Journal of Statistical Software

MMMMMM YYYY, Volume VV, Issue II. http://www.jstatsoft.org/

An Excel Tool for Statistical Analysis

Fanghu Dong
University of Hong Kong

Guosheng Yin
University of Hong Kong

Abstract

This article presents a macro-enhanced Excel file for statistical analysis, intended as a portable, WYSIWYG tool for analyzing medium sized data. The current tool also demonstrates the inner workings of several widely applied multivariate methods (as selected by Anderson (2003) and Johnson and Wichern (1992)) on a spreadsheet as a side effect. For each method, a large portion of the function chain between the raw data and the final statistics is kept to allow users to perform interactive studies such as data perturbation, formula branching (to experiment an ad-hoc idea), and visualization of internal stages of the analysis. Users can see all the key variables at the same time, allowing them to quickly identify some close relationships between the results. The current tool can be used to produce end results, to facilitate in the model construction stage, and for instructional purposes as well. It includes several plotting macros and is compatible with both Windows and Mac versions of Excel. It is a useful addition to an Excel user's statistical toolkit.

Keywords: Canonical Correlation Analysis, Excel®, Factor Analysis, General Linear Model, Longitudinal Analysis, Linear Discriminant Analysis, Multivariate ANOVA, Multivariate Regression, Principal Component Analysis, Visual Basic for Applications.

1. Introduction

To expose the method of core multivariate statistics, one needs a more transparent statistical software rather than the one used by professionals for production purpose. It is by the good software engineering principle to hide the implementation details away from the user. However, some proper amount of detail is exactly what is needed for understanding the methods, and only a deep methodological understanding could enable dexterous usage of softwares. Arguably, the ideal way to acquire such understanding is to code one's own implementation of a method like Multivariate Regression with the explicit goal of lining the outputs (coefficient estimates, standard deviation, MANOVA statistics, etc.) up with those of an established software. A good place to carry out such line-up is on a spreadsheet, where one can have a view of the entire "memory" layout and its dynamic updating that is monitored by an event system and orchestrated by the functional evaluator. Microsoft Excel is a very popular spreadsheet software. It is fully integrated with the highly productive Visual Basic for Applications (VBA) language (See Green, Bullen, Bovey, and Alexander (2007) and MSDN (2013)). VBA complements the sheet-level functional environment with procedural programming (e.g., loops, state variables, classes) and integrates with Excel so closely that it can automate literally everything that one does manually on Excel. And even the automation itself is automated. Excel carries a macro recording utility to automate the coding of manual operations. Excel also implements a set of data visualization utilities that produce several types of sophisticated plots that can be made with a few selections and clicks.

Disadvantages of Excel may include reduced speed and a hard cap of data size when facing big datasets. For Excel add-in development under higher speed and memory requirement, extensions are commonly written as `COM` `dll` using C++ and/or on the `.NET` platform using C# or VB.NET through the Visual Studio Tools for Office (`VSTO`) and therefore is currently hinged to the Windows platform. There could be other short-comings perceived with individual developer's experience. Despite of these, Excel is still a popular numerical environment for mathematical modeling. It supports basic matrix mathematics (multiplication, inversion, and determinant), includes many of the building-block functions in mathematics and statistics, and has the basic functions and utilities for text processing. Finally, for Windows users, Excel has access to unlimited number of `dll` files that exposes functions and objects to `COM`.

The software closely follows Anderson (2003) and Johnson and Wichern (1992) in choice of the methods for implementation. These include classical works of Hotelling, Fisher, and Pearson, including a multivariate version of the t–test, covariance tests, canonical correlation (as a measure of association between two sets of variables), linear discrimination (as a supervised classification algorithm), principal component (as a data orthogonalization algorithm), Factor analysis (to reduce correlation by splitting random factors of the covariance matrix), and the General Linear Model (mixture of continuous and categorical regressors predicting multivariate continuous response with multivariate ANOVA).

The entire project including this document has been setup on github at `https://github.com/josephdfh/ExcelStatTool`.

2. Examples

In this section, we explain the tool to a general readership using examples in the style of "Introduction to Statistics with Excel Tool". While we focus on the most important aspects implemented in the current software we cannot cover all of them. We have to also rely on the reader to use the magnifier function came with the electronic document viewer to view those condensed figures.

2.1. Regression with Excel Tool

This first example is reserved for multiple regression for its unsaid importance among all statistical methods. Multiple regression models the mean value of a single y-variable by the linear combination of a chosen set of x-variables. We use the "Correl" (Correl) sheet to perform multiple regression and a manual variable selection for the prostate cancer data

(Stamey, Kabalin, McNeal, Johnstone, Freiha, Redwine, and Yang (1989), Halvorsen (2015)). A copy of the dataset can be found on the "Data" (Data) sheet under name "Prostate" using the dropdown menu of cell `Data!H2`. The column "lpsa" is the y-variable; some or all of the other columns can be included as x-variables. The following steps can be followed to reproduce the figured states.

1. Copy the Prostate dataset from the sheet "Data" (Data) to the system clipboard then immediately switch to the "Pivot" (Pivot) sheet and double-click the top-left green cell at `Pivot!A1` to create a working copy of the dataset on the "Pivot" (Pivot) sheet. All preprocessing operations will be performed on this copy.

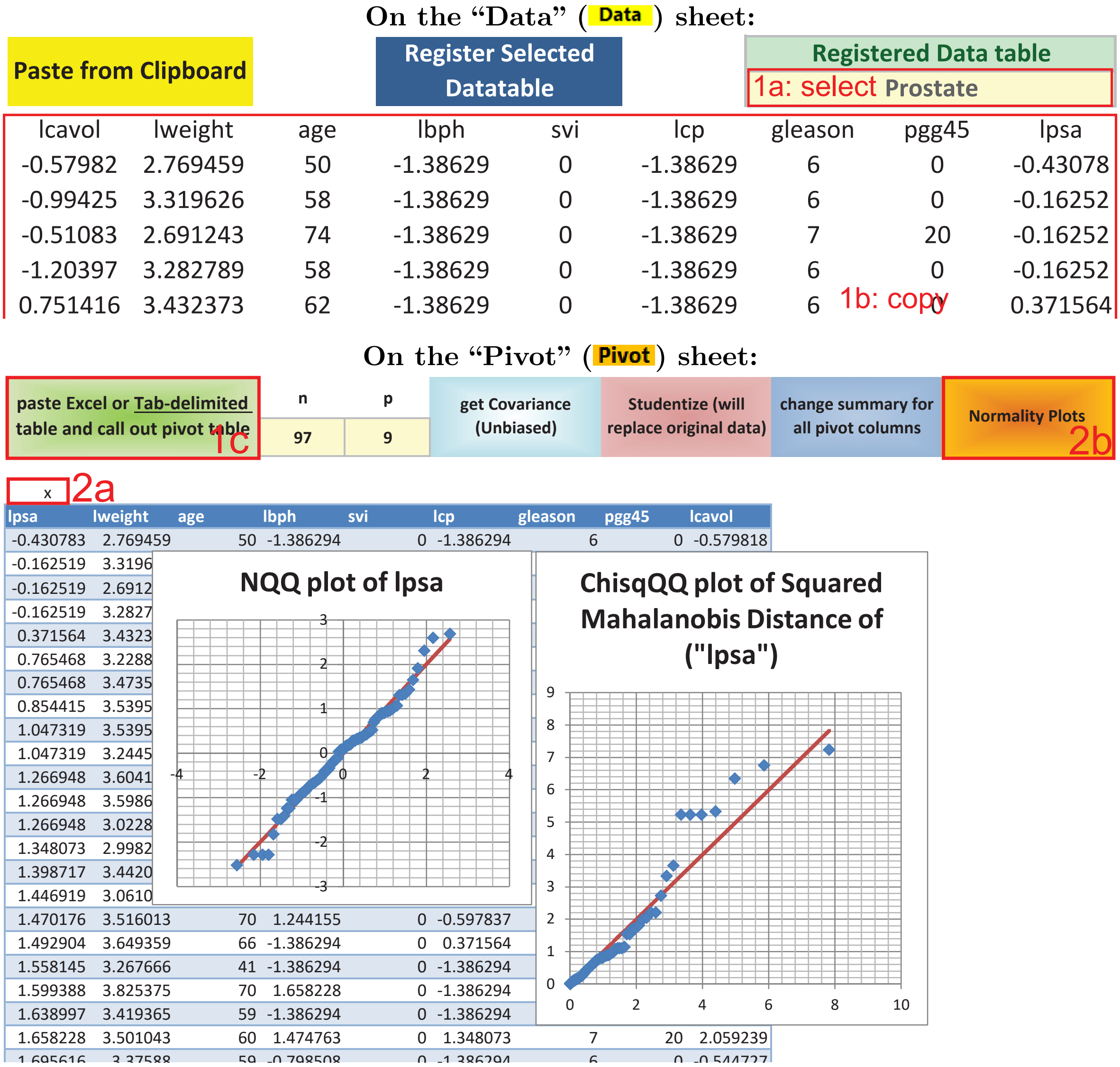
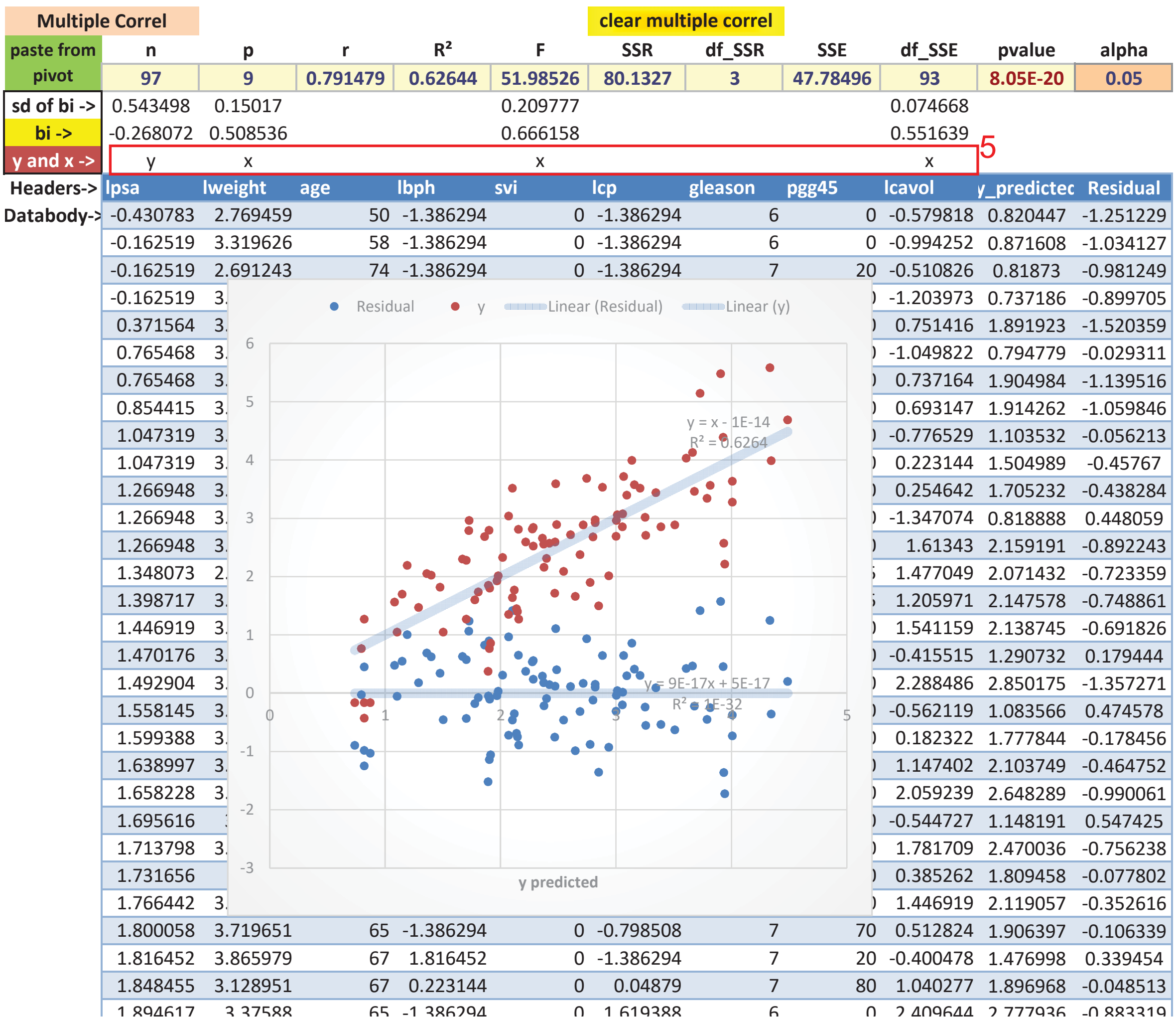


Figure 1: **On the "Data" sheet:** The dataset is registered in the dropdown box by name "Prostate". The numerical marks have corresponding descriptions in the text. **On the "Pivot" sheet:** Some initial exploration in this dataset can be made, for example, starting with normality checks.

