2014 ■ vol. 10 ■ no. 2

DOI: 10.20982/tqmp.10.2.p200

Partial Least Squares tutorial for analyzing neuroimaging data

Patricia Van Roon [a, b], Jila Zakizadeh [a, b], Sylvain Chartier [b]

[a] School of Psychology, Carleton University

[b] School of Psychology, University of Ottawa

Abstract ■ Partial least squares (PLS) has become a respected and meaningful soft modeling analysis technique that can be applied to very large datasets where the number of factors or variables is greater than the number of observations. Current biometric studies (e.g., eye movements, EKG, body movements, EEG) are often of this nature. PLS eliminates the multiple linear regression issues of over-fitting data by finding a few underlying or latent variables (factors) that account for most of the variation in the data. In real-world applications, where linear models do not always apply, PLS can model the non-linear relationship well. This tutorial introduces two PLS methods, PLS Correlation (PLSC) and PLS Regression (PLSR) and their applications in data analysis which are illustrated with neuroimaging examples. Both methods provide straightforward and comprehensible techniques for determining and modeling relationships between two multivariate data blocks by finding latent variables that best describe the relationships. In the examples, the PLSC will analyze the relationship between neuroimaging data such as Event-Related Potential (ERP) amplitude averages from different locations on the scalp with their corresponding behavioural data. Using the same data, the PLSR will be used to model the relationship between neuroimaging and behavioural data. This model will be able to predict future behaviour solely from available neuroimaging data. To find latent variables, Singular Value Decomposition (SVD) for PLSC and Non-linear Iterative PArtial Least Squares (NIPALS) for PLSR are implemented in this tutorial. SVD decomposes the large data block into three manageable matrices containing a diagonal set of singular values, as well as left and right singular vectors. For PLSR, NIPALS algorithms are used because they provide a more precise estimation of the latent variables. Mathematica notebooks are provided for each PLS method with clearly labeled sections and subsections. The notebook examples show the entire process and the results are reported in the Section 3 Examples.

Keywords ■ partial least squares, PLS, regression, correlation, Mathematica, NIPALS

patriciavanroon@gmail.ca

Introduction

Partial Least Squares (PLS) is a powerful multivariate statistical tool that estimates the predictive or causal relationship between variables. It was introduced by Herman Ole Andreas Wold in 1975 who was critical of structural equation models because these methods tended to ignore the presumption that "causality proceeds through time" (Wold, 1964) whereas recursive models addressed this deficit. A recursive model uses any variable in a sequence to relate to the previous variable in the model. The most recognized number sequence, the Fibonacci sequence (Beck & Geoghegan, 2010; Bóna, 2011; Sigler, 2002), is an example of a recursive model where variable F_n in the sequence relates to the previous variables, F_{n-1}, F_{n-2} as below:

$$F_n = F_{n-1} + F_{n-2} \quad \text{for all } n \geq 2 \text{ and } F_1 = 1, F_0 = 0$$

Wold found, for recursive causal chain models (i.e. timing order where variables affect each other), the PLS method was more efficient and exceeded other techniques (e.g., Principal Component Analysis or PCA, Multiple Linear Regression or MLR) in intrinsic properties such as correlation and data size. PCA examines the variances represented in a single set of data and describes these variances in terms of a set of factors (Brown, 2009), and MLR examines the relationship between a set of independent variables and a response variable. If its number of independent variables gets too large (e.g., greater than number of observations), its performance declines substantially. Intrinsic properties of PLS include the ability to deal with large, noisy, collinear datasets whereas MLR cannot accurately apply multi-collinear variables to predict the response variable. In addition, PLS has no issue with missing data.

Essentially, PLS consists of two components – (i) the structural, which shows the relationship between latent variables, and (ii) the measurement, which shows the relationship between latent variables and their indicators (Haenlein & Kaplan, 2004). The major

advantage of PLS is that it is much less restrictive in terms of assumptions compared to other multivariate statistical techniques such as MLR, in that, there exists no need to check normality (data can have any distribution), linearity, and independence of observations. Researchers need to be aware of the assumption surrounding the latent variables. Specifically, that each observed variable has a specific location on the latent structure and these observed variables are discrete (Henning, 1989). Furthermore, the researcher should realize that these methods are not reliable if the dataset is very small, in general, that is, less than 30 cases.

