7
Referências Bibliográficas

1. ÁLVAREZ, G. Operational Risk Quantification. Mathematical Solutions for Analyzing Loss Data. Response to the 2001 Basel Committee on Banking Supervision Consultative Document on Operational Risk, May 2001.

2. BARROS, M. Notas de Inferência Estatística. Pontifícia Universidade Católica do Rio de Janeiro (PUC-RJ), 1999.

3. Basel Committee on Banking Supervision (1998a). "Framework for the evaluation of internal control systems", Basel, January 1998.

4. Basel Committee on Banking Supervision (1998b). "Operational risk management, Basel", September 1998.

5. Basel Committee on Banking Supervision (1999). "Core principles methodology", Basel, October 1999.

6. Basel Committee on Banking Supervision (2001a). "Operational risk". Supporting document to the new Basel capital accord, Consultative document, Basel, January 2001.

7. Basel Committee on Banking Supervision (2001b). "QIS 2 - Operational loss risk data, Guidance note to Quantitative Impact Survey", Basel, May 2001.

8. Basel Committee on Banking Supervision (2001c). "Working paper on the regulatory treatment of operational risk", Basel, September 2001.

9. Basel Committee on Banking Supervision (2001d). Regulatory Treatment of Operational Risk, Basel, 2001.

10. Basel Committee on Banking Supervision (2003a). "Overview of The New Basel Capital Accord", Consultative document, Basel, April 2003.

11. Basel Committee on Banking Supervision (2003b). "The 2002 Loss Data Collection Exercise for Operational Risk: Summary of the Data Collected", Basel, March 2003.

12. Basel Committee on Banking Supervision (2004). "International Convergence of Capital Measurement and Capital Standards. A Revised Framework", Basel, June 2004.

13. BAKER, C. The Numerical Treatment of Integral Equations. Oxford: Clarendon Press, 1997.

14. BAUD, N.; FRACHOT, A.; RONCALLI, T. Internal data, external data and consortium data for operational risk measurement: How to pool data properly? Working paper, Groupe de Recherche Opérationnelle, Crédit Lyonnais, France, June 2002.

15. ______. How to Avoid Over-estimating Capital Charge for Operational Risk? Working paper, Groupe de Recherche Opérationnelle, Crédit Lyonnais, France, December 2002.

16. BEANS, K. M. How to Create a Framework for Managing Operational Risk. Community Banker, v.12, n.4, p. 22-26, April 2003.

17. CAGAN, P. The Tale of the Rogue Trader, Making a Scene… Algo Research Quarterly, v. 5, n. 2, 2002.

18. CHAPELLE, A. et al. Basel II and Operational Risk: implications for risk measurement and management in the financial sector. National Bank of Belgium (NBB Working Paper no. 51), May 2004.

19. CHANDRA, M.; SINGPURWALLA, N. D.; STEPHENS, M. A. Kolmogorov Statistics for Test of Fit for the Extreme Value and Weibull Distributions. Journal of the American Statistical Association, v. 76, p. 729-731, 1981.

20. COLEMAN, R. Using Modelling in Operational Risk Management, Conference "Operational Risk in Retail Financial Services", 19th & 20th June 2000, The Mayfair Conference Centre, London 57.

21. ______. Operational Risk: Modelling Extremes, Actuarial Teaching and Research Conference 8th and 9th July 2002 Christ Church, Oxford.

22. ______. Op risk modelling for extremes – Part 1, Operational Risk, December 2002.

23. ______. (2003a) Op risk modelling for extremes – Part 2, Operational Risk, January 2003

24. ______. (2003b) Statistical Techniques and Qualitative Adjustments for LDA, ORRFXVII at BaFin in Bonn, October 2003.

25. COLES, S.; DIXON, M. Likelihood-based Inference for Extreme Value Models, working paper, 1998.

26. CRUZ, M. Modeling, Measuring and Hedging Operational Risk. Londres, John Wiley & Sons, Ltd, 2002.

27. CURRIE, C. V. Basel II and Operational Risk – Overview of Key Concerns, IQPC Operational Risk Forum, 25th March 2004, Carlton Crest Hotel, Sydney.

28. DANIELSSON, J.; DE VRIES, C. G (1997b) Beyond the sample: extreme quantile and probability with applications to financial data. Tinbergen Institute, Rotterdan (Technical Report Financial Markets Group (FMG)), 1997.

29. DANIELSSON, J. et al. An Academic Response to Basel II. Special Paper n. 130, Financial Markets Group, London School of Economics, 2001.

30. DE FONTNOUVELLE, P. et al. Capital and Risk: New Evidence on Implications of Large Operational Losses. Working paper, Federal Reserve Bank of Boston, 2003.

31. ______; ROSENGREN, E. Implication of Alternative Operational Risk Modeling Techniques. Working paper, Federal Reserve Bank of Boston, June, 2004.

32. DEKKERS, A.; EINMAHL, J.; DE HAAN, L. A moment estimator for the index of an extreme-value distribution. Annals of Statistics, 17, p. 1833-55, 1989.

33. DELBAEN, F.; DENAULT, M Coherent allocation of risk capital, RiskLab, Working Paper, 2000.

34. DOERIG, H. Operational Risk in Financial Services. An old challenge in a new environment. Institut International D'Etudes Bancaries, London, 2000.

35. EMBRECHTS, P.; KLUPPELBERG, C.; MIKOSCH, T. Modelling Extremal Events for Insurance and Finance. Springer-Verlag, New York, 1997.

36. ______ . Extremes and Integrated Risk Management. Risk Books in association with UBS Warburg, July 2000.

37. ______; KAUFMANN, R.; SAMORONDNITSKY, G. Ruin Theory Revisited: Stochastic Models for Operational Risk. Working Paper. ETH-Zurich and Cornell University, 2002.

38. Federal Reserve Bank of Boston. Operational Risk Management: A Practical Approach and its Regulatory Implications. November 2001.

39. FELLER, W. An introduction to Probability Theory and Its Applications – Volume ? 3 ed., Wiley Series in Probability and Mathematical Statistics, John Wiley & Sons, New York, 1968 (first edition 1950).

40. ______. An introduction to Probability Theory and Its Applications – Volume ?I, 2 ed. Wiley Series in Probability and Mathematical Statistics, John Wiley & Sons, New York, 1971 (first edition 1966).

41. Financial Services Authority. Operational risk systems and controls, July 2002.

42. FRACHOT, A.; GEORGES, P.; RONCALLI, T. Loss Distribution Approach for operational risk. Working paper, Groupe de Recherche Opérationnelle, Crédit Lyonnais, France, April 2001.

43. ______; RONCALLI, T. Mixing internal and external data for managing operational risk. Working paper. Groupe de Recherche Opérationnelle, Crédit Lyonnais, France, January 2002.

44. ______; MOUDOULAUD O., RONCALLI, T. Loss Distribuition Approach in Pratice. Working paper. Groupe de Recherche Opérationnelle, Crédit Lyonnais, France, May 2003.

45. GIGERENZER, G. Reckoning with risk: learning to live with uncertainty. London: Penguin, 2002.

46. GUIMARÃES, T. A. Implementação do Método de Distribuição de Perdas para Risco Operacional. São Paulo, 2003. 59p. Dissertação

de Mestrado – Faculdade de Economia, administração e contabilidade, Universidade de São Paulo, 2003.

47. GULDIMANN, T. Capital Allocation for Operational Risks. The Federal Reserve Bank of Boston, November 15, 2001.

48. HALL, P. Using the bootstrap to estimate mean squared error and select smoothing parameter in nonparametric problems. Journal of Multivariate Analysis, v. 32, p. 177-203, 1990.

49. HECKMAN, P. E.; MEYERS, G. G. The calculation of aggregate loss distributions from claim severity and claim count distributions. Proceedings of the Casualty Actuarial Society, LXX, p. 22-61, 1983.

50. HERRING, R. J. The Basel 2 Approach to Bank Operational Risk: Regulation on the Wrong Track. Journal of Risk Finance, v. 4, n. 1, p. 42-45, 2002.

51. HIDETOSHI, N. A Case Study of Operational Risk - Measurement based on Loss Distribution Approach. 27th International Congress of Actuaries, 2002.

52. HILL, B. A Simple General Approach to Inference about the Tail of a Distribution. Annals of Statistics, 3, p. 1163-1173, 1975.