2. Select column "lpsy" by putting an "x" above the header (a reordering of columns will be triggered to prioritize the selected) and double-click the orange button "Normality Plots" (**Normality Plots**) to have a visual check of the response variable's normality condition. The plots show that the data have a little bit excess kurtosis over that of a normal distribution. Nevertheless, we will proceed for demonstration purpose.

3. Activate the "Correl" (**Correl**) sheet. On the "Correl" (**Correl**) sheet, double-click on the green cell at `Correl!A26` to copy-paste the dataset from the "Pivot" (**Pivot**) sheet and augment it with two additional columns: the fitted response "`y_predicted`" and the residuals of fitting. By creating a further copy of the preprocessed data in each analysis sheet, we are free to change the analysis copy without side-affecting other methods on the same preprocessed data.



Multiple Correl						clear multiple correl					
paste from pivot	n	p	r	R^2	F	SSR	df_SSR	SSE	df_SSE	pvalue	alpha
	97	9	0.791479	0.62644	51.98526	80.1327	3	47.78496	93	8.05E-20	0.05
sd of bi ->	0.543498	0.15017			0.209777				0.074668		
bi ->	-0.268072	0.508536			0.666158				0.551639		
y and x ->	y	x			x				x	5	
Headers->	lpsa	lweight	age	lbph	svi	lcp	gleason	pgg45	lcavol	y_predictec	Residual
Databody->	-0.430783	2.769459	50	-1.386294	0	-1.386294	6	0	-0.579818	0.820447	-1.251229
	-0.162519	3.319626	58	-1.386294	0	-1.386294	6	0	-0.994252	0.871608	-1.034127
	-0.162519	2.691243	74	-1.386294	0	-1.386294	7	20	-0.510826	0.81873	-0.981249
	-0.162519	3.							-1.203973	0.737186	-0.899705
	0.371564	3.							0.751416	1.891923	-1.520359
	0.765468	3.							-1.049822	0.794779	-0.029311
	0.765468	3.							0.737164	1.904984	-1.139516
	0.854415	3.							0.693147	1.914262	-1.059846
	1.047319	3.							-0.776529	1.103532	-0.056213
	1.047319	3.							0.223144	1.504989	-0.45767
	1.266948	3.							0.254642	1.705232	-0.438284
	1.266948	3.							-1.347074	0.818888	0.448059
	1.266948	3.							1.61343	2.159191	-0.892243
	1.348073	2.							1.477049	2.071432	-0.723359
	1.398717	3.							1.205971	2.147578	-0.748861
	1.446919	3.							1.541159	2.138745	-0.691826
	1.470176	3.							-0.415515	1.290732	0.179444
	1.492904	3.							2.288486	2.850175	-1.357271
	1.558145	3.							-0.562119	1.083566	0.474578
	1.599388	3.							0.182322	1.777844	-0.178456
	1.638997	3.							1.147402	2.103749	-0.464752
	1.658228	3.							2.059239	2.648289	-0.990061
	1.695616								-0.544727	1.148191	0.547425
	1.713798	3.							1.781709	2.470036	-0.756238
	1.731656								0.385262	1.809458	-0.077802
	1.766442	3.							1.446919	2.119057	-0.352616
	1.800058	3.719651	65	-1.386294	0	-0.798508	7	70	0.512824	1.906397	-0.106339
	1.816452	3.865979	67	1.816452	0	-1.386294	7	20	-0.400478	1.476998	0.339454
	1.848455	3.128951	67	0.223144	0	0.04879	7	80	1.040277	1.896968	-0.048513
	1.894617	3.37588	65	-1.386294	0	1.619388	6	0	2.409644	2.777936	-0.883319

Figure 2: Correl sheet: Multiple Regression

4. Enter "y" in cell `Correl!B30` and "x" to any subset of cells `Correl!C30:J30` while mointoring the R^2 at cell `Correl!E27`. After a few trials, one may quickly settle to the subset of "lweight", "svi", "lcavol" giving an $R^2 = 0.62644$. The implementation hides

the trigger of the regression computation in the cell value change event. Whenever the row above the header of the analysis copy of data has some value change, all regression results on the sheet will be refreshed. This allows interactive variable selection to be performed seemlessly. One can then quickly make some visualizations of the numbers. The "Correl" (Correl) sheet now appears as Figure 2.

One may quickly verify the three ways of computing R^2 in multiple regression:

$$R^2 = \frac{SSR}{SSR + SSE} = \mathbf{R}_{yx}\mathbf{R}_{xx}^{-1}\mathbf{R}_{xy} = \mathrm{corr}(y, \hat{y})^2$$

The second way is coded into the formula for r at cell `Correl!D27` (Figure 3). It interprets multiple regression as a process of maximizing the squared correlation between the response and a vector in the linear space spanned by the regressors. And the correlation-maximizing vector is the $\hat{y}$.

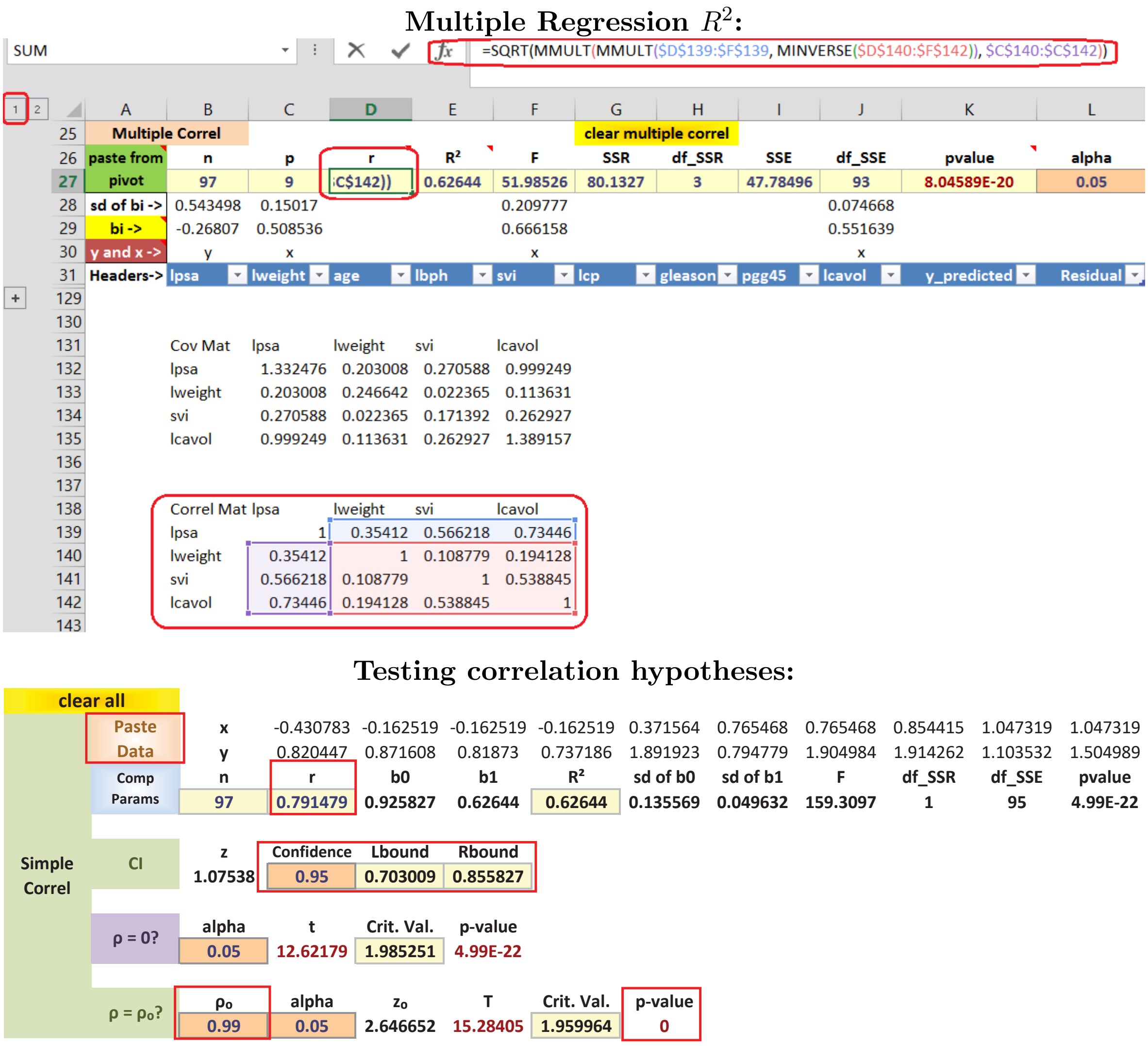


Figure 3: Multiple Regression R^2 and Hypotheses about Corrlelation

Now we test the hypothesis that R^2 is close enough to 1, or, equivalently, $H_0 : \mathrm{corr}(y, \hat{y}) = 1$.

1. Select the two columns of "y" and "`y_predicted`" holding the `ctrl` key. Copy the selection.

2. Double click cell `Correl!B2`. A few results are already shown. For example, the 95%-confidence interval of $\mathrm{corr}(y, \hat{y})$ is $[0.703, 0.856]$ based on the Fisher z-transform (Fisher 1915).

3. Enter 0.99 in cell `Correl!C14`. The p-value of testing $H_0 : \mathrm{corr}(y, \hat{y}) = 0.99$ shows up as 0 in cell `Correl!H14` indicating the model has left unexplained a non-zero portion of variability in the response variable.

2.2. Multivariate Regression Involving Categorical Variables

In a general regression setup, one frequently encounters more than one response variable and categorical variables in the regressors. The "LM" (LM) sheet is implemented for this task. LM stands for Linear Model. The initiation step is similar as before: after the data is pasted to the "Pivot" (Pivot) sheet and preprocessed there, one switch to the "LM" (LM) sheet and double-clicks the green paste-from-pivot button at the top-left corner to create a working copy of the dataset.

One then specifies a `y` ahead of each response column, an `x` ahead of each continuous regressor column and a `c` ahead of each categorical regressor column. A second specification, regarding Rectangle 3 of Figure 4, is needed to indicate which of the `x` and `c` columns will finally be used with an `x` in the cells above the Working Data. Note that the categorical variables in Rectangle 2 are auto-encoded into dummy variables of Rectangle 4 in Figure 4.

