Arithmetization of distributions and linear goal programming

José L. Vilar

Departamento de Economía Financiera y Actuarial, Universidad Complutense de Madrid, Facultad de Ciencias Económicas y Empresariales, Pabellón de 5 curso, Campus de Somosaguas, Pozuelo de Alarcón, Madrid 28223, Spain

Received 1 September 1998; received in revised form 1 December 1999; accepted 20 January 2000

Abstract

Linear goal programming can be used as a complementary technique when local moment matching method up to the second moment gives some negative mass. This could happen when manipulating a discrete or mixed type severity distribution. In that case we can avoid a simple retreat to the first moment and look for an arithmetic distribution with equal mean and the second moment closest to that of the original distribution. © 2000 Elsevier Science B.V. All rights reserved.

Keywords: Arithmetic distribution; Local moment matching; Linear goal programming; Simplex method

1. Introduction

Arithmetization of probability distribution functions is often required when the user arrives to the numerical stage. This is specially the case when using Panjer’s algorithm (Panjer and Willmot, 1992, Corollary 6.6.1) in the numerical evaluation of compound distributions with nonarithmetic severities, the latter being discrete with an irregular span, absolutely continuous or mixed types (Panjer and Willmot, 1992, p. 223). This numerical evaluation can be handled in the resolution of many problems as for example: approximating the probabilities of the distribution of total claims or some ruin probability (in both the cases of classical continuous-time with infinite horizon and fixed-time ruin probabilities), and finally, in the search of bounds or approximations of stop-loss transforms. Examples of these applications are found for instance in Gerber (1982), Gerber and Jones (1976), Panjer and Lutek (1983), Panjer and Willmot (1992), and finally De Vylder (1996) where arithmetization is studied inside the topic of optimization.

Given a distribution function, there are many ways to look for an arithmetic equivalent. If this equivalence depends on the equality (up to some order \( n \in \{0, 1, 2, \ldots\} \)) between the moments of the former and the latter, the available techniques are grouped under the denomination of local moment matching methods. In this paper we focus on the major drawback of these methods, namely the possibility of obtaining negative masses when “arithmetizing” some discrete or mixed type distribution for a given span \( h > 0 \), and we propose a complementary approach based on the nearness of local moments. The method of the nearest second moment is built in terms of linear programming (more precisely in terms of linear goal programming). Our objective is to show that this technique may furnish more confident solutions than the ones produced by local moment matching.
2. Local moment matching methods

As these methods are very well known in the actuarial literature, we will only give a very brief résumé plus an example in which we will obtain those disturbing negative masses.

Definition 1 (Panjer and Willmot, 1992, p. 24). A random variable $X$ is said to be arithmetic with span $h > 0$ if its distribution function is a step function with a countable number of discontinuities placed over the integral multiples of $h$, i.e. the set of possible values of $X$ is contained in the set \( \{ x = ih, \ i = 0, \pm 1, \pm 2, \ldots \} \).

The probability distribution function (PDF) of such a random variable will also be named arithmetic distribution.

All the PDF considered in this paper have their support contained in the half-line $[0, C_1]$, so the latter definition can be thought of only for positive multiples of the span $h$.

Given any PDF $F(x)$ with support included in $[0, +\infty)$, we want to “arithmetize” it, i.e. we have to choose a span or new monetary unit $h > 0$ (convenient to the settings of our application), to fix a nonnegative integer $n \in \{0, 1, 2, \ldots \}$, and try to determine the arithmetic PDF with span $h$ that has the property of equating $n$ moments with $F$.

To achieve the goal, the method of local moment matching tries to calculate the $n + 1$ masses $m^0_k, m^1_k, \ldots, m^n_k$, located at the points $x_k, x_k + h, \ldots, x_k + nh$, which are solutions of the linear system (Panjer and Willmot, 1992, p. 226):

\[
\sum_{j=0}^n (x_k + jh)^r m^k_j = \int_{x_k}^{x_k + nh} x^r \, dF(x), \quad r = 0, 1, 2, \ldots, n. \tag{1}
\]

In fact, we must solve (1) in each of the intervals $[0, nh], (nh, 2nh], (2nh, 3nh], \ldots$ corresponding to $x_0 = 0, x_1 = nh, x_2 = 2nh, \ldots$, etc. The final probabilities are the point masses obtained as solutions of (1) in each interval, “except that the two point masses coinciding at the end points of each interval of length $nh$ are summed” (Panjer and Willmot, 1992, p. 228). Note that if the support of the distribution function $F$ were not bounded it would be necessary to truncate it at some point, desirably at one of those $x_k$.

