WST211 Notes: Simulations in SAS

1. Introduction to simulations
2. DO LOOPS in the Data step

---

1. Introduction to simulations

Example 1.1

Generate a random sample of 5 observations from a $N(0,1)$ distribution.
Print the data.

SAS Program

```
data normal;
 do j=1 to 5;
  z=rannor(0);
  output;
 end;

proc print;
run;
```

SAS Output

Obs	j	z
1	1	-2.08410
2	2	-1.85358
3	3	-1.66878
4	4	-0.75558
5	5	1.46857

Example 1.2

Generate a random sample of 1000 observations from a $N(0,1)$ distribution.
Calculate the average and standard deviation for the data. Calculate $P(Z < 0.7)$ empirically.

SAS Program

```
data normal;
 do j=1 to 1000;
  z=rannor(0);
  if z<0.7 then ind=1; else ind=0;
  output;
 end;

proc means n mean;
var z ind;
run;
```

SAS Output

The MEANS Procedure

Variable	N	Mean
z	1000	0.0255254
ind	1000	0.7690000

Example 1.3
Generate 200 samples, each consisting of 1000 observations generated randomly from a UNIF(0,1) distribution.
Create a variable AVERAGE, which calculates the average for each sample of 1000 observations.
Do a PROC UNIVARIATE on the variable AVERAGE and test for normality of this variable.
What is the parameters for the theoretical distribution of the variable AVERAGE.
Compare the empirical results with the theoretical values.

SAS Program

```
data limit;
do i=1 to 200;
 total=0;
   do j=1 to 1000;
     u=ranuni(0);
     total=total+u;
   end;
 average=total/1000;
 output;
end;
proc univariate normal plot; var average; run;
```

SAS Output

```
                             The UNIVARIATE Procedure
                                Variable:  average
                                     Moments
         N                         200    Sum Weights               200
         Mean               0.50104796    Sum Observations   100.209591
         Std Deviation      0.00841733    Variance           0.00007085
         Skewness           -0.1491082    Kurtosis           0.09038876
         Uncorrected SS     50.2239105    Corrected SS       0.01409944
         Coeff Variation    1.67994535    Std Error Mean      0.0005952

                            Basic Statistical Measures
                  Location                    Variability
              Mean     0.501048     Std Deviation           0.00842
              Median   0.501763     Variance              0.0000709
              Mode      .           Range                   0.04896
                                    Interquartile Range     0.01123

                               Tests for Normality
            Test                  --Statistic---    -----p Value------
            Shapiro-Wilk          W     0.994485    Pr < W      0.6733
            Kolmogorov-Smirnov    D     0.043214    Pr > D     >0.1500
            Cramer-von Mises      W-Sq  0.088182    Pr > W-Sq   0.1653
            Anderson-Darling      A-Sq  0.474195    Pr > A-Sq   0.2434

                             Quantiles (Definition 5)
                             Quantile       Estimate
                             100% Max       0.523836
                             99%            0.520728
                             95%            0.514188
                             90%            0.511165
                             75% Q3         0.506733
                             50% Median     0.501763
                             25% Q1         0.495500
                             10%            0.489903
                             5%             0.486033
                             1%             0.481145
                             0% Min         0.474875
```

```
Stem Leaf                          #             Boxplot
 522 8                             1                0
 520 331                           3                |
 518 0                             1                |
 516 9                             1                |
 514 222306                        6                |
 512 10025                         5                |
 510 678901226                     9                |
 508 3770125                       7                |
 506 00123446692344445566667899   26             +-----+
 504 12571124456668               14             |     |
 502 0001225666678888901234569    25             |     |
 500 01224467890224566789         20             *--+--*
 498 123446678990145788           18             |     |
 496 01334635799                  11             |     |
 494 3456004678                   10             +-----+
 492 012000146689                 12                |
 490 00023711129                  11                |
 488 2634689                       7                |
 486 219                           3                |
 484 580889                        6                |
 482 38                            2                |
 480 0                             1                |
 478
 476
 474 9                             1                0
     ----+----+----+----+----+-
 Multiply Stem.Leaf by 10**-3
```

Test for normality:
H_0: Data have a normal distribution
H_1: Data do not have a normal distribution
Since the *p*-value for the Shapiro-Wilk statistic is greater than 0.05 it can be concluded that the distribution of the data do not deviate significantly from a normal distribution.

