Chapter 2
On Testing for Multivariate Normality

Abstract In this chapter we compare and contrast three approaches for testing multivariate normality. These are, namely, Mardia's skewness and kurtosis statistics and the Henze–Zirkler statistic. Type I errors and power are demonstrated using simulations in both the complete-data and the randomly-incomplete-data cases. In the randomly-incomplete-data case, we use Sidak's method for multiple testing. Examples are also provided.

Keywords Alternative hypothesis • Henze–Zirkler statistic • Mardia's skewness statistic • Mardia's kurtosis statistic • Multivariate • Normality • Null hypothesis • Power • Sidak's method • Type I error

2.1 The Complete-Data Case

Suppose $\mathbf{X} = \left(X_1, \ldots, X_p\right)$ is a p-dimensional ($p > 1$) random vector that has an unknown mean vector $\boldsymbol{\mu} = (\mu_1, \ldots, \mu_p)$ and a positive-definite, symmetric covariance matrix

$$\boldsymbol{\Sigma} = \begin{bmatrix} \sigma_1^2 & \sigma_{21} & \cdots & \sigma_{p1} \\ \sigma_{21} & \sigma_2^2 & \cdots & \sigma_{p2} \\ \vdots & \ldots & \ddots & \vdots \\ \sigma_{p1} & \sigma_{p2} & \cdots & \sigma_p^2 \end{bmatrix}$$

Suppose we collect n data $\mathbf{X}_i = (X_{i1}, \ldots, X_{ip}), i = 1, \ldots, n$, from the distribution of $\mathbf{X}$ above. Consider the five-variate normal distribution with mean row vector {1 2 3 4 5} and covariance matrix

T. Desai, *A Multiple-Testing Approach to the Multivariate Behrens-Fisher Problem: with Simulations and Examples in SAS®*, SpringerBriefs in Statistics,
DOI 10.1007/978-1-4614-6443-3_2, © The Author 2013

$$\mathbf{\Sigma} = \begin{bmatrix} 1 & 0.5 & 0.4 & 0.3 & 0.2 \\ 0.5 & 1 & 0.5 & 0.4 & 0.3 \\ 0.4 & 0.5 & 1 & 0.5 & 0.4 \\ 0.3 & 0.4 & 0.5 & 1 & 0.5 \\ 0.2 & 0.3 & 0.4 & 0.5 & 1 \end{bmatrix}$$

Table 2.1 below gives Type 1 error rates under the null hypothesis that a multivariate random vector is from a multinormal distribution. For each sample size considered, a 1000 simulations were run. The Type I error rates were obtained using Mardia's skewness and kurtosis statistics (Mardia 1974) and the Henze–Zirkler test (Henze and Zirkler 1990). We will denote Mardia's skewness and kurtosis statistics as S and K, respectively. The Henze–Zirkler statistic will be denoted as HZ.

Table 2.1 Type 1 error rates for complete data under the null

Sample size	S	K	HZ
6	0.000	0.000	0.000
10	0.013	0.000	0.012
20	0.061	0.008	0.022
30	0.077	0.018	0.018
40	0.073	0.021	0.027
50	0.077	0.039	0.030
75	0.060	0.027	0.038
100	0.063	0.033	0.039