A quick look at linear system (1) tells us that it has the same number of equations and unknowns, the determinant being of the Vandermonde type, so a unique solution always exists. With the help of concepts carried from numerical analysis (see Gerber (1982, p. 15) and Panjer and Willmot (1992, Theorem 6.15.1)) it is shown that this unique solution can be written in terms of the distribution $F$, namely

\[
m^k_j = \int_{x_k}^{x_k + nh} \prod_{i \neq j} \frac{x - x_k - ih}{(j - i)h} \, dF(x), \quad j = 0, 1, 2, \ldots, n. \tag{2}
\]

But nothing ensures us that this unique solution will be nonnegative. If the solution were negative in any of the subintervals, indicating that the required arithmetic distribution does not exist, we would have to adopt one of the following alternatives:

- retreat to the matching of the preceding moment (say the first if we were trying to match until the second), as counseled in Panjer and Lutek (1983, p. 177), or
- change the span $h > 0$.

For an absolutely continuous $F$, this should not be upsetting, since we will be usually trying to approximate some kind of function (for instance a ruin probability), so the span $h$ will be conveniently small and then the solution (2) will be nonnegative, as explained in De Vylder (1996, p. 337). But if $F$ had any atom, then negative solutions (2) could eventually arise stopping the calculations corresponding to the chosen span $h > 0$.

For numerical purposes, in Panjer and Willmot (1992, p. 228) we are counseled that it is enough matching until the second moment (i.e. $r = 2$ in (1)), so we will do from hereafter.
Its first, second and third order moments are

\[ f(x) = 0.0103301 \delta(x - 14) + 0.0307990 \delta(x - 15) + 0.0293511 \delta(x - 16) + 0.0103301 \delta(x - 17) \\
+ 0.0730414 \delta(x - 18) + 0.0111568 \delta(x - 19) + 0.0264554 \delta(x - 20) + 0.1002133 \delta(x - 24) \\
+ 0.0815418 \delta(x - 26) + 0.0252146 \delta(x - 28) + 0.0212857 \delta(x - 30) \\
+ 0.0254214 \delta(x - 31) + 0.0991756 \delta(x - 55) + 0.4556837 \delta(x - 60). \]  

(3)

Its first, second and third order moments are

\[ \mu_1 = 42.7611973, \quad \mu_2 = E(X^2) = 2174.3969029, \quad \mu_3 = E(X^3) = 120649.5121549. \]  

(4)

In the reference, the authors want to replace this probability function (PF) \( f \) in Panjer and Willmot (1992, p. 229). For our purposes, we note it by means of the \textit{generalized derivative} \( f \): (money unit is $1000)

\[ p_0 = -0.009861356375, \quad p_{20} = 0.41172824775, \quad p_{40} = 0.058769996125, \]
\[ p_{60} = 0.548660825, \quad p_{80} = -0.0092977125. \]  

(5)

which indicates the nonexistence of the desired solution. Repeating this process taking successively all integral spans \( h \) between 1 and 25 it is found that the only case for the solution to be effectively a probability distribution (i.e. be nonnegative) is \( h = 1 \). Therefore the solution does not exist for any appropriate span. As counseled in Panjer and Lutek (1983, p. 177), we can retreat to first moment matching taking span \( h = 20 \), thus obtaining

\[ p_0 = 0.026080495, \quad p_{20} = 0.339844545, \quad p_{40} = 0.10400956, \quad p_{60} = 0.5300654, \]  

(6)

with the following moments

\[ \mu_1 = 42.7611973, \quad \mu_2 = 2210.588554, \quad \mu_3 = 123869.4946. \]  

