
      rLevel[2,3*j+k] = qgpd(p, xi=EPM[i,4], beta=EPM[i,5], lower.tail=FALSE)
      rLevel[3,3*j+k] = qgpd(p, xi=ZS[i,4], beta=ZS[i,5], lower.tail=FALSE)
      rLevel[4,3*j+k] = qgpd(p, xi=MAPE[i,4], beta=MAPE[i,5], lower.tail=FALSE)
      rLevel[5,3*j+k] = qgpd(p, xi=MCE[i,4], beta=MCE[i,5], lower.tail=FALSE)
      rLevel[6,3*j+k] = quantile(saltos, probs=1-(1/m[k])) - u[i]
      rLevel[,3*j+k] = rLevel[,3*j+k] + u[i]
    }
  }
#-------------------------------------------------------------------------------
# --> Período de retorno dos modelos GPD, para cada u
```

```
#-------------------------------------------------------------------------------
u = unique(tab[,2])
M = c(15.38,15.50)
L = c("OR","WR")
rPer = matrix(0, nrow=5, ncol=length(u)*length(M))
colnames(rPer) = paste(rep(u,each=length(L)),"|",rep(L,times=length(u)),sep="")

for(i in 1:length(u))
  for(k in 1:length(M))
  {
      c = sum(saltos>u[i])/length(saltos)
      Mu = M[k]-u[i]
      j = i-1

      rPer[1,2*j+k] = pgpd(Mu, xi=ML[i,4], beta=ML[i,5], lower.tail=FALSE)
      rPer[2,2*j+k] = pgpd(Mu, xi=EPM[i,4], beta=EPM[i,5], lower.tail=FALSE)
      rPer[3,2*j+k] = pgpd(Mu, xi=ZS[i,4], beta=ZS[i,5], lower.tail=FALSE)
      rPer[4,2*j+k] = pgpd(Mu, xi=MAPE[i,4], beta=MAPE[i,5], lower.tail=FALSE)
      rPer[5,2*j+k] = pgpd(Mu, xi=MCE[i,4], beta=MCE[i,5], lower.tail=FALSE)
      rPer[,2*j+k] = 1/(c*rPer[,2*j+k])
  }
```

Anexo H

Melhores Marcas Femininas em Triplo Salto (Código Matlab)

```
%%=============================================================================
%%=============  MELHORES MARCAS FEMININAS EM TRIPLO SALTO  ===============
%%=============================================================================
saltos = importdata('IAAF_TripleJumpF.txt');
saltos = sort(saltos)';

%-------------------------------------------------------------------------------
% (1) Cálculo das estimativas, para cada nivel u
%-------------------------------------------------------------------------------
w = pocketAlgorithm(1e05);
ests = [];
for u = [13.8, 14.25, 14.7]

    y = saltos(saltos>u) - u;
    m = length(y);
    ests = [ests; 1, u, m, ML(y)'];
    ests = [ests; 2, u, m, EPM(y)'];
    ests = [ests; 3, u, m, ZS(y)'];
    ests = [ests; 4, u, m, MAPE(y)'];
    ests = [ests; 5, u, m, MCE(y,w)'];

end
%-------------------------------------------------------------------------------
% (2) Exportação das estimativas calculadas
%-------------------------------------------------------------------------------
dlmwrite('IAAF_TripleJumpF_Estimates.txt', ests, 'delimiter', '\t');
```

Anexo I

Montante de Sinistros Automóvel (Código R)

```
#================================================================================
#====================  MONTANTES DE SINISTROS AUTOMÓVEIS  ======================
#================================================================================
dados = read.table("SinsAutomovel_Input.txt", header=TRUE)

#--------------------------------------------------------------------------------
# --> Criação de índices anuais (fixar o ano mais recente como base)
#--------------------------------------------------------------------------------
ipc = unique(dados[,c("AnoSin","IpcSin")])
ipc = cbind(ipc, 0)
colnames(ipc) = c("AnoSin","IpcSin","Indice")

for(i in seq(dim(ipc)[1],1,-1))
{
  if(i == dim(ipc)[1])
    ipc[i,3] = 100
  else
    ipc[i,3] = ipc[i+1,3]/(1 + ipc[i+1,2]/100)
}