McIntosh, Bookstein, Haxby, & Grady (1996) first introduced PLS to neuroimaging data analysis. Event-related potentials (ERPs), Positron Emission Tomography (PET) or functional Magnetic Resonance Imaging (fMRI) are examples of experiments that generate large neuroimaging datasets. Due to the expensive nature of these experiments, often the number of cases, or observations, is less than 50. Finding relationships between these large blocks of data, with many manifesting factors and few observations, can be a challenging task. Although there are many factors, there may be a few latent (unobservable or hidden) factors that account for most of the pattern co-variation in the data blocks or most of variation in the response. PLS tries to find or extract those latent variables using techniques such as decomposition of the covariance matrix in least squares sense or NIPALS.

With respect to ERPs, this is accomplished by extracting the latent variables that better relate brain activity, specifically electroencephalographic amplitudes at specific scalp locations, to behaviour (e.g., response times, and accuracy) or experimental design (e.g., contrast tasks, such as similarities and differences).

The objectives of these guidelines are to assist in analysis and interpretation of event-related potentials (ERP) and behavioural data using partial least squares (PLS) methods, specifically correlation and regression. These methods are implemented using a high-powered statistical system known as Mathematica. The provided codes can be adapted to other languages (e.g. Matlab, R, etc.). Each block in a dataset may contain multiple variables; however, for simplification purposes, the examples in this tutorial use behavioural and neuroimaging data blocks which are limited to a few variables. The behavioural data block has two variables, reaction time and number of words recalled, and the neuroimaging data block has multiple variables, the brain electrical activities at twelve channel locations.

Materials and Methods

Svante referred to the partial least squares technique as "*projection to latent structures*" (Abdi, 2010) because each observed variable is projected onto a latent variable. In order to better understand PLS analyses, one requires tools such as eigenvectors, eigenvalues, projection, singular value decomposition, and linear algebra concepts which are described in the following sections.

Materials – Notation, Definitions and Theorems

Matrices are denoted by bold capital letters, vectors by bold lower-case letters, transpose of matrix $\mathbf{X}$ by $\mathbf{X}^T$, ith entry in the vector $\mathbf{v}$ by v_i, element (i,j) of matrix $\mathbf{X}$ by x_{ij} and dimension of matrix $\mathbf{X}$ by $_{m \times m}$ where n is the number of rows and m is the number of columns. The norm of vector $\mathbf{v}$ is denoted by $\|\mathbf{v}\|$. Finally, the matrices of the $\mathbf{X}$ latent variables and the $\mathbf{Y}$ latent variables are denoted by $\mathbf{Lx}$ and $\mathbf{Ly}$ respectively.

Projection

Let S be a Hilbert space (i.e. a vector space possessing a structure of dot product – a scalar or inner product) and M is a subspace of dimension m (here m=2). The projection of a vector $\mathbf{v} \in$ S on M is a vector $\hat{\mathbf{v}} \in$ M such that,

$$\|\mathbf{v} - \hat{\mathbf{v}}\| < \|\mathbf{v} - \tilde{\mathbf{v}}\| \quad \textbf{for all } \tilde{\mathbf{v}} \in \mathbf{M}$$

as shown in Figure 1.

Eigenvectors and eigenvalues

In general, an eigenvector is defined as a non-zero column vector that satisfies the equation:

$$\mathbf{Ax} = \lambda \mathbf{x}$$

where $\mathbf{A}$ is a $m \times n$ square matrix, $\mathbf{x}$ is a non-zero vector, and λ represents the eigenvalues of $\mathbf{A}$.

The above equation can be written as follows:

$$\mathbf{Ax} - \lambda \mathbf{x} = \mathbf{0}$$

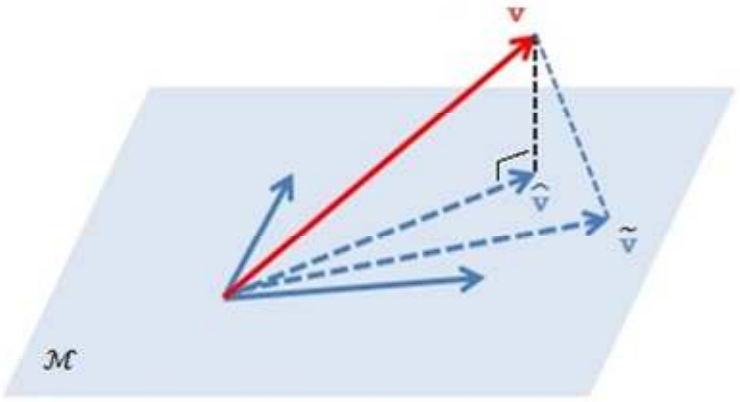


Figure 1 ■ Projection a vector v onto subspace M (www.cs.cmu.edu/~htong/pdf/KDD08-tong.ppt; adapted from Tong, Papadimitriou, Yu, & Faloutsos, 2008).

$$(\mathbf{A} - \lambda\mathbf{I})\,\mathbf{x} = \mathbf{0}$$

where **I** is an identity matrix of size *n*.

