Lecture 29:
G under Drift, Comparisons of G, Dimensionality of G

Bruce Walsh lecture notes

Synbreed course

version 11 June 2013

Drift and G

Recall there is a simple expression for the *expected* change in G given drift:

$$E[\,\mathbf{G}(t)\,] = (1 - f_t)\mathbf{G}(0) = \left(1 - \frac{1}{2N_e}\right)^t \mathbf{G}(0)$$

Complications: Can be considerable variation about the expected value of G!

Complications: If nonadditive variances are present, expected change in G complex

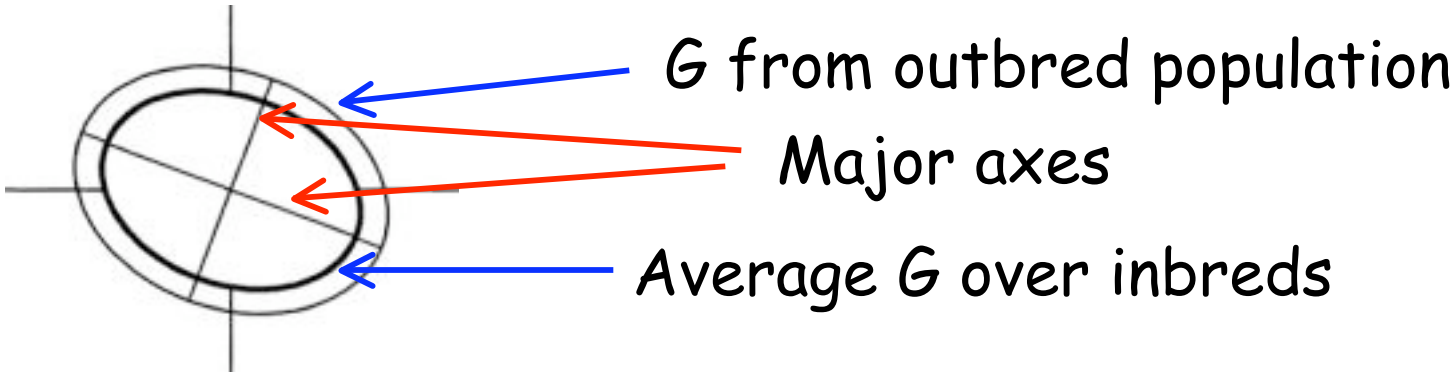
High variance about E[G] under inbreeding

Phillips, Whitlock, and Fowler (2001) examined 52 lines of *Drosophila melanogaster* that had been inbred for one generation of brother-sister mating and then expanded to a large population size by two generations of random mating. They estimated **G** for all 52 lines using 6 wing traits

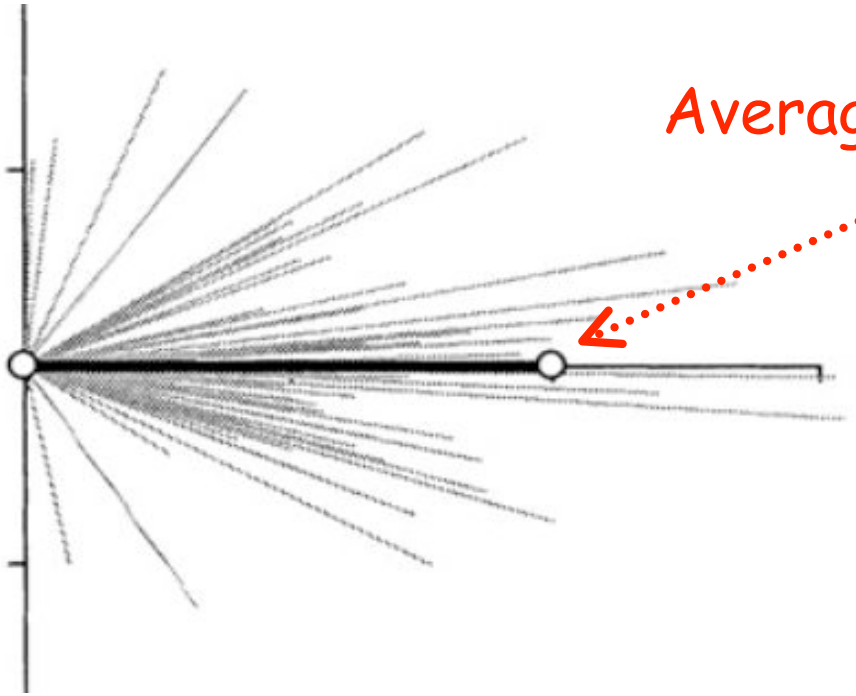
The mean **G** (the average over all 52 lines) was quite consistent with the theory, showing a proportional reduction (eigenvectors unchanged, eigenvalues reduced)

