

The qprod Toolbox

Expectations of Products of Quadratic Forms in Normal Random Variables
User Manual, Version 1.3

Raymond Kan Jiening Pan

July 24, 2026

Contents

1	Overview	2
2	Requirements and installation	2
3	Background and notation	3
4	Numerical evaluation	3
5	Symbolic expansion	4
6	Counting the number of terms	4
7	Auxiliary functions	5
8	Worked examples	5
8.1	A product of three quadratic forms	5
8.2	Symbolic expansions	5
8.3	Repeated quadratic forms	6
8.4	Complex Hermitian quadratic forms	6
8.5	Wishart product moments	6
8.6	Moments of a single quadratic form	7
8.7	Counting	7
9	Performance notes	7
10	Testing	8
11	Version history	8

1 Overview

The `qprod` toolbox computes the expectation of a product of quadratic forms in normal random vectors,

$$\mathbb{E} \left[\prod_{i=1}^k (\mathbf{z}^T \mathbf{A}_i \mathbf{z})^{s_i} \right], \quad \mathbf{z} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma}), \quad (1)$$

and its complex analogue $\mathbb{E}[\prod_{i=1}^k (\mathbf{z}^H \mathbf{A}_i \mathbf{z})^{s_i}]$ for $\mathbf{z} \sim \mathcal{CN}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ with Hermitian $\mathbf{A}_i$. The toolbox provides three kinds of routines:

1. *numerical evaluation* of (1) for given matrices (Section 4);
2. *symbolic expansion*: generation of the explicit expression of (1) as a linear combination of products of trace terms, with each distinct term listed exactly once together with its integer coefficient (Section 5);
3. *counting*: the number of distinct terms in these expansions (Section 6).

The algorithms implement the explicit formulas derived in the companion paper:

R. Kan and J. Pan, *An explicit expression for the expectation of a product of quadratic forms in multivariate normal random variables*, working paper, University of Toronto, 2026.

Section, proposition, and equation numbers cited below refer to that paper. Please cite the paper when using the toolbox. Bug reports are welcome at Raymond.Kan@rotman.utoronto.ca.

2 Requirements and installation

The toolbox is a collection of plain `.m` files with no toolbox dependencies. It runs under

- MATLAB R2019a or later, or
- GNU Octave 7 or later.

To install, copy all files into a directory and add it to the search path, e.g. `addpath('c:/qprod')`. To verify the installation, run

```
>> test_all
...
===== 63 passed, 0 failed =====
```

`nthperm.mexw64` is an optional compiled helper for 64-bit Windows MATLAB; on other platforms the plain `nthperm.m` is used automatically.

One portability note: the expression-generating functions of Section 5 return a *string array* under MATLAB and a *cell array of character vectors* under Octave (where string arrays do not exist). Both can be indexed element by element; code that should run on both platforms can normalize with `if ~iscell(ys), ys = cellstr(ys); end`.

3 Background and notation

Let $\mathbf{z} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ with $\boldsymbol{\Sigma}$ positive semidefinite, and let $\mathbf{A}_1, \dots, \mathbf{A}_k$ be symmetric. For a word $\boldsymbol{\nu} = (\nu_1, \dots, \nu_p)$ of indices, define

$$\tau_{\boldsymbol{\nu}} = \text{tr}(\mathbf{A}_{\nu_1} \boldsymbol{\Sigma} \cdots \mathbf{A}_{\nu_p} \boldsymbol{\Sigma}), \quad \theta_{\boldsymbol{\nu}} = \boldsymbol{\mu}^T \mathbf{A}_{\nu_1} \boldsymbol{\Sigma} \cdots \boldsymbol{\Sigma} \mathbf{A}_{\nu_p} \boldsymbol{\mu}. \quad (2)$$

The paper shows (Propositions 1 and 2) that (1) with all $s_i = 1$ is an integer linear combination of products of τ - and θ -terms, that the expansion can be written with no repeated terms, and that the terms can be enumerated efficiently by iterating over set partitions of $\{1, \dots, k\}$. Proposition 3 gives the complex case: the coefficients change, and every τ - and θ -term is replaced by the real part of the corresponding expression in (2) (with $\mathbf{T}$ replaced by $\mathbf{H}$). Proposition 4 covers repeated quadratic forms ($s_i > 1$), in which case the number of distinct terms drops.