#--------------------------------------------------------------------------------
# --> Atualização de montantes relativamente ao ano mais recente
#--------------------------------------------------------------------------------
for(i in 1:length(ipc[,"AnoSin"]))
{
  indiceAno = ipc[i,3]
  indiceBase = ipc[ipc[,"AnoSin"]==max(ipc[,"AnoSin"]),3]

  j = which(dados[,"AnoSin"] == ipc[i,1])
```

```
  # Atualização do valor dos montantes
  dados[j,"Montante"] = dados[j,"Montante"] * (indiceBase/indiceAno)
  # Logaritmização dos montantes
  dados[j,"Montante"] = log(dados[j,"Montante"])
}


#=============================================================================
# (1) ANÁLISE PRELIMINAR ANUAL
#=============================================================================

#-----------------------------------------------------------------------------
# --> Frequência anual de Montantes (Gráfico de Barras)
#-----------------------------------------------------------------------------
barplot(table(dados[,"AnoSin"]), ylim=c(0,1100), col="lightgrey")
title(xlab="Anos", ylab="Frequência")
abline(h=900, lty=2)
box()

#-----------------------------------------------------------------------------
# --> Variabilidade anual dos Montantes (Boxplots)
#-----------------------------------------------------------------------------
boxplot(dados[,"Montante"] ~ dados[,"AnoSin"], col="lightgrey")
title(xlab="Anos", ylab="log(Montantes) (Euros)")
box()

#-----------------------------------------------------------------------------
# --> Montantes máximos por data de referência
#-----------------------------------------------------------------------------
montMax = tapply(dados[,"Montante"], dados[,"DtSin"], max)
plot(montMax, type="h", xaxt="n", xlab="", ylab="")
title(xlab="DtReferência", ylab="log(Montantes) (Euros)")
labs = sort(unique(dados[,"DtSin"]))
i = seq(from=1, to=length(montMax), by=12)
axis(1, at=i, labels=labs[i])

#-----------------------------------------------------------------------------
# --> Gráfico de autocorrelação dos Montantes máximos por data de referência
#-----------------------------------------------------------------------------
acf(montMax, main="", xlab="", ylab="")
title(xlab="Lag", ylab="Autocorrelação")
```

```
#=============================================================================
# (2) ANÁLISE PRELIMINAR GLOBAL
#=============================================================================

#-----------------------------------------------------------------------------
# --> Determinação da melhor marca de cada atleta + "shake" dos valores
#-----------------------------------------------------------------------------
sinsauto = sort(dados[,"Montante"])
write(sinsauto, file="SinsAutomovel.txt", ncolumns=1)

#-----------------------------------------------------------------------------
# --> Frequência de sinistros por classe (Histograma)
#-----------------------------------------------------------------------------
hist(sinsauto, freq=T, main="", xlab="", ylab="", col="lightgrey")
title(xlab="log(Montantes) (Euros)", ylab="Frequência")
box()

#-----------------------------------------------------------------------------
# --> Variabilidade global das marcas (Boxplot)
#-----------------------------------------------------------------------------
boxplot(sinsauto, horizontal=TRUE, main="", xlab="", col="lightgrey")
title(xlab="log(Montantes) (Euros)")
box()

#-----------------------------------------------------------------------------
# --> QQ-Plot Exponencial
#-----------------------------------------------------------------------------
n = length(sinsauto)
p = (1:n)/(n+1)
x = qexp(p, rate=1)
plot(x, sinsauto, xlab="", ylab="")
title(xlab=expression("-log(1-p"[i:n]*")"), ylab=expression("x"[i:n]))

#-----------------------------------------------------------------------------
# --> Gráfico de Excesso Médio (ME-Plot)
#-----------------------------------------------------------------------------
mePlot = function(x,u)
{
n = length(x)
k = 1:(n-1)
```

```
me = (1/k)*cumsum(rev(x[k+1])) - x[n-k]
me = rev(me)
plot(x[k], me, main=paste("u =", u), xlab="", ylab="")
title(xlab=expression("x"["(n-k):n"]), ylab=expression("e"["k,n"]))
abline(v=u, lty=2, col="gray40")

y = x[(sum(x<=u)+1):(n-1)]
mey = me[(sum(x<=u)+1):(n-1)]
abline(lm(mey~y), lty=2, col="gray40")
}

mePlot(sinsauto, 9.2)
mePlot(sinsauto, 12.43)  #Aproximadamente quantil 98%
mePlot(sinsauto, 12.79)  #Aproximadamente quantil 99%
mePlot(sinsauto, 13.2)