High variance about E[G] under inbreeding

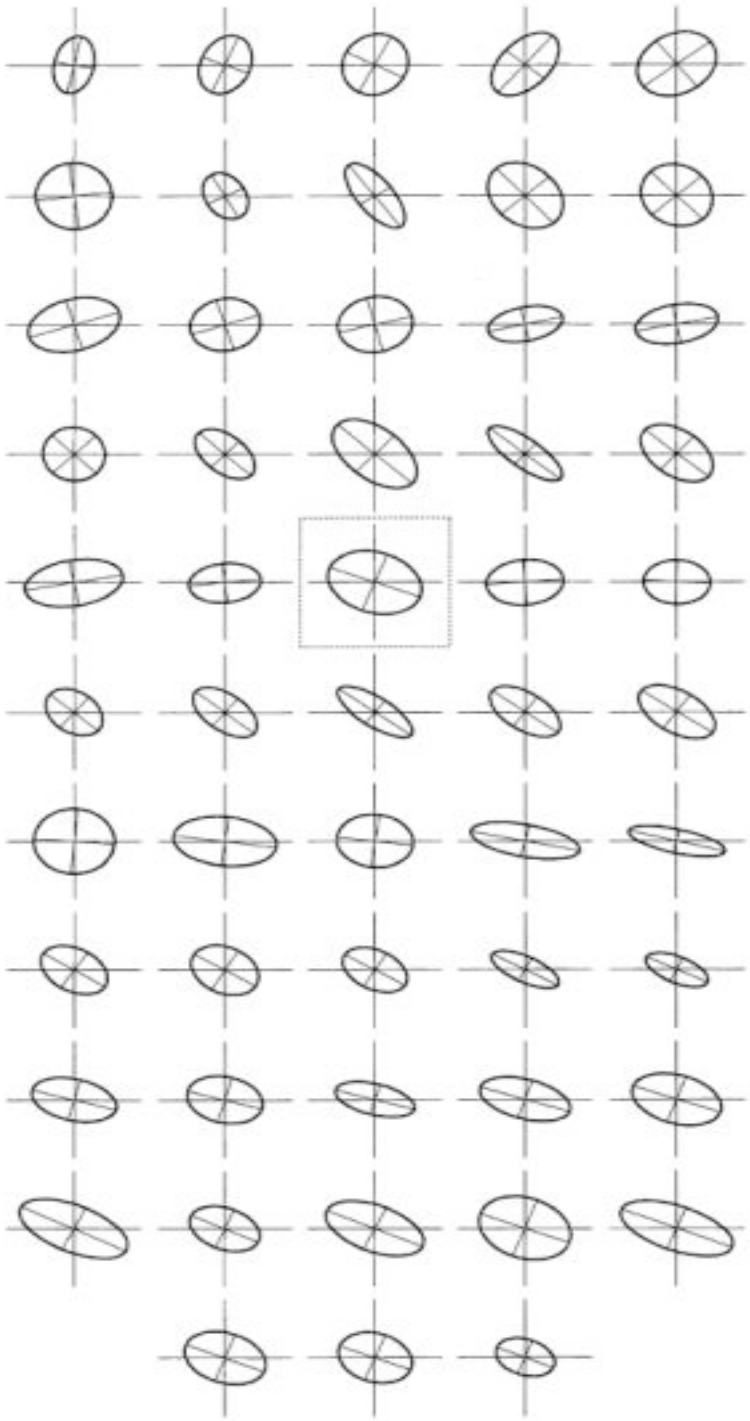
95% confidence ellipsoid for 2 traits



While the MEAN **G** agreed with theory, there was massive variation among the particular realizations.

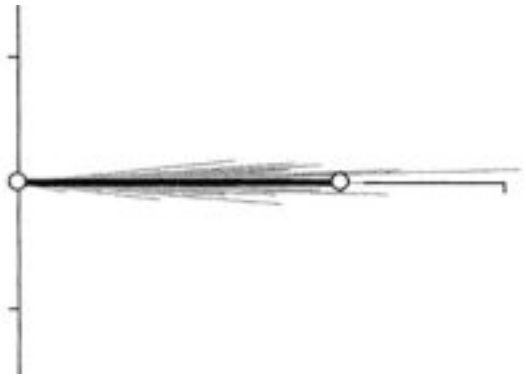


The realizations for each of the 52 lines is given by a "whisker", which shows both its length and direction relative to the major axis.

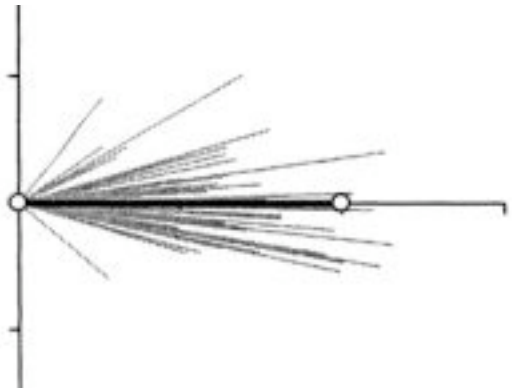


Example of hidden pleiotropy in their data

Below are plots for two sets of traits, both of which had a genetic covariance of zero in the outbred pop



little spread in the major axes -- traits still uncorrelated among the samples



Significant spread in the major axes -- traits positively and negatively correlated among various sampled G

Suggests hidden pleiotropy

Changes in G under nonadditive variance

When nonadditive (dominance, epistasis) variance present, genetic variances can actually increase (for a time) under inbreeding.

Simplest example is with Additive x Additive epistasis:

$$\sigma_t(A_1, A_2) = (1 - f_t)\sigma_0(A_1, A_2) + 4f_t(1 - f_t)\sigma_0(AA_1, AA_2)$$

Hence $\mathbf{G}_t = (1 - f_t)\mathbf{G}_0 + 4f_t(1 - f_t)\mathbf{G}_{AA,0}$

Here G is not proportionately changing with f.

Expected Eigenstructure of G under drift

Griswold et al. showed that the distribution of eigenvalues is highly non-uniform under a drift and mutation model, showing close to an exponential distribution -- a few large, and many small, eigenvalues

This skewing arises for genealogical reasons: drift imposes a dependence structure on the alleles in the sample due to shared common ancestry, and this in turn results in the distribution of the eigenvalues of **G** being highly nonuniform.