53. HOSKING, J. R. M.; WALLIS, J. R.; WOOD, E. Estimation of the Generalized Extreme-Value Distribution by the Method of Probability-Weighted Moments. Technometrics, 27, p. 251-261, 1985.

54. ______; Wallis, J. R. Regional Frequency Analysis: An Approach Based on L-Moments. Cambridge University Press, 1997.

55. HUISMAN, R. et al. Tail-index estimates in small samples, Journal of Business and Economics Statistics, 19, p. 208-216, 2001.

56. Insurance of Operational Risk Under the New Basel Capital Accord. A Working Paper submitted by Insurance Companies, November 7, 2001.

57. JORION, P. Value at risk: The new benchmark for managing financial risk. 2 ed., McGraw-Hill, New York, 2000.

58. KLUGMAN, S.; PANJER, H.; WILMOT, G. Loss Models: From Data to Decisions. Wiley Series in Probability and Statistics, Wiley, New York, 1998.

59. LANDWEHR, J.; MATALAS, N.; WALLIS, J. R. Probability weighted moment compared to some traditional techniques in estimating Gumbel parameters and quantiles. Water Resources Research, 15, p. 1055-1064, 1979.

60. LEWIS, M. A. Cause, consequence and control: towards a theoretical and practical model of operational risk. Journal of Operations Management, 2003.

61. Ludwig Report. "Report to the Boards of Allied Irish Banks, p.l.c., Allfirst Financial Inc., and Allfirst Bank concering Currency Trading Losses." by Promontory Financial Group and Wachtell, Lipton, Rosen & Katz, www.allfirst.com/Ludwig_report.pdf, March, 2002.

62. MARSHAL, C. L. Measuring and Managing Operational Risk in Financial Instituitions. John Wiley & Sons, Ltd, Asia, 2000.

63. MEDOVA, E. A.; KYRIACOU, M. N. Extremes in operational risk management. University of Cambridge, March 2001.

64. MENDES, B. V. M. Introdução à análise de eventos extremos. E-papers Serviços Editoriais Ltda, Rio de Janeiro, 2004.

65. MIDDLEMISS, J. An Emerging 'New Science'. Wall Street & Technology, July 2003.

66. Ministério da Previdência Social – MPS (2004). Modelo Brasileiro de Gerenciamento de Riscos Operacionais da Previdência Social. MPS, Março 2004.

67. MUERMANN, A.; OKTEM, U. The Near-Miss Management of Operational Risk. Journal of Risk Finance, 2002.

68. NYSTRÖM, K.; SKOGLUND, J. Quantitative Operational Risk Management. Group Financial Risk Control Swedbank, Sept. 2002.

69. PANJER, H. Recursive evaluation of compound distributions. Astin Bulletin, 12, p. 22-26, 1981.

70. ______; LUTEK, B. Practical Aspects of Stop-Loss Calculations. Insurance: Mathematics and Economics, 2, p. 159-177, 1983.

71. ______; WILLMOT, G. Insurance Risk Models. Chigaco: Society of Actuaries, 1992.

72. PICKANDS, J. III. Statistical Inference Using Extreme Order Statistics, Annals of Statistics, n.3, p.119-131, 1975.

73. ______. Bayes Quantile Estimation and Threshold Selection for Generalized Pareto Family, Extreme Value Theory and Applications, Edited by Gambalos, J.; Lechner, J.; Simiu, E. (s.1): Kluver Academic Publishers, p.123-138, 1994.

74. PICTET, O.; DACOROGNA, M.; MULLER, U. Hill, Bootstrap and Jacknife Estimators for Heavy Tails. Olsen & Associates, 1996. http:\\www.olsen.ch

75. POWER, M. The Invention of Operational Risk. University of New South Wales, February 2003.

76. POWOJOWSKI, M. R.; REYNOLDS, D.; TUENTER, H. J. H. Dependent Events and Operational Risk. Algo Research Quartely, Summer 2002.

77. REYNOLDS, D.; SYER, D. Mark to Future and Operational Risk. Algo Research Quartely, Summer 2002.

78. ______; SYER, D. "The Actuarial Approach to Loss Distributions." Algo Research Quartely, Summer 2002.

79. ______. "BIS II and Beyond". Algo Research Quartely, Summer 2002.

80. ROBERTSON, J. P. The computation of aggregate loss distributions. Proceedings of the Casualty Actuarial Society, LXXIX, p. 57-133, 1992.

81. ROEHR, A. "Modelling Operational Losses". Algo Research Quartely, Summer 2002.

82. SPARROW, A. "A Theoretical Framework for Operational risk management and opportunity realization". New Zealand Treasure, 2000.

83. SUNDT, B.; JEWELL, W. S. Further Results on Recursive Evaluation of Compound Distributions. Astin Bulletin 12, 1981.

A. Distribuição de S(i,j) condicionada a N(i,j)=n

Seja S(i,j) = X_1(i,j) + X_2(i,j) + ... + X_n(i,j), onde N(i,j) = n. Por simplicidade de notação, a partir de agora serão desconsiderados os índice i e j.

Seja $G_n(s_n)$ = Pr (S = x | N = n) a distribuição de S condicionada a N = n.

Sejam as hipóteses abaixo:

- X_1, X_2, ...X_n são variáveis aleatórias iid condicionadas a N = n.
- A distribuição das variáveis aleatórias X_1, X_2, ...X_n independe de n, condicionadas a N = n.
- N e X_k são independentes para todo o k.
- X_k é uma variável aleatória não-negativa.

Pelas hipóteses mencionadas acima se conclui que X_1, X_2, ...X_n são variáveis aleatórias iid, unificadas pela notação X, onde a função de distribuição de probabilidade acumulada desta severidade é chamada de F com domínio real não-negativo.

Logo, $G_n(s_n)$ é igual a 1 quando x = 0, e igual a $F^{n*}(x)$ quando x > 0, onde * é o operador convolução sobre funções de distribuição, e $F^{n*}(x)$ é a n-ésima auto-convolução de F(x).

Demonstração para x = 0

De acordo com a nota de rodapé [73] tem-se: x = 0 ➔ n = 0.

Logo, $G_n(s_n)$ = Pr (S = 0 | N = 0) = 1.

Demonstração para x > 0

Para esta demonstração será utilizado o princípio da indução finita (P.I.F).

Em primeiro lugar, verifica-se se a tese vale para n = 1.

$$n = 1 \qquad \text{(i)}$$

De (i) tem-se:

$$S = X_1 \Rightarrow \Pr(S \le x \mid N = 1) = \Pr(X_1 \le x \mid N = 1) \qquad \text{(ii)}$$

Por hipótese X_1 e N são independentes, então:

$$\Pr(X_1 \le x \mid N = 1) = \Pr(X_1 \le x) \times \Pr(N = 1) \qquad \text{(iii)}$$

Sendo Pr(X_1 = x) = F(x), e Pr(N = 1) = 1 quando n = 1, de (8), (ii) e (iii) tem-se:

$$G_1(s_1) = F(x) = F^{1*}(x) \qquad \text{(iv)}$$

Então, a tese vale para n = 1.

Em seguida, arbitra-se n = k > 1 pertencente aos inteiros positivos, e assume-se que a tese é verdadeira para este k. Assim têm-se as equações:

$$S_k = X_1 + \Lambda + X_k \tag{v}$$

$$G_k(s_k) = F^{k*}(x) \tag{vi}$$

Por último, para se provar que a tese é verdadeira para todo n, tenta-se provar que ela vale para n = k + 1, baseando-se nas hipóteses já assumidas e nas equações (v) e (vi). Dessa forma, para n = k + 1:

$$S_{k+1} = X_1 + \Lambda + X_k + X_{k+1} = S_k + X_{k+1} \tag{vii}$$

Pelas hipóteses apresentadas, S_k e X_{k+1} são variáveis aleatórias independentes com suas respectivas funções de distribuição de probabilidade (f.d.p) iguais a $g_k(s_k)$ e $f(x_{k+1})$, sendo então a f.d.p conjunta igual a $g_k(s_k)f(x_{k+1})$.