#-------------------------------------------------------------------------------
# --> Gráfico de estimativas (PWM) de k, para cada nível u definido
#-------------------------------------------------------------------------------
k = c()
n = length(sinsauto)
for(i in 1:(n-2))
{
u = sinsauto[i]
y = sinsauto[sinsauto>u] - u
k = c(k, -0.5*((mean(y)^2)/var(y) - 1))
}

plot(sinsauto[1:(n-2)], k, type="l", xlab="", ylab="", ylim=c(-1,0.2))
title(xlab="log(Montantes) (Euros)", ylab=expression("k"["PWM"]))
abline(h=0, lty=2, col="gray40")

u = 12.43
abline(v=u, lty=2, col="gray40")

u = 12.79
abline(v=u, lty=2, col="gray40")

u = 13.2
abline(v=u, lty=2, col="gray40")
```

```
#=============================================================================
# (3) ESTIMAÇÃO DOS PARÂMETROS DA GPD
#=============================================================================
tab = read.table("SinsAutomovel_Estimates.txt", header=FALSE)
colnames(tab) = c("Metodo","u","m","k","sigma")

ML = tab[tab[,1]==1,]
EPM = tab[tab[,1]==2,]
ZS = tab[tab[,1]==3,]
MAPE = tab[tab[,1]==4,]
MCE = tab[tab[,1]==5,]


#=============================================================================
# (4) TESTES DE AJUSTAMENTO E INFER??NCIA
#=============================================================================
install.packages("fExtremes"); library(fExtremes)
install.packages("goftest"); library(goftest)

#-----------------------------------------------------------------------------
# --> Medidas de ajustamento dos modelos GPD, para cada u
#-----------------------------------------------------------------------------
u = unique(tab[,2])
gof = matrix(0, nrow=5, ncol=3*length(u))
L = c("Cor","CvM","AD")
colnames(gof) = paste(rep(u,each=length(L)),"|",rep(L,times=length(u)),sep="")

for(i in 1:length(u))
{
y = sinsauto[sinsauto > u[i]] - u[i]
m = length(y)
p = (1:m)/(m+1)
j = i-1

gof[1,3*j+1] = cor(qgpd(p, xi=ML[i,4], beta=ML[i,5]), y)
gof[2,3*j+1] = cor(qgpd(p, xi=EPM[i,4], beta=EPM[i,5]), y)
gof[3,3*j+1] = cor(qgpd(p, xi=ZS[i,4], beta=ZS[i,5]), y)
gof[4,3*j+1] = cor(qgpd(p, xi=MAPE[i,4], beta=MAPE[i,5]), y)
gof[5,3*j+1] = cor(qgpd(p, xi=MCE[i,4], beta=MCE[i,5]), y)

gof[1,3*j+2] = cvm.test(y, "pgpd", xi=ML[i,4], beta=ML[i,5])$p.value
gof[2,3*j+2] = cvm.test(y, "pgpd", xi=EPM[i,4], beta=EPM[i,5])$p.value
```

```
gof[3,3*j+2] = cvm.test(y, "pgpd", xi=ZS[i,4], beta=ZS[i,5])$p.value
gof[4,3*j+2] = cvm.test(y, "pgpd", xi=MAPE[i,4], beta=MAPE[i,5])$p.value
gof[5,3*j+2] = cvm.test(y, "pgpd", xi=MCE[i,4], beta=MCE[i,5])$p.value

gof[1,3*j+3] = ad.test(y, "pgpd", xi=ML[i,4], beta=ML[i,5])$p.value
gof[2,3*j+3] = ad.test(y, "pgpd", xi=EPM[i,4], beta=EPM[i,5])$p.value
gof[3,3*j+3] = ad.test(y, "pgpd", xi=ZS[i,4], beta=ZS[i,5])$p.value
gof[4,3*j+3] = ad.test(y, "pgpd", xi=MAPE[i,4], beta=MAPE[i,5])$p.value
gof[5,3*j+3] = ad.test(y, "pgpd", xi=MCE[i,4], beta=MCE[i,5])$p.value
}

