Part I.

Rotation to recover factors lacking pure indicators

Wm. W. Rozeboom

Univ. of Alberta

I have progress to announce on a classic problem of Exploratory Factor Analysis. For those of you who favor Structural Equations Modelling for multivariate analysis, this may seem rather like advising computer engineers of new developments in slide rule design. But someday our methodology pendulum will swing back to a more seemly appreciation for inductivist science; and when that occurs you will be pleased to find that the art of EFA has moved far beyond the constricted routines for factor analysis now tendered by commercial data-analysis packages.

Background.

The EFA problem here addressed is rotation to simple structure when the best solution is considerably more complex than the independent-clusters ideal on which so much past study of common-factoring performance has focused. Rather than positioning this specifically in the context of psychometric research, I will put it in abstract terms that apply to any application of linear algebra wherein interpretable decompositions of Gramian matrices is an issue: Suppose we have a numerically identified symmetric real matrix $\mathbf{C}_{hh}$ of order p whose rank m is less than p. ~~In most~~ psychometric applications, $\mathbf{C}_{hh}$ ~~is the~~ usually matrix of covariances among the common parts H of p data variables Y, estimated by finding an order-p diagonal matrix $\mathbf{D}_u$ of uniquenesses such that $\mathbf{C}_{hh} + \mathbf{D}_u$ well-approximates the Y-variables' covariances $\mathbf{C}_{yy}$, while m is the number of common factors seemingly sufficient to explain the Y-covariances. (How optimal this reduction of $\mathbf{C}_{yy}$ to $\mathbf{C}_{hh}$ may be does not concern us here.) Then so long as $\mathbf{C}_{hh}$ has no negative eigenvalues, as we posit, it can be decomposed as a matrix product

$$(1) \qquad \mathbf{C}_{hh} = \mathbf{A} \cdot \mathbf{C}_{ff} \cdot \mathbf{A}'$$

where $\mathbf{A}$ and $\mathbf{C}_{ff}$ are real, m-by-m $\mathbf{C}_{ff}$ is well-conditioned symmetric with standardized diagonal (unities), and p-by-m matrix $\mathbf{A}$ has full column rank. Interpretively, factor pattern $\mathbf{A}$ estimates the coefficients by which m conjectured common factors F, whose correlations are $\mathbf{C}_{ff}$, determine scores on the H-components of variables Y; but again that doesn't concern us here except to explain our interest in what comes next.

When we hope – as factor analysts often do – to interptet the numbers in (1)'s factor pattern as manifesting the assorted degrees to which the Y-variables are differentially influenced by m common sources F, we must reckon with the algebraic complication that whenever $\mathbf{C}_{hh}$ has an m-factors decomposition of form (1) it has infinitely many others as well, namely,

$$(2) \qquad \mathbf{C}_{hh} = \mathbf{B} \cdot \mathbf{C}_{gg} \cdot \mathbf{B}' \qquad (\mathbf{B} = \mathbf{A} \cdot \mathbf{W}^{-1} , \ \mathbf{C}_{gg} = \mathbf{W} \cdot \mathbf{C}_{hh} \cdot \mathbf{W}')$$

for any invertable m-by-m real matrix $\mathbf{W}$ scaled to leave unities on the diagonal of $\mathbf{W} \cdot \mathbf{C}_{ff} \cdot \mathbf{W}'$. (Later, I will refer to the set of all Bs into which A can be transformed by some W in (2) as Rot[A].) Choosing to replace an initial solution for A by $\mathbf{B} = \mathbf{A} \cdot \mathbf{W}^{-1}$ for some choice of W other than $\mathbf{W} = \mathbf{I}$ is known for familiar geometric reasons as a "rotation" of A; and its problem for EFA is how to find Ws yielding the most intriguing configurations of coefficients in B.

What we aim to disclose by pattern rotation is a matter of taste. But there is considerable inductivist agreement that most interpretively provocative is for $\mathbf{B} \cdot \mathbf{C}_{gg} \cdot \mathbf{B}'$ to fit $\mathbf{C}_{yy} - \mathbf{D}_u$ tightly with low m/p ratio and a much higher proportion of zero (~~or~~ in practice, near-zero) elements in B than can be produced by rotations of randomly created matrices of A's order. Most classically ideal – the "independent clusters" pattern – is for each row of B to contain just one salient loading, that is, one too large to view as nearly zero. But that is far too extreme to be our exclusive target of rotation; more complicated patterns can be just as interpretively significant if not more so, and we want the ability to detect those whether they please us or not.

To discuss this situation, let each row of a factor pattern be called an "item" (more precisely, the "items" are whatever entities correspond to the pattern rows, in psychometric practice usually data variables), and say that the "complexity" of item *i* in a factor pattern **B**, however rotated, is the number of salient loadings (coefficients) in row *i* of **B**. (We leave open how large a loading should be to count as salient, and disregard that in practice the boundary between clearly salient and clearly nonsalient will be broadly fuzzy.) Then the factor "complexity" of an item in rotated pattern **B** is the number of salient loadings in its row, and rotation to achieve "simple structure" can be defined as attempting to minimize overall item complexity in the rotated pattern subject to constraints that may be imposed, notably factor orthogonality $\mathbf{C}_{gg} = \mathbf{I}$ or, more wisely, some less extreme limit on solution obliquity.