The coefficient and standard deviation estimates of the regression are output in Rectangle 6. In addition to estimation of the regression coefficients, the sheet also implements Multivariate-ANOVA tests and transformation matrices on both continuous responses and continuous regressors.

The orange cells in Rectangle 7 allows specifying two linear transformations of both the response variables (`y`) and the continuous regressors (`x`) and then running the regression on the linearly transformed data.

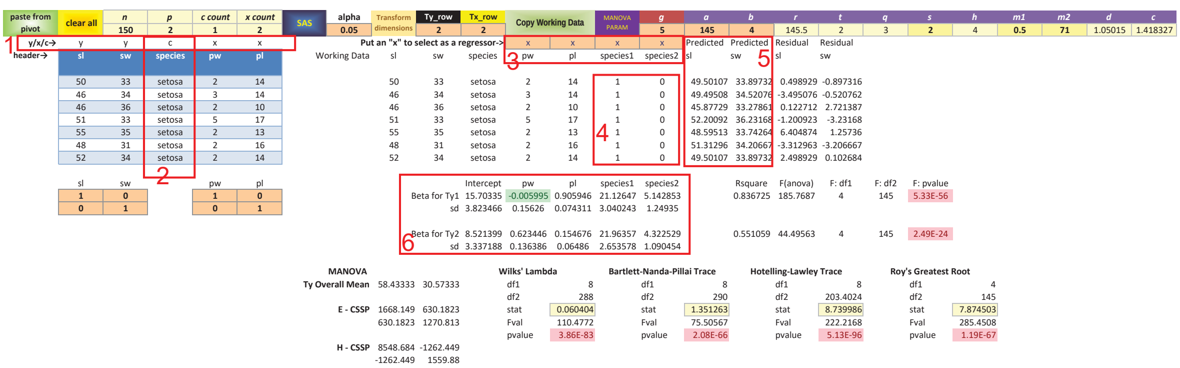


Figure 4: Output of the Linear Model sheet

2.3. Principal Component Analysis

The main purpose of principal component analysis is dimension reduction and, if one assumes multivariate normality of data, de-correlation as a side effect. The "PCA" (PCA) sheet implements principal component analysis with the multivariate normality assumption. We demonstrate its usage with the Men's track sports data stored under the name "Men" on the "Data" (Data) sheet. As before, we copy this raw data table, paste it to the "Pivot" (Pivot) sheet, switch to the "PCA" (PCA) sheet and double-click on the top-left green button with text "Paste from Pivot". The actual PCA computation is event-triggered in the same fashion as seen before on the "Correl" (Correl) sheet: whenever an `x` is put above a data column, the entire sheet is regenerated to display a new set of PCA results. The computation is very fast so one barely notice any delay. The "Men" dataset has 8 numerical columns; to enter 8 `x`'s simultaneously: Select cells `PCA!C3:J3` | press `x` on the keyboard | Windows user: press `ctrl` + `Enter` on the keyboard; Mac user: Press `command` + `Enter`. The "PCA" (PCA) sheet should appear as Figure 5 where the only inputs required are marked by Rectangles 1, 2, a, b, and d while the rest are generated content.

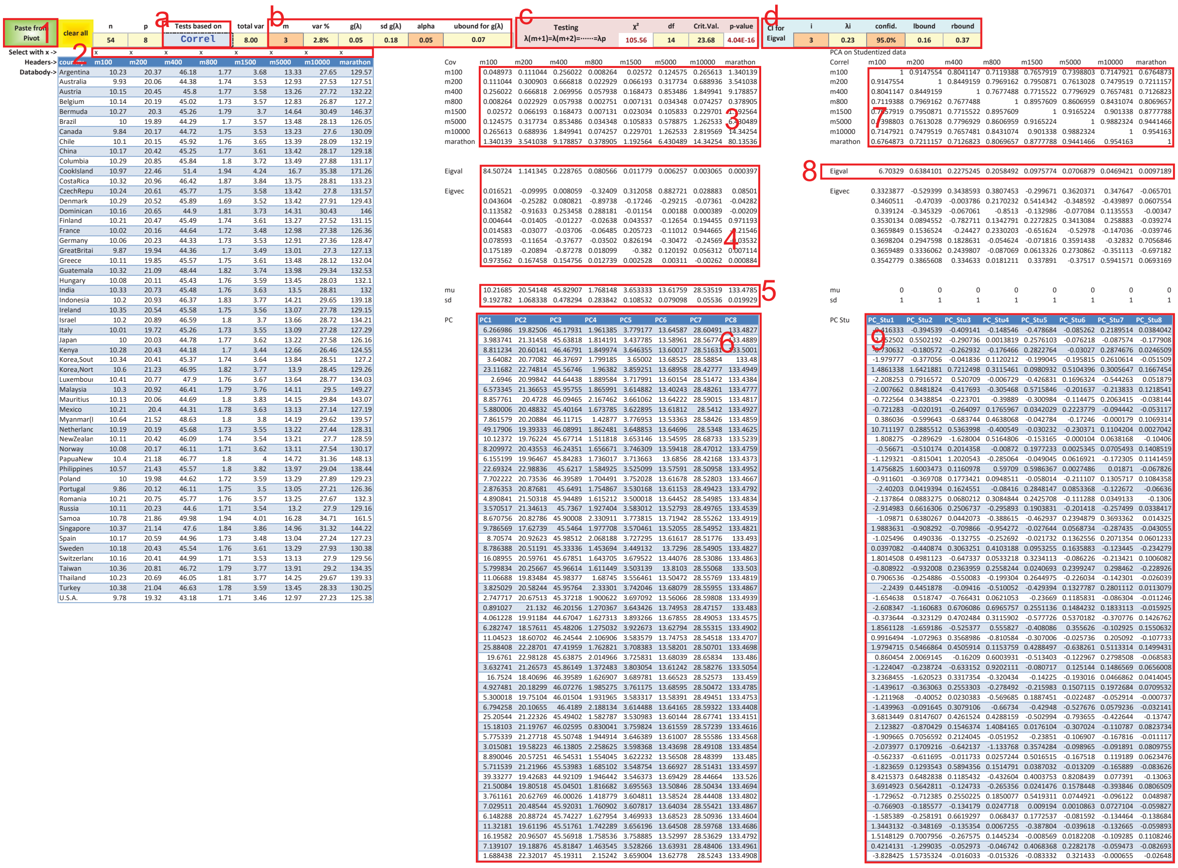


Figure 5: The "PCA" sheet.

Rectangles 3 and 7 display the covariance and correlation matrices. These are the starting points of two branches of computation. Rectangle 4 displays the eigenvalues and the corresponding eigenvectors of the covariance matrix (Rectangle 3) computed by the formula

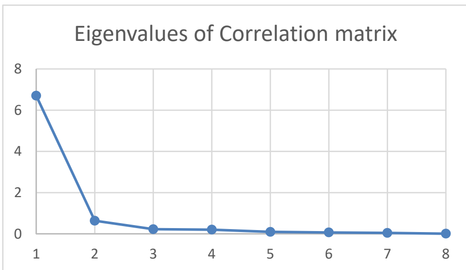
```
=CovEigenDecompQR(Cov)
```

calling the underlying VBA function. It returns as the first row the eigenvalues, followed by a blank row followed by a matrix consisted of the eigenvectors as its columns. The full signature of the function is `Function CovEigenDecompQR(A, Optional maxiter = 1000, Optional eigvec, Optional eigval, Optional returnColumnBound As Boolean = False)`. The second argument specifies the maximum iterations allowed before returning. The third and fourth arguments are call-back arguments that can be used to return the eigenvalues and eigenvector matrix in separate variables rather than the glued-together return value of the function. The fifth argument specifies whether the eigenvalues should be a row put on top of the eigenvector matrix (`False`) or a column on the left of it (`True`). Rectangle 5 displays the mean and sd of the orthogonal columns of Rectangle 6. The mean is fixed at the original mean vector. The table of Rectangle 6 holds the principle components constructed for the original data, using the affine transformation

$$\xi = \left(X - \bar{X}\right) Q + \bar{X}$$

where $\bar{X}$ is the mean vector in Rectangle 5 and Q is the eigenvector matrix in Rectangle 4. The same computation is repeated to produce Rectangle 9 from the Correlation matrix in Rectangle 7.

We next analyze m, the cut-off number of PCs to use as the "compressed" dataset. We demonstrate such analysis for Rectangle 9 which is constructed from the studentized data. Figure 6 (upper) plots the eigenvalues in Rectangle 8 and suggests retaining three PCs as they visually account for a high proportion of the total variance. The top region of the "PCA"



n	p	a: Tests based on	total var	b: m	var %	g(λ)	sd g(λ)	alpha	ubound for g(λ)	c: Testing	χ²	df	Crit.Val.	p-value	d: CI for	i	λi	confid.	lbound	rbound
54	8	Correl	8.00	3	2.8%	0.05	0.18	0.05	0.07	λ(m+1)=λ(m+2)=……=λp	105.56	14	23.68	4.04E-16	Eigval	3	0.23	95.0%	0.16	0.37

Figure 6: Determining the cut-off number: Eigenvalue plot and Hypothesis Testing.

(**PCA**) sheet implements a number of hypotheses tests (Figure 6 (lower)) to quantitatively determine m and confirm the visual check. After entering $m = 3$ in the first orange cell of Rectangle b, the other cells show that the 3rd PC accounts for 2.8% of total variance, the smallest $p - m$ PCs account for an expected $g(\lambda) = 5\%$ and maximum 7% at 95% confidence of total variance. Rectangle c tests whether the remaining $p - m$ eigenvalues are all equal while Rectangle d constructs a confidence interval for the third eigenvalue.

Discussion

Under the data normality assumption, PCA amounts to eigen-decomposition of the covariance matrix because the eigenvectors give the principal axes of the elliptic contour of the data:

$$\begin{aligned} \Sigma &= Q\Lambda Q^\intercal \\ \Lambda &= Q^\intercal \Sigma Q \end{aligned} \tag{2.3.1}$$

where Λ is the covariance matrix which is always real-symmetric and have a set of orthogonal eigenvectors (the columns of Q) with positive eigenvalues (the diagonal elements of Λ). The eigen-decomposition is computed by the method of power iteration extended with QR-decomposition to output all eigenvalues and eigenvectors at once. The QR-decomposition is computed by the VBA function `QR` with the signature `Function QR(A, Optional Q = Null, Optional R)`. The function returns the two matrices Q and R vertically stacked with a blank row separating in between. They can also be returned in separate variables using the second and third argument as callback variables. In the current software, the `QR` function is mainly consumed by the `CovEigenDecompQR` function. The latter function manages the power iteration which exploits that, when the exponent $n \to \infty$, the matrix

$$\Sigma^n = Q\Lambda^n Q^\intercal$$

has an extremely elongated ellipitical contour in the direction of the eigenvector of the maximum eigenvalue as its ratio to the rest of the eigenvalues is increasing to ∞. Choose any vector from the eigenspace and change basis to the standard basis, the resultant vector is almost surely to be along the direction of the maximum eigenvector. To simultaneously obtain other eigenvalue-vectors, we exploit the the orthogonality among the eigenvectors of a real symmetric matrix and use an iteration of QR-decompositions to force orthogonality of the Q matrix. The eigenvalues can be obtained by the second line of Eq(2.3.1). For rigorous explanations of the algorithm, see Golub and Van Loan (2012), Sauer (2011). The eigen-decomposition of the covariance matrix is an important computation at the center of a few methods implemented in the current software.

