FF_DISC_RAND_VAR_MASS2COVCOR Examples

back to Fan's Intro Math for Econ, Matlab Examples, or Dynamic Asset Repositories

This is the example vignette for function: ff_disc_rand_var_mass2covcor from the MEconTools Package.

This function calculates covariance and correlation based for two discrete random variables.

Test FF_DISC_RAND_VAR_MASS2COVCOR Defaults

Call the function with defaults

ff_disc	_rand	_var_	mass	2covc	or();								
XXXXXXXX						 xxx rray (Matr	niv d	a+c)					
XXXXXXXX			_				TX 6						
		i	idx	ndi		_	owN	colN	sum	mean	std	coefvari	min
mt f	_of_s	1	5	2		30	6	5	1	0.033333	0.035743	1.0723	3.7187
	_01_3 _of_s	2	6	2			6	5	25	0.83333	5.3051	6.3661	J., 10,
	_of_s	3	7	2			6	5	249.78	8.3259	7.1913	0.86373	-6.
TARI	m+ £	٠	~~~~										
xxx TABL		_o†_s > : 1	(XXXX	xxxxxx c2	(XXXXX)	c3		c4	c5				
							_						
r1	0 0	28917		0.0464	10/	0.022848	2	0.0036146	0.000	110			
r2		020917		0.0929		0.085679		0.024097	0.0014				
r3		080324		0.0743		0.12852		0.064259	0.0014				
r4		013387		0.029		0.096388		0.085679	0.018				
r5		911156	0	.00594		0.036146		0.057119	0.023				
r6	3.718	37e-06	0.	000475	599	0.0054218	3	0.015232	0.011	621			
TADI	□+ v	٠٠ ،	~~~~~		~~~~~	-							
XXX TABL	c1	_OT_S > c2	c3	.xxxxxx c4	(XXXXX) C5	ζ							
r1	-7	-6	-7	-6	-6								
r2	-7 -5	-3	- <i>7</i> - 5	-3	-4								
r3	-2	-1	-1	0	-1								
r4	2	2	3	4	2								
r5	6	5	5	6	5								
r6	8	9	7	9	9								
XXX TABL		_ot_s >	(XXXXX c2	XXXXXX	(XXXXX) c3	c4		c 5					
	c1		(2		CS	C4		CS					
r1	13.23		21.7		18.136			13.901					
r2	9.94		16.8		9.6914			8.6906					
r3	16.25		6.21		13.799			11.641					
r4	12.62		2.75		6.5321			13.357					
r5	5.884	14	4.03	52	6.05	0.141	L02	0.50318					

-0.72091 5.1855

r6

3.5617

CONTAINER NAME: covvar_input_map Scalars -4.4805

-6.3772

	i	idx	value
	-		
fl_x_mean	1	1	-0.11081
fl_x_sd	2	2	4.1239
fl_y_mean	3	3	8.8423
fl_y_sd	4	4	6.5845

CONTAINER NAME: covvar_output_map ND Array (Matrix etc)

	i	idx	ndim	numel	rowN	colN	sum	mean	std	coefv
	-									
<pre>mt_cov_component_weighted</pre>	1	1	2	30	6	5	-22.084	-0.73612	1.0404	-1.41
<pre>mt_x_devi_from_mean</pre>	2	2	2	30	6	5	28.324	0.94415	5.3051	5.61
mt_x_y_multiply	3	3	2	30	6	5	-939.63	-31.321	36.564	-1.16
mt_y_devi_from_mean	4	4	2	30	6	5	-15.493	-0.51644	7.1913	-13.9

	c1	c2	с3	c4	c 5
r1	-0.87434	-3.5432	-1.4628	-0.22368	-0.0035451
r2	-0.13003	-2.1607	-0.35565	-0.47814	0.00087767
r3	-0.11248	0.17365	-0.56642	-0.025838	-0.018507
r4	0.010697	-0.38241	-0.69273	-3.0184	0.17717
r5	-0.0020165	-0.14618	-0.51584	-3.0371	-0.99056
r6	-0.00015927	-0.041473	-0.14098	-2.1121	-1.4106