(7)

In fact, the authors apply the \textit{method of rounding} which \textit{only conserves the probability mass} of the original distribution in each one of the intervals \( (x_k, x_k + h) \), obtaining three solutions (depending on rounding to the lower (A), nearest (B) or upper unit (C))

\[ \begin{array}{c}
A: \quad p_0 = 0.1650085, \quad p_{20} = 0.2801322, \quad p_{40} = 0.0991756, \quad p_{60} = 0.4556837, \\
B: \quad p_0 = 0, \quad p_{20} = 0.3984336, \quad p_{40} = 0.0467071, \quad p_{60} = 0.5548593, \\
C: \quad p_0 = 0, \quad p_{20} = 0.1914640, \quad p_{40} = 0.2536767, \quad p_{60} = 0.5548593. 
\end{array} \]  

(8)

Moments of order 1, 2 and 3 are not conserved (compare (9) with (4))

\[ \begin{array}{c}
A: \quad \mu_1 = 36.9106900, \quad \mu_2 = 1911.1951600, \quad \mu_3 = 107015.9752, \\
B: \quad \mu_1 = 43.1285140, \quad \mu_2 = 2231.5928000, \quad \mu_3 = 126026.320. \\
C: \quad \mu_1 = 47.2679060, \quad \mu_2 = 2479.9618000, \quad \mu_3 = 137616.6296. 
\end{array} \]  

(9)

Now we ask ourselves: for every span \( h \), would it be possible to determine another arithmetic PF more related to the original \( f \) in that its second moment be the nearest?

---

1 Numerical calculations in all the examples have been done with the help of procedures written in MapleV release 4 language. These procedures ask for the generalized derivative \( f \) as an input. This is to unify, from the code point of view, the three cases: absolutely continuous, discrete and mixed.
3. Geometrical interpretation of the problem

Consider a PDF $F$. The linear systems \((1)\) to be solved for $n = 2$ in each of the intervals $(0, 2h]$, $(2h, 4h]$, $(4h, 6h]$, \ldots, $(nh, (n+2)h]$, \ldots (\text{n even}) are

\[
\begin{align*}
\begin{cases}
m_0^n + m_1^n + m_2^n &= c_{0n}, \\
(nh)m_0^n + (n+1)hm_1^n + (n+2)hm_2^n &= c_{1n}, \\
(nh)^2m_0^n + ((n+1)h)^2m_1^n + ((n+2)h)^2m_2^n &= c_{2n},
\end{cases}
\]

where $c_{rn} = \int_{nh}^{(n+2)h} x^r dF(x)$, $r = 0, 1, 2$, $n = 0, 2, 4, \ldots$ (integrals are considered over the intervals $(nh, (n+2)h]$). The following applies in every $(nh, (n+2)h]$:\ we can consider $F$ restricted to the interval and then normalize it dividing by $c_{0n}$ to obtain a PDF $F_n$. For simplicity we will note $m_0, m_1, m_2$, the probability masses to be placed over the points $nh$, $(n+1)h$, $(n+2)h$, and $c_1 = c_{1n}/c_{0n}$, $c_2 = c_{2n}/c_{0n}^2$ will stand for the first and second order moments of $F_n$. Then Theorem 3a in De Vylder (1996, p. 343) applies.

**Theorem 1.** There exists a discrete PDF with first and second order moments $c_1$, $c_2$ and exactly three atoms placed at $nh$, $(n+1)h$, $(n+2)h$, if and only if the point $(c_1, c_2)$ belongs to the convex polygon with vertices $(nh, (nh)^2)$, \((n+1)h, (n+1)h^2\), \((n+2)h, (n+2)h^2\).

This result is easily understandable if we look at \((10)\) having in mind the normalization already done.