The longest principal axis is the linear direction on which the data projects to maximum variance. The second longest principal axis gives the next maximum-variance linear direction, and so on. The eigenvalues have the interpretation as the variance of the multivariate data along the corresponding eigenvector direction.

There is a subjective decision on whether studentization of data is needed. PCA on studentized data is equivalent to eigen-decomposition of the correlation matrix. The main issue is whether it is desirable to retain variance ratios among the observed variables. This certainly depends on the context. Ratios among some original variables may have established interpretations and hence would be preferred to retain. In other cases, for example, one is preparing the independent variables going into the right-hand side of a regression formula, one might want to studentize the data as the regression coefficients can recover such ratio. In middle cases, a properly estimated convex combination between the two matrices might be considered. Following are some further details regarding implementation.

1. PCA on correlation matrix need not add back the mean vector because of the data is assumed to have been studentized. This reflects that PCA on correlation matrix focuses on the angular difference between coordinates and ignores the radial differences.

2. PCA on covariance matrix adds back the mean vector. The PCA on multivariate normal sample is essentially doing a rotation of the sample space about the mean vector, not

the origin. To perform that rotation via a rotation matrix, we need to first remove the mean before applying the rotation matrix formed by the eigenvectors, and may or may not add the mean vector back—we chose to add it back.

3. PCA is essentially a data orthogonalization routine and does not model the mean vector. It can be used to orthogonalize the covariates for regression if prediction is the main goal and interpretation of the new covariates' meaning is not a concern. A more important purpose is dimension reduction. Those eigenvectors with a tiny eigenvalue indicate insufficient information in those trailing dimensions and hence could be removed to avoid over-fitting.

Note that the original definition of PCA is to be a method seeking the linear directions on which data projects with maximum variance. With this definition, it is not restricted to multivariate normal data but is applicable under any sampling assumption by proper techniques. Nonetheless, if we can assume multivariate normality, the computation becomes much easier.

2.4. Canonical Correlation Analysis

The overall idea of Canonical Correlation Analysis (Hotelling 1936) is to construct a scalar correlation measure r, called the canonical correlation, that describes linear association between two vectors of multivariate random variables $\mathbf{v} \in \mathbb{R}^p$ and $\mathbf{w} \in \mathbb{R}^q$. We demonstrate canonical correlation analysis following Anderson (2003) with the "Salespeople" dataset registered on the "Data" (Data) sheet under the same name. The "Salespeople" dataset collects 7 covariates per person that can be grouped into two sets. The first 3 variables measure sales performance scores; the rest 4 variables measure general intellectual tendencies of the person.

As before, the raw data should be copied first to the "Pivot" (Pivot) sheet, and then brought to the "CanCorr" (CanCorr) sheet using the green paste-from-pivot double-click button. We will investigate the correlation structure between the 3 performance variables and the 4 skill variables. To indicate grouping, we put a letter `v` ahead each column of a performance variable and a letter `w` ahead each column of a skill variable. The "CanCorr" (CanCorr) sheet should now appear as Figure 7. As in the "PCA" (PCA) sheet, each time an `v` or `w` is typed, the whole sheet is regenerated to show a new set of results. The analysis is also branched into one based on the original data and the other based on the studentized data.

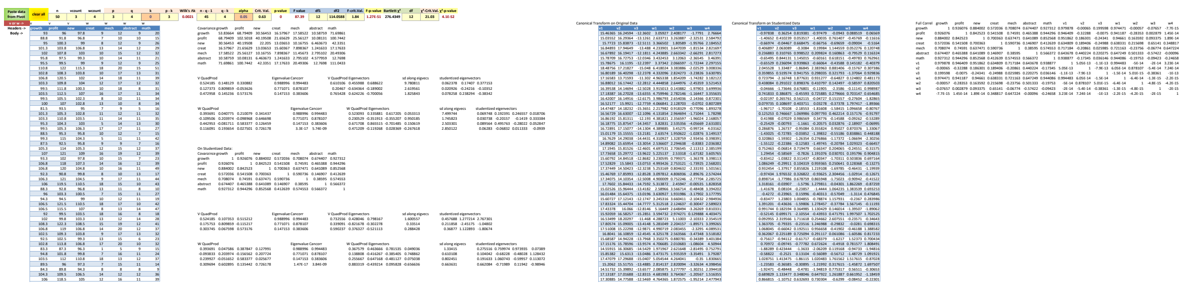


Figure 7: The "CanCorr" sheet.

The canonical correlation is constructed by finding in each space a unit directional vector such that the usual unsigned correlation (geometrically the cosine of the angle) between the two directional vectors are maximal. The derivation of the pair of optimal directional vectors shows the problem can be recast as the eigenvalue problems for two positive semi-definite matrices, which are computed in the current software by the VBA function `CovEigenDecompQR`. Figure 8 shows the formula in which the eigenvalue equations (2.4.1) are coded.

$$
\begin{aligned}
r^2\mathbf{v} &= \mathbf{S}_{vv}^{-1}\mathbf{S}_{vw}\mathbf{S}_{ww}^{-1}\mathbf{S}_{wv}\mathbf{v}\\
r^2\mathbf{w} &= \mathbf{S}_{ww}^{-1}\mathbf{S}_{wv}\mathbf{S}_{vv}^{-1}\mathbf{S}_{vw}\mathbf{w}
\end{aligned}
\tag{2.4.1}
$$

Covariance	growth	profit	new	creat	mech	abstract	math
growth	53.83664	68.79409	30.56453	16.57967	17.58522	10.58759	71.69861
profit	68.79409	Svv 102.5018	40.19508	21.65629	Svw 25.56127	10.08131	100.7442
new	30.56453	40.19508	22.205	13.03653	10.16755	6.463673	42.3351
creat	16.57967	21.65629	13.03653	15.60367	7.898367	1.241633	17.17633
mech	17.58522	Swv 25.56127	10.16755	7.898367	11.45673	2.795102	20.49306
abstract	10.58759	10.08131	6.463673	1.241633	Sww 2.795102	4.577959	12.7698
math	71.69861	100.7442	42.3351	17.17633	20.49306	12.7698	111.0433

V QuadProd

```
=MMULT(MMULT(MINVERSE($L$5:$N$7), $O$5:$R$7), MMULT(MINVERSE($O$8:$R$11), $L$8:$N$11))
```

V QuadProd			Eigenvalue	Cancorr	V QuadProd Eigenvectors			sd along eigv
								9.780811
0.127373	0.809859	-0.053626	0.771071	0.878107	0.20467	-0.634364	-0.189002	2.619561
0.472958	0.145236	0.573176	0.147153	0.383606	0.765428	0.624226	-0.700056	1.825843

W QuadProd				Eigenvalue	Cancorr	W QuadProd Eigenvectors			
0.393691	0.040775	0.210079	0.341437	0.988996	0.994483	0.523093	0.335881	0.617205	0.053313
-0.109506	0.203974	-0.098968	0.646698	0.771071	0.878107	0.230529	-0.351913	-0.355207	0.950185
0.442953	-0.081711	0.583377	0.126459	0.147153	0.383606	0.671708	0.865515	-0.701485	0.150619
0.116091	0.193654	0.027501	0.726178	3.64E-17	6.04E-09	0.471209	-0.119268	0.028369	-0.267618

Figure 8: Coding the canonical correlation equations (2.4.1).

It can be shown that the two matrices share the a same set of non-zero eigenvalues and the largest eigenvalue is the squared canonical correlation being sought. The eigenvectors corresponding to the largest eigenvalue for each matrix is a unit vector in each space. For the second-largest eigenvalue, it is the squared canonical correlation between the orthogonal complement spaces $\alpha^{\perp} \subset \mathbb{R}^{p-1}$ and $\beta^{\perp} \subset \mathbb{R}^{q-1}$ of the two directional vectors, and recursively so constructing the other smaller non-zero eigenvalues. Note that the "QuadProd" matrices are real symmetric, this means that the eigenvectors are perpendicular to each other. Together with the "maximal correlation" property, the eigenvectors can be used to re-coordinate the data columns, as done in range starting at cell `CanCorr!AG4` and `CanCorr!AP4`. Figure 9 shows first a few rows of re-coordinated data. Figure 10 shows the full correlation matrix including

Canonical Transform on Original Data

v1	v2	v3	w1	w2	w3	w4
15.46365	16.24594	-12.3602	3.05927	2.408177	-1.7791	2.76664
15.03552	16.29364	-13.1261	2.633711	3.263887	-2.32531	2.584792
15.7723	15.83873	-12.5111	3.366502	1.80589	-1.95766	2.184552
16.84893	17.94649	-13.488	4.233901	3.647039	-1.81534	2.821697
16.67892	16.19417	-12.1811	4.243885	2.663342	-2.66291	2.817273
15.78709	16.72753	-12.0346	3.432453	3.12063	-2.36545	3.46391
15.78675	16.1195	-12.2397	3.37342	2.066597	-1.72244	2.297155
18.48756	17.21827	-15.048	6.418024	3.383886	-2.02529	3.008336

Canonical Transform on Studentized Data

v1	v2	v3	w1	w2	w3	w4
-0.97838	0.36254	0.819381	-0.97479	-0.0943	0.088519	0.06569
-1.40652	0.410239	0.053517	-1.40035	0.761407	-0.45769	-0.11616
-0.66974	-0.04467	0.668475	-0.66756	-0.69659	-0.09004	-0.5164
0.406897	2.063089	-0.3084	0.19984	1.144559	0.052276	0.120748
0.236883	0.310765	0.998522	0.209824	0.160863	-0.79529	0.116324
-0.65495	0.844131	1.145015	-0.60161	0.618151	-0.49783	0.762961
-0.65529	0.236094	0.939863	-0.66064	-0.43588	0.145182	-0.40379
2.045528	1.33487	-1.86845	2.383963	0.881406	-0.15767	0.307386

Figure 9: Canonical Variates

the original dataset and the constructed canonical variates is displayed in range starting at

Full Correl	growth	profit	new	creat	mech	abstract	math	v1	v2	v3	w1	w2	w3	w4
growth	1	0.926076	0.884002	0.572036	0.708074	0.674407	0.927312	0.979878	-0.00065	0.199598	0.974471	-0.00057	-0.07657	-7.7E-15
profit	0.926076	1	0.842523	0.541508	0.74591	0.465388	0.944296	0.946409	-0.32288	-0.0075	0.941187	-0.28353	0.002879	1.45E-14
new	0.884002	0.842523	1	0.700363	0.637471	0.641089	0.852568	0.951862	0.186301	-0.24341	0.94661	0.163592	0.093375	1.89E-14
creat	0.572036	0.541508	0.700363	1	0.590736	0.146907	0.412639	0.634809	0.189406	-0.24988	0.638331	0.215698	0.65141	0.348817
mech	0.708074	0.74591	0.637471	0.590736	1	0.38595	0.574553	0.717184	-0.20861	0.025985	0.721163	-0.23756	-0.06774	0.647224
abstract	0.674407	0.465388	0.641089	0.146907	0.38595	1	0.566372	0.643678	0.440224	0.220275	0.647249	0.501333	-0.57422	-0.00096
math	0.927312	0.944296	0.852568	0.412639	0.574553	0.566372	1	0.938877	-0.17345	0.036146	0.944086	-0.19753	-0.09423	-0.24658
v1	0.979878	0.946409	0.951862	0.634809	0.717184	0.643678	0.938877	1	5.05E-13	-1.1E-13	0.994483	-5E-14	-2E-14	3.23E-14
v2	-0.00065	-0.32288	0.186301	0.189406	-0.20861	0.440224	-0.17345	5.05E-13	1	-7.9E-13	6.05E-14	0.878107	-5.4E-14	7.24E-14
v3	0.199598	-0.0075	-0.24341	-0.24988	0.025985	0.220275	0.036146	-1.1E-13	-7.9E-13	1	-1.5E-14	5.01E-15	-0.38361	-1E-13
w1	0.974471	0.941187	0.94661	0.638331	0.721163	0.647249	0.944086	0.994483	6.05E-14	-1.5E-14	1	-6.4E-14	-1.3E-15	-2.2E-15
w2	-0.00057	-0.28353	0.163592	0.215698	-0.23756	0.501333	-0.19753	-5E-14	0.878107	5.01E-15	-6.4E-14	1	-4.8E-15	-4.2E-15
w3	-0.07657	0.002879	0.093375	0.65141	-0.06774	-0.57422	-0.09423	-2E-14	-5.4E-14	-0.38361	-1.3E-15	-4.8E-15	1	-2E-15
w4	-7.7E-15	1.45E-14	1.89E-14	0.348817	0.647224	-0.00096	-0.24658	3.23E-14	7.24E-14	-1E-13	-2.2E-15	-4.2E-15	-2E-15	1

Figure 10: Full correlation matrix among the original and the canonical variates stylized.

cell `CanCorr!AY3`. This full correlation matrix has only one version as it is invariant to studentization of the input data.