#-------------------------------------------------------------------------------
# --> Nível de retorno dos modelos GPD, para cada u
#-------------------------------------------------------------------------------
u = unique(tab[,2])
m = c(100,200,1000)
rLevel = matrix(0, nrow=6, ncol=length(u)*length(m))
colnames(rLevel) = paste(rep(u,each=length(m)),"|",rep(m,times=length(u)),sep="")

for(i in 1:length(u))
for(k in 1:length(m))
{
    c = sum(sinsauto>u[i])/length(sinsauto)
    p = 1/(c*m[k])
    j = i-1

    if(p>1)
    {
      rLevel[,3*j+k] = -1
      rLevel[6,3*j+k] = quantile(sinsauto, probs=1-(1/m[k]))
    }
    else
    {
      rLevel[1,3*j+k] = qgpd(p, xi=ML[i,4], beta=ML[i,5], lower.tail=FALSE)
      rLevel[2,3*j+k] = qgpd(p, xi=EPM[i,4], beta=EPM[i,5], lower.tail=FALSE)
      rLevel[3,3*j+k] = qgpd(p, xi=ZS[i,4], beta=ZS[i,5], lower.tail=FALSE)
      rLevel[4,3*j+k] = qgpd(p, xi=MAPE[i,4], beta=MAPE[i,5], lower.tail=FALSE)
      rLevel[5,3*j+k] = qgpd(p, xi=MCE[i,4], beta=MCE[i,5], lower.tail=FALSE)
      rLevel[6,3*j+k] = quantile(sinsauto, probs=1-(1/m[k])) - u[i]
      rLevel[,3*j+k] = rLevel[,3*j+k] + u[i]
    }
}
```

```
#-------------------------------------------------------------------------------
# --> Período de retorno dos modelos GPD, para cada u
#-------------------------------------------------------------------------------
u = unique(tab[,2])
M = c(quantile(sinsauto,prob=c(0.9995,0.9999)),max(sinsauto))
L = c("Q_{0.9995}","Q_{0.9999}","Max")
rPer = matrix(0, nrow=5, ncol=length(u)*length(M))
colnames(rPer) = paste(rep(u,each=length(L)),"|",rep(L,times=length(u)),sep="")

for(i in 1:length(u))
  for(k in 1:length(M))
  {
    c = sum(sinsauto>u[i])/length(sinsauto)
    Mu = M[k]-u[i]
    j = i-1

    rPer[1,3*j+k] = pgpd(Mu, xi=ML[i,4], beta=ML[i,5], lower.tail=FALSE)
    rPer[2,3*j+k] = pgpd(Mu, xi=EPM[i,4], beta=EPM[i,5], lower.tail=FALSE)
    rPer[3,3*j+k] = pgpd(Mu, xi=ZS[i,4], beta=ZS[i,5], lower.tail=FALSE)
    rPer[4,3*j+k] = pgpd(Mu, xi=MAPE[i,4], beta=MAPE[i,5], lower.tail=FALSE)
    rPer[5,3*j+k] = pgpd(Mu, xi=MCE[i,4], beta=MCE[i,5], lower.tail=FALSE)
    rPer[,3*j+k] = 1/(c*rPer[,3*j+k])
  }
```

Anexo J

Montante de Sinistros Automóvel (Código Matlab)

```
%%=============================================================================
%%=============  MELHORES MARCAS FEMININAS EM TRIPLO SALTO  ===============
%%=============================================================================
sinsauto = importdata('SinsAutomovel.txt');
sinsauto = sort(sinsauto)';

%-------------------------------------------------------------------------
% (1) Cálculo das estimativas, para cada nivel u
%-------------------------------------------------------------------------
w = pocketAlgorithm(1e05);
ests = [];
for u = [12.43, 12.79, 13.2]

    y = sinsauto(sinsauto>u) - u;
    m = length(y);

    ests = [ests; 1, u, m, ML(y)'];
    ests = [ests; 2, u, m, EPM(y)'];
    ests = [ests; 3, u, m, ZS(y)'];
    ests = [ests; 4, u, m, MAPE(y)'];
    ests = [ests; 5, u, m, MCE(y,w)'];