From the central limit theorem the variable AVERAGE will have a normal distribution with a mean of 0.5 and a standard deviation of 0.00913.

The theoretical mean of the variable AVERAGE is 0.5.
The empirical value is 0.501.

The theoretical standard deviation of the variable AVERAGE is .00913.
The empirical value is 0.00842.

Example 1.4
The following 3 programs illustrates the different data sets created depending on the position of the OUTPUT statement in the data step. Three samples, each consisting of 5 observations from a UNIF(0,1) distribution were generated and the average calculated for each of the 3 samples.

SAS Program (1)

```
data limit;
do i=1 to 3;
 total=0;
   do j=1 to 5;
     u=ranuni(15);
     total=total+u;
   end;
 average=total/5;
 output;
end;
proc print; run;
```

SAS Output

OBS	I	TOTAL	J	U	AVERAGE
1	1	4.03815	6	0.82404	0.80763
2	2	3.40463	6	0.99850	0.68093
3	3	2.31188	6	0.35536	0.46238

SAS Program (2)

```
data limit;
do i=1 to 3;
 total=0;
   do j=1 to 5;
     u=ranuni(15);
     total=total+u;
     output;
   end;
 average=total/5;
 output;
end;
proc print; run;
```

SAS Output

OBS	I	TOTAL	J	U	AVERAGE
1	1	0.77444	1	0.77444	.
2	1	1.32577	2	0.55133	.
3	1	2.32313	3	0.99736	.
4	1	3.21411	4	0.89098	.
5	1	4.03815	5	0.82404	.
6	1	4.03815	6	0.82404	0.80763
7	2	0.53916	1	0.53916	0.80763
8	2	0.68385	2	0.14469	0.80763
9	2	1.65922	3	0.97538	0.80763
10	2	2.40613	4	0.74691	0.80763
11	2	3.40463	5	0.99850	0.80763
12	2	3.40463	6	0.99850	0.68093
13	3	0.28978	1	0.28978	0.68093
14	3	1.14784	2	0.85806	0.68093
15	3	1.94875	3	0.80091	0.68093
16	3	1.95652	4	0.00777	0.68093
17	3	2.31188	5	0.35536	0.68093
18	3	2.31188	6	0.35536	0.46238

SAS Program (3)

```
data limit;
do i=1 to 3;
 total=0;
   do j=1 to 5;
     u=ranuni(15);
     total=total+u;
     output;
   end;
 average=total/5;
end;

proc print;

proc means n mean;
var u;
by i;

run;
```

SAS Output

OBS	I	TOTAL	J	U	AVERAGE
1	1	0.77444	1	0.77444	.
2	1	1.32577	2	0.55133	.
3	1	2.32313	3	0.99736	.
4	1	3.21411	4	0.89098	.
5	1	4.03815	5	0.82404	.
6	2	0.53916	1	0.53916	0.80763
7	2	0.68385	2	0.14469	0.80763
8	2	1.65922	3	0.97538	0.80763
9	2	2.40613	4	0.74691	0.80763
10	2	3.40463	5	0.99850	0.80763
11	3	0.28978	1	0.28978	0.68093
12	3	1.14784	2	0.85806	0.68093
13	3	1.94875	3	0.80091	0.68093
14	3	1.95652	4	0.00777	0.68093
15	3	2.31188	5	0.35536	0.68093

```
Analysis Variable : U
------------------------------------------- I=1 ------------------------------------------
                                      N          Mean
                                      5          0.8076304

------------------------------------------- I=2 ------------------------------------------
                                      N          Mean
                                      5          0.6809265

------------------------------------------- I=3 ------------------------------------------
                                      N          Mean
                                      5          0.4623755
```