The fact that each V-canonical variate only respond to one of the W-canonical variates means that if we run a regression of all V's on all W's, then we know it is merely a bunch of 1-to-1 simple linear regression performed together. Finally, the first two rows implements a hypothesis test for determining the cut-off number k of canonical variates to retain, should dimension reduction be a concern, such that the association between the two sets of variables is sufficiently preserved. It constructs a Wilk's Λ statistic to test whether the discarding variates all have zero canonical correlations with their unique counterparts in the opposite set. The sampling distribution of the Wilk's Λ can be approximated by either an F distribution or a χ^2 distribution.

n	vcount	wcount	p	q	k	p - k	Wilk's Λk	n - q - 1	q - k	alpha	Crit. Val.	p-value	F value	df1	df2	F-crit.Val.	F-p-value	Bartlett χ²	df	χ²-Crit.Val.	χ²-p-value
50	3	4	3	4	1	2	0.1952	45	3	0.05	0.76	8.25E-14	11.14	9	104.8013	1.97	5.13E-12	73.5084	6	12.59	7.78E-14

2.5. Linear Factor Analysis

When all observed variable are used and the residuals are still not spherical, one ponders over the existence of latent factors. The factor model is formulated as

$$y - \mu = \Lambda F + \varepsilon$$

where F is a multivariate normal vector that can be required to satisfy

$$\mathrm{var}\,(F) = I.$$

It plays the equivalent role of the observed x-variables in a usual regression model. The ε is the new residual that is hopefully more spherical than before. The coefficient matrix Λ is called the factor loadings; it plays the equivalent role of the β coefficients in a usual regression model. Both F and Λ will need to be estimated. A further simplifying assumption makes Λ constant so that

$$\mathrm{var}\,(\Lambda F) = \Lambda \Lambda^{\intercal} \tag{2.5.1}$$

hence

$$\mathrm{var}(y - \mu) = \Lambda \Lambda^{\intercal} + \mathrm{var}\,(\varepsilon)\,. \tag{2.5.2}$$

This suggests expanding the real-symmetric left-hand side by eigen-decomposition

$$\mathrm{var}(y - \mu) = \lambda_1 v_1 v_1^{\intercal} + \lambda_2 v_2 v_2^{\intercal} + \cdots + \lambda_p v_p v_p^{\intercal} \tag{2.5.3}$$

and estimating $\Lambda\Lambda^\intercal$ by first m terms of this summation

$$\hat{\Lambda} = V_m D_m^{\frac{1}{2}} \tag{2.5.4}$$

where V_m is an m-column matrix binding together the vectors $v_1, \ldots, v_m$ and $D_m^{\frac{1}{2}}$ is a square diagonal matrix holding the square-rooted eigenvalues $\sqrt{\lambda_1}, \ldots, \sqrt{\lambda_m}$. This is the method implemented in the current software. The eigen-decomposition is computed by the VBA function `CovEigenDecompQR`.

After the factor loadings matrix is estimated, we proceed to estimate F by, for example, least square. In the current software, we follow Anderson (2003) to implement the weighted least square method of Bartlett (1938), denoting by $\hat{\Psi}$ the estimate of $\mathrm{var}(\varepsilon)$,

$$\hat{F} = \left(\hat{\Lambda}^\intercal \hat{\Psi}^{-1} \hat{\Lambda}\right)^{-1} \hat{\Lambda}^\intercal \hat{\Psi}^{-1} z \tag{2.5.5}$$

and the conditional expectation method of Thomson (1951),

$$\hat{F} = \hat{\Lambda}^\intercal \left(\hat{\Lambda}\hat{\Lambda}^\intercal + \hat{\Psi}\right)^{-1} z = \left(I + \hat{\Lambda}^\intercal \hat{\Psi} \hat{\Lambda}\right)^{-1} \hat{\Lambda}^\intercal \hat{\Psi}^{-1} z. \tag{2.5.6}$$

In the following example, we analyze the 1988 Olympic men's decathlon results data, found under name "Olympic88" on the "Data" (**Data**) sheet, using factor analysis. The dataset doesn't contain any predicting covariates such as atheletes' physical measurements or results taken at an earlier time, making it suitable to demonstrate the latent factor approach.

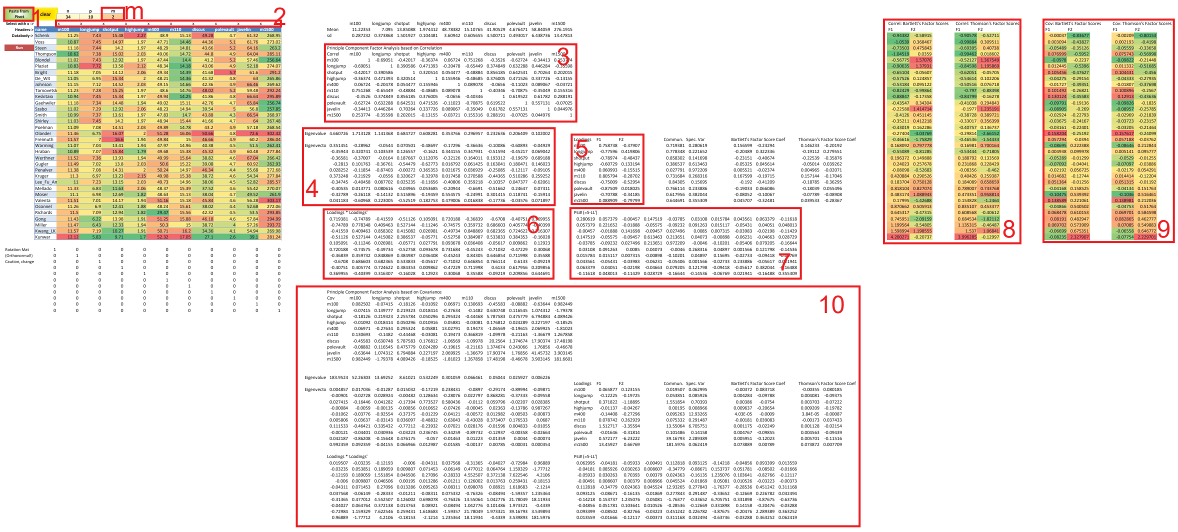


Figure 11: Output from the Factor Analysis sheet

As before, we copy the "Olympic88" raw data table on the "Data" (**Data**) sheet, paste it to the "Pivot" (**Pivot**) sheet, switch to the "Factor" (**Factor**) sheet and double-click on the top-left green button with text "Paste from Pivot". The actual computation for Factor analysis is event-triggered in the same fashion as seen before. There are two branches of computations done the original and studentized data. After entering 10 `x`'s ahead of every results columns, the "Factor" (**Factor**) sheet should now appear as Figure 11.

The computation for Factor analysis based on studentized data starts with Rectangle 3 where the correlation matrix is placed. Rectangle 4 computes the eigenvalues and eigenvectors for the correlation matrix. Rectangle 5 shows the m-column loading matrix estimate according to Eq(2.5.4). The number m is specified in the Rectangle m. Rectangle 6 computes the common covariance estimate (2.5.1). Rectangle 7 computes the specific covariance estimate $\hat{\Psi} = S - \hat{\Lambda}\hat{\Lambda}^{\intercal}$. The heat-mapped Rectangles 8 compute the latent factors by (2.5.5) and (2.5.6) for the studentized data. Rectangle 10 repeats the same computation for the original data, which produces the latent factors in Rectangle 9.

2.6. Multivariate Hypothesis Testing

Many hypotheses about the differences of *correlated* variables can be tested using Hotelling's T^2 statistic. The T^2 statistic is a multivariate generalization of Student's t statistic. It takes a quadratic form and its sampling distribution is linked to the F-distribution (Hotelling 1931):

$$n(\bar{\mathbf{x}} - \boldsymbol{\mu})^{\intercal}\mathbf{S}^{-1}(\bar{\mathbf{x}} - \boldsymbol{\mu}) \sim T^2(p, n-1) = \frac{p\,(n-1)}{n-p}F\,(p, n-p)$$

where $\mathbf{x} \in \mathbb{R}^{p\times n}$ is the data vector, $\bar{\mathbf{x}} \in \mathbb{R}^{p}$ is the sample mean vector, $\boldsymbol{\mu} \in \mathbb{R}^{p}$ is the true mean vector, n is the sample size, and $\mathbf{S}$ is the sample covariance matrix

$$\mathbf{S} = \frac{1}{n-1}\,(\mathbf{x} - \bar{\mathbf{x}}\mathbf{1}^{\intercal})\,(\mathbf{x} - \bar{\mathbf{x}}\mathbf{1}^{\intercal})^{\intercal}.$$