end

%-------------------------------------------------------------------------
% (2) Exportação das estimativas calculadas
%-------------------------------------------------------------------------
dlmwrite('SinsAutomovel_Estimates.txt', ests, 'delimiter', '\t');
```

Referências Bibliográficas

[1] Ahsanullah, M., Nevzorov, V.B. & Shakil, M. (2013). An Introduction to Order Statistics. Atlantis Press.

[2] Bazaraa, M.S., Sherali, H.D. & Shetty, C.M. (2014). Nonlinear Programming: Theory and Algorithms (Third Edition). Wiley.

[3] Beaumont, M.A., Zhang, W. & Balding, D.J. (2002). Approximate Bayesian Computation in Population Genetics. *Genetics*, 162(4), 2025—2035.

[4] Beirlant, J., Teugels, J. & Vynckier, P. (1996). Practical Analysis of Extreme Values. Leuven University Press.

[5] Beirlant, J., Goegebeur, Y., Segers, J. & Teugels, J. (2004). Statistics of Extremes: Theory and Applications. Wiley.

[6] Biau, G., Cérou, F. & Guyader, A. (2013). New Insights Into Approximate Bayesian Computation.

[7] Bishop, C. M. (2006). Pattern Recognition and Machine Learning. Springer.

[8] Bolstad, W.M. & Curran, J.M. (2017). Introduction to Bayesian Statistics (Third Edition). Wiley.

[9] Boggs, P.T. & Tolle, J.W. (1996). Sequential Quadratic Programming. *Acta Numerica*, 4(4), 1–51.

[10] Castillo, E. & Hadi, A.S. (1997). Fitting the Generalized Pareto Distribution to Data. *Journal of the American Statistical Association*, 92(440), 1609–1620.

[11] Choulakian, V. & Stephens, M.A. (2001). Goodness-of-Fit Tests for the Generalized Pareto Distribution. *Technometrics*, 43(4), 478–484.

[12] Coles, S. (2001). An Introduction to Statistical Modeling of Extreme Values. Springer.

[13] Csilléry, K., Blum, M.G.B., Gaggiotti, O.E. & François, O. (2010). Approximate Bayesian Computation (ABC) in practice. *Trends in Ecology & Evolution*, 25(7), 410–418.

[14] DuMouchel, W.H. (1983). Estimating the stable index α in order to measure tail thickness: a critique. *The Annals of Statistics*, 11(4), 1019–1031.

[15] Embrechts, P., Kluppelberg, C. & Mikosch, T. (1997). Modelling Extremal Events for Insurance and Finance. Springer-Verlag.

[16] Faraway, J.J. (2002). Practical Regression and Anova using R. University of Bath.

[17] Fonseca e Silva, A. (1994). Dicionário de Seguros. Publicações Dom Quixote.

[18] Fraga Alves, I. (2011). Notas teóricas de Estatística Computacional e Simulação. FCUL.

[19] Fu, Y.-X. & Li, W.-H. (1997). Estimating the Age of the Common Ancestor of a Sample of DNA Sequences. *Molecular Biology and Evolution*, 14(2), 195–199.

[20] Gallant, S.I. (1990). Perceptron-Based Learning Algorithms. *IEEE Transactions of Neural Networks*, 2(1), 179–191

[21] Gomes, M.I., Fraga Alves, M.I. & Neves, C. (2013). Análise de Valores Extremos: Uma Introdução. Edições SPE.

[22] Government of Western Australia (2017). Athletics – Jumping Events. `https://www.dsr.wa.gov.au/support-and-advice/facility-management/developing-facilities/dimensions-guide/sport-specific-dimensions/athletics-jumping-events`

[23] Greenwood, J.A., Landwehr, J.M., Matalas, N.C. & Wallis, J.R. (1979). Probability Weighted Moments: Definition and Relation to Parameters of Several Distributions Expressable in Inverse Form. *Water Resources Research*, 15(5), 1049—1054.

[24] Grimshaw, S.D. (1993). Computing Maximum Likelihood Estimates for the Generalized Pareto Distribution. *Technometrics*, 35(2), 185–191.

[25] Han, S.-P. (1976). Supralinearly Convergent Variable Metric Algorithms for General Nonlinear Programming Problems. *Mathematical Programming*, 11(1), 263—282.

[26] Hosking, J.R.M. & Wallis, J.R. (1987). Parameter and Quantile Estimation for the Generalized Pareto Distribution. *Technometrics*, 29(3), 339–349.

[27] Hymans, R. & Matrahazi, I. (2015). IAAF World Records Progression - 2015 Edition. IAAF.

[28] IAAF (2015). IAAF Competition Rules 2016-2017, in force from 1 November 2015. IAAF.