In order to have a non-zero **x**, the matrix (**A** - λ**I**) must be singular (i.e. its determinant is zero).

$$\det(\mathbf{A}-\lambda\mathbf{I}) = 0$$

where det(**A**-λ**I**) is called the characteristic polynomial of matrix **A**, the roots of this polynomial are eigenvalues of **A**.

Eigenvalues are important when the matrix is a transformation from one vector space onto itself.

Note that for non-square matrices, it matters on which side the **x** resides. If it is on the left it refers to a left eigenvector (i.e. a column vector). If it is on the right, it refers to the right eigenvector (i.e. a row vector).

Singular values and vectors

In general, a singular value and pair of singular vectors of a square or rectangular matrix **A** are non-negative scalar δ and two non-zero vectors **u** and **v** that satisfy the equations:

$$\mathbf{A}\mathbf{v} = \delta\mathbf{u}$$

$$\mathbf{A}^{\mathbf{T}}\mathbf{u} = \delta\mathbf{v}$$

Singular values are important in situations when the matrix is a transformation from one vector space to a different vector space, possibly with a different dimension. Under- or over-determined systems are situations where singular values are important.

Singular value decomposition

If matrix $\mathbf{A} \in \mathbb{R}^{m\times n}$ has rank $k \leq \min(m, n) < \infty$, there exists orthogonal matrices,

$$\mathbf{U} = [\mathbf{u}_1, \ldots, \mathbf{u}_m] \in \mathbb{R}^{m\times m}$$

and

$$\mathbf{V} = [\mathbf{v}_1, \ldots, \mathbf{v}_n] \in \mathbb{R}^{n\times n}$$

whose columns are the normalized singular vectors that satisfy $\mathbf{U}\mathbf{U}^{\mathbf{T}} = \mathbf{I}$ and $\mathbf{V}\mathbf{V}^{\mathbf{T}} = \mathbf{I}$, such that,

$$\mathbf{A} = \mathbf{U}\Delta\mathbf{V}^{\mathbf{T}}$$

where,

$$\Delta = \mathbf{diag}(\delta_1, \ldots, \delta_k, 0, \ldots) \in \mathbb{R}^{m\times n}$$

is a diagonal matrix for which $\delta_1, \ldots, \delta_k$ satisfy

$$\delta_1 \geq \delta_2 \geq \cdots \geq \delta_k > 0$$

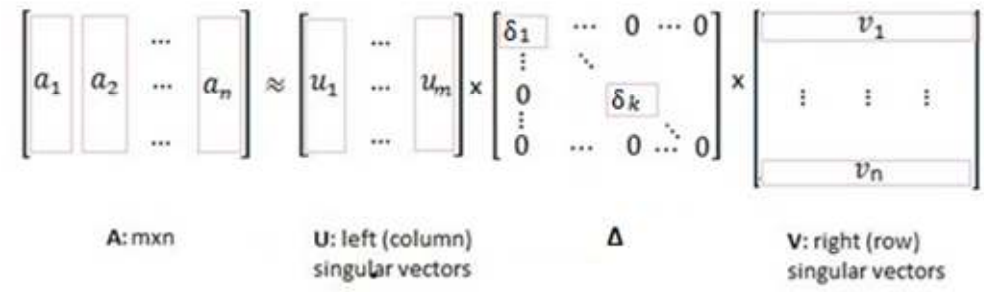


Figure 2 ■ Singular Value Decomposition – The left most matrix (A) is approximately equal to matrix U (i.e. the left singular vectors) x matrix Δ (i.e. the singular values) x matrix V (i.e. the right singular vectors).

This decomposition of **A** is called Singular Value Decomposition (SVD). δ_i s are arranged in descending order in a diagonal of **Δ** matrix and known as singular values of **A**. They are square roots of the eigenvalues of $\mathbf{AA}^T$ (if $m<n$) or of $\mathbf{A}^T\mathbf{A}$ (if $m>n$). If **A** is a symmetric matrix, its singular values are the absolute values of its eigenvalues. The columns of **U** are called left singular vectors and the columns of **V** are called right singular vectors of **A**.

SVD is one of the most elegant techniques in linear algebra (Ghazy, Hadhoud, Dessouky, El-Fishawy, & Abd El-Samie, 2008; Haykin, 1991) devised to interpret the least squares problem (See Figure 2).

