Approximations to the distributions of quadratic forms in normal variables

Sha Yu

University of North Carolina at Charlotte

December 7, 2015

Overview

- Consider the quadratic form $Q(X) = X'AX$
- X follows an n-dimensional multivariate normal distribution with mean vector μ_x and non-singular variance matrix Σ
- A is an $n \times n$ symmetric and non-negative definite matrix
- A problem of interest is to evaluate the probability

$$Pr(Q(X) > q \tag{1}$$

- The quadratic form Q can be expressed as a weighted sum of independent chi-square random variables

$$Q = X'AX = Y'\Lambda Y = \sum_{r=1}^{m} \lambda_r \chi^2_{hr}(\delta_r)$$

  - P be a $n \times n$ orthonormal matrix which converts $B = \Sigma^{1/2} A \Sigma^{1/2}$ to the diagnoal form $\Lambda = diag(\lambda_1, ..., \lambda_n) = PBP'$, where $\lambda_1 \geq ...\lambda_n \geq 0$
  - $Y = P\Sigma^{-1/2}X$ is normally distributed with mean $P\Sigma^{-1/2}\mu_x$ and variance In
  - $m = rank(A)$; $h_r = 1$; $\delta_r = \mu^2_{yr}$, and μ_{yr} is the ith component of μ_y

- Why do we study the distribution of quadratic forms?
- Estimating the tail probability of Q(x) arises in many statistical applications.
    1. The power analysis of a test procedure if the (asymptotic) distribution test statistic(e.g., Peason's chi-square statistic) takes the form Q(X).
    2. Chernoff-Lehmann test statistic for goodness-of-fit to a fixed distribution converges to quadratic forms under the null hypothese.
    3. In time series analysis, Box-Pierece-Ljung portmanteau test statistic for lack of fit in ARMA models converges in distribution to a weighted sum of non-central chi-square random variables under local alternatives.

- Why do we approximate the distribution of quadratic forms?
- Computing (1) is usually not straightforward except in some special cases.

  e.g., when $A\Sigma$ is idempotent matrix of rank p,then

$$Q(X) \sim \chi^2_{p,\lambda}$$

  where $\lambda = {\mu_x}' A \mu_x$

- Otherwise, there is no closed analytic expressions.

- Many methods have been proposed to compute (1).

1. Imhof's approximation
2. Farebrother's approximation
3. Pearson three-moment central χ^2 approximation
4. Liu-Tang-Zhang's four-moment non-central χ^2 approximation
5. Kuonen's saddle point approximation

- Imhof's approximation

- Imhof proposed this method relying on numerical inversion of the

  characteristic function of $Q(x) = \sum_{r=1}^{m} \lambda_r \chi^2_{hr}(\delta_r)$

$$F(x) = \frac{1}{2} - \frac{1}{\pi} \int_0^\infty t^{-1} g(e^{-itx}\phi(t))dt$$

  where $g(z)$ denots the imaginary part of z;

  $\phi(t)$ is the characteristic function of t

$$\phi(t) = \sum_{r=1}^{m} (1 - 2i\lambda_r t)^{-\frac{h_r}{2}} exp(i \sum_{r=1}^{m} \frac{\delta_r^2 \lambda_r t}{1 - 2i\lambda_r t})$$

- After the substitution $2t = u$ is made, we have

$$Pr(Q(X) > q \approx \frac{1}{2} + \frac{1}{\pi}\int_0^{\infty} \frac{\sin\theta(u)}{u\rho(u)}du \tag{2}$$

where

$$\theta(u) = \frac{1}{2}\sum_{r=1}^{m}[h_r \arctan(\lambda_r u) + \delta_r^2 \lambda_r u(1+\lambda_r^2 u^2)^{-1}] - \frac{1}{2}qu$$

$$\rho(u) = \prod_{r=1}^{m}(1+\lambda_r^2 u^2)^{\frac{h_r}{4}} exp(\frac{\sum_{r=1}^{m}(\delta_r \lambda_r u)^2}{2(1+\lambda_r^2 u^2)})$$

- Hard to compute but can be accurate by bonding the approximation error.
- $\lambda_r, h_r and \delta_r$ must be determinated explicitly.
- However, it is very time-consuming and challenging to convert quadratic forms into weighted sums of chi-square if A is complex and high dimensional.