In the symbolic output of the toolbox, the term $\tau_{\boldsymbol{\nu}}$ is printed as `a_ $\nu_1$ _ $\nu_2$ _...` and $\theta_{\boldsymbol{\nu}}$ as `h_ $\nu_1$ _ $\nu_2$ _...`. For example,

$$\begin{aligned} 4*\text{h_1_2} & \text{ stands for } 4\theta_{(1,2)} = 4\boldsymbol{\mu}^T \mathbf{A}_1 \boldsymbol{\Sigma} \mathbf{A}_2 \boldsymbol{\mu}, \\ 2*\text{a_1_2_2} & \text{ stands for } 2\tau_{(1,2,2)} = 2\text{tr}(\mathbf{A}_1 \boldsymbol{\Sigma} \mathbf{A}_2 \boldsymbol{\Sigma} \mathbf{A}_2 \boldsymbol{\Sigma}). \end{aligned}$$

For the single-matrix moment $\mathbb{E}[(\mathbf{z}^T \mathbf{A} \mathbf{z})^k]$ (functions `qmom*`), the indices refer instead to powers: `a_ $i$` stands for $\tilde{\tau}_i = \text{tr}((\mathbf{A} \boldsymbol{\Sigma})^i)$ and `h_ $i$` for $\tilde{\theta}_i = \boldsymbol{\mu}^T (\mathbf{A} \boldsymbol{\Sigma})^{i-1} \mathbf{A} \boldsymbol{\mu}$.

Wishart product moments. If $\mathbf{W} \sim \mathcal{W}_m(n, \boldsymbol{\Sigma}, \boldsymbol{\Lambda})$ (noncentral real Wishart with n degrees of freedom and noncentrality matrix $\boldsymbol{\Lambda}$), then $\mathbb{E}[\prod_i \text{tr}(\mathbf{W} \mathbf{A}_i)^{s_i}]$ is obtained from the same expansions by substituting (Corollaries 2 and 3 of the paper)

$$\tau_{\boldsymbol{\nu}} \rightarrow n \text{tr}(\mathbf{A}_{\nu_1} \boldsymbol{\Sigma} \cdots \mathbf{A}_{\nu_p} \boldsymbol{\Sigma}), \quad \theta_{\boldsymbol{\nu}} \rightarrow \text{tr}(\mathbf{A}_{\nu_1} \boldsymbol{\Sigma} \cdots \boldsymbol{\Sigma} \mathbf{A}_{\nu_p} \boldsymbol{\Lambda}), \quad (3)$$

with real parts taken in the complex Wishart case. Example 8.5 demonstrates this.

4 Numerical evaluation

All routines below take an $m \times m \times k$ array $\mathbf{A}$ whose i -th slice $\mathbf{A}(:, :, i)$ is $\mathbf{A}_i$ (a plain $m \times m$ matrix is accepted when $k = 1$), the covariance matrix $\mathbf{S}$ (positive semidefinite; singular matrices are supported), the mean vector `mu`, and, for the `qprods*` functions, a vector $\mathbf{s}$ of nonnegative integer powers with `length(s) = size(A,3)` (entries equal to zero are allowed and are skipped). Inputs are validated: non-square, mismatched, or non-symmetric (non-Hermitian in the complex case) inputs produce informative error messages.

