## Package ‘binomialcftp’

February 19, 2015

**Type** Package  
**Title** Generates binomial random numbers via the coupling from the past algorithm  
**Version** 1.0  
**Date** 2012-09-14  
**Author** Francisco Juretig  
**Maintainer** Francisco Juretig <fjuretig@yahoo.com>  
**Description** Binomial random numbers are generated via the perfect sampling algorithm. At each iteration dual markov chains are generated and coalescence is checked. In case coalescence occurs, the resulting number is outputted. In case not, then the algorithm is restarted from \( T(t)=2^*T(t) \) until coalescence occurs.  
**License** GPL-2  
**LazyLoad** yes  
**Repository** CRAN  
**Date/Publication** 2012-09-20 06:42:30  
**NeedsCompilation** no

### R topics documented:

- binomialcftp-package .......................... 2  
- bin_ps .................................. 3  
- draw_i .................................. 4

**Index**

1
binomialcftp-package  Generates Random Numbers according to the coupling from the past algorithm.

Description

Dual Markov Chains are generated, one starting at n and the other one at 0 (in this fashion, every possible state is trapped between these two chains) and are updated according to a Metropolis-Hastings step. The transition kernel is defined as a ratio between the density evaluated at the current iteration vs the previous iteration. In this way the chains evolve (in general) in the direction where the density is higher. Coalescence is checked at every step (meaning that both chains coalesce), and in case this happens the resulting random number is outputted at t=0. In case coalescence does not occur, the algorithm is restarted starting from a distant past twice as large as the current starting past time. Every iteration that goes through some t that has previously been traversed, uses the exact same random number used at that point.

Details

Package: binomialcftp
Type: Package
Version: 1.0
Date: 2012-09-14
License: GPL-2
LazyLoad: yes

Author(s)

Fracisco Juretig <fjuretig@yahoo.com>

References


See Also

Perfectly Random Sampling with Markov Chains http://dimacs.rutgers.edu/~dbwilson/exact/

Examples

bin_ps(1000,20,0.5)
**bin_ps**  

*CFTP Binomial Random Numbers*

**Description**
Main function, return 0-n binomial distributed random numbers according to the coupling from the past algorithm

**Usage**

```
bin_ps(x, n, p)
```

**Arguments**

- **x** sample_size: number of random numbers desired  
- **n** n= binomial parameter  
- **p** p= binomial parameter

**Details**
As usual, p should be between 0 and 1 and n should be any positive integer

**Value**

- **values** Random numbers  
- **iteration_needed** Number of iterations needed until coalescence  
- **rand_used** Number of random numbers used

**Note**
Running time is different as n and p change

**Author(s)**
Francisco Juretig <fjuretig@yahoo.com>

**References**

**Examples**
```
bin_ps(1000, 20, 0.5)
```
draw_i

**generate random integer numbers**

---

**Description**

Random +1/-1 numbers. This is a secondary function used by bin_ps

**Usage**

`draw_i()`

**Value**

+1,-1

**Author(s)**

Francisco Juretig <fjuretig@yahoo.com>
Index

*Topic CFTP
  bin_ps, 3

*Topic \textasciiitildekw1
  draw_i, 4

*Topic \textasciiitildekw2
  draw_i, 4

*Topic binomial
  bin_ps, 3
    binomialcftp-package, 2
  bin_ps, 3
  binomialcftp(binomialcftp-package), 2
  binomialcftp-package, 2

  draw_i, 4