**Example 2** (Example 1 continued). In our last example, we consider the span $h = 20$, the interval $(0, 40]$ where we have $c_{00} = 0.4451407$, and the PDF $F_0 = 1/c_{00}F$. Then we have $c_1 = 22.38734247$, $c_2 = 525.5178707$. It is not difficult to check that the point $(c_1, c_2)$ is out of the convex polygon $K$ (see Fig. 1) with vertices $(0, 0)$, $(20, 400)$, $(40, 1600)$, so the obtention of a negative mass at the origin (5) is explained (see the point $G$ in Fig. 2). Applying first order moment matching we obtain the PF $p_0 = 0.0585893$, $p_{20} = 0.7634542$, $p_{40} = 0.1779564$. Its moments are $c_1 = 22.38734247$, $c_2 = 590.11201177$, thus this last point belongs to the convex polygon $K$ though far from the point $G$.

Recall that in Example 1 moment matching method up to second order does not give any probabilistic solution for any integer span $h$ between 2 and 25, therefore producing the last geometrical situation in some of the intervals $(nh, (n+2)h] (n$ even). When working with any PDF that produces this situation, retreating to first moment matching may not furnish the best arithmetic PF to work with, as there may exist another with equal first moment and a closer second moment for the chosen span.

We could summarize the comments of the last paragraph asking ourselves: instead of retreating to first moment matching or even to rounding, would it be possible for every span to determine the arithmetic PF which conserves the first moment $\mu_1$ and has the nearest second moment to $\mu_2$? In the next section we show how linear goal programming can give a very simple and quick way to calculate this PF.

4. Application of linear goal programming

4.1. General comments on goal programming

When we model a decision problem by means of goal programming, we must clearly establish what would be the best or ideal values for our objective functions, and then look to them as goals we want to attain, keeping in mind that it is probably not possible to match them simultaneously. In other words, if a decision vector $x$ giving those best values did not exist, we would look at least for another $x$ which gave us the nearest values for the objectives. This is done as follows. (To get insight into the subject of linear goal programming we refer to Romero (1991,1993)).
Suppose that our objective functions $f_0, \ldots, f_n$ are linear. We write them as the restrictions of a new mathematical program, each one equated to its best value $v_0, \ldots, v_n$, and we add to the first member of each restriction two deviation variables $y_j^+, -y_j^-$ ($j = 0, \ldots, n$). These ones stand for positive (how much we are down to each ideal value $v_j$) or negative deviation (how much we exceed), respectively. These $n + 1$ linear equations plus nonnegative restrictions in the deviation variables $y_j^+, y_j^-$ define the feasible set of the new mathematical program. Now, we want to obtain the decision vector which gives us the nearest values to those $v_j$ ($j = 0, \ldots, n$). This is done by means of minimization of some $L_p$ distance on the deviation variables. If we choose the $L_1$ distance, our objective will be minimizing the sum of all deviation variables and our mathematical program will be a linear one. This whole process consists in the resolution of the following linear program:

$$\text{Min } \sum_{j=0}^n (y_j^+ + y_j^-)$$

$$\begin{align*}
&f_0(x) + y_0^+ - y_0^- = v_0, \\
&\vdots \\
&f_n(x) + y_n^+ - y_n^- = v_n, \\
&y_j^+, y_j^- \geq 0, \ x \geq 0.
\end{align*}$$

(11)

Coordinates of feasible solutions are $(x, y_0^+, y_0^-, \ldots, y_n^+, y_n^-)$. In an optimal feasible solution, at least one coordinate in each pair $(y_j^+, y_j^-)$ must be null. When both vanish, the ideal value $v_j$ is attained on $x$. The program (11) stands for a typical linear goal program. Its advantage relies on its linearity, so it can be solved by means of the Simplex method.
Fig. 2. The frontier of the polygon $K$ is $y = 60x - 800$. Downwards is the parabola $c_2 = c_1^2$. The coordinates of the point $G$ are the first and second moment of the PDF $F_0$ (see Example 2). As the point lies outside the polygon $K$, second order moment matching results in a negative solution, $G_2$ is the point nearest to $G$ belonging to the polygon, and with same abscissa. If we summed the distance $y^+$ to the ordinate of $G$ we would obtain the ordinate of $G_2$.