But finding simple structure in EFA is still an imperfect art. For in huge contrast to structural modelling, which posits in advance where the pattern zeros should go, exploratory factoring would like to search the entire rotation space Rot[**A**] of a received **A** to discover patterns in which item complexities are so low and provocatively configured that we cannot help but suspect that this positioning of factor axes aligns with the causal grain of these items' common sources. And since we can't inspect many patterns in Rot[**A**] individually, we need to search out a few most worthy of interpretive consideration by algorithms that optimize functions on Rot[**A**] which we take to measure overall pattern simplicity.

Distressingly, as Michael Browne addressed here last fall, rotation algorithms that can dependably find simple structure appreciably more complex than independent clusters are still in their infancy. Given a to-be-rotated pattern **A**, let us say that a pattern **T** in Rot[**A**] so simple-structuredly appealing that we would like to study it is a "target" for rotation of **A**. (Never mind that if one pattern in Rot[**A**] is provocative enough to qualify as a rotation target there will be a range of others that also qualify.) When a target **T** in Rot[**A**] contains several complexity-1 markers for each factor (that is, for each pattern column, several items with salient loading in that column only) and few items of higher complexity, it can pretty well be guaranteed that most currently standard oblique rotation algorithms, especially direct Oblimin and Promax, will retrieve a decent approximation to **T** from Rot[**A**]. But the quality of that approximation deteriorates if increasingly many pure indicators in the target are converted to higher complexity, especially complexity-2; and not many complexity increases are needed before the patterns these algorithms wrest from **A** scarcely hint at **T**. Indeed, I recently found to my dismay that this has also been true of my Hyball rotation procedures despite their exceptional proficiency at recovery of complex simple structure. But happily, I now also find that a simple adaptive weighting technique — we may as well call it "Comp2 weighting" — can substantially mitigate the complexity-2 problem in Hyball, and there is no evident reason why other rotation procedures cannot similarly profit from it. Before describing the computational nature of Comp2 weighting, I'll first try to convince you that this is well worth your attention.

Recovery of target patterns wherein Complexity-2 is prominent

For study of Comp2 weighting's effectiveness, consider the template in Table 1 for semi-random creation of target patterns with stipulated item complexities. Each 'x' in this template is a salient loading initially randomized in size between stipulated medium-to-large bounds while the remaining unmarked coefficients are randomized within a small stipulated interval around zero. Subsequent constrained-random assignment of communalities and factor correlations modifies these initial pattern elements somewhat, but preserves the contrast between large-to-medium salient and modest-to-negligible nonsalient loadings. The Table 1 template is for patterns that realize all different ways to assign complexities 1, 2, and 3 to the loadings of 25 items on 5 factors. Table 2*a* gives an instance of this schema, which for easy comprehension is shown in Table 2*b* rounded to one decimal with decimal points and zeros omitted. I will now show you assorted successes and failures in rotational recovery first of target 2*a*, and next of the 20-item pattern derived from 2*a* by deleting its complexity-1 items.

It has long been known, though still not adequately appreciated, that what a rotation algorithm retrieves from Rot[**A**] may well be not the pattern that globally optimizes its solution-quality measure but only a local optimum most accessible to the pattern in Rot[**A**] on which the algorithm is started. And its start pattern needn't be the received **A**; it can just as well be a previous rotation of **A** by another method, by the same method with altered parameter settings, or by random rotation of some pattern in Rot[**A**] already in hand. To test the upper limit of a rotation method's ability to recover a target pattern **T** that a simulation study knows at outset, we can give it **T** as input and observe the extent to which its output diverges from that. The rotations returned when the Table 2*a* pattern – hereafter the specific referent of "**T**" is rotated respectively by the Quartimin variant of direct Oblimin and Hyball's default indigenous procedure are shown as 1st-digit schemata in Tables 3*a*,*b* and with greater detail in Table 3-*alternate*. The Hyball rotation of **T** differs only trivially from **T** itself apart from yielding slightly cleaner zeros, and Oblimin rotation too preserves **T** decently though clearly not as well as Hyball. And in this particular instance, both Oblimin and default Hyball also get pretty much these same respective target recoveries under random sampling of other start positions.

> Note. Table 3-*alternate* is included with Tables 3*a*-3*c* here partly to exhibit the perspicuous pattern comparisons afforded by Hyball (any two factor patterns, not necessarily with the same number of factors or items, can be so juxtaposed if they have some rawdata items in common), but also to show you numerical examples of pattern similarity measure Divergence (Div) in comparison to the more familiar but less standardized RMS difference (RMSd) measure of agreement between pattern columns. Div and RMSd both increase as pattern match deteriorates in approximately linear proportion to one other; but the constant of proportionality is influenced by the shape and variability of terms in the vectors compared. You need some understanding of Div to appreciate Tables 5ff. below.