General issues in comparing **G**

One immediate issue with **G** is that it is typically <u>not</u> estimated as a product-moment covariance

P can be directly estimated

$$\widehat{P}_{jk} = \frac{1}{n-1}\sum_{i=1}^{n}(x_{ji} - \overline{x}_j)(x_{ki} - \overline{x}_k)$$

(co)variances of breeding values, on the other hand, are indirectly estimated:

$$\widehat{\sigma}^2_A = \frac{1}{r}\widehat{\sigma}^2_B = \frac{1}{r}\frac{M_B - M_W}{n}$$

$$\widehat{\mathbf{G}} = \frac{1}{r}\frac{\mathbf{M}_B - \mathbf{M}_W}{n}$$

Less precision on individual elements of **G** than of **P**

Estimated **G** may not be a covariance matrix

No guarantee that **G** is non-negative definite (contains no negative eigenvalues)

If **G** contains at least one negative eigenvalue, then there is some combination of traits such that $Var(\Sigma a_i g_i) < 0$

Indeed (Hill and Thompson) the probability of a negative eigenvalue for a sample covariance matrix is very high unless the sample size is very large.

Power:

The basic sampling unit when constructing **G** is the family.

Thus, sample size power is more a function of number of families vs. number of individuals

Further, distribution theory for tests of matrix differences usually built around product-moment, as opposed to variance-component, matrices

"Robust" statistical approaches

Many matrix comparison approaches use "robust" methods (some of these are called distribution-free or nonparametric).

Most powerful: Randomization tests. Basic idea (due to Fisher) is to construct our test statistic for matrix difference on the original sample.

One then randomizes the independent sampling unit (unrelated families) over groups and recomputes the test statistic. This is done thousands of times generating a distribution under the null of equality.

"Robust" statistical approaches (cont)

Suppose n of our N randomization test statistics are more extreme than our sample. The p value for equality is just (n+1)/(N+1). The +1 occurs because we also include our original observation.

When one can identify independent sampling units, randomization tests are bullet-proof. However, finding these independent units can be very subtle. For example, if families are related, they are not independent! Great care must be taken, with randomization approaches, as it is very easy to use the wrong sampling unit.

While randomization can give p values, what about standard errors and approximate confidence intervals?

Two resampling procedures widely used: the Bootstrap and the jackknife. CARE must be taken when using these, as they do not always work!

The idea behind the bootstrap is that the sample itself provides information on the sample variance
Suppose n families are used in constructing **G**, and we want a CI on det(**G**)

A bootstrap sample for **G** is obtained by sampling families with replacement from the original families to generate a sample of n families. **G** is constructed, det(**G**) found

The sample variance in det(**G**) in the bootstrap samples is the estimate of the true sample variance, the lower2.5% and upper 97.5% values set the 95% CI

Jackknife methods also involve resampling. Here one removes each observation and constructs n "pseudovalues"

$$\phi_i = n\widehat{\theta} - (n-1)\widehat{\theta}_{-i}$$

Mean of the n pseudovalues is the jackknife estimate of the parameter, their variance is the estimate of the sample variance, and t-test are used for hypothesis testing given the estimate mean and variance.

Again, the sampling unit here is the family.

Comparing two G matrices

Most obvious approach is element-by element

One could use standard tests (or robust methods) to test, separately, the equality of each element.

For example, are the heritabilties of trait one the same in the two groups we are comparing? What about trait 2, etc.

Multiple comparison issues: is the collection of tests, as a whole, significant?

Especially problematic if tests are correlated, as occurs when estimates are correlated.

One simple approach for multiple comparisons is based on the binominal --- given the p values, is there an excessive number of significant tests?

Paulsen found 9 of 45 comparison of heritabilities between 2 species of buckeye butterflies significant at the 5% level. Does this indicate a significant difference in **G**?

$$\sum_{i=9}^{45} \Pr(k \text{ significant tests}) = \sum_{k=9}^{45} \frac{45!}{k!(45-k)!}\, 0.05^k\, 0.95^{45-k} = 0.0003$$

Hence, an excess of significant tests. The general problem is that variance estimates for **G** are correlated and hence the tests are not independent.

What does it mean if matrices are not "equal"

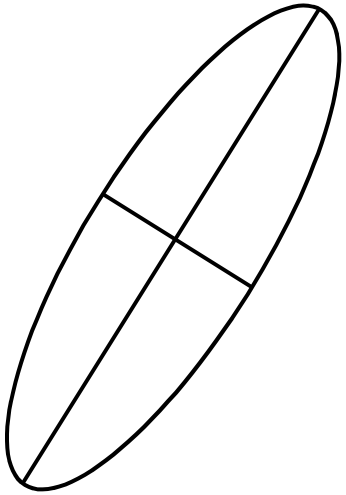
What does it mean if a test for equality fails?

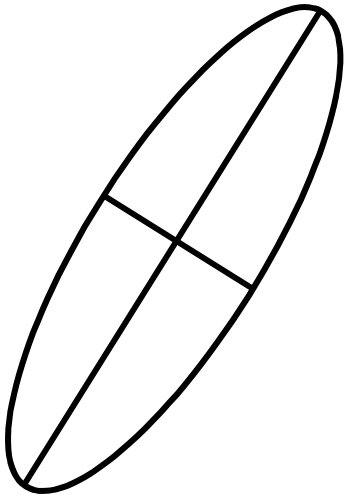
A matrix is a complex geometric object, and two matrices can be very similar geometrically, but not equal.

Key: We really want to compare elements of matrix geometry.

