```
*==================================================================================
* PROGRAMS

*----------------------------------------------
* Program to sample from a Wishart
* uses the WinBUGS parameterisation
*----------------------------------------------
program rwishart
   args T R k

   tempname L B
   local p = rowsof(`R')
   matrix `L' = cholesky(invsym(`R'))
   matrix `B' = J(`p',`p',0)
   forvalues r=2/`p' {
      local q = `r' - 1
      forvalues c=1/`q' {
         matrix `B'[`r',`c'] = rnormal()
      }
   }
   forvalues r=1/`p' {
      matrix `B'[`r',`r'] = sqrt(rchi2(`k'-`r'+1))
   }
   matrix `T' = `L'*`B'*(`L'*`B')'
end

*----------------------------------------------
* Program to sample from a Multivariate Normal
* an alternative to drawnorm that is more
* efficient when we just want one sample at a time
*----------------------------------------------
program rmnormal
   args Y M V

   tempname L Z
   local p = rowsof(`V')
   matrix `L' = cholesky(`V')
   matrix `Z' = J(`p',1,0)
   forvalues r=1/`p' {
      matrix `Z'[`r',1] = rnormal()
   }
   matrix `Y' = `M' + `L'*`Z'
end

*----------------------------------------------
* Program for Gibbs Sampling
* parameters passed in row vector b
* dummy replace an unneeded program for evaluating the log-posterior
* and is included so that mcmcrun can call viralGibbs
*----------------------------------------------
program viralGibbs
   args dummy b

   tempname MU V T MS VT TR S TT TM
*-----------------------
* unpack b
*-----------------------
   matrix `MU' = J(2,1,0)
   matrix `V' = J(2,2,0)
   matrix `MU'[1,1] = `b'[1,1]
   matrix `MU'[2,1] = `b'[1,2]
   matrix `V'[1,1] = `b'[1,3]
   matrix `V'[2,1] = `b'[1,4]
   matrix `V'[1,2] = `b'[1,4]
   matrix `V'[2,2] = `b'[1,5]
   matrix `T' = invsym(`V')
*-----------------------
* update mean
*-----------------------
   matrix `MS' = J(2,1,0)
   qui su plasma
   matrix `MS'[1,1] = r(sum)
   qui su saliva
   matrix `MS'[2,1] = r(sum)
   matrix `TT' = MT + 195*`T'
   matrix `VT' = invsym(`TT')
   matrix `TM' = `VT'*(MT*M+`T'*`MS')
```

```
   rmnormal `MU' `TM' `VT'
*-----------------------
* update precision
*-----------------------
   tempvar r1 r2
   qui gen `r1' = plasma - `MU'[1,1]
   qui gen `r2' = saliva - `MU'[2,1]
   qui matrix accum `S' = `r1' `r2' , nocon
   matrix `TR' = R + `S'
   local k = $df + 195
   rwishart `T' `TR' `k'
   matrix `V' = invsym(`T')
*-----------------------
* update missing
*-----------------------
   local LD = log10(80)
   local m1 = `MU'[1,1]
   local s1 = sqrt(`V'[1,1])
   local m2 = `MU'[2,1]
   local s2 = sqrt(`V'[2,2])
   local r = `V'[1,2]/(`s1'*`s2')
   forvalues i=1/195 {
*-----------------------
* plasma missing
*-----------------------
      if cplasma[`i'] > 0 & csaliva[`i'] < 0 {
         local m12 = `m1' + `r'*`s1'*(saliva[`i']-`m2')/`s2'
         local s12  = `s1'*sqrt(1-`r'*`r')
         qui replace plasma = `m12'+ invnorm(runiform()* ///
            normal((`LD'-`m12')/`s12'))*`s12' in `i'
      }
*-----------------------
* saliva missing
*-----------------------
      else if cplasma[`i'] < 0 & csaliva[`i'] > 0 {
         local m21 = `m2' + `r'*`s2'*(plasma[`i']-`m1')/`s1'
         local s21  = `s2'*sqrt(1-`r'*`r')
         qui replace saliva = `m21'+ invnorm(runiform()* ///
            normal((`LD'-`m21')/`s21'))*`s21' in `i'
      }
*-----------------------
* both missing
*-----------------------
      else if cplasma[`i'] > 0 & csaliva[`i'] > 0 {
         qui replace plasma = `m1'+ invnorm(runiform()* ///
            normal((`LD'-`m1')/`s1'))*`s1' in `i'
         local m21 = `m2' + `r'*`s2'*(plasma[`i']-`m1')/`s1'
         local s21  = `s2'*sqrt(1-`r'*`r')
         qui replace saliva = `m21'+ invnorm(runiform()* ///
            normal((`LD'-`m21')/`s21'))*`s21' in `i'
      }
   }
*-----------------------
* repack b
*-----------------------
   matrix `b'[1,1] = `MU'[1,1]
   matrix `b'[1,2] = `MU'[2,1]
   matrix `b'[1,3] = `V'[1,1]
   matrix `b'[1,4] = `V'[2,1]
   matrix `b'[1,5] = `V'[2,2]
   matrix `b'[1,6] = saliva[9]
end

*==================================================================================
* ANALYSIS


*-----------------------
* Read Data
*-----------------------
use hiv.dta, clear
gen cplasma = -1 + 2*(plasma==.)
gen csaliva = -1 + 2*(saliva==.)
*---------------------------------------
* Set Priors
*    MU~ N(M,MV)  (precision MT)
*    T ~ Wishart(R,df)   V=inv(T)
```

```
*--------------------------------------
matrix M = ( 4 \ 3)
matrix MV = ( 1 , 0.5 \ 0.5, 1)
matrix MT = invsym(MV)
matrix R = ( 40 , 30 \ 30, 40)
global df = 10
*--------------------------------------
* Initial values
*--------------------------------------
qui replace plasma = 80/sqrt(2) if plasma == .
qui replace saliva = 80/sqrt(2) if saliva == .
qui replace plasma = log10(plasma)
qui replace saliva = log10(saliva)
matrix MU = M
matrix V = R/$df
matrix b = J(1,6,0)
matrix b[1,1] = MU[1,1]
matrix b[1,2] = MU[2,1]
matrix b[1,3] = V[1,1]
matrix b[1,4] = V[2,1]
matrix b[1,5] = V[2,2]
matrix b[1,6] = saliva[9]
*--------------------------------------
* Use mcmcrun to control the simulations
* calls viralGibbs, which uodates all 6 parameters
* within each call
*--------------------------------------
mcmcrun dummy b using hivMCMC.csv, replace ///
  sampler( ( viralGibbs , dim(6) )  ) ///
  update(5000) burn(1000) par(mu1 mu2 v11 v12 v22 sv9)
*--------------------------------------
* Read results and summarize
*--------------------------------------
insheet using hivMCMC.csv , clear
gen s1 = sqrt(v11)
gen s2 = sqrt(v22)
gen r = v12/(s1*s2)
mcmcstats *
mcmctrace mu1 mu2 v11 v22 v12 sv9
mcmcsection mu1 mu2 v11 v22 v12 sv9
```