	CI	C2	C3	С4	C5	
r1	-6.8892	-5.8892	-6.8892	-5.8892	-5.8892	
r2	-4.8892	-2.8892	-4.8892	-2.8892	-3.8892	
r3	-1.8892	-0.88919	-0.88919	0.11081	-0.88919	
r4	2.1108	2.1108	3.1108	4.1108	2.1108	
r5	6.1108	5.1108	5.1108	6.1108	5.1108	
r6	8.1108	9.1108	7.1108	9.1108	9.1108	

xxx TABLE:mt_x_y_multiply xxxxxxxxxxxxxxxxx

c1	c2	с3	c4	c 5	
-30.237	-76.225	-64.023	-61.882	-29.792	
-5.396	-23.242	-4.151	-19.842	0.59004	
-14.003	2.3348	-4.4073	-0.40209	-2.4884	
7.9905	-12.854	-7.1868	-35.23	9.5287	
-18.075	-24.568	-14.271	-53.172	-42.62	
-42.83	-87.129	-26.003	-138.66	-121.38	
	-30.237 -5.396 -14.003 7.9905 -18.075	-30.237 -76.225 -5.396 -23.242 -14.003 2.3348 7.9905 -12.854 -18.075 -24.568	-30.237 -76.225 -64.023 -5.396 -23.242 -4.151 -14.003 2.3348 -4.4073 7.9905 -12.854 -7.1868 -18.075 -24.568 -14.271	-30.237 -76.225 -64.023 -61.882 -5.396 -23.242 -4.151 -19.842 -14.003 2.3348 -4.4073 -0.40209 7.9905 -12.854 -7.1868 -35.23 -18.075 -24.568 -14.271 -53.172	

	c1	c2	c 3	c4	c 5
r1	4.389	12.943	9.2933	10.508	5.0587
r2	1.1037	8.0444	0.84902	6.8677	-0.15171
r3	7.4123	-2.6258	4.9566	-3.6286	2.7985
r4	3.7855	-6.0898	-2.3103	-8.57	4.5142
r5	-2.9579	-4.8071	-2.7924	-8.7013	-8.3392
r6	-5.2806	-9.5633	-3.6568	-15.22	-13.323

 fl_cov

```
-22.0835
fl_cor
-0.8133
```

Test FF_DISC_RAND_VAR_MASS2COVCOR Four States-Points

Over some (a,z) states that is 2 by 2, c matrix, and y matrix, find correlation. Positively related.

Same as before, but now inputs are single column:

```
% Call Function
[fl_cov_xy, fl_cor_xy] = ff_disc_rand_var_mass2covcor(...
    mt_c_of_s(:), mt_y_of_s(:), mt_f_of_s(:), bl_display_drvm2covcor);
display(['cov=' num2str(fl_cov_xy) ',cor=', num2str(fl_cor_xy)]);
```

cov=1.4446,cor=0.65723

cov=1.4446,cor=0.65723

Test FF_DISC_RAND_VAR_MASS2COVCOR Two Random Vectors

Generate two random vectors, with random or even mass, correlation should be zero:

cov=-57.6533,cor=-0.062023

Test FF_DISC_RAND_VAR_MASS2COVCOR Provide Mean and SD

Same as above, but now provide means and sd for x andy directly. The results are the same as when mean and sd are calculated inside the function.

```
% Set Parameters
rng(4567);
mt_c_of_s = rand([20,1])*100;
mt_y_of_s = rand([20,1])*100;
mt_f_of_s = rand(size(mt_c_of_s));
mt_f_of_s = mt_f_of_s/sum(mt_f_of_s, 'all');
fl c mean = sum(mt f of s.*mt c of s);
fl_c_sd = sqrt(sum(mt_f_of_s.*(mt_c_of_s-fl_c_mean).^2));
fl_y_mean = sum(mt_f_of_s.*mt_y_of_s);
fl_y_sd = sqrt(sum(mt_f_of_s.*(mt_y_of_s-fl_y_mean).^2));
bl_display_drvm2covcor = false;
% Call Function
[fl_cov_xy, fl_cor_xy] = ff_disc_rand_var_mass2covcor(...
    mt_c_of_s, mt_y_of_s, mt_f_of_s, ...
    fl_c_mean, fl_c_sd, ...
    fl_y_mean, fl_y_sd, bl_display_drvm2covcor);
display(['cov=' num2str(fl_cov_xy) ',cor=', num2str(fl_cor_xy)]);
```

cov=-57.6533,cor=-0.062023