4.2. The method of the nearest second moment

Coming back to arithmetization, let us consider the local moment matching up to the second order as a main objective that could be impossible to attain. This is particularly true when working with discrete or mixed types PDFs, as seen in Example 1. Our goal is to obtain an arithmetic distribution that is near the original one in that its first moment be equal and the second be the closest. For example considering the first interval $[0, 40]$ in Example 2, this would be equivalent to identify the PDF associated with the point $G_2$ represented in Fig. 2. Let us consider the following linear programs, each one related to its corresponding interval $(nh, (n + 2)h)$:

Min $y^+_2 + y^-_2$

\[
\begin{aligned}
m^0_n + m^n_1 + m^n_2 &= c_0n, \\
nhm^0_n + (n + 1)hm^n_1 + (n + 2)hm^n_2 &= c_1n, \\
(nh)^2m^n_0 + ((n + 1)h)^2m^n_1 + ((n + 2)h)^2m^n_2 + y^+_2 - y^-_2 &= c_2n, \\
y^+_2, y^-_2 &\geq 0, \quad m^0_n, m^n_1, m^n_2 \geq 0.
\end{aligned}
\]

The decision vector is $d = (m^0_n, m^n_1, m^n_2, y^+_2, y^-_2)$. The three restrictions model our goals, i.e. locally matching the mass $(c_{0n})$ and the local first moment $(c_{1n})$, and be as close as possible to the local second moment $(c_{2n})$. This last requirement is modeled through the objective function, which consists of the minimization of the $L_1$ distance for the two deviation variables. In order to set an efficient algorithm, the following properties will be very useful.
Proposition 1 (Feasibility). The linear program (12) is feasible (i.e. the feasible set is not empty) if and only if $c_{1n}/c_{0n} \in [nh, (n+2)h]$ $\forall n \in \{0, 2, 4, 6, \ldots \}$.

Proof. This is a direct consequence of Theorem 1c in De Vylder (1996, p. 336 and 341). Let us divide by $c_{0n}$ the first and second restrictions in (12). Writing $m_j = m_j^0/c_{0n}$, we have just obtained the linear system

$$m_0 + m_1 + m_2 = 1, \quad nhm_0 + (n+1)hm_1 + (n+2)hm_2 = c_{1n}/c_{0n}.$$  \hspace{1cm} (13)

Any nonnegative solution of (13) can be interpreted as a PF with atoms placed on $nh, (n+1)h, (n+2)h$, and mathematical expectation equal to $c_{1n}/c_{0n}$. By the referred theorem, solutions exists if and only if $c_{1n}/c_{0n} \in [nh, (n+2)h]$. Taking one of these, the triplet with coordinates $m_j^0 = c_{0n}m_j$ $(j = 0, 1, 2)$, will be a nonnegative solution of the linear system

$$m_0^0 + m_1^0 + m_2^0 = c_{0n}, \quad nhm_0^0 + (n+1)hm_1^0 + (n+2)hm_2^0 = c_{1n}.$$  \hspace{1cm} (14)

Now joining to (14) the third restriction and the nonnegativity restrictions of (12), we obtain the definition of the feasible set.

$$m_0^1 + m_1^1 + m_2^1 = c_{0n}, \quad nhm_0^1 + (n+1)hm_1^1 + (n+2)hm_2^1 = c_{1n}, \quad (nh)^2m_0^2 + ((n+1)h)^2m_1^2 + ((n+2)h)^2m_2^2 + y_{2n}^2 = c_{2n}, \quad y_{2n}^+ y_{2n}^- \geq 0, \quad m_0^1, m_1^1, m_2^1 \geq 0.$$ \hspace{1cm} (15)

Completing every nonnegative solution of (14) with any nonnegative values for the deviation variables $y_{2n}^+, y_{2n}^-$ we will obtain points of the feasible set of program (12).

Proposition 2 (Moment matching is a particular case). A nonnegative solution of the linear system (10) is an optimal basic feasible solution of program (12).