Function	Computes
<code>y = qproda(A, S, mu)</code>	$\mathbb{E}[\prod_i \mathbf{z}^T \mathbf{A}_i \mathbf{z}], \mathbf{z} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$
<code>y = qprodac(A, S)</code>	same with $\boldsymbol{\mu} = \mathbf{0}$
<code>y = qprods(A, S, mu, s)</code>	$\mathbb{E}[\prod_i (\mathbf{z}^T \mathbf{A}_i \mathbf{z})^{s_i}]$
<code>y = qprodsc(A, S, s)</code>	same with $\boldsymbol{\mu} = \mathbf{0}$
<code>y = qproda_comp(A, S, mu)</code>	$\mathbb{E}[\prod_i \mathbf{z}^H \mathbf{A}_i \mathbf{z}], \mathbf{z} \sim \mathcal{CN}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$
<code>y = qprodac_comp(A, S)</code>	same with $\boldsymbol{\mu} = \mathbf{0}$
<code>y = qprods_comp(A, S, mu, s)</code>	$\mathbb{E}[\prod_i (\mathbf{z}^H \mathbf{A}_i \mathbf{z})^{s_i}]$
<code>y = qprodsc_comp(A, S, s)</code>	same with $\boldsymbol{\mu} = \mathbf{0}$

The following routines implement alternative algorithms. They are slower and are provided mainly for verification: `qproda_slow` and `qprodac_slow` evaluate the admissible-permutation expansion (Proposition 1) instead of the set-partition expansion (Proposition 2); `qproda_lm` and `qproda_lm_comp` implement the method of Letac and Massam (2008); `qprodac_glm` and `qprodac_glm_comp` implement the formulas of Graczyk, Letac, and Massam (2003, 2005). `qprods_sim(A,S,mu,s,nobs)` estimates (1) by Monte Carlo with `nobs` draws (default 10^5), which is useful as an order-of-magnitude sanity check.

5 Symbolic expansion

Function	Expression generated
<code>ys = qprodastr(k)</code>	$E[\prod_{i=1}^k \mathbf{z}^T \mathbf{A}_i \mathbf{z}]$, noncentral real
<code>ys = qprodastrc(k)</code>	central real
<code>ys = qprodastr_comp(k)</code>	noncentral complex
<code>ys = qprodastrc_comp(k)</code>	central complex
<code>ys = qprodsstr(s)</code>	$E[\prod_i (\mathbf{z}^T \mathbf{A}_i \mathbf{z})^{s_i}]$, noncentral real
<code>ys = qprodsstrc(s)</code>	central real
<code>ys = qprodsstr_comp(s)</code>	noncentral complex
<code>ys = qprodsstrc_comp(s)</code>	central complex
<code>[c,ys] = qmom(k)</code>	$E[(\mathbf{z}^T \mathbf{A} \mathbf{z})^k]$, noncentral (one term per partition)
<code>c = qmomc(k)</code>	coefficients for $E[(\mathbf{z}^T \mathbf{A} \mathbf{z})^k]$, central
<code>[c,ys] = qmom_rec(k), c = qmomc_rec(k)</code>	as above, by recursion

Each element of the output is one term of the expansion, e.g. "8*a_1_2_3"; the full expression is the sum of all elements. Every term appears exactly once, so the number of elements equals the counts reported by the functions of Section 6. For the complex functions, each `a/h` factor denotes the *real part* of the corresponding trace expression; this is essential for $k \geq 3$ (see Section 6 of the paper). Under MATLAB the output is a string array; under Octave it is a cell array of character vectors.

6 Counting the number of terms

Function	Number of distinct terms in	Paper
<code>qcount(k)</code>	$E[\prod_{i=1}^k \mathbf{z}^T \mathbf{A}_i \mathbf{z}]$, $\boldsymbol{\mu} \neq \mathbf{0}$	$M(k)$, Eq. (18)
<code>qcountc(k)</code>	$E[\prod_{i=1}^k \mathbf{z}^T \mathbf{A}_i \mathbf{z}]$, $\boldsymbol{\mu} = \mathbf{0}$	$N(k)$, Eq. (21)
<code>qcountc_rec(k)</code>	as <code>qcountc</code> , via the recurrence of Section 4	
<code>qcounts(s)</code>	$E[\prod_i (\mathbf{z}^T \mathbf{A}_i \mathbf{z})^{s_i}]$, $\boldsymbol{\mu} \neq \mathbf{0}$	$M(\mathbf{s})$, Eq. (45)
<code>qcountsc(s)</code>	$E[\prod_i (\mathbf{z}^T \mathbf{A}_i \mathbf{z})^{s_i}]$, $\boldsymbol{\mu} = \mathbf{0}$	

For orientation: $M(k) = 2, 6, 24, 123, 774, 5,778, 49,824$ and $N(k) = 1, 2, 5, 17, 73, 388, 2,461$ for $k = 1, \dots, 7$. The counts grow quickly, so symbolic expansion is practical only for moderate k (or for repeated powers, where the counts are much smaller).