The above table suggests that, when the null hypothesis of normality is true, K and HZ tend to be conservative, while S tends to be a bit anti-conservative. The SAS® code that generated the null distribution and Table 2.1 is given as follows. Note that the simulation uses a sample size of 100. To use a different sample size, modification of the code is simple and is left as an exercise for the reader.

```
proc iml;
sim = j(1000000, 5, 0);
sigma1 = {1 0.5 0.4 0.3 0.2,
          0.5 1 0.5 0.4 0.3,
          0.4 0.5 1 0.5 0.4,
          0.3 0.4 0.5 1 0.5,
          0.2 0.3 0.4 0.5 1};
mu1 = {1 2 3 4 5};
do i = 1 to 1000000;
   sim[i, 1] = mu1[1, 1] + (sqrt(1) * normal(0));
   mu = mu1[1, 2] + (sigma1[2, 1] * inv(sigma1[1, 1]) * (sim[i, 1] − mu1[1, 1]));
   s_sq = sigma1[2, 2] − (sigma1[2, 1] * inv(sigma[1, 1]) * sigma1[1, 2]);
   sim[i, 2] = mu + (sqrt(s_sq) * normal(0));
   mu = mu1[1, 3] + (sigma1[3, 1 : 2] * inv(sigma1[1 : 2, 1 : 2]) * (sim[i, 1 : 2]^t − mu1[1, 1 : 2]^t));
   s_sq = sigma1[3, 3] − (sigma1[3, 1 : 2] * inv(sigma1[1 : 2, 1 : 2]) * sigma1[1 : 2, 3]);
   sim[i, 3] = mu + (sqrt(s_sq) * normal(0));
```

```
    mu = mu1[1,4] + (sigma1[4,1:3] * inv(sigma1[1:3,1:3]) * (sim[i,1:3]` - mu1[1,1:3]`));
    s_sq = sigma1[4,4] - (sigma1[4,1:3] * inv(sigma1[1:3,1:3]) * sigma1[1:3,4]);
    sim[i,4] = mu + (sqrt(s_sq) * normal(0));
    mu = mu1[1,5] + (sigma1[5,1:4] * inv(sigma1[1:4,1:4]) * (sim[i,1:4]` - mu1[1,1:4]`));
    s_sq = sigma1[5,5] - (sigma1[5,1:4] * inv(sigma1[1:4,1:4]) * sigma1[1:4,5]);
    sim[i,5] = mu + (sqrt(s_sq) * normal(0));

end;
create complete_type1 from sim [colname = {'x1' 'x2' 'x3' 'x4' 'x5'}];
append from sim;
run;
quit;


%macro test;


%do t = 1 %to 1000;
ods listing close;


proc iml;
index = %sysevalf(((&t - 1) * 100) + 1) : %sysevalf(&t * 100);
use complete_type1;
read point index var {x1 x2 x3 x4 x5} into sim;
create sim from sim [colname = {'x1' 'x2' 'x3' 'x4' 'x5'}];

append from sim;
run;
quit;


proc model data = sim;
x1 = parm1;
x2 = parm2;
x3 = parm3;
x4 = parm4;
x5 = parm5;
fit x1 x2 x3 x4 x5 / normal;
ods output NormalityTest = nt
run;
quit;


proc iml;
use nt;
read all var {Prob} into p;
est = p[6,]||p[7,]||p[8,];
create est&t from est [colname = {'s' 'k' 'hz'}];
append from est;
run;
quit;


%end;
```

```
%mend test;

%test;

%macro cat;
   %do i = 1 %to 1000;
    est&i
   %end;
%mend cat;

data out_comp_tp1;
set %cat;
i = _n_;
run;

data _null_;
set out_comp_tp1 end = last;
if s < 0.05 then c1 + 1;
if k < 0.05 then c2 + 1;
if hz < 0.05 then c3 + 1;
if last then do; put c1 c2 c3; end;
run;
```

Now we consider power under an alternative. Consider the mixture of normals which has mean {1 2 3 4 5} with probability 0.5 and {-1 -2 -3 -4 -5} with probability 0.5. Let the covariance matrices be the same in both cases. The SAS® code that generates this data set is as follows:

```
proc iml;
sim = j(1000000, 5, 0);
sigma1 = {1 0.5 0.4 0.3 0.2,
          0.5 1 0.5 0.4 0.3,
          0.4 0.5 1 0.5 0.4,
          0.3 0.4 0.5 1 0.5,
          0.2 0.3 0.4 0.5 1};
do i = 1 to 1000000;
   r = rantbl(i * 10, 0.5, 0.5);
   if r = 1 then mu1 = {1 2 3 4 5};
    else mu1 = {-1 -2 -3 -4 -5};
   sim[i, 1] = mu1[1, 1] + (sqrt(1) * normal(0));
   mu = mu1[1, 2] + (sigma1[2, 1] * inv(sigma1[1, 1]) * (sim[i, 1] - mu1[1, 1]));
   s_sq = sigma1[2, 2] - (sigma1[2, 1] * inv(sigma[1, 1]) * sigma1[1, 2]);
   sim[i, 2] = mu + (sqrt(s_sq) * normal(0));
   mu = mu1[1, 3] + (sigma1[3, 1 : 2] * inv(sigma1[1 : 2, 1 : 2]) * (sim[i, 1 : 2]` - mu1[1, 1 : 2]`));
   s_sq = sigma1[3, 3] - (sigma1[3, 1 : 2] * inv(sigma1[1 : 2, 1 : 2]) * sigma1[1 : 2, 3]);
   sim[i, 3] = mu + (sqrt(s_sq) * normal(0));
   mu = mu1[1, 4] + (sigma1[4, 1 : 3] * inv(sigma1[1 : 3, 1 : 3]) * (sim[i, 1 : 3]` - mu1[1, 1 : 3]`));
   s_sq = sigma1[4, 4] - (sigma1[4, 1 : 3] * inv(sigma1[1 : 3, 1 : 3]) * sigma1[1 : 3, 4]);
   sim[i, 4] = mu + (sqrt(s_sq) * normal(0));
```

```
mu = mu1[1, 5] + (sigma1[5, 1 : 4] * inv(sigma1[1 : 4, 1 : 4]) * (sim[i, 1 : 4]^t - mu1[1, 1 : 4]^t));
s_sq = sigma1[5, 5] - (sigma1[5, 1 : 4] * inv(sigma1[1 : 4, 1 : 4]) * sigma1[1 : 4, 5]);
sim[i, 5] = mu + (sqrt(s_sq) * normal(0));

end;
create complete_power from sim [col name = {'x1' 'x2' 'x3' 'x4' 'x5'}];
append from sim;
```