- Farebrother's approach

- Farebrother based on the results of Ruben (1962) exploit that (1) can be written as an infinite series of central chi-square distributions.

$$Pr(Q(X) > q) = \sum_{k=0}^{\infty} c_k Pr(\chi^2_{2k+\tilde{h}} > \frac{t}{\beta}) \qquad (3)$$

For any $0 < \beta < min(\lambda_1, ...\lambda_m)$, where $\tilde{h} = \sum_{r=1}^{m} h_r$, $\gamma_r = 1 - \beta/\lambda_r$

$g_k = [\sum_{r=1}^{m} h_r \gamma_r^k + k \sum_{r=1}^{m} \delta_r \gamma_r^{k-1}(1-\gamma_r)]/2$

$c_0 = \prod r = 1^m (\frac{\beta}{\lambda_r})^{\frac{h_r}{2}}; exp(-\frac{\sum_{r=1}^{m} \delta_r}{2})$; $c_k = k^{-1} \sum_{k=o}^{k-r} g_{k-r} c_r$ for $k \geq 1$

- Share the similar properties with Imhof's approximation

- Pearson three-moment central χ^2 approximation
- Pearson's method essentially uses the central χ^2 approximation by reqiring the match of the third-order moment.

$$Pr(Q(X) > q) \approx Pr(\chi^2_{l^*} > l^* + t^*\sqrt{2l^*}) \tag{4}$$

where $l^* = 1/s_1^2$ is determined so that Q(X) and $\chi^2_{l^*}$ have equal skewness.

$t^* = \frac{t-\mu_Q}{\sigma_Q}$ and skewness of Q(X) is $\beta_1 = \sqrt{8}s_1$

- This method is easy to compute and no need to convert quadratic forms into the weighted sum of chi-square.

- Liu-Tang-Zhang's four-moment non-central χ^2 approximation

$$Pr(Q(X) > q) \approx Pr(\chi_l^2(\delta) > t^*\sigma_X + \mu_X)) \tag{5}$$

where $t^* = \frac{t-\mu_Q}{\sigma_Q}$, $\mu_x = l + \delta$, $\sigma_X = \sqrt{2(l+2\delta)}$

- THe parameter δ and l are determined so that the skewness of Q(X) and χ_l^2 are equal and the difference between the kurtosis is minimized.
- This approach does not involve inverting a matrix and it has more accuracy than Pearson's approximation.

- How to decide δ and l ?
- skewness of Q(X) is $\beta_1 = \frac{\kappa_3}{\kappa_2^{3/2}} = \sqrt{8}s_1$kurtosis of Q(X) is $\beta_2 = \frac{\kappa_4}{\kappa_2^2} = 12s_2$

  k_{th} cumulant of Q(X) is $\kappa_k = 2^{k-1}(k-1)!(\sum_{r=1}^{m} \lambda_r^k h_r + k\sum_{r=1}^{m} \lambda_r^k \delta_r)$

  1. If $s_1^2 > s - 2$

     $a = 1/(s_1 - \sqrt{s_1^2 - s_2})$, $\delta = s_1 a^3 - a^2$ and $l = a^2 - 2\delta$

  2. If $s_1^2 \leq s_2$

     $a = 1/s_1$, $\delta = s_1^3 - a^2 = 0$ and $l = 1/s_1^2$

     In the case, the result is the same as Peason's approximation.