Proof. If we substitute a nonnegative solution of (10) in the restrictions of program (12) we will obtain $y_{2n}^+ = y_{2n}^- = 0$. These are the nonbasic coordinates. Now, recall that the coefficients of variables $m_j^n$ $(j = 0, 1, 2)$ form a Vandermonde matrix, so its determinant is different from 0. These are the basic coordinates of the basic feasible solution. Optimality follows directly from the fact that the deviation variables are null. These two properties allow us to think that the resolution of the linear program (12) in each interval $(nh, (n+2)h]$ supplies

- the nonnegative solution obtained by means of local moment matching, or
- when local moment matching does not have a nonnegative solution, the triplet is $(m_0^0, m_1^0, m_2^0)$, whose second order local moment is closest to the given $c_{2n}$.

We can implement the method of the nearest second moment encoding the following steps:

Step 1. Choose a span $h > 0$ and consider the intervals $(nh, (n+2)h]$ $n \in \{0, 2, 4, 6, \ldots \}$. If the support of the PDF is not bounded, truncate the PDF at some point $nh$ (n even) so as to have $F(nh) < \varepsilon$, with $\varepsilon$ conveniently small. If the support were bounded the end point of the last interval would be $\min((n+2)h/n \text{ even}, \ F(nh) = 1)$. Step 2. Solve in each interval the linear program (12) by means of Simplex method. Sum the two point masses $m_1^n, m_2^n$ coinciding at the end of each interval of length $2h$.

The method of the nearest second moment is applied to the PDFs presented in Examples 1 and 2.

Example 3.

1. We can come back to Example 2 and try to find the PF corresponding to the point $G_2$ in Fig. 2. We have to solve the linear program (12) over the interval $(0, 40]$. Its solution is

$$p_0 = 0, \quad p_{20} = 0.880632876113, \quad p_{40} = 0.119367123667$$

with mean 22.38734247 and second order moment 543.2405483, which are the coordinates of the point $G_2$. 


Table 1
Values of the compound Poisson distribution with claims frequency 4.841423259

<table>
<thead>
<tr>
<th>x</th>
<th>Exact</th>
<th>Rounding to nearest unit</th>
<th>Nearest second moment</th>
<th>Moment matching third order</th>
<th>Moment matching first order</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0.0078</td>
<td>0.0078</td>
<td>0.0078</td>
<td>0.0078</td>
<td>0.0078</td>
</tr>
<tr>
<td>40</td>
<td>0.0297</td>
<td>0.0396</td>
<td>0.0400</td>
<td>0.0389</td>
<td>0.0355</td>
</tr>
<tr>
<td>80</td>
<td>0.0096</td>
<td>0.1226</td>
<td>0.1236</td>
<td>0.1203</td>
<td>0.1103</td>
</tr>
<tr>
<td>120</td>
<td>0.2199</td>
<td>0.2363</td>
<td>0.2406</td>
<td>0.2375</td>
<td>0.2141</td>
</tr>
<tr>
<td>160</td>
<td>0.3517</td>
<td>0.3850</td>
<td>0.3905</td>
<td>0.3846</td>
<td>0.3437</td>
</tr>
<tr>
<td>200</td>
<td>0.5130</td>
<td>0.5431</td>
<td>0.5493</td>
<td>0.5415</td>
<td>0.4841</td>
</tr>
<tr>
<td>240</td>
<td>0.6609</td>
<td>0.6789</td>
<td>0.6863</td>
<td>0.6783</td>
<td>0.6046</td>
</tr>
<tr>
<td>280</td>
<td>0.7739</td>
<td>0.7918</td>
<td>0.7979</td>
<td>0.7881</td>
<td>0.7020</td>
</tr>
<tr>
<td>320</td>
<td>0.8630</td>
<td>0.8739</td>
<td>0.8786</td>
<td>0.8678</td>
<td>0.7736</td>
</tr>
<tr>
<td>360</td>
<td>0.9217</td>
<td>0.9272</td>
<td>0.9308</td>
<td>0.9196</td>
<td>0.8198</td>
</tr>
<tr>
<td>400</td>
<td>0.9572</td>
<td>0.9608</td>
<td>0.9630</td>
<td>0.9511</td>
<td>0.8483</td>
</tr>
<tr>
<td>440</td>
<td>0.9783</td>
<td>0.9800</td>
<td>0.9813</td>
<td>0.9690</td>
<td>0.8646</td>
</tr>
<tr>
<td>480</td>
<td>0.9894</td>
<td>0.9902</td>
<td>0.9910</td>
<td>0.9784</td>
<td>0.8732</td>
</tr>
<tr>
<td>520</td>
<td>0.9951</td>
<td>0.9954</td>
<td>0.9958</td>
<td>0.9831</td>
<td>0.8776</td>
</tr>
<tr>
<td>560</td>
<td>0.9978</td>
<td>0.9980</td>
<td>0.9982</td>
<td>0.9853</td>
<td>0.8797</td>
</tr>
<tr>
<td>600</td>
<td>0.9991</td>
<td>0.9991</td>
<td>0.9992</td>
<td>0.9863</td>
<td>0.8806</td>
</tr>
</tbody>
</table>