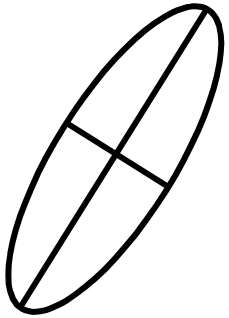
Want to compare eigenstructure: do they share common eigenvectors, eigenvalues?

Target Matrix
For Comparison

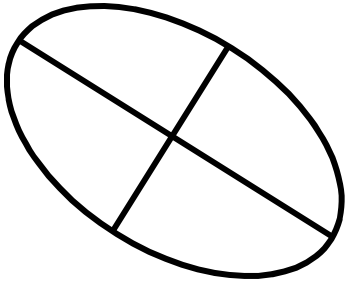




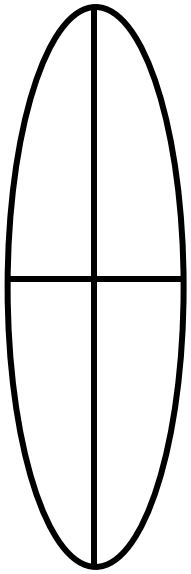
A
Identical



B
Proportional



C
Same Orientation
Different Scaling



D
Same Scaling
Different Orientation

Random Skewers

One approach for comparing matrices is to compare the similarities of response to random directions of selection, **R** = **G**β

The method of random skewers generates a large number of random (unit length) β and then projects these through both **G** matrices ($\mathbf{R}_i = \mathbf{G}_i\beta$) and then measures the angle or distance between them.

$$\cos(\theta) = \frac{\mathbf{R}_1^T\mathbf{R}_2}{||\mathbf{R}_1||\ ||\mathbf{R}_2||} = \frac{\beta^T\mathbf{G}_1\mathbf{G}_2\beta}{||\mathbf{G}_1\beta||\ ||\mathbf{G}_2\beta||}$$

$$d = ||\mathbf{G}_1\beta - \mathbf{G}_2\beta|| = \sqrt{(\mathbf{G}_1\beta - \mathbf{G}_2\beta)^T(\mathbf{G}_1\beta - \mathbf{G}_2\beta)}$$
$$= \sqrt{\beta^T(\mathbf{G}_1 - \mathbf{G}_2)^T(\mathbf{G}_1 - \mathbf{G}_2)\beta}$$

Random Skewers

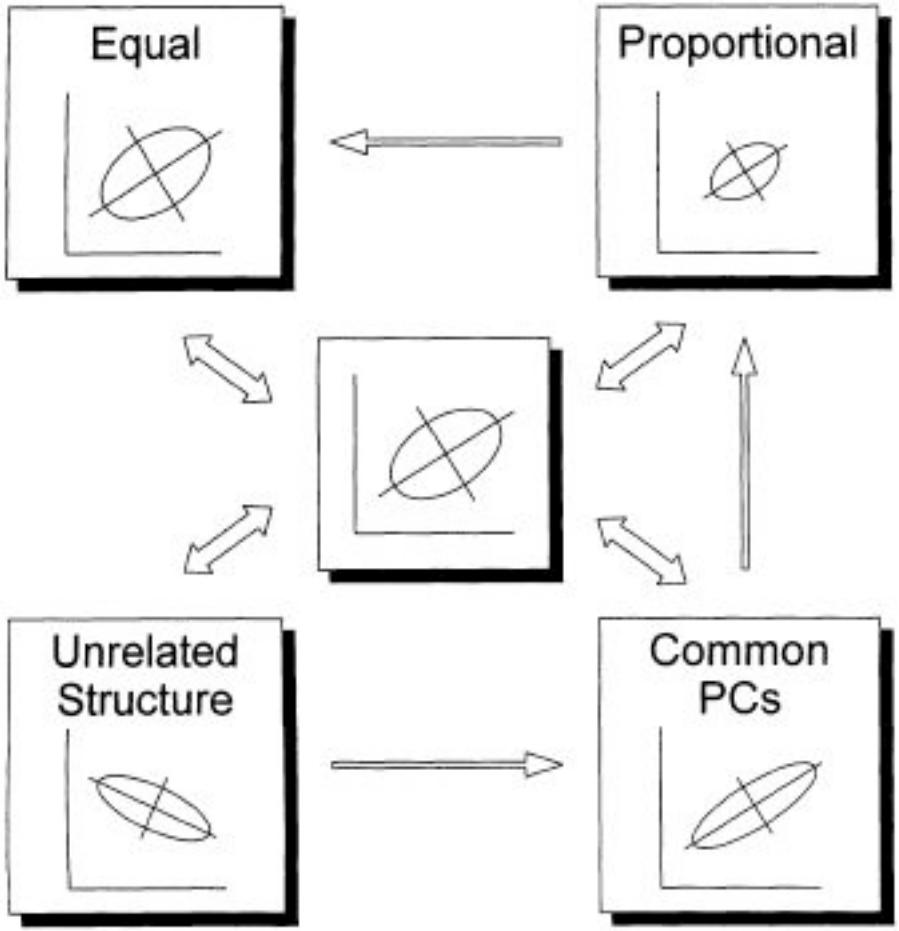
One generates a large number of random skewers and then compares the distribution for these two (or other) test statistics, for example by a nonparametric method such as a KS test for comparing two distributions