Sendo $S_{k+1} = S_k + X_{k+1}$ e $W = X_{k+1}$, pela transformação inversa $s_k = s_{k+1} - w$, $x_{k+1} = w$ e o Jacobiano:

$$J = \begin{vmatrix} \dfrac{\partial s_k}{\partial s_{k+1}} & \dfrac{\partial s_k}{\partial w} \\ \dfrac{\partial x_{k+1}}{\partial s_{k+1}} & \dfrac{\partial x_{k+1}}{\partial w} \end{vmatrix} = \begin{vmatrix} 1 & -1 \\ 0 & 1 \end{vmatrix} = 1 \tag{viii}$$

Então aplicando a regra de transformação de variáveis contínuas, a f.d.p conjunta de S_{k+1} e W é:

$$g(s_{k+1}, w) = g_k(s_{k+1} - w) \times f(w) \times |J| = g_k(s_{k+1} - w) \times f(w) \tag{ix}$$

Assim, pelas equações (7) e (ix), a f.d.p marginal de S_k é:

$$g_{k+1}(s_{k+1}) = \int_{-\infty}^{+\infty} g_k(s_{k+1} - w) \times f(w) \times dw = g_k * f \tag{x}$$

Logo, pelas equações (7), (8), (vi) e (x) tem-se:

$$g_{k+1} = g_k * f = f^{k*} * f = f^{(k+1)*} \Rightarrow G_{k+1} = F^{(k+1)*} \tag{xi}$$

De (xi) prova-se que a tese também vale para n = k + 1. Assim pelo P.I.F, a tese vale para todo n.

B. Métodos de Discretização

O método de discretização deve preservar as propriedades da distribuição original tal como a forma geral da distribuição, e ao mesmo tempo preservar as quantidades globais tais como os momentos.

Método do Arredondamento (Dispersão de Massa)

Seja f_j a probabilidade em jh, sendo h a unidade monetária e j = 0, 1, 2, ... Tem-se:

$$\begin{aligned} f_0 &= \Pr(X < \frac{h}{2}) = F(\frac{h}{2}) \\ f_j &= \Pr(jh - \frac{h}{2} \le X < jh + \frac{h}{2}) = F(jh + \frac{h}{2}) - F(jh - \frac{h}{2}), \qquad j = 1, 2, \ldots \end{aligned} \tag{i}$$

Este método divide a probabilidade entre (j+1)h e jh, e a atribui a j+1 e j.

Método do Casamento do Momento Local

O objetivo deste método é casar p momentos da distribuição discretizada com os respectivos momentos da distribuição original.

Considere um intervalo arbitrário de tamanho ph, denotado por $[x_k, x_k+ph)$. Atribuem-se pontos de massa $m_0^k, m_1^k, \ldots, m_p^k$ aos pontos x_k, x_k+h, ..., x_k+ph, tal que os primeiros p momentos sejam preservados. Logo, o sistema de p+1 equações refletindo esta condição é:

$$\sum_{j=0}^{p} (x_k + jh)^r m_j^k = \int_{x_k - 0}^{x_k + ph - 0} x^r \, dF(x), \qquad r = 0, 1, 2, \ldots, p \tag{ii}$$

A notação “-0” nos limites da integral indica que uma probabilidade discreta em x_k deve ser incluída, porém em x_k+ph não.

Arranje os intervalos tal que $x_{k+1} = x_k+ph$, e os pontos extremos coincidam. Logo, os pontos de massa dos pontos extremos são somados. Sendo $x_0 = 0$, a distribuição discreta tem as seguintes probabilidades sucessivas:

$$\begin{array}{llll} f_0 = m_o^o, & f_1 = m_1^o, & f_2 = m_2^o, & \cdots \\ f_p = m_p^o + m_o^1, & f_{p+1} = m_1^1, & f_{p+2} = m_2^1, & \cdots \\ \vdots & \vdots & \vdots & \vdots \\ f_{np} = m_{np}^{n-1} + m_o^n, & f_{np+1} = m_1^n, & f_{np+2} = m_2^n, & \cdots \end{array} \tag{iii}$$

A solução do sistema de equações (ii) é:

$$m_j^k = \int_{x_k-0}^{x_k+ph-0} \prod_{i \neq j} \frac{x - x_k - ih}{(j-i)h} dF(x), \qquad j = 0,1,2,\ldots,p \qquad \text{(iv)}$$

Panjer & Lutek (1983) [70] descobriram que 2 momentos são geralmente suficientes, e que a adição do terceiro momento somente contribui marginalmente para a acurácia.

C. Distribuições da Classe (a, b, 0)

A classe (a, b, 0) representa a família de distribuições de freqüência que satisfazem à recursão:

$$p(k) = \left(a + \frac{b}{k}\right) p(k-1), \qquad k = 1,2,3,4,\mathrm{K} \qquad \text{(i)}$$

As distribuições a seguir são os únicos membros dessa família:

a) Poisson

$$p(k) = \frac{e^{-l}\, l^{k}}{k!}, \quad k = 0,1,2,\mathrm{K}$$
$$\frac{p(k)}{p(k-1)} = \frac{l}{k}, \quad p(0) = e^{-l} \qquad \text{(ii)}$$
$$a = 0, \qquad b = l$$

b) Binomial

$$p(k) = \binom{N}{k} p^{k} (1-p)^{N-k}, \qquad k = 0,1,2,\mathrm{K}\,, N$$
$$p(k) = 0, \qquad k = N+1, N+2, \mathrm{K}$$
$$\frac{p(k)}{p(k-1)} = \frac{(N-k+1)p}{[n(1-p)]}, \qquad p(0) = (1-p)^{N} \qquad \text{(iii)}$$
$$a = p, \qquad b = \frac{(N+1)p}{(1-p)}$$

c) Binomial Negativa

$$p(k) = \binom{a+k-1}{k} p^{k} (1-p)^{a}, \qquad k = 0,1,2,\mathrm{K}$$
$$\frac{p(k)}{p(k-1)} = \frac{(a+k-1)p}{k}, \qquad p(0) = (1-p)^{a} \qquad \text{(iv)}$$
$$a = p, \qquad b = (a-1)p$$

d) Geométrica

$$p(k) = (1-p)p^{k}, \quad k = 0,1,2,\mathrm{K}$$
$$\frac{p(k)}{p(k-1)} = p, \qquad p(0) = 1-p \qquad \text{(v)}$$
$$a = p, \qquad b = 0$$

D. Distribuições da Classe (a, b, 1)

A classe (a, b, 1) representa a família de distribuições de freqüência que satisfazem à recursão:

$$p(k) = \left(a + \frac{b}{k} \right) p(k-1) \quad k = 2,3,4,\ldots \qquad \text{(i)}$$

Seja p(k) = Pr (N = k) pertencente à classe (a, b, 0), $p^T(k)$ = Pr (N = k) pertencente a classe (a, b, 1) com $p^T(0) = 0$, e $p^M(k)$ = Pr (N = k) pertencente à classe (a, b, 1) com $p^M(0) > 0$.

As relações apresentadas abaixo são demonstradas em Klugman et al. (1998) [58]:

$$p^M(k) = \frac{1 - p^M(0)}{1 - p(0)} p(k) \quad k = 1,2,\ldots \qquad \text{(i)}$$

$$p^T(k) = \frac{1}{1 - p(0)} p(k) \quad k = 1,2,\ldots \qquad \text{(ii)}$$

E. Método dos Momentos Ponderados (PWM)

Seja Q(p) a função quantil, onde Q(p) = x é a função inversa da função F(x) = P(X = x) = 1-p.

O r^th momento ponderado teórico é:

$$w_r = \int_0^1 Q(p)\,p^r\,dp = \mathrm{E}\left[Q(p)\,p^r\right] \qquad \text{(i)}$$

, onde p tem uma distribuição uniforme sobre [0,1].

Para a estatística de ordem $x_{1:n}>x_{2:n}>...>x_{n:n}$ de uma amostra aleatória de tamanho n, o momento ponderado amostral é:

$$\hat{w}_r = \frac{1}{n}\sum_{j=1}^{n} x_{j:n}\,p_{j:n}^r \qquad \text{(ii)}$$

onde

$$p_{j:n} = \frac{n-j+0.5}{n} \quad ou \quad \frac{n-j+1}{n+1}\left(=\mathrm{E}\left[U_{j:n}\right]\right) \quad ou \quad u_{j:n} \qquad \text{(iii)}$$

$$p_{j:n}^r = \left(p_{j:n}\right)^r \quad ou \quad \mathrm{E}\left[\left(U_{j:n}\right)^r\right] \qquad \text{(iv)}$$

e $\{u_{j:n}\}$ são as estatísticas de ordem $u_{1:n}>u_{2:n}>...>u_{n:n}$ de uma amostra aleatória de tamanho n para uma distribuição Uniforme (0,1).

