Subject: Re: Resampling from cumulative probability distribution
Posted by David Fanning on Sat, 26 Jan 2008 14:58:45 GMT
View Forum Message <> Reply to Message

Klemens writes:

> I am computing a bootstrap analysis where the resampling routine from
> a cumulative probability distribution needs the most cpu time. May be
> you have some ideas how to eliminate the loop and speed up the
> routine ...
>
> function get_bootstrap_pdf, np, cdf
>
> zz = randomu(S, np) ; np is the number of
> samples
>
> cdfa = cdf[0:n_elements(cdf)-2] ; cdf is the cumulative
> probability distribution
> cdfb = cdf[1:n_elements(cdf)-1]
>
> b = fltarr(n_elements(cdf)) ; b will be the
> resampled distribution
> b[*] = 0.00
>
> for i = 0, n_elements(cdf)-2 do begin ;
> loop through all bins
> index = where((zz ge cdfa[i]) and (zz lt cdfb[i]))
> if (max(index) ge 0) then begin
> b[i] = n_elements(index)
> endif else begin
> b[i] = 0.00
> endelse
> endfor
>
> total_b = total(b)
> b = b / total(b)
>
> return, b
>
> end
>
> Thanks for your help in advance !

I think you want something like this. I've added a common block
for your seed. Without it, I think you will find your results
not all that random.:-)

function get_bootstrap_pdf, np, cdf

```
  common seed, s

  zz =  randomu(S, np)

  cdfa = cdf[0:n_elements(cdf)-2]
  cdfb = cdf[1:n_elements(cdf)-1]

  I = findgen(n_elements(cdfa))
  b = ((zz ge cdfa) and (zz lt cdfb)) * I

  return, b / total(b)

  end
```

Cheers,

David
--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.dfanning.com/
Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Subject: Re: Resampling from cumulative probability distribution
Posted by Klemens on Sat, 26 Jan 2008 15:52:08 GMT
View Forum Message <> Reply to Message

Hi David,

Thanks for your response !

Unfortunately your code does not produce what I need ...

What I need I would receive when the histogram function would work on
bins that are not equidistant...
Picking up the pdf from the cdf by generating the series of random
numbers.

Cheers,

Klemens

Subject: Re: Resampling from cumulative probability distribution
Posted by David Fanning on Sat, 26 Jan 2008 16:19:04 GMT

View Forum Message <> Reply to Message

Klemens writes:

> Unfortunately your code does not produce what I need ...
>
> What I need I would receive when the histogram function would work on
> bins that are not equidistant...
> Picking up the pdf from the cdf by generating the series of random
> numbers.

Do you mean the code I sent you doesn't work, or the
algorithm you asked me to implement doesn't work?

Cheers,

David
--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.dfanning.com/
Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Subject: Re: Resampling from cumulative probability distribution
Posted by Klemens on Sat, 26 Jan 2008 16:42:20 GMT
View Forum Message <> Reply to Message

Your agorithm works, but not in this way I need it ...

Cheers,

Klemens

Subject: Re: Resampling from cumulative probability distribution
Posted by jschwab@gmail.com on Sun, 27 Jan 2008 20:38:06 GMT
View Forum Message <> Reply to Message

I didn't test it, but if I understand what you're doing, the following
code should work.

Cheers,
Josiah
--

function get_bootstrap_pdf, np, cdf

```
zz =  randomu(S, np)       ; np is the number of samples

n_cdf = n_elements(cdf)    ; ncdf is the number of points in the cdf

;; the next step finds where each zz falls in the cdf,
;; which is of course monotonically nondecreasing

zz_locs = value_locate(cdf, zz)

;; value locate returns a list of indices of the element such that
;; cdf[indices[j]] < zz[j] < cdf[indices[j]+1]
;; read the documentation if that's too terse an explanation

;; since the indices are evenly spaced, unlike the cdf values
;;(which is presumably why you couldn't just use histogram in the
;; first place) we can use the histogram command

;; the number of zeros, that is the first element of zz_hist,
;; tells you how many zz values fell between cdf[0] and cdf[1]

zz_hist = histogram(zz_locs, min = 0 , max = ncdf - 2, binsize = 1)

;; now just divide by the total of zz_hist, which presumably is np

zz_norm = zz_hist / nb

return, zz_norm

end
```

On Jan 26, 9:09 am, Klemens <jokulhl...@web.de> wrote:
> Hallo together,
>
> I am computing a bootstrap analysis where the resampling routine from
> a cumulative probability distribution needs the most cpu time. May be
> you have some ideas how to eliminate the loop and speed up the
> routine ...
>
> function get_bootstrap_pdf, np, cdf
>
> zz = randomu(S, np)   ; np is the number of
> samples
>
> cdfa = cdf[0:n_elements(cdf)-2]   ; cdf is the cumulative
> probability distribution

```
>  cdfb = cdf[1:n_elements(cdf)-1]
>
>  b = fltarr(n_elements(cdf))                    ; b will be the
>  resampled distribution
>  b[*] = 0.00
>
>  for i = 0, n_elements(cdf)-2 do begin                          ;
>  loop through all bins
>    index = where((zz ge cdfa[i]) and (zz lt cdfb[i]))
>    if (max(index) ge 0) then begin
>      b[i] = n_elements(index)
>     endif else begin
>      b[i] = 0.00
>    endelse
>  endfor
>
>  total_b = total(b)
>  b = b / total(b)
>
>  return, b
>
>  end
>
>  Thanks for your help in advance !
>
>  Klemens
```

Subject: Re: Resampling from cumulative probability distribution
Posted by Klemens on Mon, 28 Jan 2008 07:21:53 GMT
View Forum Message <> Reply to Message

Hi Josiah,

haven't tested your code but it looks great !
I have never seen the value_locate command before, but it will
eliminate a lot of loops in my programs !

Thanks for your help !

Cheers,

Klemens