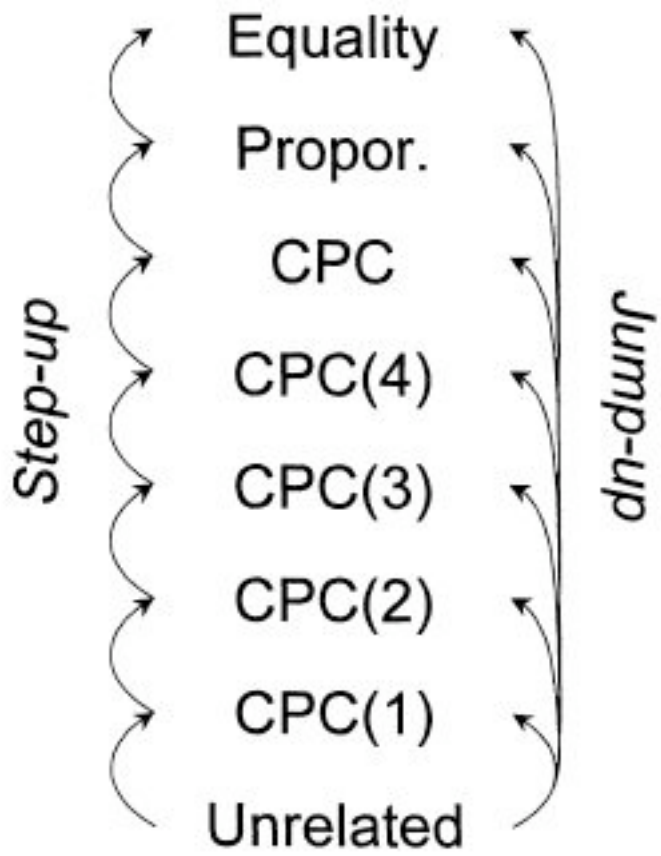
One especially nice feature about random skewers is that it mimics our real question of interest --- for random selection, do the two matrices essentially give the same responses.

Flury's CPC

A more formal approach to compare aspects of shared geometry is the method of CPC -- common principal components, proposed by Flury

Idea is that there is a hierarchy of relatedness (The Flury hierarchy). At the bottom are unrelated (no common PCs), next they share 1 PC, next they share =2 PC, .., finally share all PCs, then are proportionate, and finally (at the top of the hierarch) they are equal

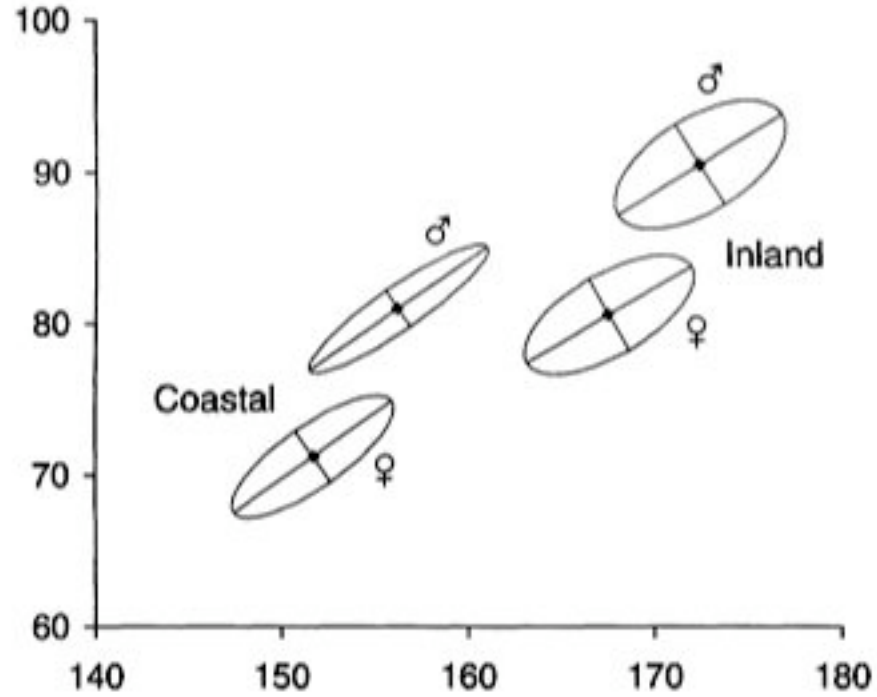




LR tests used. One can step-up (one at a time), jump-up (multiple step jump possible) or use model comparison (AIC)

As an example, consider the follows data from Arnold on two populations of snakes.

Population	λ_1	λ_2	% first 2 PCs	trace($\mathbf{G}$)
Coastal males	7.61 (61%)	3.42 (28%)	88.69	12.43
Inland males	10.16 (53%)	6.51 (34%)	88.06	18.93
Coastal females	5.73 (45%)	5.24 (41%)	86.04	12.75
Inland females	5.85 (46%)	5.27 (41%)	87.28	12.74



Step-up

Comparison	df	χ^2	p
CPC(1) vs. Unrelated	5	5.10	0.4035
CPC(2) vs. CPC(1)	4	2.00	0.7357
CPC(3) vs. CPC(2)	3	7.02	0.0714
CPC(4) vs. CPC(3)	2	0.54	0.7648
Full CPC vs. CPC(4)	1	6.30	0.0121*

Jump-up

Hierarchy	df	χ^2	p
Proportionality	20	73.77	<0.0001
Full CPC	15	20.95	0.1384
CPC(4)	14	14.65	0.4020
CPC(3)	12	14.12	0.2931
CPC(2)	9	7.10	0.6264
CPC(1)	5	5.10	0.4035

Model comparison (smallest AIC)

Model:	CPC(1)	CPC(2)	CPC(3)	CPC(4)	FullCPC	Proportionality
AIC value	37.1	31.1	32.1	28.7	33.0	75.8

Krzanowski Subspace comparison

Two matrices may have most of their variation in the same subspace, but have very different eigenvectors

CPC would score this as no relationship, but there clearly is a relationship when sets, as opposed to individual, PCs used

Krzanowski proposed a method for comparison of such subspaces

Compute a subspace projection matrix **B** for each matrix by taking the first k < n/2 eigenvectors

$$\mathbf{B}(i) = (\mathbf{e}_1(i) \quad \cdots \quad \mathbf{e}_k(i))$$

Next, compute the matrix $\mathbf{S} = \mathbf{B}^T(1)\,\mathbf{B}(2)\,\mathbf{B}^T(2)\,\mathbf{B}(1)$

The smallest angle between any two orthogonal axes is given by $\cos^{-1}\sqrt{\lambda_1}$, where λ_1 is the leading eigenvalue of $\mathbf{S}$.