Logo, estimam-se os parâmetros da distribuição da mesma forma que no método dos momentos tradicional, sendo que neste método utilizam-se os momentos ponderados (teóricos e amostrais).

Observe que para r = 0, w_0 é igual a média amostral, w_0 é igual a esperança de X.

Como exemplo, o método é aplicado a GEV e a GPD, sendo ? < 1. Considere: $m_1 = 2w_1 - w_0$ e $m_2 = 3w_2 - w_0$.

GEV

$$x = 7.8590c - 2.9554c^2 \quad ,onde \quad c = \frac{\ln 2}{\ln 3} - \frac{m_1}{m_2}$$

$$s = \frac{m_1 x}{\left(2^x - 1\right)\Gamma\left(1-x\right)} \qquad \text{(v)}$$

$$m = w_0 + \frac{s}{x}\left\{1-\Gamma\left(1-x\right)\right\}$$

, onde G é a função gamma.

GPD

$$\begin{aligned} x &= 3-2\left(\frac{m_2}{m_1}-1\right)^{-1} \\ s &= m_1(2-x)(1-x) \\ m &= w_0-\frac{s}{1-x} \end{aligned} \tag{vi}$$

F. Listagem de Programas MatLab

dbimportLDA.m

```
function AA = dbimportLDA(strSql,dblTimeout)
%DBIMPORT Importa dados para o Matlab da conexao ODBC LDA.

%   Parametros:
%       strSql      - Query no banco
%       dblTimeout  - tempo de timeout em segundos

%   Version 1.0  26-05-2005
%   Author: Lyra Jr, Jose Luis

% Set maximum time allowed for establishing a connection.

timeoutA=logintimeout(dblTimeout);

% Connect to a database.

connA=database('LDA',",");

% Check the database status.

ping(connA);

% Open cursor and execute SQL statement.

cursorA=exec(connA,strSql);

% Fetch all rows of data.

cursorA=fetch(cursorA);

% Display the data.

AA=cursorA.Data;
```

```
% Close the cursor and the connection.

close(cursorA);
close(connA);
```

geraeventos.m

```
function geraeventos()
%GERAEVENTOS Gera eventos de perdas na Tabela Tbl_Evento.

%   Version 1.0  25-05-2005
%   Author(s): Lyra Junior, Jose Luis

% Quantidade de dias de eventos de perda
n = 500;

% Distribuicao de Frequencia
N = poissrnd(50,1,n);

for i = 1:n

    if N(1,i) == 0

        X = 0;

        Registros = 1;

    else

        % Distribuicao de Severidade
        X = lognrnd(8,2.2,N(1,i),1);

        Registros = N(1,i);

    end

    if i == 1
```

```
        Data=datestr(datenum('1/1/2002'),20);

        Perdas(1:Registros,1)={'Descricao Causa Evento'};
        Perdas(1:Registros,2)={'Descricao Impacto Evento'};
        Perdas(1:Registros,3)=num2cell(X(:,1));
        Perdas(1:Registros,4)={Data};
        Perdas(1:Registros,5)={Data};
        Perdas(1:Registros,6)=num2cell(1);
        Perdas(1:Registros,7)=num2cell(1);

    else

    Tamanho = size(Perdas);
    Inicio = Tamanho(1)+1;
    Fim = Inicio + Registros -1;

    Data=datestr(datenum('1/1/2002')+i-1,20);

    Perdas(Inicio:Fim,1)={'Descricao Causa Evento'};
    Perdas(Inicio:Fim,2)={'Descricao Impacto Evento'};
    Perdas(Inicio:Fim,3)=num2cell(X(:,1));
    Perdas(Inicio:Fim,4)={Data};
    Perdas(Inicio:Fim,5)={Data};
    Perdas(Inicio:Fim,6)=num2cell(1);
    Perdas(Inicio:Fim,7)=num2cell(1);

    end

end

% Connect to a database.
connA=database('LDA','','');

% Define a string array of column names in table where data will be exported.
```

```
colnames={'Dsc_Causa_Evento','Dsc_Impacto_Evento','Mo_Severidade_Evento'
,'Dt_Criacao_Evento','Dt_Atualizacao_Evento','Fk_Tipo_Efeito','Fk_Unidade_Ris
co'};

% Determine autocommit status.
get(connA, 'autocommit');

% Export data into Avg_Freight_Cost table.
insert(connA, 'Tbl_Evento', colnames, Perdas);

% Close the connection.
close(connA);
```

cell2num.m

```
function A = cell2num(B)

% Tamanho da Matriz B
S = size(B);

% Converte Cell Array em Numeric Array
for i=1:S(1)
        for j=1:S(2)
    C(i,j)=B{i,j};
        end;
end;

A = C;
```

binofit2.m

```
function [mhat, qhat] = binofit2(x)
%BINOFIT2 Parameter estimates and confidence intervals for BINO data.
%   BINOFIT2(X) Returns the estimate of the parameter of the BINO
%   distribution give the data X.
%
```

```
%   [MHAT, QHATY, MCI, QCI] = BINOFIT2(X,ALPHA) gives MLE and 100(1-ALPHA)
%   percent confidence intervals given the data. By default, the
%   optional parameter ALPHA = 0.05 corresponding to 95% confidence intervals.


if nargin < 2
   alpha = 0.05;
end


% Passo 1: m igual a maior observacao
m = max(x);

l = binolike(m,x);

lant = l;

lpos = Inf;

while  lpos >= lant

   m = m + 1;

   lant = l;

   % Calcula a funcao negativa do log da verossimilhanca
        l = binolike(m,x);

   lpos = l;

end

mhat = m -1;

% Calcular qhat
```

```
O = (0:max(x));
FO = histc(x,O);
N = sum(x);
qhat = (inv(mhat)*O*FO)/N;
```

binolike.m

```
function [logL,avar] = binolike(params,data)
%BINOLIKE Negative Binomial log-likelihood function.
%    L = BINOLIKE(PARAMS,DATA) returns the negative of the Binomial log-likelihood
%   function for the parameters PARAMS(1) = M and PARAMS(2) = Q, given
%   DATA. The length of the vector LOGL is the length of the vector DATA.
%
%   [LOGL, AVAR] = BINOLIKE(PARAMS,DATA) adds the inverse of Fisher's
%   information matrix, AVAR.  If the input parameter values in PARAMS
%   are the maximum likelihood estimates, the diagonal elements of AVAR
%   are their asymptotic variances.  AVAR is based on the observed
%   Fisher's information, not the expected information.
%
%   BINOLIKE is a utility function for maximum likelihood estimation.
%   See also BETALIKE, GAMLIKE, MLE, NORMFIT, WEIBLIKE.



if nargin < 2,
    error('Requires at least two input arguments');
end

[n, m] = size(data);

if nargout == 2 & max(m,n) == 1
    error('To compute the 2nd output, the 2nd input must have at least two elements.');
end

if n == 1
   data = data';
```

```
  n = m;
end

m = params(1);

% Calcular q
O = (0:max(data));
FO = histc(data,O);
N = sum(data);
q = (inv(m)*O*FO)/N;

m = m(ones(n,1),:);
q = q(ones(n,1),:);

x = binopdf(data,m,q);

logL = -sum(log(x));
```

nbinfit.m

```
function [rhat, betahat] = nbinfit(x)
%NBINFIT Parameter estimates and confidence intervals for NBIN data.
%   NBINFIT(X) Returns the estimate of the parameter of the NBIN
%   distribution give the data X.
%
%   [RHAT, BETAHAT, RCI, BETACI] = NBIN(X,ALPHA) gives MOM and 100(1-
ALPHA)
%   percent confidence intervals given the data. By default, the
%   optional parameter ALPHA = 0.05 corresponding to 95% confidence
intervals.


if nargin < 2
   alpha = 0.05;
end
```

```
% Resolucao atraves de metodo dos momentos
rhat = ((mean(x))^2)/(var(x)-mean(x));
betahat = (var(x)-mean(x))/(mean(x));
```