contents of hiv.dta

id	plasma	saliva
1	38000	10000
2	11000	3400
3	3400	510
4	2000	150
5	93	
6		
7	4400	980
8	65000	6400
9	120000	
10	190000	1600
11	130000	1300
12	1000000	440
13		
14	280	
15	37000	
16	5100	4300
17		
18		
19		
20	9800	
21		
22	2100	
23	3700	
24	640	
25		
26	15000	2100
27	2600	150
28	100	
29		
30	2000	2900
31	12000	1700
32	19000	4900

33	140000	4900
34	2100	
35		300
36		
37		
38		
39		
40		
41	120	
42		
43	910	
44	95	
45		
46		
47		
48		
49	310	
50	32000	20000
51		
52		
53		
54		
55	49000	21000
56	11000	6700
57	9100	7300
58	960	160
59	2200	
60	120	
61	210	
62	1500	
63		
64	83	
65		
66	160000	510
67	2800	
68		
69		
70		
71		
72	420	
73	880000	41000
74	190000	12000
75		
76		
77		
78		
79	16000	
80	38000	410
81	64000	
82	30000	36000
83	29000	3800
84		
85		
86		
87		240
88	42000	1500
89	41000	39000
90		
91		
92	4900	2800
93	2700	3600
94	11000	5400
95	7400	360
96	100000	16000
97	1100	
98	50000	570
99		
100		
101		
102		
103	5700	4700
104	5200	880

105	850	
106	39000	550
107		
108		
109		
110	2200	430
111	1900	790
112	2000	
113	470	
114	12000	1600
115	610	
116		
117		
118	190	
119	6500	
120		
121		510
122		
123		
124		
125		
126		
127		81
128	780	
129		
130	800	
131		
132	870000	10000
133	410000	16000
134	730000	30000
135	440000	2200
136		
137		
138		88
139	230	
140		
141	94000	200
142	59000	
143	6000	
144	4900	
145	160000	990
146	28000	
147	59000	
148	7400	
149	1300	
150		
151		97
152	290	
153		
154	17000	17000
155	1700	
156	4400	
157	230	620
158		130
159	3200	130
160	140	
161	35000	
162	750	
163		
164		83
165	690	
166		
167		
168	210	7500
169		380
170		
171		
172	280000	
173	590000	
174	8100	170
175	100	
176	670	

177	670	
178	310000	
179	170	
180	19000	
181	4000	
182	48000	390
183	48000	190
184	32000	580
185	130000	58000
186	1600	
187	17000	
188	1800000	4500
189	840000	4000
190	840	
191	63000	2600
192	56000	1700
193	13000	
194		
195		