Theory in linear algebra

An mxn linear system with $m>n$ is over-determined (i.e. system has more equations than unknowns),

$$\mathbf{Ax} = \mathbf{b}$$

where **A** is a matrix of mxn

This equation does not have any exact solution but has a unique least-squares solution, $\hat{\boldsymbol{x}}$, of the smallest norm. The $\hat{\boldsymbol{x}}$ solution can be found in terms of the pseudo-inverse matrix, $\mathbf{A}^+$ of **A**, which is obtained from the singular value decomposition of **A** (Legendre, 1806).[1]

If $\mathbf{A} = \mathbf{U\Delta V}^T$ with $\mathbf{\Delta}$=diag(δ_1, ..., δ_r, 0, ..., 0) where **Δ** is an mxn matrix and $\delta_i>0$, letting $\mathbf{\Delta}^+$= diag(1/ δ_1, ..., 1/ δ_r, 0, ..., 0) be an nxm matrix, the pseudo-inverse of $\mathbf{A}^+$ is defined as

$$\mathbf{A}^+ = \mathbf{V\Delta}^+\mathbf{U}^T$$

Therefore the minimum norm solution for the above system, $\mathbf{Ax} = \mathbf{b}$, will be

$$\hat{\boldsymbol{x}} = \mathbf{A}^+\mathbf{b} = \sum_{i=1}^{r} \frac{u_i^T b}{\delta_i} \boldsymbol{v}_i$$

Methods

EEG/ERP background information

Voltage differences recorded as amplitudes over time are known as brain wave recordings or electroencephalography (EEG). These recordings reflect the synchronous activity of several thousand neurons.

Event-related potentials (ERP) are EEG recordings of an individual's response to some external or internal stimulus (e.g. auditory, visual, somatosensory, any combination of these, etc.). The stimulus is sent to a recording computer as a trigger of a specific event and these triggers are averaged together for several trials in order to reduce the background EEG noise and obtain a high signal to noise ratio and, thereby, a cleaner signal. Ocular and other artefacts are corrected or removed prior to averaging (Picton, Lins, & Scherg, 1995).

Often only a subset of the electrodes and the number of points would be used for peak analysis. The entire average files, which contain amplitude and latency information for each electrode, can be brought into the software such as SPSS, Matlab, or Mathematica

[1] Legendre's (1806) least squares method is part of an appendix attached to his 1805 work on determining comet orbits. The last part of this appendix is a treatise on the determination of the degree of deviation of the Earth's elliptical orbit, and thereby, the establishment of the length of the mètre.

