MATLAB FUNCTION `fmpolMAP`, V. 2.0

SØREN ASMUSSEN AND MOGENS BLADT

This function provides calculations of fractional moments and polynomial Gram-Charlier expansions (CGE) of the count $N(T)$ in a possibly time-inhomogeneous Markovian arrival process (MAP), similar to the numerical examples in our paper

Moments and Polynomial Expansions in Discrete Matrix-Analytic Models

submitted to the special issue of *Stochastic Processes and Applications* in honour of Larry Shepp. It represents in no way a full numerical implementation of the methodology in that paper, but only a rough sketch allowing the user to run some simple examples of his own and facilitating him to further develop the scope, should he so want.

Overview

A call has the structure

`[mu,sig2,fm,M,CGE] = fmpolMAP(C0,D0,alpha,T,kms,R)`

or

`[mu,sig2,fm,M,CGE] = fmpolMAP(C0,D0,alpha,T,kms,R,optvct)`

Input parameters. In the time-homogeneous case, `C0=C` and `D0=D` are both $p \times p$ and self-explanatory. and p is identified by the function so it needs not be specified. `T` denotes the time horizon. In the inhomogeneous case, $\boldsymbol{C}(t)$ and $\boldsymbol{D}(t)$ are assumed to be piecewise constant on m equidistant intervals

$$[0,h),[h,2h),\ldots,[(m-1)h,mh) \quad \text{where } h = T/m,\ [(m-1)h,mh) = [(m-1)h,T),$$

and `C0(:,:,j)`, `D0(:,:,j)` are then the values for $t \in [(j-1)T/m, jT/m)$. Thus `C0` and `D0` are $p \times p \times m$. One does not need to specify whether the dimensions are $p \times p \times m$ or $p \times p$ or the values of p, m. `alpha` is the p-vector of initial probabilities.

`kms` is an ordered vector of J =1, 2 or 3 components like `[7]`, `[5 9]` or `[6 8 10]`. Its components are the degrees of the polynomials in the GCE expansions that are calculated.

`R` is the number of replications in the simulation comparison. Most often, we have taken `R`$= 10^5$ ourselvs, but sometimes a larger value is required for sufficient smoothness of the results, sometimes a smaller will do. The simulations typically constitute the most time consuming part when calling the function.

`optvct` is a vector of flexible length specifying options to the defaults of the function. These are described in detail under Code Details below and deal with formatting of the figure and which reference distribution to use. If no such options are desired, `optvct` can be omitted in the call as indicated above.

Date: August 29, 2021.

Output. `mu` and `sig2` are the mean μ and variance σ^2 of $N(0,T]$. The $1\times K$ vector `fm` contains fractional moments $k=1,\dots,K$ of $N(T)$ where $K=\max($`kms`$)$, and `M(i,j,k)` is the matrix of the $(k-1)$th fractional moment conditioned on initial Markov state i and contingent on terminal state j. The $(J+1)\times\overline{x}$ matrix `CGE` contains the CGE approximations in its first J rows and the empirical values in row $J+1$. More precisely, `CGE`(j,x) is the CGE approximation for $\mathbb{P}(N[0,t]=x)$ with number of terms given by the jth element of `kms` for $j\le J$.. Similarly, `CGE`$(J+1,x)$ is the simulated value. The upper limit $\overline{x}$ is set as the rounded 99% quantile $\mu+2.58\,\sigma$ of the normal approximation.

Error messages on input. Not provided in this version 1.0. Possible error could be that the dimensions of `C` and `D` do not match, that the constraints on signs and zero row sums are not satisfied, that `kms` has more than 3 elements or is not ordered, etc.

Methodology. The function uses a Poisson reference distribution F_0 with parameter $\lambda=\mu$ if $\sigma^2<\mu$ (underdispersion). and a negative binomial one with the same mean and variance as $N(T)$ if $\sigma^2\ge\mu$ (overdispersion). These defaults can be overruled by options, see below.