VatRisk.m

```
function v = VatRisk(m,n,h)
        % Calcula o VaR.

   c = cumsum(m); % Soma progressiva
        q = find(c > n); % Encontra o primeiro indice cuja a probabilidade acumulada eh maior que o Nivel n
        v = (q(1,1)-1)*h+1; % Converte indice em perda operacional
```

medida.m

```
function V = medida(Id_Modelo_Medida)

% Query dos Parametros Entrada de Dados
strPar = ['SELECT Fk_Unidade_Risco, Fk_Agregacao_Frequencia, Fk_Frequencia_Distribuicao, Fk_Severidade_Distribuicao, Fk_Metodo_LDA, Db_Param1_Modelo_LDA_Modelo_Medida, Db_Param2_Modelo_LDA_Modelo_Medida, Db_Param3_Modelo_LDA_Modelo_Medida, Db_Param4_Modelo_LDA_Modelo_Medida, Db_Param5_Modelo_LDA_Modelo_Medida, Nm_Modelo_Medida FROM Tbl_Modelo_Medida WHERE Id_Modelo_Medida=',int2str(Id_Modelo_Medida)];

% Parametros do Modelo de Medida
Parametros = dbimportLDA(strPar,5);

FkUnidRisco=Parametros{1,1};
FkAgregFreq=Parametros{1,2};
FkFreqDist=Parametros{1,3};
FkSevDist=Parametros{1,4};
FkModeloLDA=Parametros{1,5};
DbParam1=Parametros{1,6};
```

```
DbParam2=Parametros{1,7};
DbParam3=Parametros{1,8};
DbParam4=Parametros{1,9};
DbParam5=Parametros{1,10};
NmModeloMedida=Parametros{1,11};


% Query de Severidade para uma Unidade de Risco
strSev = ['SELECT Mo_Severidade_Evento FROM Tbl_Evento WHERE Fk_Unidade_Risco=',int2str(FkUnidRisco)];

% Selecao da Query de Frequencia para uma Unidade de Risco e para um tipo de Agregacao Frequencia
 switch FkAgregFreq
    case 1 %Agregacao Frequencia Diaria
        strFreq = ['SELECT Count(Id_Evento) AS Frequencia FROM Tbl_Evento WHERE Fk_Unidade_Risco=',int2str(FkUnidRisco),' GROUP BY Dt_Atualizacao_Evento'];
    case 2 %Agregacao Frequencia Mensal
         strFreq = ['SELECT Count(Id_Evento) AS Frequencia FROM Tbl_Evento WHERE Fk_Unidade_Risco=',int2str(FkUnidRisco),' GROUP BY DateSerial(Year([Dt_Atualizacao_Evento]),Month([Dt_Atualizacao_Evento]),1)'];
    case 3 %Agregacao Frequencia Anual
         strFreq = ['SELECT Count(Id_Evento) AS Frequencia FROM Tbl_Evento WHERE Fk_Unidade_Risco=',int2str(FkUnidRisco),' GROUP BY DateSerial(Year([Dt_Atualizacao_Evento]),1,1)'];
    otherwise
       error('Tipo de Agregacao inexistente!')
 end

 % Severidade observada
 X = dbimportLDA(strSev,5);

 % Frequencia observada
 N = dbimportLDA(strFreq,5);

 % Converte Cell Array Severidade observada em Numeric Array
```

```
 X = cell2num(X);

  % Converte Cell Array Frequencia observada em Numeric Array
 N = cell2num(N);


% Selecao do Modelo LDA que nao dependem da Dist de Fre e Sev
switch FkModeloLDA

        case 1 % Metodo Aproximacao Normal

        % Estimacao dos Parametros
        mediaS = mean(X)*mean(N);
        varS = mean(N)*var(X)+var(N)*((mean(X))^2);

        muhat = mediaS;
        sigmahat = sqrt(varS);

        % Cria uma instancia de figura
        h=figure('Name','Distribuicao de Perdas Operacionais','Visible','off');

   % Distribuicao de Perdas Operacionais Ajustada
   x = 0:100:norminv(0.99999,muhat,sigmahat);
   y = normpdf(x,muhat,sigmahat);
   plot(x,y);
   str(1) = {['VaR95.0 = ',num2str(norminv(0.95,muhat,sigmahat),'%6.2f')]};
   str(2) = {['VaR97.5 = ',num2str(norminv(0.975,muhat,sigmahat),'%6.2f')]};
   str(3) = {['VaR99.0 = ',num2str(norminv(0.99,muhat,sigmahat),'%6.2f')]};
   str(4) = {['VaR99.9 = ',num2str(norminv(0.999,muhat,sigmahat),'%6.2f')]};
   text(max(x),max(y),str,'HorizontalAlignment','right','VerticalAlignment','top');
   title(['Aproximacao Normal : \mu = ',num2str(muhat,'%6.2f'),' e \sigma =
',num2str(sigmahat,'%6.2f')]);
        xlabel('Perdas Operacionais'); ylabel('Probabilidade');

        saveas(h,['c:\sigrop\relatorios\',strrep(datestr(now,26),'/',''),strrep(datestr(n
ow,13),':',''),'_Modelo_Medida_',NmModeloMedida,'.tif']);
```

```
        case 2 % Metodo Aproximacao Log Normal

        % Estimacao dos Parametros
        mediaS = mean(X)*mean(N);
        varS = mean(N)*var(X)+var(N)*((mean(X))^2);
        expS2 = varS + ((mediaS)^2);

        muhat = (4*log(mediaS)-log(expS2))/2;
        sigmahat = sqrt(log(expS2)-2*log(mediaS));

        % Cria uma instancia de figura
        h=figure('Name','Distribuicao de Perdas Operacionais','Visible','off');

        % Distribuicao de Perdas Operacionais Ajustada
        x = 0:0.1:log(logninv(0.99999999,muhat,sigmahat));
        y = lognpdf(exp(x),muhat,sigmahat);
        plot(x,y);
        str(1) = {['VaR95.0 = ',num2str(logninv(0.95,muhat,sigmahat),'%6.2f')]};
        str(2) = {['VaR97.5 = ',num2str(logninv(0.975,muhat,sigmahat),'%6.2f')]};
        str(3) = {['VaR99.0 = ',num2str(logninv(0.99,muhat,sigmahat),'%6.2f')]};
        str(4) = {['VaR99.9 = ',num2str(logninv(0.999,muhat,sigmahat),'%6.2f')]};
        text(max(x),max(y),str,'HorizontalAlignment','right','VerticalAlignment','top')
;
        title(['Aproximacao Log Normal : \mu = ',num2str(muhat,'%6.2f'),' e \sigma
= ',num2str(sigmahat,'%6.2f')]);
        xlabel('Perdas Operacionais (escala logaritmica)'); ylabel('Probabilidade');

        saveas(h,['c:\sigrop\relatorios\',strrep(datestr(now,26),'/',''),strrep(datestr(n
ow,13),':',''),'_Modelo_Medida_',NmModeloMedida,'.tif']);

    otherwise % Distribuicao de Fre e Sev

    % Selecao da Distribuicao de Severidade a ser ajustada
    switch FkSevDist
      case 5 %Log Normal

        % Estimacao dos Parametros
```