The sum of the eigenvalues of $\mathbf{S}$ is the sum of squared cosines between the sets of orthogonal axes in the two projection matrices. If these are completely aligned, this sum equals k, while if there is no shared orientation, this sum is zero. Thus, the sum of eigenvalues in the Krzanowski matrix $\mathbf{S}$ is a measure of the number of shared dimensions (of the k tested) between $\mathbf{B}(1)$ and $\mathbf{B}(2)$. When the matrices being compared are two estimates of $\mathbf{G}$, a randomization test (randomizing families over groups and then computing the sum of the eigenvalues of $\mathbf{S}$) can generate an appropriate threshold for significance.

Dimensionality of **G**

A final issue in the comparison of **G** is its rank (number of positive eigenvalues)

As mentioned, estimated **G** matrices are expected to be of less than full rank

Also, much of the variation in **G** often concentrated in the first few PCs

Leads to the related issue of reduced-rank estimates of **G**

Estimation of Eigenvalues

Leading eigenvalues tend to be overestimated, minor eigenvalues underestimated

$$E[\widehat{\lambda}_i] \simeq \lambda_i \left(1 + \frac{f_i}{N}\right), \quad \text{where} \quad f_i = \sum_{j \neq i}^{k} \left(\frac{\lambda_i}{\lambda_j} - 1\right)^{-1}$$

This arises because fitting the first eigenvector trys to account for as much variation as possible, leading to overfitting. This is made-up for by underfitting of minor eigenvalues.

Estimation of Eigenvalues (cont)

Example: suppose 6 x 6 with λ as below

		Estimated values		
λ	f_i	N=20	N=50	N=100
10	0.852	10.852	10.170	10.085
3	1.306	3.392	3.078	3.039
2	-2.864	1.427	1.885	1.943
1	-3.500	0.650	0.930	0.965
0.5	-5.330	0.234	0.447	0.473
0.1	-5.450	0.046	0.089	0.095

$$E[\widehat{\lambda}_i] \simeq \lambda_i \left(1 + \frac{f_i}{N}\right), \quad \text{where} \quad f_i = \sum_{j \neq i}^{k} \left(\frac{\lambda_i}{\lambda_j} - 1\right)^{-1}$$

Bias with bootstrap CI for eigenvalues

One approach to estimating rank is to compute the confidence intervals of the eigenvalues, and then declare rank as the number of such intervals excluding zero.

Bootstrapping has been used to estimate these CIs.

Here, a bootstrap sample is generated by resampling families with replacement, and the eigenvalues computed for this sample.

Assigning these eigenvalues and repeating this in thousands of bootstrap samples generates approximate CIs

The problem arises in the ASSIGNMENT of eigenvalues. eigenvectors are fairly unstable over bootstrap samples, so one approach is to use rank-ordering to assign: the largest in the sample corresponds to λ_1, the next to λ_2, etc

Problem: Rank ordering generates CIs that are too small.

	True Values				Inferred values				True Var	Inferred Var
λ_1	4	3	4	6	4	4	4	6	1.58	1.00
λ_2	2	4	3	2	2	3	3	3	0.92	0.25
λ_3	1	2	1	3	1	2	1	2	0.92	0.33

Better approach: Projection

Obtain the bootstrap estimate for the ith eigenvalue by projecting the sample ***G*** onto the i-th original eigenvector

$$\hat{\lambda}_i = || \hat{\mathbf{G}}_B \mathbf{e}_i ||$$

A cleaner bootstrap estimate of rank is simply to look at the distribution of rank in the samples.

Again, generate a set of bootstrap samples, for example $\boldsymbol{G}_1, \boldsymbol{G}_2, \cdots, \boldsymbol{G}_{5000}$. This generates 5000 bootstrap values for rank, from which appropriate CIs and sampling variances can be obtained.

Diagonalization of Sample **G**

Recall that diagonalization of a matrix decomposes it into a diagonal matrix of eigenvalues and matrices of associated eigenvectors,

$$\mathbf{A} = \mathbf{U}\boldsymbol{\Lambda}\mathbf{U}^T$$

$$\mathbf{U} = (\mathbf{e}_1, \ldots, \mathbf{e}_p), \quad \boldsymbol{\Lambda} = \mathbf{diag}(\lambda_1, \cdots, \lambda_p)$$

Such a diagonalization for the sample estimate of **G** provides insight into how negative eigenvalues occur, offers a test for dimension, and generates a reduced-rank estimate of **G**.