The macro *test* which we used for examining Type I errors can be again used here except that we replace the data set *complete_type*1 with the data set generated above, namely, *complete_power*.

Then we have the following power table:

Table 2.2 Power under an alternative for complete data

Sample size	S	K	HZ
6	0.000	0.000	0.000
10	0.012	0.000	0.028
20	0.034	0.015	0.092
30	0.037	0.041	0.233
40	0.031	0.071	0.420
50	0.030	0.102	0.658
75	0.027	0.166	0.982
100	0.036	0.217	1.000

Note that in Table 2.2, S doesn't seem to converge to any value, while K increases very slowly compared to the HZ statistic.

2.1.1 Example: Rao's Cork Data

This data set of (Rao, 1948; excerpted from Khattree and Naik, 1999) consists of weights of cork borings in four directions for 28 trees. According to Khattree and Naik, E.S. Pearson believed that this data is asymmetric. The data and the ensuing tests of normality are provided in the SAS® code below:

```
data rao;
input n e s w;
cards;
72 66 76 77
60 53 66 63
56 57 64 58
41 29 36 38
32 32 35 36
30 35 34 26
39 39 31 27
```

```
42 43 31 25
37 40 31 25
33 29 27 36
32 30 34 28
63 45 74 63
54 46 60 52
47 51 52 43
91 79 100 75
56 68 47 50
79 65 70 61
81 80 68 58
78 55 67 60
46 38 37 38
39 35 34 37
32 30 30 32
60 50 67 54
35 37 48 39
39 36 39 31
50 34 37 40
43 37 39 50
48 54 57 43
;
run;

proc model data = rao;
n = parm1;
e = parm2;
s = parm3;
w = parm4;
fit n e s w / normal;
run;
quit;
```

The p-values for Mardia's skewness and kurtosis statistics, along with the p-value for the Henze–Zirkler statistic, are, respectively, 0.2369, 0.6904, and 0.0222. Thus, at the 5 % significance level, the Henze–Zirkler statistic rejects four-variate normality, while Mardia's statistics fail to reject four-variate normality.