- Kuonen's saddle point approximation

- By contrast to the Pearson's method, this method use the entire cumulant generating function.

$$Pr(Q(X) > q) \approx 1 - \Phi(w + \frac{1}{w} log(\frac{v}{w})) \tag{6}$$

where $w = sign(\hat{\xi})[2(\hat{\xi}q - K(\hat{\xi}))]^{\frac{1}{2}}$, $v = \hat{\xi}(K^{''}(\hat{\xi}))^{\frac{1}{2}}$

$\hat{\xi} = \hat{\xi}(q)$ is the saddle point which satisfies the equation $K^{'}(\hat{\xi}) = q$

$K(\xi) = -\frac{1}{2}\sum_{r=1}^{m} h_r log(1 - 2\xi\lambda_r) + \sum_{r=1}^{m} \frac{\delta_r \lambda_r \xi}{1-2\xi\lambda_r}$, $\xi < \frac{1}{2} min\lambda_r^{-\frac{1}{2}}$

- This method gives highly accurate approximations.

  $\lambda_1 \geq ... \geq \lambda_m$ must be determined explicity.

- We consider five quadratic forms and compute the values of (1) given different values of q. Use Farebrother's method as gold standard.

1. $Q_1 = 0.5\chi_1^2(1) + 0.4\chi_2^2(0.6) + 0.1\chi_1^2(0.8)$
2. $Q_2 = 0.9\chi_1^2(0.2) + 0.1\chi_2^2(10)$
3. $Q_3 = 0.1\chi_1^2(0.2) + 0.9\chi_2^2(10)$
4. $Q_4 = 0.5\chi_1^2(1) + 0.4\chi_2^2(0.6) + \sum_{r=1}^{10} 0.01\chi_r^2(0.8)$
5. $Q_5 = \chi_1^1(1.0) + (0.6)^4\chi_1^2(7.0)$

   If a time series follows a first-order ARMA model with parameter α, the Box-Pirece-Ljung test statistic asymptotic follows $\chi_{m-1}^2 + \alpha^{2m}\chi_1^2$.

   We set $m = 2$, $\alpha = 0.6$.

- R demo
- Package CompQuadForm (2015) by P.Lafaye
  - $imhof(q, \lambda, h, \delta, epsabs = 10^{-6}, epsrel = 10^{-6})$
  - $farebrother(q, \lambda, h, \delta, epsabs = 10^{-6}, eps = 10^{-6})$
  - $liu(q, \lambda, h)$

- R demo of Pearson's approximation

```
### Pearson three-moment chi-square approximation ###

## simulation for Q1 ##
t <- seq(1,11,0.1); P1<-c()
for(i in 1:length(t))
{
lambda1 <- 0.5
lambda2 <- 0.4
lambda3 <- 0.1
h1 <- 1
h2 <- 2
h3 <- 1
delta1 <- 1
delta2 <- 0.6
delta3 <- 0.8

c1 <- lambda1*h1+lambda2*h2+lambda3*h3+lambda1*delta1+lambda2*delta2+lambda3*delta3
c2 <- lambda1^2*h1+lambda2^2*h2+lambda3^2*h3+2*lambda1^2*delta1+2*lambda2^2*delta2+2*lambda3^2*delta3
c3 <- lambda1^3*h1+lambda2^3*h2+lambda3^3*h3+3*lambda1^3*delta1+3*lambda2^3*delta2+3*lambda3^3*delta3

s1 <- c3/(c2^(3/2))
l_star <- 1/(s1^2)
t_star <- (t[i]-c1)/sqrt(2*c2)
q <- l_star + t_star*sqrt(2*l_star)
df <- l_star

P1[i] <- pchisq(q, df, ncp=0, lower.tail=FALSE)
}
P11 <- rbind(P1[11], P1[31], P1[51], P1[71],P1[91])
```