PUC-Rio - Certificação Digital Nº 0310428/CA

```
[muhat,sigmahat,muci,sigmaci] = normfit(log(X(find(X~=0),1)));

% Cria uma instancia de figura
h = figure('Name','Distribuicao de Severidade Ajustada','Visible','off');

% Subplot da Distribuicao Ajustada
x = 0:0.1:log(logninv(0.99999999,muhat,sigmahat));
y = lognpdf(exp(x),muhat,sigmahat);
subplot(2,1,1);
plot(x,y);
title(['Log Normal : \mu = ',num2str(muhat),' e \sigma = ',num2str(sigmahat)]);
xlabel('Severidade (escala logaritmica)'); ylabel('Probabilidade');

% Subplot QQ-Plot (Distribuicao Ajustada x Dados)
p = 0:0.001:1;
y = logninv(p,muhat,sigmahat);
subplot(2,1,2);
qqplot(X(find(X~=0),1),y);

saveas(h,['c:\sigrop\relatorios\',strrep(datestr(now,26),'/',''),strrep(datestr(now,13),':',''),'_Modelo_Medida_',NmModeloMedida,'_Severidade.tif']);

case 6 %GEV

% Estimacao dos Parametros com base no dados X e no blocksize DbParam3
out = gev(X,DbParam3);

xihat = out.par_ests(1);
sigmahat = out.par_ests(2);
muhat = out.par_ests(3);


% Cria uma instancia de figura
h = figure('Name','Distribuicao de Severidade Ajustada','Visible','off');
```

```
        % Subplot da Distribuicao Acumulada Ajustada
        x = 0.001:0.001:1;
        y = qgev(x,xihat,muhat,sigmahat);
        subplot(2,1,1);
        plot(y,x);
        title(['GEV Acumulada: \xi = ',num2str(xihat),' e \mu = ',num2str(muhat),' e
\sigma = ',num2str(sigmahat)]);
        xlabel('Severidade'); ylabel('Probabilidade');

        % Subplot QQ-Plot (Distribuicao Ajustada x Dados)
        subplot(2,1,2);
        qqplot(X,y);



        saveas(h,['c:\sigrop\relatorios\',strrep(datestr(now,26),'/',''),strrep(datestr(n
ow,13),':',''),'_Modelo_Medida_',NmModeloMedida,'_Severidade.tif']);

      otherwise
        error('Distribuicao da Severidade inexistente!')
    end



    % Selecao da Distribuicao de Frequencia a ser ajustada
    switch FkFreqDist
      case 1 %Poisson

        % Estimacao dos Parametros
        [lambdahat,lambdaci] = poissfit(N);

        % Cria uma instancia de figura
        h = figure('Name','Distribuicao da Frequencia Ajustada','Visible','off');

        % Subplot da Distribuicao Ajustada
        x = 0:(poissinv(0.99999999,lambdahat));
        y = poisspdf(x,lambdahat);
        subplot(2,1,1);
        stem(y,'fill','-');
```

```
title(['Poisson : \lambda = ',num2str(lambdahat)]);
xlabel('Frequencia'); ylabel('Probabilidade');

% Subplot QQ-Plot (Distribuicao Ajustada x Dados)
p = 0:0.001:1;
y = poissinv(p,lambdahat);
subplot(2,1,2);
qqplot(N,y);



saveas(h,['c:\sigrop\relatorios\',strrep(datestr(now,26),'/',''),strrep(datestr(now,13),':',''),'_Modelo_Medida_',NmModeloMedida,'_Frequencia.tif']);

case 2 %Binomial Negativa

% Estimacao dos Parametros
[rhat, betahat] = nbinfit(N);

% Cria uma instancia de figura
h = figure('Name','Distribuicao da Frequencia Ajustada','Visible','off');

% Subplot da Distribuicao Ajustada
x = 0:(nbininv(0.999,rhat,inv(1+betahat)));
y = nbinpdf(x,rhat,inv(1+betahat));
subplot(2,1,1);
stem(y,'fill','-');
title(['Binomial Negativa : r = ',num2str(rhat),' e \beta = ',num2str(betahat)]);
xlabel('Frequencia'); ylabel('Probabilidade');

% Subplot QQ-Plot (Distribuicao Ajustada x Dados)
p = 0.001:0.001:0.999;
y = nbininv(p,rhat,inv(1+betahat));
subplot(2,1,2);
qqplot(N,y);
```

```
        saveas(h,['c:\sigrop\relatorios\',strrep(datestr(now,26),'/',''),strrep(datestr(now,13),':',''),'_Modelo_Medida_',NmModeloMedida,'_Frequencia.tif']);

    case 4 %Binomial

        % Estimacao dos Parametros
        [mhat, qhat] = binofit2(N);

        % Cria uma instancia de figura
        h = figure('Name','Distribuicao da Frequencia Ajustada','Visible','off');

        % Subplot da Distribuicao Ajustada
        x = 0:(binoinv(0.99999999,mhat,qhat));
        y = binopdf(x,mhat,qhat);
        subplot(2,1,1);
        stem(y,'fill','-');
        title(['Binomial : m = ',num2str(mhat),' e q = ',num2str(qhat)]);
        xlabel('Frequencia'); ylabel('Probabilidade');

        % Subplot QQ-Plot (Distribuicao Ajustada x Dados)
        p = 0:0.001:1;
        y = binoinv(p,mhat,qhat);
        subplot(2,1,2);
        qqplot(N,y);


        saveas(h,['c:\sigrop\relatorios\',strrep(datestr(now,26),'/',''),strrep(datestr(now,13),':',''),'_Modelo_Medida_',NmModeloMedida,'_Frequencia.tif']);

    otherwise
      error('Distribuicao de Frequencia inexistente!')
  end

 end

if (FkModeloLDA ~= 1)&(FkModeloLDA ~= 2)
```

```
% Selecao do Modelo LDA que dependem da Dist de Fre e Sev
switch FkModeloLDA

case 3 %Simulacao

% Selecao da Distribuicao de Frequencia a ser ajustada
switch FkFreqDist

    case 1 % Poisson
    R = poissrnd(lambdahat,1,DbParam1);

    case 2 % Binomial Negativa
    R = nbinrnd(rhat,inv(1+betahat),1,DbParam1);

    case 4 % Binomial
    R = binornd(mhat,qhat,1,DbParam1);

end

for i = 1:DbParam1

% Selecao da Distribuicao de Severidade a
ser ajustada
switch FkSevDist

    case 5 % Log Normal
A = lognrnd(muhat,sigmahat,1,R(1,i));

    case 6 % GEV
A = rgev(R(1,i),xihat,muhat,sigmahat);

end

S(1,i)=sum(A);
```

```
end

% Cria uma instancia de figura
h = figure('Name','Distribuicao de Perdas Operacionais','Visible','off');

% Distribuicao Acumulada de Perdas Operacionais Ajustada
y = 0:1:100;
x = prctile(S,y);

% Distribuicao de Probabilidade de Perdas Operacionais Ajustada
str(1) = {['VaR95.0 = ',num2str(log(prctile(S,95)),'%6.2f')]};
str(2) = {['VaR97.5 = ',num2str(log(prctile(S,97.5)),'%6.2f')]};
str(3) = {['VaR99.0 = ',num2str(log(prctile(S,99)),'%6.2f')]};
str(4) = {['VaR99.9 = ',num2str(log(prctile(S,99.9)),'%6.2f')]};
plot(log(x),y);

text(max(log(x)),min(y),str,'HorizontalAlignment','right','VerticalAlignment','bottom');
title(['Simulacao com ',num2str(DbParam1),' cenarios']);
ylabel('Probabilidade (%)');
xlabel('Perdas Operacionais (escala logaritmica)');


saveas(h,['c:\sigrop\relatorios\',strrep(datestr(now,26),'/',''),strrep(datestr(now,13),':',''),'_Modelo_Medida_',NmModeloMedida,'_LDA.tif']);

case 4 %Inversao FFT

% Encontra a quantidade de pontos, divindo o maximo (DbParam2) pelo passo da discretizacao (DbParam1)
Pontos = DbParam2/DbParam1;

% Quantidade de Pontos na proxima Potencia de 2
T = 2^nextpow2(Pontos);
```

```
% Discretiza Severidade Ajustada pelo Metodo de
Arredondamento
switch FkSevDist

    case 5 % Log Normal

x(1,1)=logncdf(0.5*DbParam1,muhat,sigmahat);
j = 1:(T);
x(1,j)=  logncdf((j+0.5)*DbParam1,muhat,sigmahat)-
logncdf((j-0.5)*DbParam1,muhat,sigmahat);

    case 6 % GEV

x(1,1)=pgev(0.5*DbParam1,xihat,muhat,sigmahat);
j = 1:(T);
x(1,j)=pgev((j+0.5)*DbParam1,xihat,muhat,sigmahat)-
pgev((j-0.5)*DbParam1,xihat,muhat,sigmahat);

end

% Garante a soma das probabilidaes discretizadas da 1
x(1,T)=1-sum(x);

% Calcula funcao caracteristica
y = fft(x,T);

% Aplica a equacao 16 da dissertacao (fgp da Frequencia)
% encontrando a funcao caracteristica da distribuicao agregada
switch FkFreqDist

    case 1 % Poisson
        w = exp(lambdahat*(y-1));

    case 2 % Binomial Negativa
        w = (1 - betahat*(y-1))^(-rhat);

    case 4 % Binomial
```

```
                              w = (1 + qhat*(y-1))^(mhat);

                    end

          % Inverte a funcao caracteristica da ditribuicao agregada
                    g = real(ifft(w));

        % Cria uma instancia de figura
                    h = figure('Name','Distribuicao de Perdas
Operacionais','Visible','off');

          % Monta o grafico na escala logaritmica
          j = 1:T;
          plot(log((j-1)*DbParam1+1),g(1,j));

str(1) = {['VaR95.0 = ',num2str(log(VatRisk(g,0.95,DbParam1)),'%6.2f')]};
str(2) = {['VaR97.5 = ',num2str(log(VatRisk(g,0.975,DbParam1)),'%6.2f')]};
str(3) = {['VaR99.0 = ',num2str(log(VatRisk(g,0.99,DbParam1)),'%6.2f')]};
str(4) = {['VaR99.9 = ',num2str(log(VatRisk(g,0.999,DbParam1)),'%6.2f')]};
text(max(log((j-1)*DbParam1+1)),max(g(1,j)),str,'HorizontalAlignment','right',
'VerticalAlignment','top');
title('Invsersao FFT');
xlabel('Perdas Operacionais (escala logaritmica)'); ylabel('Probabilidade');

        saveas(h,['c:\sigrop\relatorios\',strrep(datestr(now,26),'/',''),strrep(datestr(n
ow,13),':',''),'_Modelo_Medida_',NmModeloMedida,'_LDA.tif']);

     otherwise
       error('Modelo LDA inexistente!')

        end

end
```