But as increasingly many complexity-1 markers are deleted from pattern **T**, target recovery diminishes with increasing severity, especially for Oblimin. Since there isn't time to take you through all steps in this progressive descent into rotation Hell, I will dive directly to bottom for results from the 20-by-5 target pattern – call it "**TT**" for "tough target" – derived from Table 2*a* by deleting the five **T** items assigned complexity-1 by its production template. In Table 4 you see the counterpart of Table 3 for **TT**, namely, easy-to-grasp roundings of **TT** and its rotations by default-style Hyball (Table 4*b*) and direct Oblimin (Table 4*c*). Hyball's recovery of **TT** from this ideal start is still excellent, but Oblimin's is largely a failure except in its column 5 and fairly decent column 4. (~~Quantitative~~ details are given in Table 5.) It now remains to show you how these methods perform (*a*) from practical start positions, and (*b*) when Comp2 weighting is put into play.

Divergence

First, however, I had better clarify what happens on Hyball runs. When Hyball starts its interactive work on a new input pattern (alternatively it can reload the log of a prior run to resume that) it conducts an open-ended sequence of rotations, pausing at end of each to log or maybe delete the latest result and allow, among other continuation options, a revised choice of solution style and control parameters (some of which affect the algorithm's to-be-optimized measure of hyperplane fit) as well as retrieval of a prior rotation in this run's log store (the list can be exhibited on screen with hyperplane appraisals and divergences among them) to serve as the next start position. And one of the continuation options is Spin search, which carries out a series of "Trys" – each a rotation from randomized start seeking to optimize the currently selected hyperplane-fit criterion – which are ranked in current-criterion quality, filtered to delete Trys closely matching ones of higher quality, and saved in a temporary buffer pending user decision on how many of the top-ranked Trys (the "Cream" of this Spin series) to log. (Deleting one or more of the rotations logged most recently is always a continuation option, so you can temporarily save more of a Spin search's Cream than you expect to want and make a final cut after comparing these along with previously logged patterns for hyperplane strengths and divergences.) Finally, after termination of this rotation run, Hyball-supplement programs can print appraisals of its logged rotations on various measures of pattern character (none shown here) along with detailed divergence comparisons such as their edited output excerpts in Table 3-*alternate* and Table 5 next to be discussed. (Square-bracketed words therein are edit additions.)

Table 5 reports on a log of rotations in Rot[TT] by various Hyball procedures from assorted start positions. Since rotations of extraction patterns often begin in practice with Varimax or less commonly Equamax, an orthogonalization of TT was rotated by Varimax and again by Equamax to be records No. 2 and No. 3 in this Hyball log starting with TT (No. 1). (Note that Equamax approximated TT substantially better than did Varimax, a superiority that has appeared consistently in my simulation studies with complex targets though not for simple ones.) The next three rotations in Table 5a from TT start hold no great interest beyond showing, after we average footers in Table 3-*alternate*, that whereas loss of Table 2*a*'s complexity-1 items degrades best-possible target recovery by SCAN/P from 6.1° to 9.2° Divergence (still quite good), Oblimin's best-possible goes from 14.8° (decent) to 35.8° (worthless), and STEP/P too does badly. (STEP/P's severe inferiority in No. 5 is atypical – as illustrated by the <7,8> comparison in Table 5*b*, seldom does same-start STEP veer far from SCAN – and some data bearing on the disparity in this case will surface in Part II.) Even so, rotation No. 4 by SCAN/P demonstrates that from a good start position Hyball's default rotation style can indeed recover TT quite well.

But what of success when rotation starts from a position available in practice, notably Varimax or better Equamax? Oblimin reaches almost exactly the same result from Equamax as it gets from ideal start (see Table 5*b*) or indeed from most positions in Rot[TT] (more evidence later) – which however has little value when, for higher-complexity targets, that recovery is so wretched. But neither are Hyball's rotations from Equamax start (Nos. 7,8) any better than Oblimin's in this difficult case.

However, what I have previously described as the "spin agenda"[1] – procedures for panning gold from the gravel of local optima when rotating for simple structure – is not merely practical but has great prowess in recovering elusively complex simple structure, especially when refined by Comp2 weighting. As already mentioned, Spin search is a series of rotations by any selected method variant (style and misfit measure) from random start positions from which the ranked and repetition-filtered Cream (best by criterion) are logged for further study. Table 5*a* exhibits the match to TT of Cream from six Spin searches of Rot[TT]. These prove little, since Spin scarcely ever replicates a search exactly; but they illustrate tendencies that Table 6 in Part II will make more perspicuous. And don't fret that you don't know what "Comp2 = .8" and "Comp2 = 1.8" mean – these are variants of Complexity-2 weighting now at brink of explanation. The salient tendencies:

a) Almost all returns of Spin search by Oblimin (cf. Nos. 10-12) are virtually identical (see the divergences among Nos. 6,9,10,12 in Table 5*b* and Part II's data on default Oblimin in Table 7); and the occasional deviant (cf. No. 11) matches TT just as worthlessly as the others.

b) Nos. 13-15: The best patterns in Spin's Cream under SCAN/P (Hyball's default style) without Comp2 weighting match TT rather nicely, nearly as good as SCAN/P from TT start (No. 4); but none of these – not even the TT start – recover all five target factors well.

c) Nos. 18-21: One Try in the top Cream of Spin this search by SCAN/P using strong Comp2 weighting without KS-norming (Comp2 = .8), matches TT excellently on all five factors. But it is ranked lower in criterion quality than three pronouncedly inferior matches to TT.

d) No. 22: The rank-1 return of STEP/P Spin search using strong Comp2 weighting without KS-norming, recovered only three axes of TT decently, and no other patterns in the Cream of this Spin series were any more accurate.

e) Nos. 23-30: The rank-1 return of Spin search by both SCAN/P and STEP/P under strong Comp2 weighting with KS-norming (Comp2 = 1.8) is an excellent match to TT on all factors. (A BEST BUY)

---

[1] Rozeboom, W. W. (1992). The glory of suboptimal factor rotation: Why local minima in analytic optimization of simple structure are more blessing than curse. Multivariate Behavioral Research, 27, 585-599.