Each column corresponds to a different secondary distribution: the original PF $f$ (see Example 1), and the PFs obtained by means of rounding to nearest unit, nearest second moment, third moment matching, and first moment matching.

2. Coming back to Example 1, we make use of Steps 1 and 2 above to find the arithmetic PF with span $h = 20$ with equal mean and nearest second moment. We obtain the PF

$$p_0 = 0, \quad p_{20} = 0.392005535, \quad p_{40} = 0.077929065, \quad p_{60} = 0.5300654$$

with moments, up to the third

$$\mu_1 = 42.7611973, \quad \mu_2 = 2189.724158, \quad \mu_3 = 122617.63084.$$ (16)

Compare (16) with (4), (7) and (9). The last three are the moments corresponding to the original PF $f$ and those of the PFs obtained by means of moment matching up to the first order, and rounding. Finally, the method works for every integral span, indicating the feasibility of the linear programs engaged in the calculations.

4.3. Numerical illustration

As done in Panjer and Willmot (1992, p. 230), we have reported in Table 1 the exact values of the compound Poisson distribution taking the original PDF (3) as secondary distribution. Then come the values of the compound distribution taking successively as secondary the PF (8.B) (rounding to the nearest unit), (15) (nearest second moment), (6) (first order local moment matching). We have also included the values obtained using as secondary distribution the one supplied by third order moment matching. The span is $h = 20$ in all cases. Poisson claim frequency has mean equal to 4.841423259.

We can see that the method of the nearest second moment provides values

- With an accuracy similar to that of rounding which is sufficiently good, as observed in Panjer and Willmot (1992, p. 230).
- Comparing with the values provided by first order local moment matching, we conclude that we obtain a better accuracy.
- Comparing with the values provided by third order local moment matching, accuracy is better in the tail of the distribution.
4.4. Generalization to higher order moments

It is possible to set the generalization of the method of the nearest second order moment in a parallel way it is
done for the moments matching method. This consists in applying lexicographic goal programming, i.e. establishing
pre-emptive priorities among the matching of successive moments. As an example, we build on a sequential method
in order to obtain an arithmetic PF with equal mean and then nearest second moment and finally nearest third
moment. If moment matching were possible, this PF would be the same as the one provided by local moment
matching method up to the order 3.

We choose a span \( h > 0 \) and consider the intervals \((0, 3h] \), \((3h, 6h]\), \ldots , \((nh, (n + 3)h]\), \ldots \) (\( n \) as a multiple of
3). In each interval we want to place the probability masses \( m^n_0 \), \( m^n_1 \), \( m^n_2 \), \( m^n_3 \), over the points \( nh \), \( (n + 1)h \), \( (n + 2)h \), \( (n + 3)h \), such that the resulting arithmetic PF locally conserves the mass and the first moment, then we have
local second moment as the closest, and finally local third moment closest to those of the given distribution. Then
we have to solve sequentially the following linear programs in each interval (with the obvious modification in the
limits of the integrals \( c_{irn} \)):

\[
\begin{align*}
\text{Min} & \quad y^+_2 + y^-_2 \\
\text{s.t.} & \quad m^n_0 + m^n_1 + m^n_2 + m^n_3 = c_{0n}, \\
& \quad nhm^n_0 + (n + 1)hm_1^n + (n + 2)hm_2^n + (n + 3)hm_3^n = c_{1n}, \\
& \quad (nh)^2m_0^n + ((n + 1)h)^2m_1^n + ((n + 2)h)^2m_2^n + ((n + 3)h)^2m_3^n + y^+_2 - y^-_2 = c_{2n}, \\
& \quad y^+_2, y^-_2 \geq 0, \quad m^n_0, m^n_1, m^n_2, m^n_3 \geq 0.
\end{align*}
\]