Diagonalization of Sample G (cont)

Balanced one-way ANOVA/MANOVA

$$\widehat{\sigma}_A^2 = \frac{1}{r}\widehat{\sigma}_B^2 = \frac{1}{r}\frac{M_B - M_W}{n}$$

Univariate

$$\widehat{\mathbf{G}} = \frac{1}{r}\frac{\mathbf{M}_B - \mathbf{M}_W}{n}$$

Multivariate

$$\mathbf{M}_B - \mathbf{M}_W = \mathbf{T}\left(\boldsymbol{\Lambda}_Q - \mathbf{I}\right)\mathbf{T}^T$$

$$\mathbf{Q} = \left(\mathbf{M}_W^{-1/2}\right)\mathbf{M}_B\left(\mathbf{M}_W^{-1/2}\right)^T$$

$$\mathbf{T} = \mathbf{W}^{1/2}\mathbf{U}_Q$$

$$\mathbf{M}_B - \mathbf{M}_W = \mathbf{T}\left(\boldsymbol{\Lambda}_Q - \mathbf{I}\right)\mathbf{T}^T \qquad \mathbf{Q} = \left(\mathbf{M}_W^{-1/2}\right)\mathbf{M}_B\left(\mathbf{M}_W^{-1/2}\right)^T$$

For **G** estimated by one-way MANOVA, negative eigenvalues arise from eigenvalues of **Q** less than one

Hence, one test for rank is the number of eigenvalues of **Q** that are significantly greater than one (much easier to test than significantly greater than zero).

Amemiya's LR test:

$$Y = (M+N)\sum_{i=b+1}^{k} \ln\left(\frac{M\lambda_i + N}{M+N}\right) - M\sum_{i=b+1}^{k} \ln(\lambda_i)$$

Amemiya's reduced-rank estimator of G

$$\mathbf{M}_B - \mathbf{M}_W = \mathbf{T}\,(\boldsymbol{\Lambda}_Q - \mathbf{I})\,\mathbf{T}^T$$

$$\boldsymbol{\Lambda}_Q = \begin{pmatrix} \boldsymbol{\Lambda}_k & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\Lambda}_l \end{pmatrix}$$

$$\mathbf{T} = (\,(\mathbf{e}_1 \quad \cdots \quad \mathbf{e}_k) \quad (\mathbf{e}_{k+1} \quad \cdots \quad \mathbf{e}_n)\,) = (\,\mathbf{T}_k \quad \mathbf{T}_l\,)$$

$$\boldsymbol{\Lambda}_k = \mathbf{diag}(\lambda_1, \cdots, \lambda_k), \quad \textbf{and} \quad \boldsymbol{\Lambda}_l = \mathbf{diag}(\lambda_{k+1}, \cdots, \lambda_p)$$

$$\mathbf{T}\,(\boldsymbol{\Lambda}_Q - \mathbf{I})\,\mathbf{T}^T = (\,\mathbf{T}_k \quad \mathbf{T}_l\,)\left(\begin{pmatrix} \boldsymbol{\Lambda}_k & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\Lambda}_l \end{pmatrix} - \begin{pmatrix} \mathbf{I}_k & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_{p-k} \end{pmatrix}\right)(\,\mathbf{T}_k \quad \mathbf{T}_l\,)^T$$

$$= \mathbf{T}_k\,(\boldsymbol{\Lambda}_k - \mathbf{I}_k)\,\mathbf{T}_k^T + \mathbf{T}_l\,(\boldsymbol{\Lambda}_l - \mathbf{I}_{p-k})\,\mathbf{T}_l^T$$

Hence, we can write the estimate of ***G*** as positive-definite part (eigenvalues of **Q** > 1) and a part with all zero or negative eigenvalues

$$\widehat{\mathbf{G}} = \widehat{\mathbf{G}}_R + \widehat{\mathbf{G}}_N$$

The first part is a reduced-rank estimator of ***G***,

$$\widehat{\mathbf{G}}_R = \frac{1}{nr}\mathbf{T}_k\left(\boldsymbol{\Lambda}_k - \mathbf{I}_k\right)\mathbf{T}_k^T = \frac{1}{nr}\sum_{i=1}^{k}\left(\lambda_{Q,i} - 1\right)\mathbf{e}_i^T\mathbf{e}_i$$

We can choose k as the number of eigenvalues of **Q** observed to be > 1. This is equivalent to setting any negative eigenvalues of the estimated ***G*** to zero.

$$\widehat{\mathbf{G}}_R = \frac{1}{nr}\,\mathbf{T}_k\,(\boldsymbol{\Lambda}_k - \mathbf{I}_k)\,\mathbf{T}_k^T = \frac{1}{nr}\sum_{i=1}^{k}(\lambda_{Q,i} - 1)\,\mathbf{e}_i^T\mathbf{e}_i$$

Alternatively, the number of included dimensions can be less than the observed rank of ***G***.

For example, k could be the number of statistically supported eigenvalues of **Q** > 1

Likewise, we might fix the number of dimensions to consider (since majority of variation often restricted to the first few PCs).

Problem with last: It has a bias that does not decrease with increasing sample size

$$E[\mathbf{G}_r] = \mathbf{G} - \frac{1}{r}\sum_{i=k+1}^{p} E[\,\lambda_i\,\mathbf{e}_i\mathbf{e}_i^T\,]$$

Factor-analytic modeling: Direct estimation of PCs of *G*

Kirkpatrick and Meyer proposed that instead of estimating ***G*** first and extracting the few few eigenvalue, instead one could estimate the eigenvalues directly (without going through an estimated ***G***)

Motivation: Consider the spectral decomposition of a matrix ***A***,

$$\mathbf{A} = \lambda_1 \mathbf{e}_1 \mathbf{e}_1^T + \lambda_2 \mathbf{e}_2 \mathbf{e}_2^T + \cdots + \lambda_n \mathbf{e}_n \mathbf{e}_n^T$$