7 Auxiliary functions

These combinatorial helpers are used internally and may be of independent interest.

Function	Purpose
<code>bell(k)</code>	Bell number B_k (number of set partitions of $\{1, \dots, k\}$)
<code>setpart(n)</code>	all set partitions of $\{1, \dots, n\}$
<code>ip_desc(k)</code>	integer partitions of k in descending lexicographic order
<code>necklace(v)</code>	number of necklaces with content $\mathbf{v} = (v_1, \dots, v_p)$
<code>bracelet(v)</code>	number of bracelets with content $\mathbf{v}$ (Lemma 8 of the paper)
<code>fixbrace(v)</code>	bracelets with content $\mathbf{v}$, with multiplicity coefficients
<code>fixbracef(k)</code>	bracelets of $(1, \dots, k)$, coded as factoradic indices
<code>nthperm(v, n)</code>	n -th lexicographic permutation of $\mathbf{v}$
<code>nmp(p)</code>	next multiset permutation
<code>istheta(p)</code>	tests whether an index word is a canonical θ -word
<code>taulist(v), thetalist(v)</code>	canonical τ -/ θ -words and multiplicities for a block
<code>totient(n), divisor(n), gcdv(v), factor2(n)</code>	elementary number theory

8 Worked examples

All outputs below were produced by the toolbox; the script `demo_run.m` is not distributed, but every example can be pasted directly into the command window. The scripts `example1.m` to `example5.m` included in the package provide further demonstrations and timing comparisons.

8.1 A product of three quadratic forms

Take $m = 2$ and

$$\mathbf{A}_1 = \begin{pmatrix} 2 & 1 \\ 1 & 3 \end{pmatrix}, \quad \mathbf{A}_2 = \begin{pmatrix} 1 & 0 \\ 0 & 2 \end{pmatrix}, \quad \mathbf{A}_3 = \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}, \quad \boldsymbol{\Sigma} = \begin{pmatrix} 1 & 0.5 \\ 0.5 & 1 \end{pmatrix}, \quad \boldsymbol{\mu} = \begin{pmatrix} 1 \\ 2 \end{pmatrix}.$$

```
>> A1 = [2 1; 1 3]; A2 = [1 0; 0 2]; A3 = [1 1; 1 1];
>> S = [1 0.5; 0.5 1]; mu = [1; 2];
>> A = cat(3, A1, A2, A3);
>> y = qproda(A, S, mu) % E[Q1*Q2*Q3], z ~ N(mu, S)
y = 14256
>> yc = qprodac(A, S) % E[Q1*Q2*Q3], z ~ N(0, S)
yc = 594
```

(The value 14,256 can be verified independently by differentiating the moment generating function; both values are exact integers here because the inputs are integer/half-integer.)

8.2 Symbolic expansions

The explicit expression for $k = 2$ (Example 4 of the paper) and the central expression for $k = 3$:

```
>> qprodastr(2)
"2*a_1_2" % 2*tau_(1,2)
"4*h_1_2" % 4*theta_(1,2)
"a_1*a_2"
```

```

"h_1*a_2"
"a_1*h_2"
"h_1*h_2"
>> qprodastrc(3)
"8*a_1_2_3"
"2*a_1_2*a_3"
"2*a_1_3*a_2"
"2*a_1*a_2_3"
"a_1*a_2*a_3"

```

The five central terms match $N(3) = 5$, and the six noncentral $k = 2$ terms match $M(2) = 6$.