Comp2 Weighting Revealed

Start of Part II ?

In the process of developing this presentation, I discovered having had at my fingertips an enormously powerful way to ascertain Spin-search tendencies not just in Hyball rotation but for any method of numerical analysis that seeks to optimize a loss-function having many local minima. Before detailing that, however, I had better give you specifics on Comp2 weighting before my presentation time runs out.

Complexity-2 items cause trouble for rotation to simple structure by any algorithm that converges to its solution through an iteration of planar rotations because any item with large loadings on both factors defining the plane in which simple structure is currently under improvement is a powerful magnet for attracting a hyperplane. Other items that load strongly on one but not the other pre-shift axis of this plane may suffice to anchor the hyperplanes their way; but lacking those, items that for best simple structure should remain at complexity-2 are apt to take control of hyperplanes over which by rights they should have no say. In principle, there is a simple answer to this problem: If, when rotating plane $\langle j,k \rangle$ during an iteration of planar rotations, we don't care how a particular item i's loadings should be apportioned between this plane's axes, then our rotation algorithm should disregard item i during this plane's rotation adjustments. And if i's loadings are large on both pre-shifted axes of plane $\langle j,k \rangle$ but tiny on all the other factors, then we may well have no objection to its remaining at complexity-2 in this plane after axis repositioning – which recommends assigning a weight to i that temporarily shrinks its loadings during $\langle j,i \rangle$ rotation to an extent proportionate to how strongly we want i's vote on axis positioning in this plane to be discounted. But care is needed in this lest markers for the axes we want in this plane also get emasculated.

Here is the weighting recipe with which I have tested this idea, with low startup expectations giving way to delight at how well it seemed to work on patterns having the full Table 1 layout:

> **Comp2 theory**: For each item i, let Σ_i^2 be the sum of i's squared loadings over all factors in the current pattern **A** while $\Sigma_{i,jk}^2$ is their sum only on factors j and k. Then $P_i =_{def} \Sigma_{i,jk}^2/\Sigma_i^2$ measures the prominence of factor plane jk in i, while $R_i =_{def} \mathrm{Min}(|A_{ij}|,|A_{ik}|)/\mathrm{Max}(|A_{ij}|,|A_{ij}|)$ reflects the degree to which these two factors are equally prominent in item i. So $\mathrm{Comp2}(i) =_{def} P_i \cdot R_i$ equals 1.0 when i lies entirely in plane jk with the same weight on both factors while decreasing to zero as either P or R decreases. Finally, with parameter Q set in the unit interval, item weight $W_i =_{def} 1 - Q \cdot \mathrm{Comp2}(i)$ diminishes i's effect on the rotation in plane jk with intensity of diminution damped by Q. Full rotation influence ($W_i = 1$) of this weight occurs when one of i's loadings in this plane jk is zero; from there, its influence decreases linearly with increasing $\mathrm{Comp2}(i)$ to a minimum of $W_i = Q$ when $\mathrm{Comp2}(i) = 1$. [Note: In Table 5, "Comp2" gives just the Q-level thereof.]

However, my initial elation over this apparent breakthrough on rotation to complex simple structure was rudely deflated when I tested its recovery of tough target TT. As noted above as tendency (c), although Spin search under Comp2 weights did indeed find near-perfect matches to TT, they were recovered unreliably and seldom in top Cream. But then I noticed that the production parameters I had chosen to make T more difficult than my previous tests had put large variation into the item communalities. For reasons unimportant here, I surmised that such inequalities might work to drag down the benefit of Comp2 weighting. And if so, that should be rectified by what I will call "KS-norming" (since it was popularized by Kaiser but first proposed, ~~Bob Pruzek informs me~~, by Saunders), namely, rescaling all the items to have unit variance (any other positive constant would serve as well) in common-factor space. So KS-norming is now an additional option when Hyball does Comp2 weighting, currently activated (efficienly even if ungainly) by adding 1 to the chosen value of parameter Q. We shouldn't expect that KS-norming will always enhance Comp2 prowess when communalities vary (that should depend on which items get juiced); but as tendency (e) points out above, it is certainly a winner in this particular application.

I have learned

In **PART II**, I will confirm tendencies (a)-(e) using a powerful way to exploit Spin search – megaSpin – whose full potential I had previously underappreciated. More broadly, I will urge that megaSpin should be a valuable addition to the toolbox of multivariate analysts who, when solving for model parameters by minimizing a function over parameter space that measures the quality of model fit,must deal with issues of local optima. [End of **PART I**]

TABLE 1. A template for generating common-factor patterns of 25 items on 5 factors realizing all item complexities at levels 1, 2, and 3. Locations of salient loadings are marked "x" and nonsalients by "."