Setting $\mathbf{f}_i = \sqrt{\lambda_i}\mathbf{e}_i$

We can write this as $\mathbf{A} = \sum_{i=1}^{P} \mathbf{f}_i \mathbf{f}_i^T$

Factor-analytic modeling: Direct estimation of PCs of *G*

Motivation: Consider the spectral decomposition of a matrix ***A***,

$$\mathbf{A} = \lambda_1 \mathbf{e}_1 \mathbf{e}_1^T + \lambda_2 \mathbf{e}_2 \mathbf{e}_2^T + \cdots + \lambda_n \mathbf{e}_n \mathbf{e}_n^T$$

Setting $\mathbf{f}_i = \sqrt{\lambda_i}\mathbf{e}_i$

We can write this as $$\mathbf{A} = \sum_{i=1}^{p} \mathbf{f}_i \mathbf{f}_i^T$$

Note that f_i is the direction of PC_i, while the square of its length is the amount of variation it accounts for, as $$||\mathbf{f}_i||^2 = \mathbf{f}_i^T \mathbf{f}_i = \lambda_i \, \mathbf{e}_i^T \mathbf{e}_i = \lambda_i \, ||\mathbf{e}_i||^2 = \lambda_i$$

Thus, if we can directly estimate the first k PCs, a reduced-rank estimator of G, guaranteed to be positive-definite, is given by $\widehat{\mathbf{G}}_R = \sum_{i=1}^{k} \widehat{\mathbf{f}}_i \widehat{\mathbf{f}}_i^T$

Question: How do we directly estimate the f_i?

Consider the animal model, $\mathbf{z}_i = \boldsymbol{\mu} + \mathbf{a}_i + \mathbf{e}_i$

One can write the vector of breeding values a_i for individual a as the sum of the contributions over f_j, $\mathbf{a}_i = \sum_{j=1}^{p} \alpha_{i,j} \mathbf{f}_j$

Hence, the animal model becomes $\mathbf{z}_i = \boldsymbol{\mu} + \sum_{j=1}^{p} \alpha_{i,j} \mathbf{f}_j + \mathbf{e}_i$

Hence, one can start by estimating the first PC,

$$\mathbf{z}_i = \boldsymbol{\mu} + \alpha_{i,1}\mathbf{f}_1 + \mathbf{e}_i$$

Where $\alpha_{i,1}$ is individual i's breeding value for the first PC.

One can test for improvement of fit over the no PC model, and if the first is significant, move to a model with 2 PCs,

$$\mathbf{z}_i = \boldsymbol{\mu} + \alpha_{i,1}\mathbf{f}_1 + \alpha_{i,2}\mathbf{f}_2 + \mathbf{e}_i$$

Here $\alpha_{i,2}$ is i's BV for PC 2. This is repeated until the addition of new factors does not improve fit,

If k factors are fitted, the reduced-rank estimate becomes

$$\hat{\mathbf{G}}_R = \sum_{i=1}^{k} \hat{\mathbf{f}}_i \hat{\mathbf{f}}_i^T$$

Caution! When using this approach. The factors change values as new ones are added, and the eigenvalue for the last PC estimated is typically estimated with bias.

It can happen that "PC1" is really some other PC, and only by the addition of extra factors does this become clear.

"step-down" approach best -- start with m factors and see if there is a significant loss of fit moving to m-1.

Another check is to make sure that the trace is well-behaved, $\mathrm{tr}\left(\hat{\mathbf{G}}_{r,m+1}\right) - \mathrm{tr}\left(\hat{\mathbf{G}}_{r,m}\right) = \lambda_{m+1}$

Other eigenvalue-based measures of matrix dimension

Rank (number of positive eigenvalues) is in some sense a crude measure of dimension

As an extreme example, consider two $\mathbf{G}$ matrices both of rank 10. $\mathbf{G}_1$ has $\lambda_1 = 100, \lambda_2 = 10, \lambda_2 = \cdots = \lambda_{10} = 0.125$, while $\mathbf{G}_2$ has $\lambda_1 = \cdots = \lambda_{10} = 11$. Both matrices have the same rank and the same eigenvalue mean $\overline{\lambda} = 11$, but clearly there are far more constraints associated with $\mathbf{G}_1$.

Measures of the dispersion of the eigenvalues attempt to quantity this, as the more spread among the λ_i, the more potential constraints.

Wagner considered a measure based on the variance of the eigenvalues of the correlation matrix,

$$n_{d,W} = n - \sigma^2(\lambda)$$

Motivation is that for a diagonal matrix, all eigenvalues are one, hence zero variance and dim = n

With only a single positive eigenvalue, var = n-1, and dim =1

Kirkpatrick: Mean-standardize the traits, and then look at the fraction of variation accounted for by PC1,

$$n_{d,K} = \sum_{i=1}^{p} \frac{\lambda_i}{\lambda_1} = \frac{\text{trace}(\mathbf{G})}{\lambda_1}$$

Motivation: Total variance can be written as

$$\sum_{i=1}^{n} \lambda_i = \lambda_1 n_{d,K}$$

Kirkpatrick: Mean-standardize the traits, and then look at the fraction of variation accounted for by PC1,

$$n_{d,K} = \sum_{i=1}^{p} \frac{\lambda_i}{\lambda_1} = \frac{\mathrm{trace}(\mathbf{G})}{\lambda_1}$$

Motivation: Total variance can be written as

$$\sum_{i=1}^{n} \lambda_i = \lambda_1 n_{d,K}$$

With this metric, Kirkpatrick found that (for the traits he examined), all had effective dimensions less than 2!