causal.m

```
function causal(Id_Modelo_Causal)
```

```
% Query dos Parametros das Variaveis Dependentes
strParVD = ['SELECT Nm_Modelo_Causal, Fk_Unidade_Risco, Fk_Agregacao_Frequencia, Fk_Frequencia_Distribuicao, Fk_Severidade_Distribuicao FROM Tbl_Modelo_Causal WHERE Id_Modelo_Causal=',int2str(Id_Modelo_Causal)];

% Parametros das Variaveis Dependentes
ParametrosVD = dbimportLDA(strParVD,5);

NmModeloCausal=ParametrosVD{1,1};
FkUnidRisco=ParametrosVD{1,2};
FkAgregFreq=ParametrosVD{1,3};
FkFreqDist=ParametrosVD{1,4};
FkSevDist=ParametrosVD{1,5};

%Query dos Parametros das Variaveis Explicativas
strParVE = ['SELECT Tbl_Variavel.Id_Variavel, Tbl_Variavel.Nm_Variavel, Tbl_Variavel_Modelo_Causal.Fk_Distribuicao, Tbl_Variavel_Modelo_Causal.Fk_Agregacao_Variavel, Tbl_Variavel_Modelo_Causal.Db_Valor_Modelo_Causal FROM Tbl_Variavel INNER JOIN (Tbl_Modelo_Causal INNER JOIN Tbl_Variavel_Modelo_Causal ON Tbl_Modelo_Causal.Id_Modelo_Causal = Tbl_Variavel_Modelo_Causal.Fk_Modelo_Causal) ON Tbl_Variavel.Id_Variavel = Tbl_Variavel_Modelo_Causal.Fk_Variavel WHERE Tbl_Variavel_Modelo_Causal.Fk_Modelo_Causal=',int2str(Id_Modelo_Causal)];

% Parametros das Variaveis Explicativas
ParametrosVE = dbimportLDA(strParVE,5);

IdVar=ParametrosVE(:,1);
NmVar=ParametrosVE(:,2);
FkVarDist=ParametrosVE(:,3);
FkVarAgreg=ParametrosVE(:,4);
DbValor=ParametrosVE(:,5);

% Converte Cell Array IdVar em Numeric Array
```

```
IdVar = cell2num(IdVar);

% Converte Cell Array FkVarDist em Numeric Array
FkVarDist = cell2num(FkVarDist);

% Converte Cell Array FkVarAgreg em Numeric Array
FkVarAgreg = cell2num(FkVarAgreg);

% Converte Cell Array DbValor em Numeric Array
DbValor = cell2num(DbValor);

% Monta Array com os Valores das Variaveis Dependentes
for i=1:max(size(IdVar))

strVarValor = ['SELECT Db_Valor_Variavel FROM Vw_Evento_Valor WHERE
Fk_Unidade_Risco=',int2str(FkUnidRisco),' AND
Fk_Variavel=',int2str(IdVar(i,1))];

   % Parametros das Valor das Variaveis Explicativas
        ValorVE(:,i) = dbimportLDA(strVarValor,5);

end

% Converte Cell Array ValorVE em Numeric Array
ValorVE = cell2num(ValorVE);

% Query de Severidade para uma Unidade de Risco
strSev = ['SELECT Mo_Severidade_Evento FROM Tbl_Evento WHERE
Fk_Unidade_Risco=',int2str(FkUnidRisco)];

% Severidade observada
X = dbimportLDA(strSev,5);

% Converte Cell Array Severidade observada em Numeric Array
X = cell2num(X);

% Inclui Coluna de 1s para o Modelo de Regressao
```

```
Y = [ones(size(X)) ValorVE];

% Regressao Multipla pelo Metodo dos Minimos Quadrados
[b,bint,r,rint,stats] = regress(X,Y);

% Estimacao dos Parametros das Variaveis e Preparacao para Test de Estresse
pcres = 0:0.0001:0.9999;
pdecres = 1-pcres;

for i=1:max(size(IdVar))

  V = ValorVE(:,i);

      switch FkVarDist(i)

    case 1 %Poisson

      % Estimacao dos Parametros
      [lambdahat,lambdaci] = poissfit(V);

      % Cria uma instancia de figura
      h = figure('Name','Distribuicao Ajustada','Visible','off');

      % Subplot da Distribuicao Ajustada
      x = 0:(poissinv(0.99999999,lambdahat));
      y = poisspdf(x,lambdahat);
      subplot(2,1,1);
      stem(y,'fill','-');
      title(['Poisson: \lambda = ',num2str(lambdahat)]);
      xlabel(NmVar{i}); ylabel('Probabilidade');

      % Subplot QQ-Plot (Distribuicao Ajustada x Dados)
      p = 0:0.001:1;
      y = poissinv(p,lambdahat);
      subplot(2,1,2);
      qqplot(V,y);
```

```
saveas(h,['c:\sigrop\relatorios\',strrep(datestr(now,26),'/',''),strrep(datestr(now,13),':',''),'_Modelo_Causal_',NmModeloCausal,'_Variavel_',NmVar{i},'.tif']);

%Monta Array Para Stress Test
if b(i+1)>0
   ValorStress(:,i) = poissinv(pdecres,lambdahat)';
else
   ValorStress(:,i) = poissinv(pcres,lambdahat)';
end

case 2 %Binomial Negativa

% Estimacao dos Parametros
[rhat, betahat] = nbinfit(V);

% Cria uma instancia de figura
h = figure('Name','Distribuicao Ajustada','Visible','off');

% Subplot da Distribuicao Ajustada
x = 0:(nbininv(0.999,rhat,inv(1+betahat)));
y = nbinpdf(x,rhat,inv(1+betahat));
subplot(2,1,1);
stem(y,'fill','-');
title(['Binomial Negativa : r = ',num2str(rhat),' e \beta = ',num2str(betahat)]);
xlabel(NmVar{i}); ylabel('Probabilidade');

% Subplot QQ-Plot (Distribuicao Ajustada x Dados)
p = 0.001:0.001:0.999;
y = nbininv(p,rhat,inv(1+betahat));
subplot(2,1,2);
qqplot(V,y);


saveas(h,['c:\sigrop\relatorios\',strrep(datestr(now,26),'/',''),strrep(datestr(now,13),':',''),'_Modelo_Causal_',NmModeloCausal,'_Variavel_',NmVar{i},'.tif']);
```

```
        %Monta Array Para Stress Test
        if b(i+1)>0
            ValorStress(:,i) = nbininv(pdecres,rhat,inv(1+betahat))';
        else
            ValorStress(:,i) = nbininv(pcres,rhat,inv(1+betahat))';
        end

    case 4 %Binomial

        % Estimacao dos Parametros
        [mhat, qhat] = binofit2(V);

        % Cria uma instancia de figura
        h = figure('Name','Distribuicao Ajustada','Visible','off');

        % Subplot da Distribuicao Ajustada
        x = 0:(binoinv(0.99999999,mhat,qhat));
        y = binopdf(x,mhat,qhat);
        subplot(2,1,1);
        stem(y,'fill','-');
        title(['Binomial : m = ',num2str(mhat),' e q = ',num2str(qhat)]);
        xlabel(NmVar{i}); ylabel('Probabilidade');

        % Subplot QQ-Plot (Distribuicao Ajustada x Dados)
        p = 0:0.001:1;
        y = binoinv(p,mhat,qhat);
        subplot(2,1,2);
        qqplot(V,y);


        saveas(h,['c:\sigrop\relatorios\',strrep(datestr(now,26),'/',''),strrep(datestr(n
ow,13),':',''),'_Modelo_Causal_',NmModeloCausal,'_Variavel_',NmVar{i},'.tif']);

        %Monta Array Para Stress Test
        if b(i+1)>0
            ValorStress(:,i) = binoinv(pdecres,mhat,qhat)';
```

```
else
   ValorStress(:,i) = binoinv(pcres,mhat,qhat)';
end


case 5 %Log Normal

   % Estimacao dos Parametros
   [muhat,sigmahat,muci,sigmaci] = normfit(log(V(find(V~=0),1)));

   % Cria uma instancia de figura