Complexity-1						Complexity-2						Complexity-3					
1:	x	.	.	.	.	6:	x	x	.	.	.	16:	x	x	x	.	.
2:	.	x	.	.	.	7:	x	.	x	.	.	17:	x	x	.	x	.
3:	.	.	x	.	.	8:	x	.	.	x	.	18:	x	x	.	.	x
4:	.	.	.	x	.	9:	x	.	.	.	x	19:	x	.	x	x	.
5:	.	.	.	.	x	10:	.	x	x	.	.	20:	x	.	x	.	x
						11:	.	x	.	x	.	21:	x	.	.	x	x
						12:	.	x	.	.	x	22:	.	x	x	x	.
						13:	.	.	x	x	.	23:	.	x	x	.	x
						14:	.	.	x	.	x	24:	.	x	.	x	x
						15:	.	.	.	x	x	25:	.	.	x	x	x

TABLE 2.

Table 2*a*

A target pattern with Table-1 structure, item communalities in parentheses

1.	(.72)	.87	.04	-.05	.02	-.01
2.	(.46)	-.10	.63	.06	.0	-.14
3.	(.38)	.17	.09	.45	-.03	-.14
4.	(.33)	-.11	.11	.08	.54	-.06
5.	(.74)	.02	-.06	-.01	-.04	.85
6.	(.72)	.51	.62	.09	-.05	.08
7.	(.29)	.14	.0	.46	.02	.02
8.	(.35)	.42	.01	-.04	.41	-.02
9.	(.54)	.49	.05	-.04	.01	.55
10.	(.58)	-.04	.38	.60	-.03	.02
11.	(.62)	-.11	.36	.11	.71	-.04
12.	(.36)	.04	.31	.07	.05	.54
13.	(.67)	.07	.06	.57	.31	-.05
14.	(.68)	.04	-.07	.43	.09	.83
15.	(.70)	-.04	-.02	-.03	.62	.56
16.	(.61)	.41	.15	.47	-.04	-.02
17.	(.75)	.58	.48	.07	.37	.03
18.	(.47)	.27	.45	-.08	-.02	.47
19.	(.64)	.46	-.03	.38	.18	-.04
20.	(.27)	.36	.0	.16	-.01	.31
21.	(.50)	.30	.05	.01	.41	.47
22.	(.46)	-.06	.16	.29	.49	-.02
23.	(.65)	.02	.25	.45	-.01	.77
24.	(.66)	-.01	.52	-.01	.58	.46
25.	(.30)	.04	.03	.34	.29	.17

with factor correlations

Factor 1.	1.00				
Factor 2.	.0	1.00			
Factor 3.	.45	.26	1.00		
Factor 4.	.10	-.21	.47	1.00	
Factor 5.	.0	-.06	-.36	.0	1.00

Table 2*b*

Loadings over .10 in pattern 2a rounded to one decimal

1:[1]	9	.	.	.	.
2:[2]	.	6	.	.	-1
3:[3]	2	.	4	.	-1
4:[4]	-1	1	.	5	.
5:[5]	.	.	.	.	9
6:[6]	5	6	.	.	.
7:[7]	1	.	5	.	.
8:[8]	4	.	.	4	.
9:[9]	5	.	.	.	6
10:[10]	.	4	6	.	.
11:[11]	-1	4	1	7	.
12:[12]	.	3	.	.	5
13:[13]	.	.	6	3	.
14:[14]	.	.	4	.	8
15:[15]	.	.	.	6	6
16:[16]	4	2	5	.	.
17:[17]	6	5	.	4	.
18:[18]	3	4	.	.	5
19:[19]	5	.	4	2	.
20:[20]	4	.	2	.	3
21:[21]	3	.	.	4	5
22:[22]	.	2	3	5	.
23:[23]	.	3	4	.	8
24:[24]	.	5	.	6	5
25:[25]	.	.	3	3	2

TABLE 3.

Table 3*a*
1-decimal schema of target pattern 2b

1:[1]	9	.	.	.	.
2:[2]	.	6	.	.	-1
3:[3]	2	.	4	.	-1
4:[4]	-1	1	.	5	.
5:[5]	.	.	.	.	9
6:[6]	5	6	.	.	.
7:[7]	1	.	5	.	.
8:[8]	4	.	.	4	.
9:[9]	5	.	.	.	6
10:[10]	.	4	6	.	.
11:[11]	-1	4	1	7	.
12:[12]	.	3	.	.	5
13:[13]	.	.	6	3	.
14:[14]	.	.	4	.	8
15:[15]	.	.	.	6	6
16:[16]	4	2	5	.	.
17:[17]	6	5	.	4	.
18:[18]	3	4	.	.	5
19:[19]	5	.	4	2	.
20:[20]	4	.	2	.	3
21:[21]	3	.	.	4	5
22:[22]	.	2	3	5	.
23:[23]	.	3	4	.	8
24:[24]	.	5	.	6	5
25:[25]	.	.	3	3	2