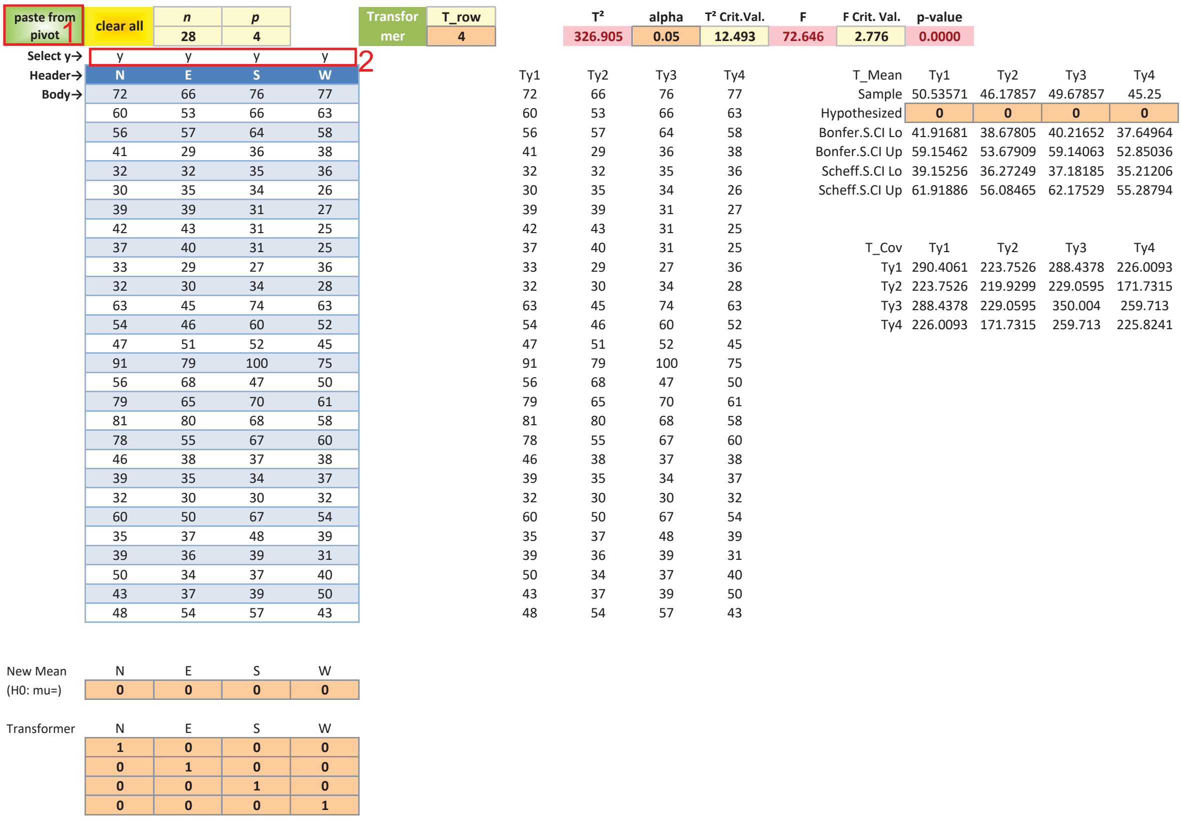
If the sample covariance $\mathbf{S}$ is replaced by the true covariance $\boldsymbol{\Sigma}$ then the resulting pivotal quantity $n(\bar{\mathbf{x}} - \boldsymbol{\mu})^{\intercal}\boldsymbol{\Sigma}^{-1}(\bar{\mathbf{x}} - \boldsymbol{\mu})$ is χ^2-distributed. Since any linear transformation of a normal random vector remains normal, the test statistic remains T^2. One then focuses on the design of a linear transform C that represents the hypothesis of interest:

$$H_0 : C\bar{\mathbf{x}} - \boldsymbol{\xi}_0 = \mathbf{0}.$$

Since the design of the linear transform is essentially a process of choosing the proper basis for a re-coordinatization of data, the whole process can be intuitively understood as finding the most direct "angle" to view the data such that the hypothesis is settled by judging whether the distance between a pair of points is too much for them to be considered the same point. We demonstrate the methodology using the classic example of Rao (1948) with our Excel tool.

Example: Bark deposit from 4 directions of the trunk of 28 Oak trees.

Rao (1948) exhibited a dataset containing measurements of the weights of cork borings taken from 4 directions on the trunk of 28 Oak trees. The hypothesis of interest was that the deposit is uniform in the North–South directions and also uniform but *less* in the East–West directions. The dataset is registered under name "Cork borings" on the "Data" (**Data**) sheet. As before, a copy is pasted to the "Pivot" (**Pivot**) sheet for any preprocessing and initial exploration. Then a further copy for analysis is pasted to the "Tsquare" (**Tsquare**) sheet by double-clicking the green paste button placed at the top-left cornor of each sheet. The initial setup should appear as Figure 12. The analysis is carried out in the following steps.



paste from pivot	clear all	n	p	Transformer	T_row
1		28	4		4

T^2	alpha	T^2 Crit.Val.	F	F Crit. Val.	p-value
326.905	0.05	12.493	72.646	2.776	0.0000

	N	E	S	W		Ty1	Ty2	Ty3	Ty4
Select y→	y	y	y	y	2				
Header→	N	E	S	W		Ty1	Ty2	Ty3	Ty4
Body→	72	66	76	77		72	66	76	77
	60	53	66	63		60	53	66	63
	56	57	64	58		56	57	64	58
	41	29	36	38		41	29	36	38
	32	32	35	36		32	32	35	36
	30	35	34	26		30	35	34	26
	39	39	31	27		39	39	31	27
	42	43	31	25		42	43	31	25
	37	40	31	25		37	40	31	25
	33	29	27	36		33	29	27	36
	32	30	34	28		32	30	34	28
	63	45	74	63		63	45	74	63
	54	46	60	52		54	46	60	52
	47	51	52	45		47	51	52	45
	91	79	100	75		91	79	100	75
	56	68	47	50		56	68	47	50
	79	65	70	61		79	65	70	61
	81	80	68	58		81	80	68	58
	78	55	67	60		78	55	67	60
	46	38	37	38		46	38	37	38
	39	35	34	37		39	35	34	37
	32	30	30	32		32	30	30	32
	60	50	67	54		60	50	67	54
	35	37	48	39		35	37	48	39
	39	36	39	31		39	36	39	31
	50	34	37	40		50	34	37	40
	43	37	39	50		43	37	39	50
	48	54	57	43		48	54	57	43

T_Mean	Ty1	Ty2	Ty3	Ty4
Sample	50.53571	46.17857	49.67857	45.25
Hypothesized	0	0	0	0
Bonfer.S.CI Lo	41.91681	38.67805	40.21652	37.64964
Bonfer.S.CI Up	59.15462	53.67909	59.14063	52.85036
Scheff.S.CI Lo	39.15256	36.27249	37.18185	35.21206
Scheff.S.CI Up	61.91886	56.08465	62.17529	55.28794

T_Cov	Ty1	Ty2	Ty3	Ty4
Ty1	290.4061	223.7526	288.4378	226.0093
Ty2	223.7526	219.9299	229.0595	171.7315
Ty3	288.4378	229.0595	350.004	259.713
Ty4	226.0093	171.7315	259.713	225.8241

New Mean (H0: mu=)	N	E	S	W
	0	0	0	0

Transformer	N	E	S	W
	1	0	0	0
	0	1	0	0
	0	0	1	0
	0	0	0	1

Figure 12: Tsquare sheet: initial setup.

1. Perform the following three steps exactly: Select cells `Tsquare!B3:E3` | press `y` on the keyboard | Windows user: press `ctrl` + `Enter` on the keyboard; Mac user: Press `command` + `Enter`. By doing these steps, you have entered 4 "y"s simultaneously.

2. We start with testing the univariate null hypothesis $H_0 : N = S$. The following step computes the p-value = 0.5740 in cell `Tsquare!N2` for the test and the null hypothesis is accepted.

   Enter -1 in cell `Tsquare!D39` such that the first row of the matrix of our linear transform becomes $(1, 0, -1, 0)$. Enter 1 in cell `Tsquare!G2` to indicate we use only the first row of the transformation matrix. The "Tsquare" (**Tsquare**) sheet should now appear as the upper-left view in Figure 13.

3. Next we test the another univariate hypothesis $H_0 : E = W$ by simply modifying the first row of the transformation matrix near `Tsquare!B39:E39` into $(0, 1, 0, -1)$. The "Tsquare" (**Tsquare**) sheet should now appear as the upper-right view in Figure 13. This results in a p-value=0.6310 and therefore the null hypothesis is accepted.

4. Next we test the two hypotheses jointly: $H_0 : N = S$ and $E = W$. The following step results in an increased p-value=0.8194 and therefore null hypothesis accepted.

   Change `Tsquare!G2` to 2 and modify the first 2 rows of the transformation matrix near `Tsquare!B39:E40` into

$$\begin{bmatrix} 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & -1 \end{bmatrix}$$

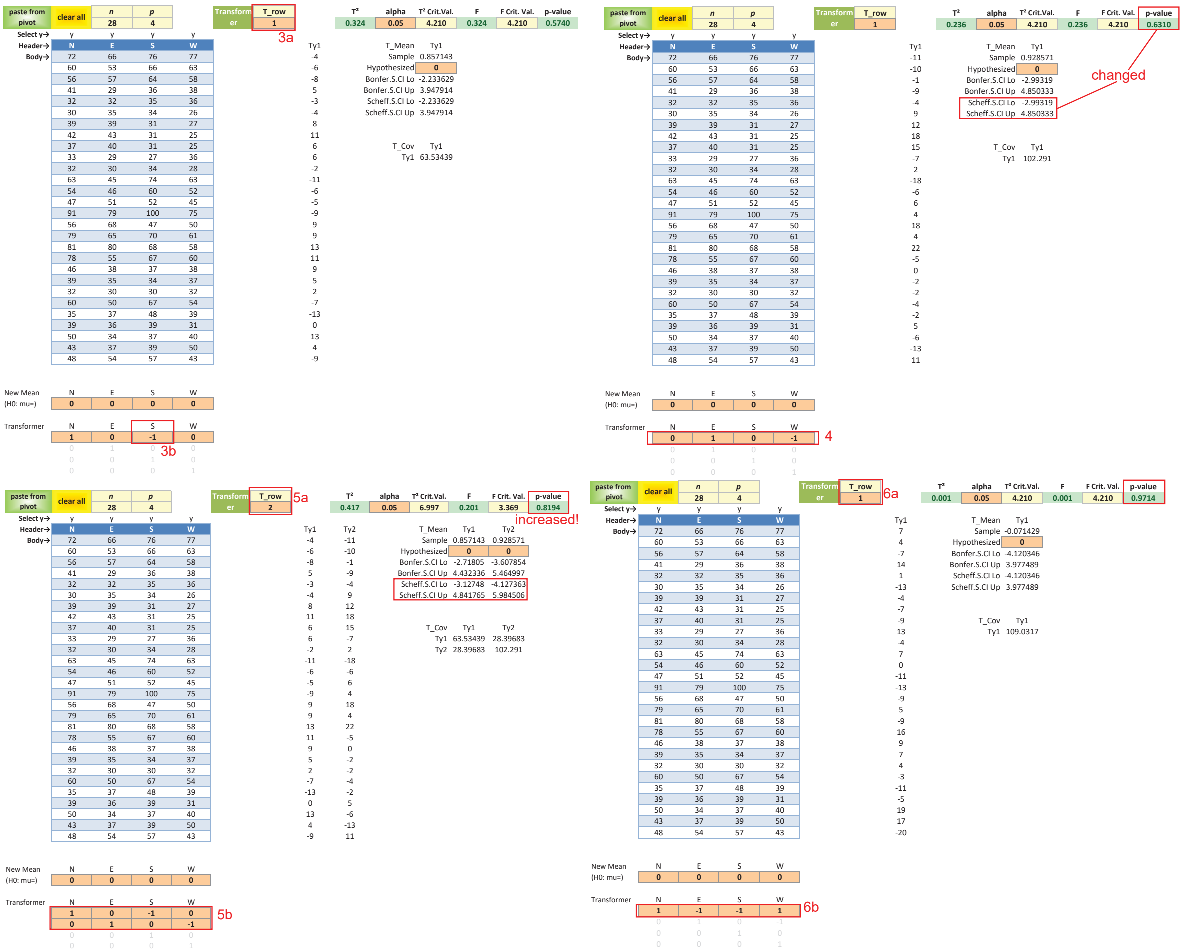


Figure 13: Upper-Left, $H_0 : N = S$; Upper-Right, $H_0 : E = W$; Lower-Left, $H_0 : N = S$ `and` $E = W$; Lower-Right, $H_0 : N - S = E - W$.

The "Tsquare" (Tsquare) sheet should now appear as the lower-left view in Figure 13.

5. To understand why the p-value has increased in the previous step, we plot the covariance matrix of the transformed data. Now perform the following steps: copy `L14:M15`, the covariance of the transformed data (`Ty1, Ty2`), switch to the "Cov2Correl" (Cov2Correl) sheet, double-click on the wide green cell `Cov2Correl!A3`, double-click on the orange cell `Cov2Correl!Q1`, follow the instruction on the popup to pick the covariance range at `Cov2Correl!B4:C5` and click done. Figure 14 should appear on the "Cov2Correl" (Cov2Correl) sheet now.