- R demo of Kuonen's saddle point approximation

```
## simulation for Q1 ##
t <- seq(1,11,0.1); S1 <- c()
for ( i in 1:length(t))
{
lambda1 <- 0.5
lambda2 <- 0.4
lambda3 <- 0.1
h1<- 1
h2 <- 2
h3 <- 1
delta1 <- 1
delta2 <- 0.6
delta3 <- 0.8

K <-function(x)
(-(1/2)*h1*log(1-2*x*lambda1)-(1/2)*h2*log(1-2*x*lambda2)-(1/2)*h3*log(1-2*x*lambda3)
+ delta1*lambda1*x/(1-2*x*lambda1)+delta2*lambda2*x/(1-2*x*lambda2)+delta3*lambda3*x/(1-2*x*lambda3))
g <- function(x) {}       ## first derivative ##
body(g) <- D(body(K), 'x')
h <- function(x) {}       ## second derivative ##
body (h) <- D(body(g), 'x')

g.new <- function(x) (g(x)-t[i])
epsi <- uniroot.all(g.new, c(-10,10))[uniroot.all(g.new, c(-10,10))<min(1/(2*lambda1),1/(2*lambda2),
1/(2*lambda3))]
w <- sign(epsi)*(2*(epsi*t[i]-K(epsi)))^(1/2)
v <- epsi*(h(epsi))^(1/2)
S1[i] <-pnorm((w+(1/w)*log(v/w)),mean=0, sd=1, lower.tail=FALSE)
}
S11 <- rbind(S1[11], S1[31], S1[51], S1[71], S1[91])
```

- R output

	Quadratic form	q	Farebrother	Imhof	AE_I	Pearson	AE_P	LTZ	AE_L	Saddle	AE_S
1	Q1	2	0.45746068	0.45746066	0.0000000	0.4589672107	0.0015065	0.45775299	0.0002923	0.45947844	0.0020178
2	Q1	4	0.12894336	0.12894327	0.0000001	0.1284892697	0.0004541	0.12894184	0.0000015	0.13017604	0.0012327
3	Q1	6	0.03110907	0.03110888	0.0000002	0.0309291459	0.0001799	0.03107919	0.0000299	0.03149375	0.0003847
4	Q1	8	0.00688557	0.00688536	0.0000002	0.0069078478	0.0000223	0.00688262	0.0000029	0.00698636	0.0001008
5	Q1	10	0.00144056	0.00144037	0.0000002	0.0014750467	0.0000345	0.00144230	0.0000017	0.00146387	0.0000233
6	Q2	2	0.45424639	0.45424635	0.0000000	0.4262063860	0.0280400	0.42620639	0.0280400	0.43591378	0.0183326
7	Q2	4	0.11949795	0.11949789	0.0000001	0.1268720507	0.0073741	0.12687205	0.0073741	0.12287075	0.0033728
8	Q2	6	0.03712131	0.03712123	0.0000001	0.0390284424	0.0019071	0.03902844	0.0019071	0.03946821	0.0023469
9	Q2	8	0.01226277	0.01226249	0.0000003	0.0121693371	0.0000934	0.01216934	0.0000934	0.01321391	0.0009511
10	Q2	10	0.00414786	0.00414755	0.0000003	0.0038232768	0.0003246	0.00382328	0.0003246	0.00449971	0.0003519
11	Q3	2	0.97455514	0.97455505	0.0000001	0.9716175584	0.0029376	0.97454343	0.0000117	0.97493203	0.0003769
12	Q3	6	0.78391229	0.78391227	0.0000000	0.7880156847	0.0041034	0.78391149	0.0000008	0.78488589	0.0009736
13	Q3	10	0.50245578	0.50245554	0.0000002	0.5040683417	0.0016126	0.50246012	0.0000043	0.50323813	0.0007823
14	Q3	14	0.26728633	0.26728654	0.0000002	0.2658453208	0.0014410	0.26728725	0.0000009	0.26797605	0.0006897
15	Q3	18	0.12291737	0.12291727	0.0000001	0.1215023756	0.0014150	0.12291637	0.0000010	0.12324844	0.0003311
16	Q4	2	0.58404624	0.58404618	0.0000001	0.2853088807	0.2987374	0.58439862	0.0003524	0.58626157	0.0022153
17	Q4	4	0.17303507	0.17303503	0.0000000	0.0393036993	0.1337314	0.17299841	0.0000367	0.17459753	0.0015625
18	Q4	6	0.04274713	0.04274705	0.0000001	0.0048637379	0.0378834	0.04272981	0.0000173	0.04326675	0.0005196
19	Q4	8	0.00959839	0.00959834	0.0000000	0.0005742492	0.0090241	0.00960121	0.0000028	0.00973690	0.0001385
20	Q4	10	0.00202783	0.00202773	0.0000001	0.0000660179	0.0019618	0.00202959	0.0000018	0.00206031	0.0000325
21	Q5	2	0.56323112	0.56323120	0.0000001	0.5548498412	0.0083813	0.55382310	0.0094080	0.28132044	0.2819107
22	Q5	4	0.24984294	0.24984287	0.0000001	0.2543967464	0.0045538	0.25419378	0.0043508	0.08118358	0.1686594
23	Q5	6	0.11504293	0.11504291	0.0000000	0.1156894950	0.0006466	0.11584461	0.0008017	0.01685425	0.0981887
24	Q5	8	0.05293334	0.05293302	0.0000003	0.0524102404	0.0005231	0.05255468	0.0003787	0.00336670	0.0495666
25	Q5	10	0.02413091	0.02413087	0.0000000	0.0236894779	0.0004414	0.02375866	0.0003723	0.00336670	0.0207642