Table 3*b*
1-decimal schema of the target's Hyball rotation

1:[1]	9	.	.	.	.
2:[2]	.	6	.	.	-1
3:[3]	1	.	5	.	-1
4:[4]	.	.	.	5	.
5:[5]	.	.	.	.	9
6:[6]	5	6	.	.	.
7:[7]	.	.	5	.	.
8:[8]	5	.	.	4	.
9:[9]	5	.	.	.	5
10:[10]	.	4	6	.	.
11:[11]	.	3	.	7	.
12:[12]	.	3	.	.	6
13:[13]	.	.	6	3	.
14:[14]	.	.	5	.	9
15:[15]	.	.	.	6	6
16:[16]	4	1	5	.	.
17:[17]	6	4	.	4	.
18:[18]	3	5	-1	.	5
19:[19]	4	-1	4	1	.
20:[20]	3	.	2	.	3
21:[21]	3	.	.	4	5
22:[22]	.	.	3	5	.
23:[23]	.	3	5	.	8
24:[24]	.	5	.	6	5
25:[25]	.	.	4	3	2

Table 3*c*
1-decimal schema of the target's Oblimin rotation

1:[1]	9	.	.	-1	.
2:[2]	-2	6	1	.	-2
3:[3]	2	.	5	.	.
4:[4]	.	.	1	5	-1
5:[5]	.	.	-3	.	8
6:[6]	4	6	1	.	.
7:[7]	2	.	4	.	.
8:[8]	5	-2	.	3	.
9:[9]	4	.	-2	.	5
10:[10]	.	4	6	.	.
11:[11]	.	1	2	7	-1
12:[12]	.	3	.	.	5
13:[13]	2	.	6	3	.
14:[14]	.	-1	2	.	8
15:[15]	.	-2	-2	6	4
16:[16]	4	1	5	.	.
17:[17]	6	3	1	3	.
18:[18]	2	4	-2	.	4
19:[19]	5	-1	4	1	.
20:[20]	3	.	.	.	3
21:[21]	3	-1	.	3	4
22:[22]	.	.	3	5	.
23:[23]	.	2	2	.	7
24:[24]	.	3	.	6	3
25:[25]	.	.	3	3	2

Table 3-alternate

[On left, juxtaposed columns of the target pattern (#2) and its Hyball rotation (#1).]
[On right, juxtaposed columns of the target pattern (#2) and its Oblimin rotation (#1).]

Each vertical cell of the table headed 'M N' comprises column M of #1 followed by its best-matching column N of #2. Pattern loadings are given to 2 decimals with point omitted; values larger than .99 are rounded down; and paired values both smaller than parameter CUT [here .10] are blanked. At the table's foot, Div is the column-pair's congruence divergence while RMSd is the root-mean-square difference (× 100) of their corresponding elements. If paired factors are negatively congruent as received, #1 is reflected when computing Div and RMSd. (Reminder. The "divergence" of two conforming real vectors is the angle in degrees whose cosine is their unsigned congruence coefficient.)

NOTE: The variables' sourcefile indices are substituted for their unavailable names.

?

Why blank?

Hyball rotation (#1) of Target (#2)

#1 #2	1 1	2 2	3 3	4 4	5 5
[1]	87 87	. .	. .	. .	. .
[2]	. .	64 63	. .	. .	-13-14
[3]	12 17	. .	50 45	. .	-13-14
[4]	-6-11	4 11	. .	54 54	. .
[5]	. .	. .	. .	. .	87 85
[6]	55 51	62 62	. .	. .	. .
[7]	8 14	. .	52 46	. .	. .
[8]	46 42	. .	. .	40 41	. .
[9]	48 49	. .	. .	. .	55 55
[10]	. .	35 38	64 60	. .	. .
[11]	-2-11	28 36	6 11	72 71	. .
[12]	. .	35 31	. .	. .	56 54
[13]	. .	. .	62 57	29 31	. .
[14]	. .	. .	52 43	. .	87 83
[15]	. .	. .	. .	61 62	59 56
[16]	35 41	~~10 15~~	~~52 47~~	. .	. .
[17]	65 58	42 48	. .	38 37	. .
[18]	31 27	49 45	-10 -8	. .	47 47
[19]	41 46	-10 -3	43 38	16 18	. .
[20]	32 36	. .	20 16	. .	32 31
[21]	32 30	. .	. .	40 41	49 47
[22]	. .	~~9~~ 16	29 29	49 49	. .
[23]	. .	28 25	51 45	. .	81 77
[24]	. .	50 52	. .	60 58	49 46
[25]	. .	. .	37 34	28 29	20 17
Div	9.0°	9.7°	5.5°	2.8°	2.6°
RMSd	5.0	4.8	4.3	1.5	2.3

Oblimin rotation (#1) of Target (#2)