[29] Kim, T.-H. & White, H. (2004). On more robust estimation of skewness and kurtosis. *Finance Research Letters*, 1(1), 56–73.

[30] Landwehr, J.M., Matalas, N.C. & Wallis, J.R. (1979). Estimation of Parameters and Quantiles of Wakeby Distributions. *Water Resources Research*, 15, 1361–1379.

[31] Marsaglia, G. & Marsaglia, J. C. W. (2004). Evaluating the Anderson-Darling Distribution. *Journal of Statistical Software*, 9(2), 1–5.

[32] Mathworks (2017). Optimization Toolbox™ User's Guide (R2017a). Mathworks.

[33] McNeil, A.J. (1997). Estimating the Tails of Loss Severity Distributions Using Extreme Value Theory. *ASTIN Bulletin*, 27(1), 117–137.

[34] Mendes, B.V.M. & Lopes, H.D. (2004). Data driven estimates for mixtures. *Computational Statistics & Data Analysis*, 47(3), 583–598.

[35] Nocedal, J. & Wright, S.J. (2006). Numerical Optimization (Second Edition). Springer.

[36] Powell, M.J.D. (1978). A fast algorithm for nonlinearly constrained optimization calculations. *Numerical Analysis*, 630, 144–157.

[37] Pritchard, J.K., Seielstad, M.T., Perez-Lezaun, A. & Feldman, M.W. (1999). Population Growth of Human Y Chromosomes: A Study of Y Chromosome Microsatellites. *Mol. Biol. Evol.*, 16(12), 1791–1798.

[38] Rosenblatt, F. (1962). The Principles of Neurodynamics: Perceptrons and the Theory of Brain Mechanisms. Spartan.

[39] Rubin, D.B. (1984). Bayesianly Justifiable and Relevant Frequency Calculations for the Applies Statistician. *The Annals of Statistics*, 12(4), 1151–1172.

[40] Severino, E. (2007). Notas teóricas de Processos Estocásticos e Simulação. FCUL.

[41] Smith, R.L. (1984). Threshold methods for sample extremes. *Statistical Extremes and Applications*, 131, 621–638.

[42] Statistics4U (2012). Fundamentals of Statistics – Structure of Measured Data. `http://www.statistics4u.info/fundstat_eng/cc_data_structure.html`

[43] Tavaré, S., Balding, D. J., Griffiths, R. C. & Donnelly, P. (1997). Inferring Coalescence Times From DNA Sequence Data. *Genetics*, 145(2), 505–518.

[44] Thompson, G. (2009). Statistical Literacy Guide - How to adjust for inflation. House of Commons Library.

[45] Van de Schoot, R., Kaplan, D., Denissen, J., Asendorpf, J.B., Neyer, F.J. & Van Aken, M.A.G. (2014). A Gentle Introduction to Bayesian Analysis: Applications to Developmental Research. *Child Development*, 85(3), 835-–1321

[46] Weglarczyk, S., Strupczewski, W.G. & Singh, V.P. (2005). Three-parameter discontinuous distributions for hydrological samples with zero values. *Hydrological Processes*, 19(15), 2899-2914.

[47] Wilson, R.B. (1963). A Simplicial Algorithm for Concave Programming. Ph.D. Dissertation, Graduate School of Business Administration, Harward University, Boston.

[48] de Zea Bermudez, P. & Kotz, S. (2010). Parameter estimation of the generalized Pareto distribution – Part I. *Journal of Statistical Planning and Inference*, 140(6), 1353–1373.

[49] de Zea Bermudez, P., Mendes, J., Pereira, J.M.C., Turkman, K.F. & Vasconcelos, M.J.P. (2009). Spatial and temporal extremes of wildfire sizes in Portugal (1984-2004). *International Journal of Wildland Fire*, 18, 983–991.

[50] de Zea Bermudez, P., Turkman, M.A.A. & Turkman, K.F. (2001). A Predictive Approach to Tail Probability Estimation. *Extremes*, 4(4), 295—314.

[51] de Zea Bermudez, P. & Turkman, M.A.A. (2003). Bayesian Approach to Parameter Estimation of the Generalized Pareto Distribution. *Test*, 12(1), 259–277.

[52] Zhang, J. (2007). Likelihood Moment Estimation for the Generalized Pareto Distribution. *Australian & New Zealand Journal of Statistics*, 49(1), 69–77.

[53] Zhang, J. (2010). Improving on Estimation for the Generalized Pareto Distribution. *Technometrics*, 52(3), 335–339.

[54] Zhang, J. & Stephens, M.A. (2009). A New and Efficient Estimation Method for the Generalized Pareto Distribution. *Technometrics*, 51(3), 316–325.