The DO Loop and Random Number Generation

SP4R04d01.sas

1. Create a data table with 10 random deviates from four different probability distributions. Use the random seed **123** and drop the DO loop index variable. Let the distributions be a normal distribution with a mean of 20 and standard deviation of 5, a Bernoulli distribution with a probability of 0.25, a uniform distribution from 0 to 10, and an exponential distribution with a mean of 5. Print the data table.

 The uniform distribution and the exponential distribution have no inputs.

```
data sp4r.random (drop=i);
   call streaminit(123);
   do i=1 to 10;
      rnorm = rand('Normal',20,5);
      rbinom = rand('Binomial',.25,1);
      runif = rand('Uniform')*10;
      rexp = rand('Exponential')*5;
      output;
   end;
run;

proc print data=sp4r.random;
run;
```

Obs	rnorm	rbinom	runif	rexp
1	20.4197	0	3.87845	5.55589
2	17.6145	0	3.37595	8.88334
3	19.8906	1	8.46208	5.39953
4	22.1470	1	0.18005	4.14268
5	24.5320	1	7.51251	3.26894
6	25.0477	1	4.59901	0.40410
7	19.8704	0	2.95333	2.64741
8	21.0267	1	9.21826	2.31966
9	20.3133	1	5.06738	0.52389
10	22.7076	0	7.32552	1.20990

2. Add an additional variable to the data table above. Generate random numbers from a geometric distribution with a probability parameter of 0.1. Use the same random seed of 123. Print the data table upon completion.

```
data sp4r.random;
   call streaminit(123);
   set sp4r.random;
   rgeom = rand('Geometric',.1);
run;

proc print data=sp4r.random;
run;
```

Obs	rnorm	rbinom	runif	rexp	rgeom
1	20.4197	0	3.87845	5.55589	2
2	17.6145	0	3.37595	8.88334	1
3	19.8906	1	8.46208	5.39953	6
4	22.1470	1	0.18005	4.14268	1
5	24.5320	1	7.51251	3.26894	9
6	25.0477	1	4.59901	0.40410	4
7	19.8704	0	2.95333	2.64741	33
8	21.0267	1	9.21826	2.31966	4
9	20.3133	1	5.06738	0.52389	1
10	22.7076	0	7.32552	1.20990	1

3. Create a data table with 15 random deviates from two different probability distributions. Use the random seed 123. Let the distributions be Poisson with a mean of 25 and a Beta with parameters 0.5 and 0.5. Group these 15 observations into five different groups of three observations each. Finally, create a sequence from 1 to 15 to be included in the data table. Print the data table upon completion.

```
data sp4r.doloop (drop=j);
   call streaminit(123);
   do group=1 to 5;
      do j=1 to 3;
         rpois = rand('Poisson',25);
         rbeta = rand('Beta',.5,.5);
         seq+1;
         output;
      end;
   end;
run;

proc print data=sp4r.doloop;
run;
```

Obs	group	rpois	rbeta	seq
1	1	25	0.95447	1
2	1	31	0.73901	2
3	1	30	0.07951	3
4	2	24	0.00319	4
5	2	22	0.27194	5
6	2	31	0.42317	6
7	3	29	0.94307	7
8	3	20	0.98216	8
9	3	26	0.30177	9
10	4	26	0.97667	10
11	4	15	0.08009	11
12	4	27	0.57148	12
13	5	29	0.03174	13
14	5	28	0.97330	14
15	5	19	0.00528	15

4. Use a DO loop to create quantiles from -3 to 3 by 0.5. This creates 13 iterations. For the remaining arguments, use a normal distribution with a mean of 0 and a standard deviation of 1. For each iteration, identify the density and the cumulative density and create new variables, **PDF** and **CDF**.

Finally, use the new **CDF** variable to create a quantile variable that mirrors the DO loop values. Print the data table upon completion.

```
data sp4r.quants;
   do q=-3 to 3 by .5;
      pdf = pdf('Normal',q,0,1);
      cdf = cdf('Normal',q,0,1);
      quantile = quantile('Normal',cdf,0,1);
      output;
   end;
run;

proc print data=sp4r.quants;
run;
```

You can use function names as variable names in SAS.

Obs	q	pdf	cdf	quantile
1	-3.0	0.00443	0.00135	-3.0
2	-2.5	0.01753	0.00621	-2.5
3	-2.0	0.05399	0.02275	-2.0
4	-1.5	0.12952	0.06681	-1.5
5	-1.0	0.24197	0.15866	-1.0
6	-0.5	0.35207	0.30854	-0.5
7	0.0	0.39894	0.50000	0.0
8	0.5	0.35207	0.69146	0.5
9	1.0	0.24197	0.84134	1.0
10	1.5	0.12952	0.93319	1.5
11	2.0	0.05399	0.97725	2.0
12	2.5	0.01753	0.99379	2.5
13	3.0	0.00443	0.99865	3.0