#1 #2	1 1	2 2	3 3	4 4	5 5
[1]	85 87	. .	. .	-12 2	. .
[2]	-17-10	63 63	11 6	. .	-15-14
[3]	20 17	. .	48 45	. .	-8-14
[4]	-4-11	-8 11	12 8	54 54	-11 -6
[5]	. .	. .	-26 -1	. .	79 85
[6]	41 51	61 62	12 9	. .	. .
[7]	18 14	. .	44 46	. .	. .
[8]	47 42	-15 1	. .	31 41	. .
[9]	43 49	. .	-16 -4	. .	51 55
[10]	. .	37 38	57 60	. .	. .
[11]	-5-11	10 36	16 11	73 71	-12 -4
[12]	. .	28 31	. .	. .	48 54
[13]	15 7	. .	58 57	34 31	. .
[14]	. .	-13 -7	18 43	. .	80 83
[15]	. .	-25 -2	-16 -3	57 62	44 56
[16]	42 41	13 15	48 47	. .	. .
[17]	56 58	32 48	14 7	31 37	. .
[18]	16 27	44 45	-18 -8	. .	41 47
[19]	52 46	-13 -3	41 38	13 18	. .
[20]	34 36	. .	9 16	. .	31 31
[21]	31 30	-12 5	. .	33 41	39 47
[22]	. .	-2 16	31 29	51 49	. .
[23]	. .	22 25	22 45	. .	74 77
[24]	. .	30 52	. .	59 58	33 46
[25]	. .	. .	29 34	29 29	16 17
Div	10.3°	25.3°	21.3°	10.4°	6.8°
RMSd	5.7	12.1	10.7	5.6	5.6

Qualitative interpretation of Div:
- Under 5°; near-perfect match
- Circa 10°; excellent match
- Circa 20°; decent but not great match
- Circa 30°; poor match
- Over 35°; forget it

X **TABLE 4.** Cutoff in the table was inadvertently .20 rather than the intended .10

Table 4a
1-decimal schema of target pattern TT

1:[6]	5	6	.	.	.
2:[7]	x	.	5	.	.
3:[8]	4	.	.	4	.
4:[9]	5	.	.	.	6
5:[10]	.	4	6	.	.
6:[11]	x	4	.	7	.
7:[12]	.	3	.	.	5
8:[13]	.	.	6	3	.
9:[14]	.	.	4	.	8
10:[15]	.	.	.	6	6
11:[16]	4	.	5	.	.
12:[17]	6	5	.	4	.
13:[18]	3	4	.	.	5
14:[19]	5	.	4	.	.
15:[20]	4	.	.	.	3
16:[21]	3	.	.	4	5
17:[22]	.	.	3	5	.
18:[23]	.	3	4	.	8
19:[24]	.	5	.	6	5
20:[25]	.	.	3	3	.

Table 4b
1-decimal schema of TT's SCAN/P Hyball rotation

1:[6]	5	6	.	.	.
2:[7]	.	.	5	.	.
3:[8]	5	.	.	4	.
4:[9]	5	.	.	.	6
5:[10]	.	4	6	.	.
6:[11]	.	.	.	7	.
7:[12]	.	4	.	.	5
8:[13]	.	.	6	3	.
9:[14]	.	.	5	.	9
10:[15]	.	.	.	6	6
11:[16]	4	.	5	.	.
12:[17]	6	4	.	4	.
13:[18]	2	5	.	.	4
14:[19]	5	.	4	.	.
15:[20]	3	.	.	.	3
16:[21]	3	.	.	4	5
17:[22]	.	.	3	5	.
18:[23]	.	3	5	.	8
19:[24]	.	4	.	6	5
20:[25]	.	.	4	3	2

Table 4c
1-decimal schema of TT's Oblimin rotation

1:[6]	.	8	.	.	.
2:[7]	.	.	5	.	.
3:[8]	3	.	5	.	.
4:[9]	3	3	.	-2	5
5:[10]	-5	2	3	.	.
6:[11]	.	.	.	7	.
7:[12]	x	3	-2	.	4
8:[13]	.	.	6	3	.
9:[14]	.	.	.	.	8
10:[15]	3	-2	.	4	5
11:[16]	.	3	6	.	.
12:[17]	.	5	4	2	.
13:[18]	.	5	-2	.	3
14:[19]	.	.	8	.	.
15:[20]	.	.	3	.	3
16:[21]	3	.	2	.	4
17:[22]	.	.	3	5	.
18:[23]	.	.	.	.	7
19:[24]	.	3	.	6	3
20:[25]	.	.	3	2	2

TABLE 5a

PATTERN DIVERGENCES

In each row M of the congruence report for a pattern pair <L,M>, entry J is the angle in degrees of congruence divergence between the Jth column of pattern L and its counterpart in the best-matching permutation of pattern M.

[Notes: SCAN/P, STEP/P, SCAN/S, and STEP/S are different Hyball algorithms for optimizing its selected hyperplane-misfit measure by iterating planar rotations. (SCAN differs from STEP in within-plane procedure; P vs. S are alternative ways to package planar shifts.) Each bar-connected block of consecutive solutions reported below comprises the leading Trys in a Spin series ordered by hyperplane quality as rated by the currently-selected measure ("criterion") of this. More specifically, each Spin series initially ranked the 99 best-by-criterion of 500 Trys, followed by final choice of how many leaders to log after pruning out ones nearly identical to better ones. For brevity, all but one of the present Spin series have been cropped after the first or second good recovery of TT. No. 22 is the only one logged from its series because no lower-ranked Try therein matched TT appreciably better.]