Code details

0.1. **Optional parameters.**

`optvct(1)=n` gives the figure produced by the function number `n`, not 1.
`optvct(2)=xm` deactivates the automatic choice of upper x-limit in the figure and CGE table and makes it `xm`.
`optvct(3)=ym`: similar for the y-limit in the figure.
`optvct(4)-optvct(8)` deactivate the automatic choice between Poisson or negative binomial reference distribution F^*. In detail:
`optvct(4)` makes F^* Poisson with the same mean as the MAP target.
`optvct(5)` makes F^* Poisson with the same variance as the MAP target. `optvct(6)=m` makes F^* Poisson with mean `m`.
`optvct(7)=m` makes F^* negative binomial but changes the default mean to `m`.
`optvct(8)=s2` makes F^* negative binomial but changes the default variance to `s2`.

Variables. The indexing is complicated by Matlab's requirement of the index in an entry in a vector or matrix to always start by 1. Thus a set $a_0,\dots,a_n$ of numbers like the coefficients of a nth degree polynomial must be stored in a vector `a` with `a(i)`$=a_{i-1}$, etc. This caused ourselves a lot silly mistakes and consequently debugging!

`Tsteps` number of constancy intervals of $\boldsymbol{C}(t)$

`ks` the number of elements in `kms`

`kmax` maximal polynomial degree `kms(ks)`$=\max_i$`kms(i)`

`km` $=$ `km+1` , cf. the above indexing issues.

`nn` vector storing the `repl=1:R` simulated values of $N(T)$

`F` is the matrix $\boldsymbol{M}_k$ of the paper

`H` equals $\exp(\boldsymbol{M}_k T)$ where T is replaced by the step length in the inhomogeneous case.

`M` equals $\exp(\boldsymbol{M}_k T)$ where T is the time horizon in the homogeneous case. In the inhomogeneous case, `M` is the final product integral.

`poisson` logical variable, $= 1$ if the reference distribution is Poisson and $= 0$ if it is negative binomial.

`xmax` the maximal x-values to be returned in the `CGE` approximations and the figure for $\mathbb{P}(N(T) = x)$.

`xs` values `0:1:xmax` on x-axis

`f0x` point probabilities at `0:1:xmax` for reference distribution

`fsimx` empirical (simulated) point probabilities at `0:1:xmax` for $N(T)$

Subfunctions. In subfunctions `ortopolPC` and `ortopolNB`, d is the maximal polynomial degree and for $n \le d$, p_n is the nth orthonormal polynomial $\sum_{k=0}^{n} a(n,k)x_{(k)}$ of reference distribution F_0. Here F_0 is Poisson(λ) in `ortopolPC` and negative binomial in `ortopolNB`. The matrix `A`; $(d+1) \times (d+1)$ in output has `A(n,k)`$=$ $a(n-1, k-1)$

`[A,c]=orthopolPC(lambda,kmax,fms)` returns the orthonormal polynomials (Gram-Charlier) for the Poisson distribution with parameter `lambda` in the matrix `A`. The coefficients in the CGE expansions for a distribution with factorial moments contained in `fms` are returned in `c`.

`[r,rho,A,c]=orthopolNB(mu,sig2,kmax,fms)` returns the orthonormal polynomials (normalized Meixner) for the negative binomial distribution with mean `mu` and variance `sig2` in the matrix `A`. The coefficients in the CGE expansions for a distribution with factorial moments `fms` are returned in `c`, and the standard parameters of the negative binomial distribution in `r,rho`.

`[pdf,cdf] = expansion(km,A,c,t,f0x)` returns the CGE approximation of degree `km` for the p.d.f. and c.d.f. for the reference p.d.f. `f0x`, with the polynomial coefficients specified in `A` and the degree k polynomial given weight `c(k+1)`.

`[n,j]=MAPsim(p,C,D,j0,T)`: the return `n` contains one simulated value of $N(T)$ for a MAP with parameters `C`,`D` (both of dimension `p` and initial Markov state `j0`, whereas the terminal Markov state is returned in `j`. The algorithm uses uniformization with reference Poisson rate `pint`.