2.2 The Randomly-Incomplete-Data Case

To investigate this case, we generated the data set specified in the SAS® code below. Note that with probability 0.7, all components are observed; with probability 0.10, the fifth component is set to missing; and with probability 0.2, the fourth and fifth components are set to missing. We approach this case using multiple imputation (Rubin 1987) and multiple testing. Five imputations were generated for each simulation. For each of the three statistics, since the five p-values corresponding to five imputations are not mutually independent, one cannot use the FDR method of Benjamini and Hochberg (1995). Rather, we chose to use Sidak's procedure (Sidak, 1967). The code for generating the null data set is as follows:

```
proc iml;
sim = j(1000000, 6, 0);
sigma1 = {1 0.5 0.4 0.3 0.2,
          0.5 1 0.5 0.4 0.3,
          0.4 0.5 1 0.5 0.4,
          0.3 0.4 0.5 1 0.5,
          0.2 0.3 0.4 0.5 1};
mu1 = {1 2 3 4 5};
do i = 1 to 1000000;
   sim[i, 1] = mu1[1, 1] + (sqrt(1) * normal(0));
   mu = mu1[1, 2] + (sigma1[2, 1] * inv(sigma1[1, 1]) * (sim[i, 1] - mu1[1, 1]));
   s_sq = sigma1[2, 2] - (sigma1[2, 1] * inv(sigma[1, 1]) * sigma1[1, 2]);
   sim[i, 2] = mu + (sqrt(s_sq) * normal(0));
   mu = mu1[1, 3] + (sigma1[3, 1 : 2] * inv(sigma1[1 : 2, 1 : 2]) * (sim[i, 1 : 2]` - mu1[1, 1 : 2]`));
   s_sq = sigma1[3, 3] - (sigma1[3, 1 : 2] * inv(sigma1[1 : 2, 1 : 2]) * sigma1[1 : 2, 3]);
   sim[i, 3] = mu + (sqrt(s_sq) * normal(0));
   mu = mu1[1, 4] + (sigma1[4, 1 : 3] * inv(sigma1[1 : 3, 1 : 3]) * (sim[i, 1 : 3]` - mu1[1, 1 : 3]`));
   s_sq = sigma1[4, 4] - (sigma1[4, 1 : 3] * inv(sigma1[1 : 3, 1 : 3]) * sigma1[1 : 3, 4]);
   sim[i, 4] = mu + (sqrt(s_sq) * normal(0));
   mu = mu1[1, 5] + (sigma1[5, 1 : 4] * inv(sigma1[1 : 4, 1 : 4]) * (sim[i, 1 : 4]` - mu1[1, 1 : 4]`));
   s_sq = sigma1[5, 5] - (sigma1[5, 1 : 4] * inv(sigma1[1 : 4, 1 : 4]) * sigma1[1 : 4, 5]);
   sim[i, 5] = mu + (sqrt(s_sq) * normal(0));
  r = rantbl(0, 0.7, 0.1, 0.2);
  if r = 2 then sim[i, 5] = .;
  if r = 3 then do; sim[i, 4] = .; sim[i, 5] = .; end;
  sim[i, 6] = r;

end;
create incomplete_type1 from sim [colname = {'x1' 'x2' 'x3' 'x4' 'x5'}];
append from sim;
run;
quit;
```

The estimated Type I errors using five imputations are given below in Table 2.3.

Table 2.3 Type I error using five imputations

Sample size	Statistic		
n	S	K	HZ
20	0.040	0.000	0.013
30	0.052	0.001	0.016
40	0.058	0.002	0.017
50	0.050	0.004	0.017
75	0.052	0.007	0.019
100	0.046	0.008	0.023

The above table illustrates that the S statistic tends to remain near the nominal level, whereas the K and HZ statistics tend to approach the nominal level extremely slowly from below. However, none of the three statistics is anti-conservative.

The SAS® code for the macro that generated the above table is as follows:

```
%macro incomplete;

%do t = 1 %to 1000;
ods listing close;

proc iml;
index = %sysevalf(((&t − 1) * 100) + 1) : %sysevalf(&t * 100);
use incomplete_type1;
read point index var {x1 x2 x3 x4 x5 r} into x;
create x from x [colname = {'x1' 'x2' 'x3' 'x4' 'x5' 'r'}];
append from x;
run;
quit;

proc mi data = x nimpute = 5 out = bayes;
var x1 x2 x3 x4 x5;
mcmc;
run;

proc sort data = bayes;
by _Imputation_;
run;

proc model data = bayes;
x1 = parm1;
x2 = parm2;
x3 = parm3;
x4 = parm4;
x5 = parm5;
fit x1 x2 x3 x4 x5 / normal;
by _Imputation_;
ods output NormalityTest = nt1;
run;
quit;

proc sort data = nt1;
by test;
run;

proc iml;
use nt1;
```

```
read all var {prob} into p;
est = p[1 : 15,]`;
create est&t from est [colname = {'hz1' 'hz2' 'hz3' 'hz4' 'hz5' 'mk1' 'mk2' 'mk3' 'mk4' 'mk5'
                                   'ms1' 'ms2' 'ms3' 'ms4' 'ms5'}];
append from est;
run;
quit;


%end;
%mend incomplete;


%incomplete;


data out_incomp_tp1;
set %cat;
i = _n_;
run;


data _null_;
set out_incomp_tp1 end = last;
k = 1 - (0.95 * *(1/5));
if ms1 < k or ms2 < k or ms3 < k or ms4 < k or ms5 < k then c1 + 1;
if mk1 < k or mk2 < k or mk3 < k or mk4 < k or mk5 < k then c2 + 1;
if hz1 < k or hz2 < k or hz3 < k or hz4 < k or hz5 < k then c3 + 1;
if last then do; put c1 c2 c3; end;
run;
```

Table 2.4 below displays the power under the mixture alternative considered for the complete-data case. Missing data was simulated in exactly the same manner as for the null case considered above.