   Now we see that the covariance is elongated along the positive sloped direction, making it possible that the mean vector (`0.857142857, 0.928571429`), displayed at range `Tsquare!L5:L6`, has a smaller Mahalanobis distance from the center than both its projections on the two standard basis coordinates. This is a version of the Stein paradox (Stein (1956), Casella and Hwang (2012)). In the case here it can be understood as that the two hypotheses mutually corroborate. The data has indicated that it is more natural to have $N = S$ and $E = W$ happening together than separately; it is rather strange to observe uniformity in only one of the directions but not in the other. The

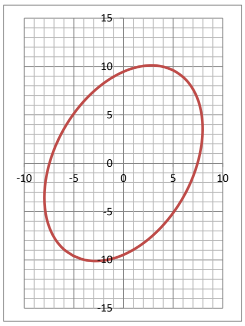


Figure 14: Cov2Correl sheet: Covariance plot of the transformed responses.

corroboration effect would not have been captured had we been testing only univariate procedures.

6. To test that the $E - W$ direction is less uniform than the $N - S$ direction, we use the linear combination $(N - S) - (E - W)$ and the null hypothesis that the two directions are equally uniform so that linear combination has zero mean under the null. On the "Tsquare" (Tsquare) sheet, we change back to use only the 1st row of the transform matrix by setting `Tsquare!G2` to 1 and enter $(1, -1, -1, 1)$ as the 1st row at `Tsquare!B39:E39`. The "Tsquare" (Tsquare) sheet should now appear as the lower-right view in Figure 13. The test result accepts the null with an even bigger p-value=0.9714 (`Tsqaure!N2`). Looking into the data, we do find a number of points where the difference between $N - S$ is greater than that between $E - W$.

   Note because $(1, -1, -1, 1) = (1, 0, -1, 0) - (0, 1, 0, -1)$, therefore we cannot test all three hypotheses in one transformation as that would create a singular covariance matrix for the transformed data and then no T^2 statistic could be constructed.

3. Storing Data and Generating Random Sample

The data import/export of the tool is delegated to Excel's own data i/o utilities. The user can add a blank sheet to import the dataset from various original sources. Next, all datasets need to be transformed into the *data frame* format, i.e., a matrix of data with column header texts. A row in the data frame is a multivariate sample vector jointly observed for all the variables and a column is a univariate sample observed repeatedly for a single variable. The number of rows in the data frame is the sample size.

After importing and transforming into the data frame format, the dataset should be registered on the sheet "Data" (Data) and archived there for future usage. The sheet "Data" (Data) can be navigated via a dropdown menu near cell `Data!A2`, which lists all registered datasets.

A working copy of a dataset should be put on the "Pivot" (Pivot) sheet.

Following is an example of generating multivariate normal random sample using the "Rand" (Rand) sheet and then registering and storing it on the "Data" (Data) sheet.

1. Activate sheet "Rand" (Rand) .

2. Put 3000 to `Rand!C2` to specify the sample size.

3. Enter the mean vector as a column vector right below cell `Rand!D1`: $[0, 0, 0]^\intercal$

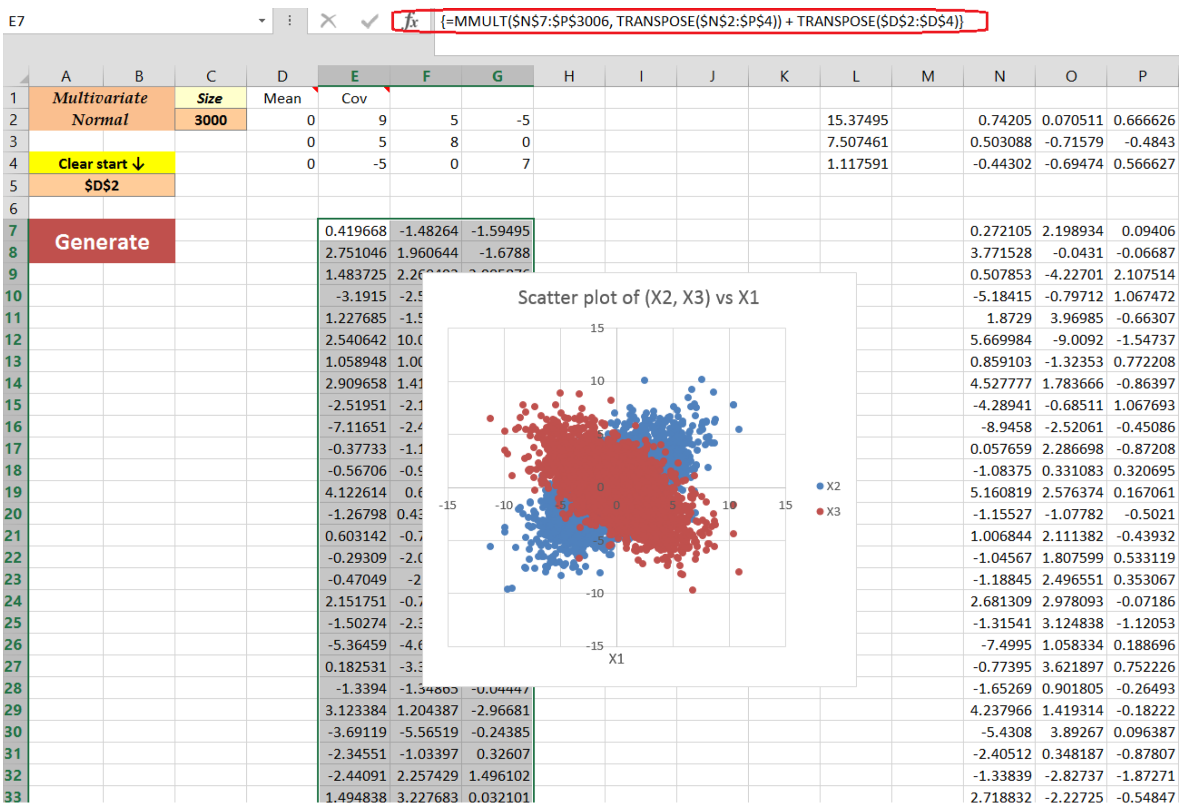
4. Enter the covariance matrix as a symmetric positive-definite matrix right below cell `Rand!E1` and extend to the right:

$$\begin{bmatrix} 9 & 5 & -5 \\ 5 & 8 & 0 \\ -5 & 0 & 7 \end{bmatrix}$$

Sheet "Rand" (Rand) should now appear as

	A	B	C	D	E	F	G	H
1	*Multivariate*		*Size*	Mean	Cov			
2	*Normal*		3000	0	9	5	-5	
3				0	5	8	0	
4	Clear start ↓			0	-5	0	7	
5	$D$2							

Now if you double-click on the cell `Rand!A7` (Generate) some equation will be entered to the sheet by a VBA macro (`shtRand.generate`) triggered on the double-click event you just performed to the cell `Rand!A7`. The 3000×3 range `Rand!E7:G3006` is also automatically selected so that you can directly press the scatter plot button to have a visual check as I am doing. Sheet "Rand" (Rand) should now appear as



The quick visual check confirms that: (i) the mean location is near the origin, (ii) both data exhibits the elliptical contour consistent with the positive definite quadratic form embedded in the MVN density, and (iii) the positive correlation between X1 and X2 gives the $+45°$ rotation of the blue sample while the negative correlation between X1 and X2 gives the $-45°$ rotation of the red sample.

A very important feature here is that the output is a function of the input and is connected to input via a formula chain. As a result, if the user changes the covariance input, then immediately the plot will update. This reveals the many upsides of using Excel to do mathematical modeling on small-to-medium sized data: it is a functional environment; it has a robust event system; it has a lot of productive utilities to operate the data; and it lets you monitor all variables at the same time. These are all conducive to (self-)teaching core multivariate statistics.

Next we will register the generated random sample to the Data sheet. Note that the following step of storing and registering dataset on the tool is the same for any data as long as it is presented in the data frame format. One can leverage Excel's own utilities to prepare the raw data into the data frame format.

1. Add names to the 3 columns by typing into cells `Rand!E6:E8` "X1", "X2", "X3". During the process the sample may be regenerated.

2. Press ctrl+a on Windows or command+a on Mac to select the entire 3001×3 data range `Rand!E6:G3006` (now with a header row)

3. Press ctrl+c to copy to clipboard.

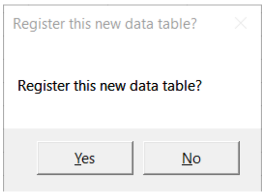
4. Launch a simple text editor and paste the data there to make it plain text.

5. Copy everything in the simple text editor to clipboard

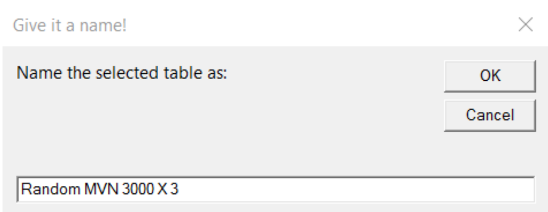
6. Activate sheet "Data" (**Data**)

7. Double click on `Data!I1` (**Paste from Clipboard**) to initiate pasting and registration of a new dataset

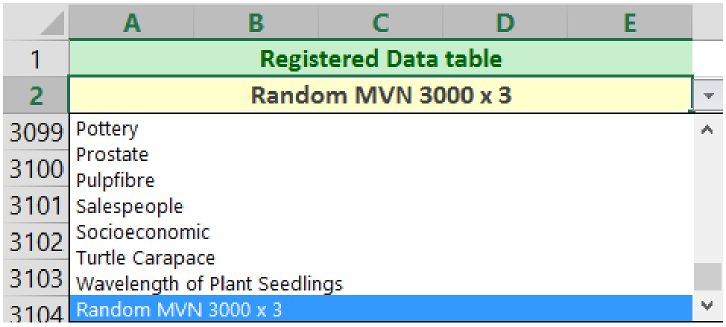
8. Click `Yes` to confirm registration of this dataset to sheet "Data" (**Data**)



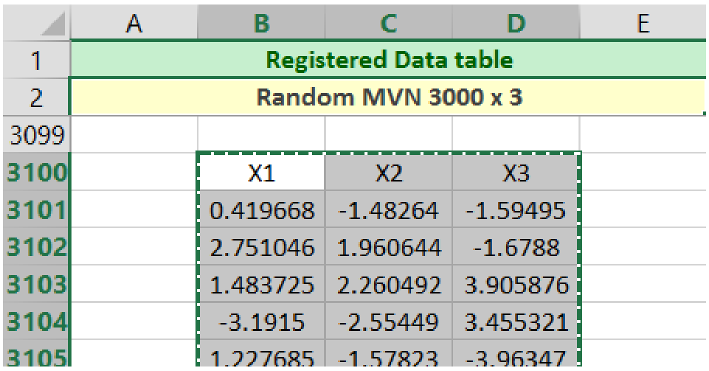
9. Enter "Random MVN 3000 x 3" to name the dataset being registered