8.3 Repeated quadratic forms

The powers are passed as the vector $\mathbf{s}$: below we evaluate $E[(\mathbf{z}^T \mathbf{A}_1 \mathbf{z})^2 (\mathbf{z}^T \mathbf{A}_2 \mathbf{z})]$ numerically ($\mathbf{s} = (2, 1)$) and expand $E[(\mathbf{z}^T \mathbf{A}_1 \mathbf{z})(\mathbf{z}^T \mathbf{A}_2 \mathbf{z})^2]$ symbolically ($\mathbf{s} = (1, 2)$):

```

>> qprods(cat(3,A1,A2), S, mu, [2 1]) % E[Q1^2*Q2]
ans = 27540
>> qprodsstr([1 2]) % E[Q1*Q2^2], symbolic
"8*a_1_2_2"      "16*h_1_2_2"      "8*h_2_1_2"
"4*a_1_2*a_2"    "8*h_1_2*a_2"    "4*a_1_2*h_2"    "8*h_1_2*h_2"
"2*a_1*a_2_2"    "2*h_1*a_2_2"    "4*a_1*h_2_2"    "4*h_1*h_2_2"
"a_1*a_2*a_2"    "h_1*a_2*a_2"    "2*a_1*a_2*h_2"  "2*h_1*a_2*h_2"
"a_1*h_2*h_2"    "h_1*h_2*h_2"

```

(Output rearranged into rows to save space.) There are 17 distinct terms, in agreement with `qcounts([1 2])`.

8.4 Complex Hermitian quadratic forms

```

>> B1 = [2 1-1i; 1+1i 3]; B2 = [1 2i; -2i 2];
>> Sc = [2 1i; -1i 2]; muc = [1+1i; 2];
>> qproda_comp(cat(3,B1,B2), Sc, muc) % z ~ CN(muc,Sc)
ans = 935
>> qprodac_comp(cat(3,B1,B2), Sc) % z ~ CN(0,Sc)
ans = 127

```

8.5 Wishart product moments

Let $\mathbf{W} \sim \mathcal{W}_2(n, \mathbf{\Sigma}, \mathbf{\Lambda})$ with $n = 5$ and $\mathbf{\Lambda} = \boldsymbol{\mu}\boldsymbol{\mu}^T$. To compute $E[\text{tr}(\mathbf{W} \mathbf{A}_1) \text{tr}(\mathbf{W} \mathbf{A}_2)]$, apply the substitution rule (3) to the $k = 2$ expansion $2\tau_{(1,2)} + 4\theta_{(1,2)} + \tau_1\tau_2 + \theta_1\tau_2 + \tau_1\theta_2 + \theta_1\theta_2$ (read off from `qprodastr(2)`):

```

>> n = 5; Lam = mu*mu';
>> tau12 = n*trace(A1*S*A2*S); theta12 = trace(A1*S*A2*Lam);
>> tau1 = n*trace(A1*S); theta1 = trace(A1*Lam);
>> tau2 = n*trace(A2*S); theta2 = trace(A2*Lam);
>> w = 2*tau12 + 4*theta12 + tau1*tau2 + theta1*tau2 ...
      + tau1*theta2 + theta1*theta2
w = 1453.5

```

As a check, the same number is obtained by writing each $\text{tr}(\mathbf{W}\mathbf{A}_i)$ as a quadratic form in the stacked vector of the n underlying normal vectors and calling `qproda` directly:

```
>> In = eye(n); AA = cat(3, kron(In,A1), kron(In,A2));
>> mus = [mu; zeros(2*(n-1),1)]; % mu_1 = mu, mu_2 = ... = mu_n = 0
>> qproda(AA, kron(In,S), mus)
ans = 1453.5
```

For a *complex* Wishart matrix, use the `_comp` expression generators and take real parts in (3). For instance, the expression for $E[\text{tr}(\mathbf{W}\mathbf{A}_1)\text{tr}(\mathbf{W}\mathbf{A}_2)^2]$, $\mathbf{W} \sim \mathcal{CW}_m(n, \mathbf{\Sigma}, \mathbf{\Lambda})$, is generated by `qprodsstr_comp([1 2])` (this reproduces the corrected version of Example 3 of Di Nardo (2014); see Section 8 of the paper and `example5.m`).