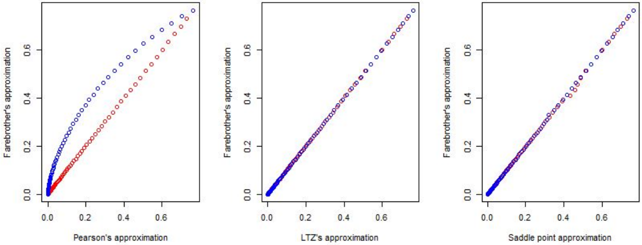
- R output

	Quadratic form	q	Farebrother	Imhof	RE_I(%)	Pearson	RE_P(%)	LTZ	RE_L(%)	Saddle	RE_S(%)
1	Q1	2	0.45746068	0.45746066	0.00000	0.4589672107	0.32932	0.45775299	0.06390	0.45947844	0.44109
2	Q1	4	0.12894336	0.12894327	0.00008	0.1284892697	0.35217	0.12894184	0.00116	0.13017604	0.95600
3	Q1	6	0.03110907	0.03110888	0.00064	0.0309291459	0.57829	0.03107919	0.09611	0.03149375	1.23662
4	Q1	8	0.00688557	0.00688536	0.00290	0.0069078478	0.32387	0.00688262	0.04212	0.00698636	1.46393
5	Q1	10	0.00144056	0.00144037	0.01388	0.0014750467	2.39490	0.00144230	0.11801	0.00146387	1.61742
6	Q2	2	0.45424639	0.45424635	0.00000	0.4262063860	6.17286	0.42620639	6.17286	0.43591378	4.03583
7	Q2	4	0.11949795	0.11949789	0.00008	0.1268720507	6.17090	0.12687205	6.17090	0.12287075	2.82248
8	Q2	6	0.03712131	0.03712123	0.00027	0.0390284424	5.13748	0.03902844	5.13748	0.03946821	6.32224
9	Q2	8	0.01226277	0.01226249	0.00245	0.0121693371	0.76166	0.01216934	0.76166	0.01321391	7.75600
10	Q2	10	0.00414786	0.00414755	0.00723	0.0038232768	7.82572	0.00382328	7.82572	0.00449971	8.48389
11	Q3	2	0.97455514	0.97455505	0.00001	0.9716175584	0.30143	0.97454343	0.00120	0.97493203	0.03867
12	Q3	6	0.78391229	0.78391227	0.00000	0.7880156847	0.52345	0.78391149	0.00010	0.78488589	0.12420
13	Q3	10	0.50245578	0.50245554	0.00004	0.5040683417	0.32094	0.50246012	0.00086	0.50323813	0.15570
14	Q3	14	0.26728633	0.26728654	0.00007	0.2658453208	0.53912	0.26728725	0.00034	0.26797605	0.25804
15	Q3	18	0.12291737	0.12291727	0.00008	0.1215023756	1.15118	0.12291637	0.00081	0.12324844	0.26937
16	Q4	2	0.58404624	0.58404618	0.00002	0.2853088807	51.14961	0.58439862	0.06034	0.58626157	0.37930
17	Q4	4	0.17303507	0.17303503	0.00000	0.0393036993	77.28572	0.17299841	0.02121	0.17459753	0.90300
18	Q4	6	0.04274713	0.04274705	0.00023	0.0048637379	88.62209	0.04272981	0.04047	0.04326675	1.21552
19	Q4	8	0.00959839	0.00959834	0.00000	0.0005742492	94.01685	0.00960121	0.02917	0.00973690	1.44295
20	Q4	10	0.00202783	0.00202773	0.00493	0.0000660179	96.74365	0.00202959	0.08876	0.00206031	1.60270
21	Q5	2	0.56323112	0.56323120	0.00002	0.5548498412	1.48807	0.55382310	1.67036	0.28132044	50.05240
22	Q5	4	0.24984294	0.24984287	0.00004	0.2543967464	1.82267	0.25419378	1.74141	0.08118358	67.50617
23	Q5	6	0.11504293	0.11504291	0.00000	0.1156894950	0.56205	0.11584461	0.69687	0.01685425	85.34962
24	Q5	8	0.05293334	0.05293302	0.00057	0.0524102404	0.98822	0.05255468	0.71543	0.00336670	93.63966
25	Q5	10	0.02413091	0.02413087	0.00000	0.0236894779	1.82919	0.02375866	1.54283	0.00336670	86.04814