[Good or at least decent matches to TT are marked >> or >]

Congruence match (degrees divergence) of pattern No. 1 [target TT] to pattern

```
   2: (Av = 34.1)  34.4 45.2 48.9 28.2 13.9   Varimax rotation of orth'd TT
   3: (Av = 24.6)  36.8 29.6 28.1 22.4  6.1   Equamax rotation of orth'd TT
```

[Best possible without Comp2 weights:]

```
 > 4: (Av =  9.2)   8.3 20.0  5.8  7.1  5.0   SCAN/P rotation of TT (No. 1)
   5: (Av = 30.7)  46.4 24.2 36.6 41.9  4.5   STEP/P rotation of TT (No. 1)
   6: (Av = 35.8)  70.2 31.8 41.8 24.3 10.8   OBLMIN rotation of TT (No. 1)
```

[Recovery from standard start without Comp2 weights:]

	7:	(Av = 35.6)	46.8	45.9	39.6	41.2	4.7	SCAN/P rotation of Equamax (No. 3)
	8:	(Av = 35.3)	47.0	47.5	36.8	40.7	4.6	STEP/P rotation of Equamax (No. 3)
	9:	(Av = 33.6)	62.7	32.2	37.4	24.9	10.7	OBLMIN rotation of Equamax (No. 3)

[Recovery by Spin search without Comp2 weighting:]

	10:	(Av = 34.6)	65.3	31.7	39.4	26.0	10.4	OBLMIN Spin
	11:	(Av = 36.4)	25.3	56.1	62.4	29.6	8.7	[Same Spin series, 2rd best]
	12:	(Av = 33.2)	61.4	32.1	37.5	24.2	10.9	[" " " 3rd "]
>	13:	(Av = 10.0)	16.3	15.1	4.9	8.2	5.7	SCAN/P Spin, Comp2 = nil
	14:	(Av = 13.8)	9.7	40.3	5.6	8.5	5.1	[Same Spin series, 2rd best]
>	15:	(Av = 11.1)	14.9	21.0	4.6	8.8	6.0	[" " " 3rd "]
	16:	(Av = 22.1)	8.1	47.1	8.8	41.8	4.8	[" " " 4th "]
	17:	(Av = 23.6)	7.3	39.9	37.1	27.9	5.9	[" " " 5th "]

[Recovery by Spin search with Comp2 weighting but no KS-norming:]

	18:	(Av = 36.3)	61.0	43.7	38.4	35.5	2.9	SCAN/P Spin, Comp2 = .8
	19:	(Av = 38.1)	64.6	43.9	39.1	39.1	4.1	[Same Spin series, 2rd best]
	20:	(Av = 21.5)	44.2	44.7	6.7	8.7	3.2	[" " " 3rd "]
>>	21:	(Av = 6.8)	7.4	9.4	5.8	7.1	4.1	[" " " 4th "]
	22:	(Av = 23.7)	47.5	49.2	7.3	10.0	4.7	STEP/P Spin, Comp2 = .8

[Recovery by Spin search with Comp2 weighting and KS-norming:]

>>	23:	(Av = 6.6)	9.4	8.0	6.2	4.8	4.4	SCAN/P Spin, Comp2 = 1.8
	24:	(Av = 22.4)	45.6	48.8	7.2	6.4	4.1	[Same Spin series, 2rd best]
	25:	(Av = 32.0)	50.1	49.6	6.9	5.6	47.6	[" " " 3rd "]
	26:	(Av = 14.8)	9.5	47.2	6.7	5.1	5.6	[" " " 4th "]
>>	27:	(Av = 6.9)	10.1	8.2	5.5	5.6	5.0	STEP/P Spin, Comp2 = 1.8
	28:	(Av = 29.4)	10.7	77.6	7.0	4.8	46.7	[Same Spin series, 2rd best]
	29:	(Av = 22.9)	47.8	49.2	7.2	6.1	4.3	[" " " 3rd "]
	30:	(Av = 13.0)	8.7	9.1	34.4	8.5	4.5	[" " " 4th "]

TABLE 5*b*.

Congruence Summary [an illustrative excerpt of information available]:
For each pair of these rotations, average divergence of their matched columns

1:	.0														
2:	34.1	.0													
3:	24.6	25.1	.0												
4:	9.2	34.4	25.7	.0											
5:	30.7	40.2	26.8	26.1	.0										
6:	35.8	38.2	25.6	33.0	24.6	.0									
7:	35.6	34.1	26.2	32.3	21.3	23.5	.0								
8:	35.3	33.5	26.4	32.0	21.7	23.7	1.9	.0							
9:	33.6	40.1	22.6	31.1	21.4	4.5	20.5	20.7	.0						
10:	34.6	39.0	23.7	31.8	22.4	2.9	21.1	21.3	2.5	.0					
11:	36.4	34.7	37.9	34.4	35.8	23.3	28.1	27.9	25.3	23.8	.0				
12:	33.2	40.0	22.5	30.7	21.1	4.7	20.6	20.8	.9	2.9	25.8	.0			
13:	10.0	34.0	25.9	7.7	26.9	32.6	30.2	29.8	30.2	31.2	35.8	29.9	.0		
14:	13.8	39.0	36.7	8.3	32.0	33.9	36.7	36.0	32.4	33.4	35.5	32.5	13.9	.0	
15:	11.1	33.5	24.5	3.2	24.0	31.3	30.8	30.5	29.2	29.9	35.0	28.8	5.7	10.4	.0

. . . .

. . . .