   h = figure('Name','Distribuicao Ajustada','Visible','off');

   % Subplot da Distribuicao Ajustada
   x = 0:0.1:log(logninv(0.99999999,muhat,sigmahat));
   y = lognpdf(exp(x),muhat,sigmahat);
   subplot(2,1,1);
   plot(x,y);
   title(['Log Normal : \mu = ',num2str(muhat),' e \sigma = ',num2str(sigmahat)]);
   xlabel([NmVar{i},' (escala logaritmica)']); ylabel('Probabilidade');

   % Subplot QQ-Plot (Distribuicao Ajustada x Dados)
   p = 0:0.001:1;
   y = logninv(p,muhat,sigmahat);
   subplot(2,1,2);
   qqplot(V(find(V~=0),1),y);


   saveas(h,['c:\sigrop\relatorios\',strrep(datestr(now,26),'/',''),strrep(datestr(now,13),':',''),'_Modelo_Causal_',NmModeloCausal,'_Variavel_',NmVar{i},'.tif']);

   %Monta Array Para Stress Test
   if b(i+1)>0
      ValorStress(:,i) = logninv(pdecres,muhat,sigmahat)';
   else
      ValorStress(:,i) = logninv(pcres,muhat,sigmahat)';
```

```
        end

    case 6 %GEV

         % Estimacao dos Parametros com base no dados X e no blocksize
DbParam3
        out = gev(V,DbParam3);

        xihat = out.par_ests(1);
        sigmahat = out.par_ests(2);
        muhat = out.par_ests(3);

        % Cria uma instancia de figura
        h = figure('Name','Distribuicao Ajustada','Visible','off');

        % Subplot da Distribuicao Acumulada Ajustada
        x = 0.001:0.001:1;
        y = qgev(x,xihat,muhat,sigmahat);
        subplot(2,1,1);
        plot(y,x);
        title(['GEV Acumulada: \xi = ',num2str(xihat),' e \mu = ',num2str(muhat),' e
\sigma = ',num2str(sigmahat)]);
        xlabel(NmVar{i}); ylabel('Probabilidade');

        % Subplot QQ-Plot (Distribuicao Ajustada x Dados)
        subplot(2,1,2);
        qqplot(V,y);

        saveas(h,['c:\sigrop\relatorios\',strrep(datestr(now,26),'/',''),strrep(datestr(n
ow,13),':',''),'_Modelo_Causal_',NmModeloCausal,'_Variavel_',NmVar{i},'.tif']);

        %Monta Array Para Stress Test
        if b(i+1)>0
           ValorStress(:,i) = qgev(pdecres,xihat,muhat,sigmahat)';
```

```
        else
           ValorStress(:,i) = qgev(pcres,xihat,muhat,sigmahat)';
        end


    case 7 %Normal

        % Estimacao dos Parametros
        [muhat,sigmahat,muci,sigmaci] = normfit(V);

        % Cria uma instancia de figura
        h = figure('Name','Distribuicao Ajustada','Visible','off');

        % Subplot da Distribuicao Ajustada
        x = 0:0.01:norminv(0.99999999,muhat,sigmahat);
        y = normpdf(x,muhat,sigmahat);
        subplot(2,1,1);
        plot(x,y);
        title(['Normal : \mu = ',num2str(muhat),' e \sigma = ',num2str(sigmahat)]);
        xlabel(NmVar{i}); ylabel('Probabilidade');

        % Subplot QQ-Plot (Distribuicao Ajustada x Dados)
        p = 0:0.001:1;
        y = norminv(p,muhat,sigmahat);
        subplot(2,1,2);
        qqplot(V,y);


        saveas(h,['c:\sigrop\relatorios\',strrep(datestr(now,26),'/',''),strrep(datestr(n
ow,13),':',''),'_Modelo_Causal_',NmModeloCausal,'_Variavel_',NmVar{i},'.tif']);

        %Monta Array Para Stress Test
        if b(i+1)>0
           ValorStress(:,i) = norminv(pdecres,muhat,sigmahat)';
        else
           ValorStress(:,i) = norminv(pcres,muhat,sigmahat)';
        end
```

```
case 8 %Beta

    % Estimacao dos Parametros
    p = betafit(V);
    alfa = p(1);
    beta = p(2);

    % Cria uma instancia de figura
    h = figure('Name','Distribuicao Ajustada','Visible','off');

    % Subplot da Distribuicao Ajustada
    x = 0:0.01:betainv(0.99999999,alfa,beta);
    y = betapdf(x,alfa,beta);
    subplot(2,1,1);
    plot(x,y);
    title(['Beta : \alpha = ',num2str(alfa),' e \beta = ',num2str(beta)]);
    xlabel(NmVar{i}); ylabel('Probabilidade');

    % Subplot QQ-Plot (Distribuicao Ajustada x Dados)
    p = 0:0.001:1;
    y = betainv(p,alfa,beta);
    subplot(2,1,2);
    qqplot(V,y);


    saveas(h,['c:\sigrop\relatorios\',strrep(datestr(now,26),'/',''),strrep(datestr(n
ow,13),':',''),'_Modelo_Causal_',NmModeloCausal,'_Variavel_',NmVar{i},'.tif']);

    %Monta Array Para Stress Test
    if b(i+1)>0
       ValorStress(:,i) = betainv(pdecres,alfa,beta)';
    else
       ValorStress(:,i) = betainv(pcres,alfa,beta)';
    end
```

```
  otherwise
    error('Distribuicao inexistente!')
 end

end

% Perdas sob Estresse
Y = [ones(size(pcres')) ValorStress];
PerdaStress = Y*b;

% Cria uma instancia de figura
h = figure('Name','Distribuicao de Perdas Operacionais','Visible','off');

% Distribuicao Acumulada de Perdas Operacionais Ajustada
x = 0:0.1:100;
y = prctile(PerdaStress,x);

% Distribuicao de Probabilidade de Perdas Operacionais Ajustada
str(1) = {['VaR95.0 = ',num2str(prctile(PerdaStress,95),'%6.2f')]};
str(2) = {['VaR97.5 = ',num2str(prctile(PerdaStress,97.5),'%6.2f')]};
str(3) = {['VaR99.0 = ',num2str(prctile(PerdaStress,99),'%6.2f')]};
str(4) = {['VaR99.9 = ',num2str(prctile(PerdaStress,99.9),'%6.2f')]};
plot(x,y);
text(min(x),max(y),str,'HorizontalAlignment','left','VerticalAlignment','top');
title('Perdas Operacionais sob Estresse');
xlabel('Probabilidade (%)'); ylabel('Perdas Operacionais');

% Monta Equacao da Regressao com R2
str2(1) = {['Perda Operacional = ',num2str(b(1),'%6.2f')]};
for i=1:max(size(IdVar))
   str2(i+1) = {[num2str(b(i+1),'%+6.2f'),'*',NmVar{i}]};
end
str2(max(size(IdVar))+2)={['+ Erro']};
str2(max(size(IdVar))+3)={['R2 = ',num2str(stats(1)*100,'%6.2f'),'%']};
text(max(x),min(y),str2,'HorizontalAlignment','right','VerticalAlignment','bottom');

% Simulacao de Cenario
```

```
str3(1) = {'Simulacao de Cenario:'};
for i=1:max(size(IdVar))
   str3(i+1) = {[NmVar{i},'=',num2str(DbValor(i),'%6.2f')]};
end
% Valor da Perda
Y = [1 DbValor'];
ValorPerda = Y*b;
str3(max(size(IdVar))+2)={['Perda=',num2str(ValorPerda,'%6.2f')]};
text((max(x)+
min(x))/2,max(y),str3,'HorizontalAlignment','left','VerticalAlignment','top');

saveas(h,['c:\sigrop\relatorios\',strrep(datestr(now,26),'/',''),strrep(datestr(now,13),'
:',''),'_Modelo_Causal_',NmModeloCausal,'.tif']);
```