2. DO LOOPS in the Data step

DO, Iterative

DO index-variable = specification -1 <,... specification-n>;

index-variable

Names a variable whose value governs execution of the DO group. Unless dropped, the index variable is included in the data set being created.

specification

Denotes an expression or series of expressions in the following form:
start <TO stop> <BY increment> <WHILE | UNTIL(expression)>

start
specifies the initial value of the index variable. When used with TO stop or BY increment, start must be a number or an expression that yields a number.

TO stop
specifies the ending value of the index variable. Stop can be a number or an expression that yields a number.

BY increment
specifies a number (or an expression that yields a number) to control incrementing of of a index-variable.

WHILE(expression)
UNTIL(expression)
evaluates, either before or after execution of DO group, any SAS expression you specify enclosed in parentheses.

DO UNTIL (expression);
DO WHILE (expression);

Example 2.1: DO, TO, BY

Write a SAS program which generates the values 2, 4, 6, …, 20.

SAS Program

```
data example1;
  do j=2 to 20 by 2;
  output;
  end;
proc print; run;
```

SAS Output

```
OBS     J
  1     2
  2     4
  3     6
  4     8
  5    10
  6    12
  7    14
  8    16
  9    18
 10    20
```

Example 2.2: DO, UNTIL

Write a SAS Program which generates values from a UNIF(0, 1) distribution. Stop once a value greater than 0.9 is generated.

SAS Program

```
data example2;
  do until (t>0.9);
  t=ranuni(0);
  output;
  end;
proc print;
run;
```

SAS Output

```
OBS         T
1        0.01576
2        0.85953
3        0.34019
4        0.38242
5        0.72255
6        0.16656
7        0.63342
8        0.85155
9        0.91175
```

Example 2.3: DO, TO, UNTIL

Write a SAS Program which generates values from a UNIF(0, 1) distribution. Stop once a value greater than 0.9 is generated with a maximum of 5 values generated.

SAS Program

```
data example3;
  do j=1 to 5 until (t>0.9);
  t=ranuni(0);
  output;
  end;
proc print;
run;
```

SAS Output

```
OBS    J       T
 1     1    0.03701
 2     2    0.20093
 3     3    0.91544
```

Example 2.4: DO, WHILE

Write a SAS Program which generates values from a UNIF(0, 1) distribution. Stop when 4 values greater than 0.4 have been generated.

SAS Program

```
data example4;
  count=0;
  do while (count<4);
  t=ranuni(0);
  if t>0.4 then count=count+1;
  output;
  end;
proc print; run;
```

SAS Output

OBS	COUNT	T
1	1	0.66793
2	1	0.00068
3	1	0.09333
4	1	0.33756
5	1	0.26585
6	2	0.80662
7	2	0.32132
8	2	0.03510
9	3	0.83897
10	4	0.60180

Example 2.5: IF THEN ELSE, IF AND

Write a SAS Program which generates values from a UNIF(0, 1) distribution. Determine the frequencies of values in the interval (0; 0.4] and (0.4; 1). Stop when at least 4 values have been generated from both these intervals.

SAS Program

```
data example5;
  count1=0;
  count2=0;
  true=0;
  do until (true=1);
  t=ranuni(0);
  if t>0.4 then count1=count1+1; else count2=count2+1;
  if count1>3 and count2>3 then true=1;
  output;
  end;
proc print;
run;
```

SAS Output

OBS	COUNT1	COUNT2	TRUE	J	T
1	1	0	0	1	0.78427
2	2	0	0	2	0.58021
3	3	0	0	3	0.92821
4	3	1	0	4	0.13608
5	4	1	0	5	0.99199
6	5	1	0	6	0.55421
7	5	2	0	7	0.26266
8	5	3	0	8	0.25433
9	5	4	1	9	0.07796