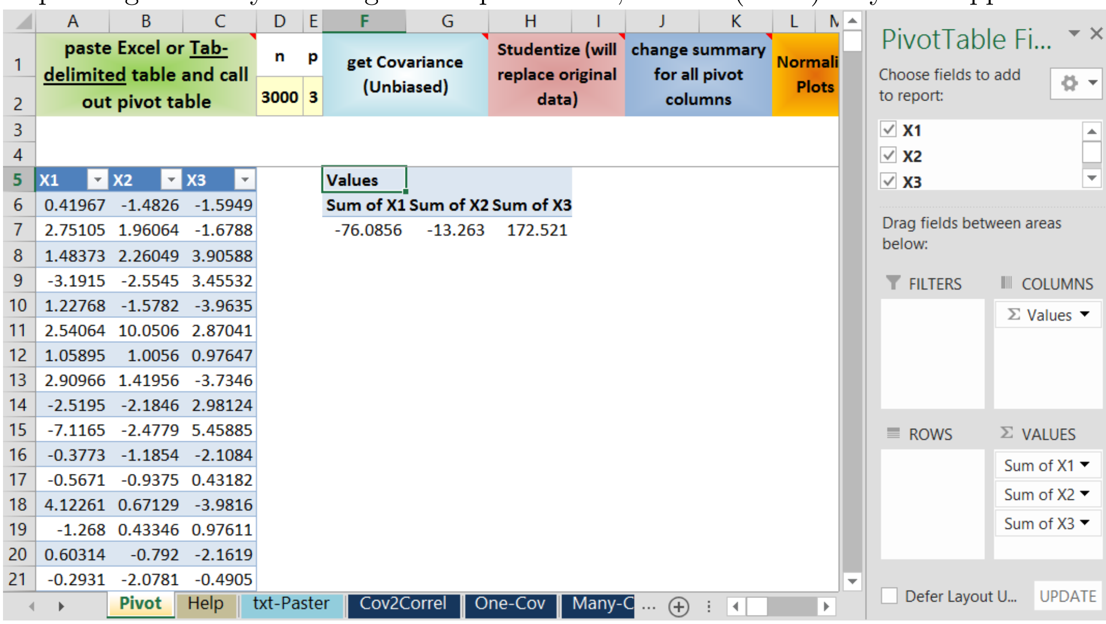
10. Roll out the dropdown menu at cell `Data!A2` and look for the newly registered dataset and select it. After select, the screen will auto-navigate to the dataset and select it.



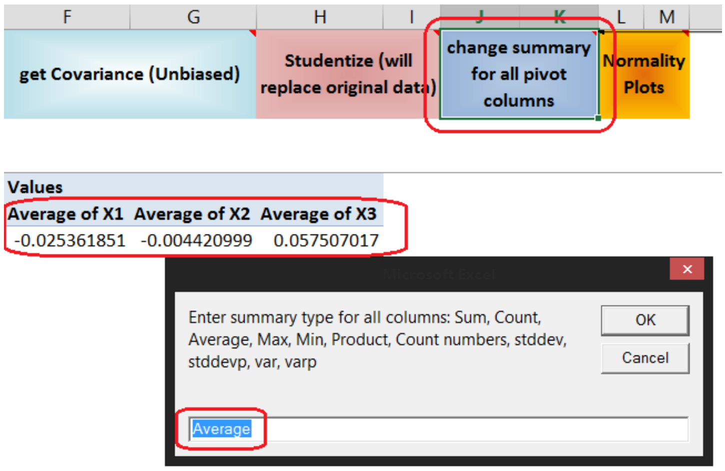
11. You can now press the keyboard shortcut to copy the selection to the system clipboard.



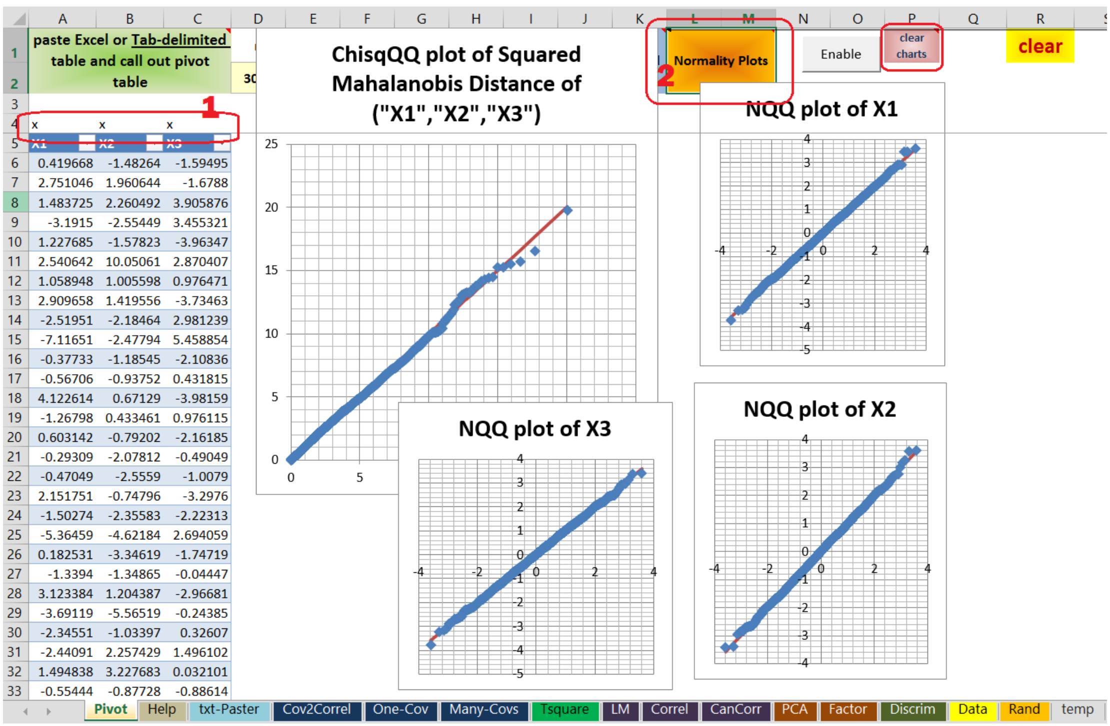
12. Once the data is on the clipboard, switch to the "Pivot" (**Pivot**) sheet and immediately double-click on the top-left green button. After double-click, the dataset will be pasted to the "Pivot" (**Pivot**) sheet as an Excel table and hence it enjoys all Excel table associated utilities such as filtering with complex conditions (SQL select in disguise), multi-column sort, pivot table (aggregation, a bit like R apply), and graphics. Depending on how you configure the pivot table, "Pivot" (**Pivot**) may now appear as



13. The cell `Pivot!J1` can help you change the aggregate function to one of `Sum`, `Count`, `Average`, `Max`, `Min`, `Product`, `Count numbers`, `stddev`, `stddevp`, `var`, and `varp`. This is a quick way to get the mean vector and the sd vector.



14. The orange button at cell `Pivot!L1` makes normal and χ^2 QQ plots to help visually check marginal and joint normality. To do this, put an "x" in cells `Pivot!A4:C4` above the column headers and then double click on the orange button. The "Pivot" (**Pivot**) sheet may now appear as



The normal quantile-quantile plots are made by the VBA macro `NormalQQplot`. The chi-squared quantile-quantile plot for inspecting violation of joint normality is made by the macro `MahalanobisChisqQQplot`. The Mahalanobis distance is defined as

$$D = \sqrt{(\mathbf{x} - \bar{\mathbf{x}})^{\intercal} \mathbf{S}^{-1} (\mathbf{x} - \bar{\mathbf{x}})}$$

Its square has an asymptotic $\chi(p)$ distribution where p is the number of variables ($p = 3$ here).

4. Collaboration

We have ported the entire project to github. Readers with Excel development experience who find the project meaningful and would like to contribute can directly fork the project on github, add a new analysis sheet, write new functions or libraries, and all kinds of features, and not forgetting to request for pulling by the master branch.

As can be seen from the examples in the previous section, the current Excel tool has manifested a sheet-oriented, event-triggered design. As it has become, currently, there are four types of sheets:

1. Data storage sheet ("Data" (**Data**)), to become the raw data interface of the software.
2. Random sample generation sheet ("Rand" (**Rand**)), to become the simulation workspace of the software.
3. Data preprocessing sheet ("Pivot" (**Pivot**)), to become the preprocessing workspace of the software.
4. Analysis sheet (e.g., "LM" (LM) , etc), to become the main modeling and analysis workspace of the software.

The "Pivot" (**Pivot**) sheet contains the data in analysis-ready state. The analysis sheets all implement a paste-from-pivot button at the top-left corner to create its own copy of the analysis-ready data and then builds formula chains to arrive at results. All sheets can use built-in Excel functionalities as well as custom addin functions written in VBA, XLL(COM), or .NET(VSTO). The transparency of the computation together with Excel's own tools around formula building, tracing, and checking allows complex models to be studied, understood, and used without tears.

References

Anderson TW (2003). *An Introduction to Multivariate Statistical Analysis*. Wiley Series in Probability and Statistics, 3rd edition. Wiley.

Bartlett MS (1938). "Further aspects of the theory of multiple regression." *Proceedings of the Cambridge Philosophical Society*, **34**, 33–40.

Casella G, Hwang JTG (2012). "Shrinkage Confidence Procedures." *Statist. Sci.*, **27**(1), 51–60. doi:10.1214/10-STS319. URL http://dx.doi.org/10.1214/10-STS319.

Fisher R (1915). "Frequency distribution of the values of the correlation coefficient in samples of an indefinitely large population." *Biometrika*, **10**, 507–521.

Golub G, Van Loan C (2012). *Matrix Computations*. Johns Hopkins Studies in the Mathematical Sciences, 4th edition. Johns Hopkins University Press.

Green J, Bullen S, Bovey R, Alexander M (2007). *Excel 2007 VBA Programmer's Reference*. Wiley.

Halvorsen KB (2015). *ElemStatLearn: Data Sets, Functions and Examples from the Book: "The Elements of Statistical Learning, Data Mining, Inference, and Prediction" by Trevor Hastie, Robert Tibshirani and Jerome Friedman*. R package version 2015.6.26, URL http://CRAN.R-project.org/package=ElemStatLearn.

Hotelling H (1931). "The Generalization of Student's Ratio." *The Annals of Mathematical Statistics*, **2**(3), 360–378. doi:10.1214/aoms/1177732979. URL http://dx.doi.org/10.1214/aoms/1177732979.

Hotelling H (1936). "Relations between two sets of variates." *Biometrika*, **28**(3/4), 321–377. 1936.

Johnson RA, Wichern DW (1992). *Applied Multivariate Statistical Analysis*. 4th edition. Prentice hall Englewood Cliffs, NJ.

MSDN (2013). "Excel VBA reference." URL https://msdn.microsoft.com/en-us/library/ee861528.aspx.

Rao CR (1948). "Tests of significance in multivariate analysis." *Biometrika*, **35**(1/2), 58–79. 1948.

Sauer T (2011). *Numerical Analysis*. Featured Titles for Numerical Analysis, 2nd edition. Pearson.

Stamey TA, Kabalin JN, McNeal JE, Johnstone IM, Freiha F, Redwine EA, Yang N (1989). "Prostate specific antigen in the diagnosis and treatment of adenocarcinoma of the prostate. II. Radical prostatectomy treated patients." *The Journal of Urology*, **141**(5), 1076–1083.

Stein C (1956). "Inadmissibility of the Usual Estimator for the Mean of a Multivariate Normal Distribution." In *Proceedings of the Third Berkeley Symposium on Mathematical Statistics and Probability, Volume 1: Contributions to the Theory of Statistics*, pp. 197–206. University of California Press, Berkeley, Calif. URL http://projecteuclid.org/euclid.bsmsp/1200501656.

Thomson GH (1951). *The Factorial Analysis of Human Ability*. 5th edition. University of London, London.

Affiliation:

Fanghu Dong
Department of Statistics and Actuarial Science
Faculty of Science
University of Hong Kong
Hong Kong

Guosheng Yin
Department of Statistics and Actuarial Science
Faculty of Science
University of Hong Kong
Hong Kong