Table 2.4 Power using five imputations

Sample size	Statistic		
n	S	K	HZ
20	0.021	0.000	0.049
30	0.021	0.000	0.160
40	0.031	0.008	0.317
50	0.032	0.018	0.527
75	0.019	0.059	0.936
100	0.031	0.092	0.999

Table 2.4 shows that of the three statistics, HZ is the most powerful. It is not clear if the power of the S statistic increases with increase in sample size, whereas the power of the K statistic increases very slowly with increasing sample size.

2.2.1 Example: Audiology Growth Data

Nunez-Anton and Woodworth (1994) present and analyze data from the Iowa Cochlear Implant Project (Gantz et al. 1988). The data consist of percentages of correct scores on a sentence test administered to two groups of deaf patients fitted with two different cochlear implants. Measurements were made 1, 9, 18, and 30 months after the fitting of implants. The data is presented in the SAS® data step below. Note that Nunez-Anton and Woodworth use data for only those patients who reached at least 5 % understanding. Thus, of the total sample size of 44, data for 9 patients were deleted bringing the total sample size down to 35. Following the data step, the code for the tests of normality done using five Bayesian imputations is presented for group 1. The corresponding code for group 0 is similar and not presented. In the data step below, note that the variable r indicates the type of missing-data pattern:

```
data audio;
input group x1 x2 x3 x4 r;
id = _n_;
cards;
1 28.57 53.00 57.83 59.22 1
1 . 13.00 21.00 26.50 2
1 60.37 86.41 . . 4
1 33.87 55.60 61.06 . 3
1 1.61 0.69 . . 4
1 26.04 61.98 67.28 . 3
1 . 59.00 66.80 83.20 2
1 11.29 38.02 . . 4
1 0.00 0.00 0.00 2.76 1
1 . 35.10 37.79 54.80 2
1 16.00 33.00 45.39 40.09 1
1 40.55 50.69 41.70 52.07 1
1 3.90 11.06 4.15 14.90 1
1 1.80 2.30 2.53 2.53 1
1 0.00 17.74 44.70 48.85 1
1 64.75 84.50 92.40 95.39 1
1 38.25 81.57 89.63 . 3
1 67.50 91.47 92.86 . 3
1 45.62 58.00 . . 4
1 0.00 0.00 37.00 . 3
1 51.15 66.13 . . 4
1 0.00 48.16 . . 4
1 0.00 0.92 . . 4
0 . 0.00 0.90 1.61 2
```

```
0 0.00 0.00 0.00 . 3
0 0.00 0.00 . . 4
0 8.76 24.42 . . 4
0 0.00 20.79 27.42 31.80 1
0 2.30 12.67 28.80 24.42 1
0 12.90 28.34 . . 4
0 . 45.50 43.32 36.80 2
0 68.00 96.08 97.47 99.00 1
0 20.28 41.01 51.15 61.98 1
0 65.90 81.30 71.20 70.00 1
0 0.00 8.76 16.59 14.75 1
0 0.00 0.00 0.00 0.00 1
0 9.22 14.98 9.68 . 3
0 11.29 44.47 62.90 68.20 1
0 30.88 29.72 . . 4
0 29.72 41.40 64.00 . 3
0 0.00 43.55 48.16 . 3
0 0.00 0.00 . . 4
0 8.76 60.00 . . 4
0 8.00 25.00 30.88 55.53 1
;
run;

data audio_;
set audio;
if _n_ in (5, 9, 14, 23, 24, 25, 26, 36, 42) then delete;
run;

data audio1;
set audio_;
if group = 0 then delete;
run;

proc mi data = audio1 nimpute = 5 out = bayes1;
var x1 x2 x3;
mcmc;
run;

proc sort data = bayes1;
by _Imputation_;
run;
```

```
proc model data = bayes1;
x1 = parm1;
x2 = parm2;
x3 = parm3;
fit x1 x2 x3 / normal;
by _Imputation_;
ods output NormalityTest = nt1;
run;
quit;


proc sort data = nt1;
by test;
run;
```

Five imputations were drawn. The decisions reached by the S, K, and HZ statistics using Sidak's procedure for both groups are presented in Table 2.5 below.

Table 2.5 Decisions for tests of multivariate normality of the audiology data

	Statistic		
Group	S	K	HZ
1	Re*ject* = *No*	Re*ject* = *No*	Re*ject* = *No*
0	Re*ject* = *No*	Re*ject* = *No*	Re*ject* = *No*

Springer

http://www.springer.com/978-1-4614-6443-3

A Multiple-Testing Approach to the Multivariate Behrens-Fisher Problem
with Simulations and Examples in SAS®
Desai, T.
2013, V, 55 p.,
ISBN: 978-1-4614-6443-3