(17)

Substituting the values of the deviation variables of the optimum of (17), \( y^+_2 = y^+_2 \), \( y^-_2 = y^-_2 \), we can write the
second linear program to be solved

\[
\begin{align*}
\text{Min} & \quad y^+_3 + y^-_3 \\
\text{s.t.} & \quad m^n_0 + m^n_1 + m^n_2 + m^n_3 = c_{0n}, \\
& \quad nhm^n_0 + (n + 1)hm_1^n + (n + 2)hm_2^n + (n + 3)hm_3^n = c_{1n}, \\
& \quad (nh)^2m_0^n + ((n + 1)h)^2m_1^n + ((n + 2)h)^2m_2^n + ((n + 3)h)^2m_3^n + y^+_3 - y^-_3 = c_{2n}, \\
& \quad (nh)^3m_0^n + ((n + 1)h)^3m_1^n + ((n + 2)h)^3m_2^n + ((n + 3)h)^3m_3^n + y^+_3 - y^-_3 = c_{3n}, \\
& \quad y^+_3, y^-_3 \geq 0, \quad m^n_0, m^n_1, m^n_2, m^n_3 \geq 0.
\end{align*}
\]

(18)

Observe that when the first linear program is feasible, the second is feasible too. A result similar to Proposition 1
could be established about the feasibility of (17). It is important to note that (18) would provide a different solution
than the one obtained in (17) if and only if the latter had more than one optimum (i.e. the optimum were not strict).
If the optimum obtained solving (17) were strict, the second program would be said to be redundant, furnishing the
same solution. As this last situation will happen so often, it is not worth to set a lexicographic ordering among the
objectives, because the resulting PF will be in almost all the cases the same as the one obtained solving (17).

5. Conclusions

When trying to replace a discrete or mixed type PDF by an arithmetic one, it can happen that local moment
matching method up to the second order fails for every appropriate span, giving some negative mass. Instead of a
simple retreat to first order local moment matching, we propose to look for the arithmetic PF with equal expectation
and closest second moment to the original PDF. The resolution of this problem is done in a fast and quite simple
way.
The determination of this arithmetic PF is done by means of linear goal programming, which makes possible the junction of both the ideas of moment matching and closest second moment in a single method called here the method of the nearest second moment. The linearity of the mathematical programs (12) permits a fast resolution of this method, because we can make use of simplex algorithm.

From the numerical point of view, the solution provided by this method gives in the example an accuracy similar to that given by the PF calculated through rounding to the nearest unit (which is quite good), and better than the one provided by the PF supplied by first order local moment matching. In the tail of the compound distribution, the accuracy provided by our solution is better than the one supplied by the PF which matches the original PF up to the third moment. This is a fact that may have important consequences, for example when evaluating fixed-time ruin probabilities. This is probably due to the fact that when we apply the moment matching method up to third order, the solution obtained is “less local”, as we only have to solve one linear system (1) on the whole interval [0, 60]. Therefore the opposite strategy to that of retreating (i.e. going to the matching of higher order moments) does not guarantee a better approximation of the compound distribution.

Having in mind that the principal goal of an arithmetization method is to calculate a PF as similar as possible to the original one, we can say that the method of the nearest second moment constitutes a complementary methodology to that of moment matching that may provide us with a solution which could be used with more confidence in the subsequent numerical calculations.

Finally the method is a direct generalization of local moment matching up to the second order.

On the other hand, generalization of the method to higher order moments through lexicographic goal programming may be useless, because of the strictness of the optimum found in the first linear program written in the sequential method. Another way of generalization to higher orders could be performed by means of weighted linear goal programming, but this has not been done because we think that the lexicographic order models better the objectives sought when “arithmetizing” a PDF.

References