- Compare Q_1 and Q_4

  $Q_1 = 0.5\chi_1^2(1) + 0.4\chi_2^2(0.6) + 0.1\chi_1^2(0.8)$ (red dots)

  $Q_4 = 0.5\chi_1^2(1) + 0.4\chi_2^2(0.6) + \sum_{r=1}^{10} 0.01\chi_r^2(0.8)$ (blue dots)

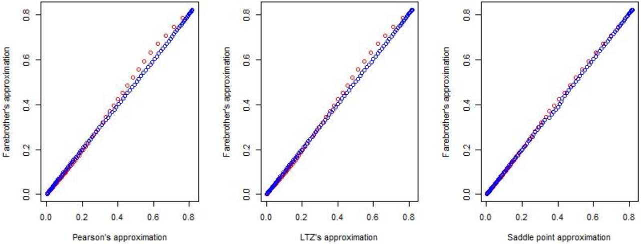


- Pearson's approximation performs poorly when the number of terms of chi-square increases.
- The number of terms of chi-square does not affact the accuracy of LTZ's method and Saddle point method.

- Compare Q_2 and Q_3

$Q_2 = 0.9\chi_1^2(0.2) + 0.1\chi_2^2(10)$ (red dots)

$Q_3 = 0.1\chi_1^2(0.2) + 0.9\chi_2^2(10)$ (blue dots)

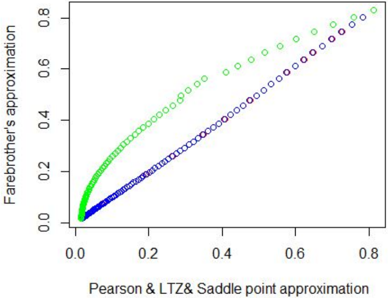


- Pearson's approximation performs better if the chi-square term with large non-centralility parameter has large weight.
- Weight of the con-central chi-square term does not affact the accuracy of saddle point approximation.

- What about Q5?

$Q_5 = \chi_1^1(1.0) + (0.6)^4\chi_1^2(7.0)$

Pearson-red dots, LTZ-blue dots; Saddle point-green dots



- Saddle point approximation performs poorly for Q5.

- Both of LTZ's approximation and Pearson's approximation have high accuracy.

- Conclusion

1. Farebrother's method and Imhof's method differ very little.
2. Pearson's method performs better when the chi-square term with large non-centrality parameter has large weight. It is not appliable when the number of chi-square terms is large.
3. LTZ's method perfoms much better compared to Pearson's method and it is the most accuracy except the Imhof's method. This method show great advantage if A is complex and high dimensional and thus has great realistic meaning.
4. Saddle point also has high accuracy but it can not be applied if the sum of coefficeints of chi-square terms is not 1.(Box-Pierce-Ljung statistic)