8.6 Moments of a single quadratic form

`qmom(k)` returns the coefficients (and, optionally, the expression) of $E[(\mathbf{z}^T \mathbf{A} \mathbf{z})^k]$ organized by integer partition, in terms of $\tilde{\tau}_i$ (`a_i`) and θ_i (`h_i`):

```
>> [c, ys] = qmom(3); ys
"8*a_3+24*h_3"
"6*a_2*a_1+6*a_2*h_1+12*h_2*a_1+12*h_2*h_1"
"a_1^3+3*a_1^2*h_1+3*a_1*h_1^2+h_1^3"
```

The three rows correspond to the partitions (3), (2, 1), and (1, 1, 1) of $k = 3$.

8.7 Counting

```
>> [qcount(3) qcountc(3) qcounts([2 2]) qcountsc([2 2])]
ans =
    24     5    54    10
```

That is, $E[\prod_{i=1}^3 \mathbf{z}^T \mathbf{A}_i \mathbf{z}]$ has $M(3) = 24$ distinct terms (5 in the central case), and $E[(\mathbf{z}^T \mathbf{A}_1 \mathbf{z})^2 (\mathbf{z}^T \mathbf{A}_2 \mathbf{z})^2]$ has 54 distinct terms (10 in the central case).

9 Performance notes

- Prefer `qproda/qprodac` (set-partition algorithm, Proposition 2) over `qproda_slow/qprodac_slow` (admissible-permutation algorithm, Proposition 1). The former iterates over the B_k set partitions, the latter effectively over $k!$ permutations; for $k = 10$, $B_{10} = 115,975$ versus $10! = 3,628,800$.
- When the same matrix appears several times in the product, use `qprods/qprodsc` with the powers in `s` rather than repeating slices of `A`: the number of terms (and the running time) can be smaller by orders of magnitude.
- The dominant cost per term is a chain of $m \times m$ matrix products, so running time scales roughly as the term count times m^3 . Counting first (Section 6) tells you what to expect before generating an expansion.
- All routines accept singular (positive semidefinite) $\mathbf{\Sigma}$; no inversion of $\mathbf{\Sigma}$ is ever performed.

10 Testing

`test_all.m` runs 63 checks: it validates every expression-generating function against the corresponding numerical evaluator on random inputs (by parsing the generated terms and evaluating them), cross-checks the independent algorithms (`_slow`, `_lm`, `_glm`) against the main routines, verifies the counting functions against the tables of the paper, and exercises the input validation and edge cases ($k = 1$, scalar `s`, zeros in `s`, row-vector `mu`). A successful run ends with `==== 63 passed, 0 failed =====`.

11 Version history

1.0 (8/15/2025) Initial release.

1.1 (9/2/2025) `qprods.m`, `qprodsc.m`, `qprods_comp.m`, `qprodsc_comp.m`: fewer coefficient computations.

1.2 (9/8/2025) Added `example5.m`.

1.3 (7/24/2026) Maintenance and portability release: corrected the function name declared inside `qprodac_slow.m`; fixed a typo in `thetalist.m` and a missing semicolon in `qprodsstrc.m`; the expression-generating functions now also run under GNU Octave; added input validation to the main evaluation routines; added `test_all.m` and this manual.

References

- [1] E. Di Nardo, On a symbolic representation of non-central Wishart random matrices with applications, *J. Multivariate Anal.* 125 (2014) 121–135.
- [2] P. Graczyk, G. Letac, H. Massam, The complex Wishart distribution and the symmetric group, *Ann. Statist.* 31 (2003) 287–309.
- [3] P. Graczyk, G. Letac, H. Massam, The hyperoctahedral group, symmetric group representations and the moments of the real Wishart distribution, *J. Theoret. Probab.* 18 (2005) 1–42.
- [4] R. Kan, J. Pan, An explicit expression for the expectation of a product of quadratic forms in multivariate normal random variables, working paper, University of Toronto, 2026.
- [5] S. Karim, J. Sawada, Z. Alamgir, S. M. Husnine, Generating bracelets with fixed content, *Theoret. Comput. Sci.* 475 (2013) 103–112.
- [6] G. Letac, H. Massam, The noncentral Wishart as an exponential family, and its moments, *J. Multivariate Anal.* 99 (2008) 1393–1417.