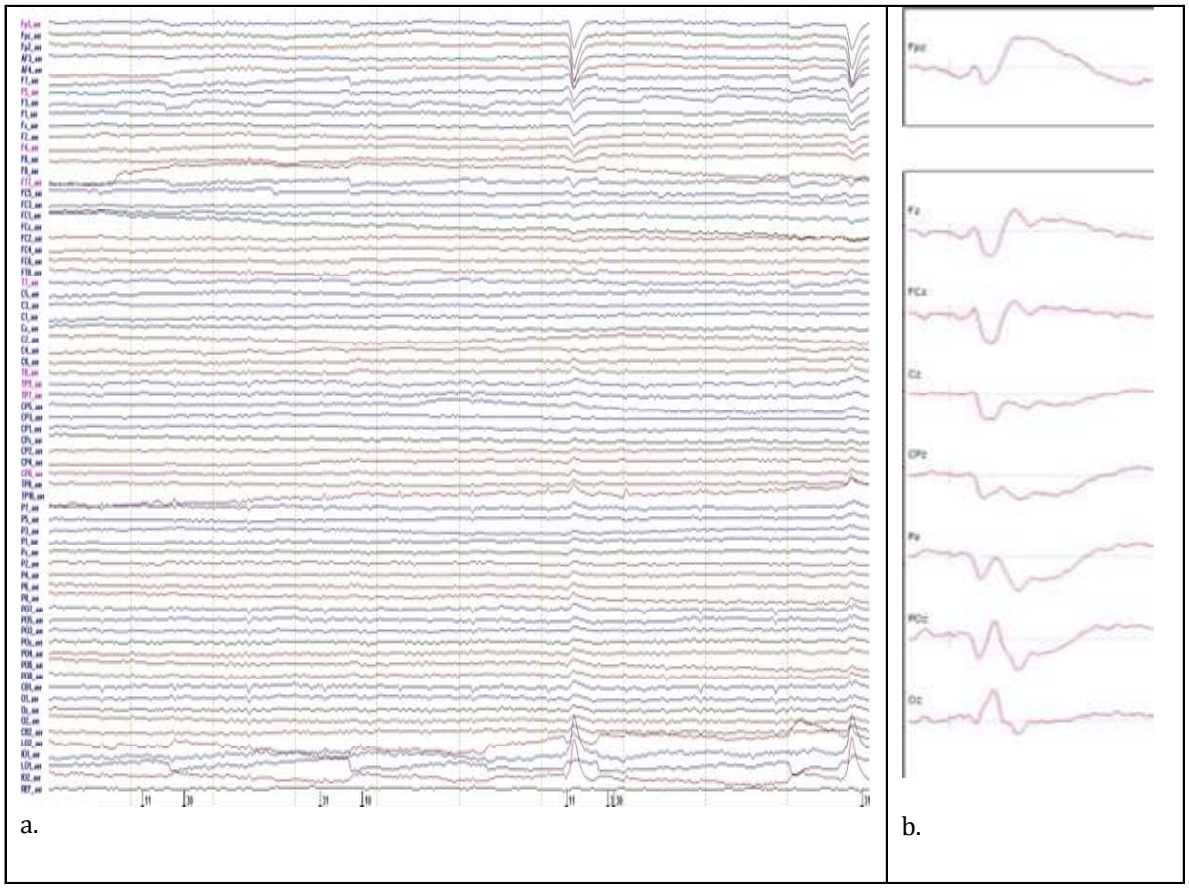


Figure 3 ■ a. Continuous EEG recording with triggers for averaging ERPs; b. Averaged ERP waveform of centre line electrodes.

for further statistical analysis. In this case, PLS methods in Mathematica will be used for all analyses here.

PLS methods – PLS correlation

Two major PLS techniques include correlation and regression (Abdi & Williams, 2013). *Path modeling* methods of PLS can directly follow these two methods but will not be covered in this article (for information on this technique see Tenenhaus, Esposito Vinzi, Chatelin, & Lauro, 2005; for a review refer to Esposito Vinzi, Trinchera, & Amato, 2010).

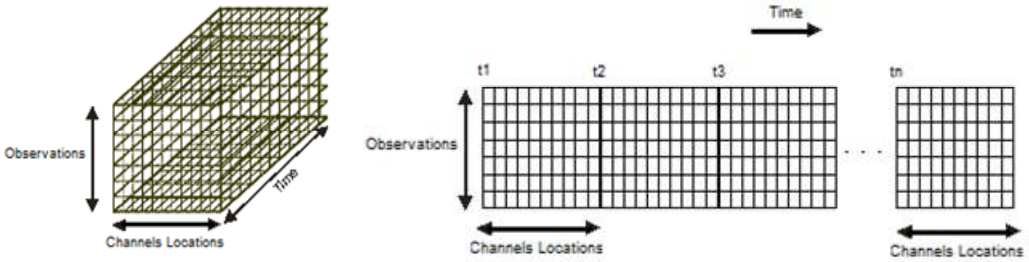
Why and when is PLSC applied? PLS Correlation is used to explore and describe any data structure. It can handle very large datasets and adapt to the experimental design. It allows the exploration of the correlation between two matrices.

How does it work? The primary goal is to analyze the communalities between the two matrices. Communality is a measurement of the percent variance of a given observed variable explained by all the latent variables together and reflects the reliability of the measured variable. Variables with high communalities are well explained while those with low communalities are not.

Let's matrix $\mathbf{X}_{n\times m}$ be the brain activity data for n number of participants and m data points of the neuroimaging data (averaged ERP amplitude for each channel as variables), and matrix $\mathbf{Y}_{n\times k}$ be the behavioural data for these n participants and k number of behaviour variables (such as reaction time and number of recalled words). The relationship between centred **X** and **Y** (i.e. zero mean) is determined by the covariance matrix. Since data has mixed units such as latency (ms) and amplitude (μV), the matrices need to be normalized column-wise as well:

$$\frac{x_{ij}-\bar{x}_j}{\|x_{ij}-\bar{x}_j\|}, \frac{y_{ij}-\bar{y}_j}{\|y_{ij}-\bar{y}_j\|}$$

where x_{ij}, y_{ij} are the (i,j) elements of matrix of **X** and **Y** respectively**.** The $\bar{x}_j$, $\bar{y}_j$ are the mean values of column



a. b.

Figure 4 ■ Schematic representation of data arrangement for PLS Correlation. a. Arrangement in terms of space, time, and conditions/participants/groups; b. Row-wise concatenation of the matrices (Vallesi, 2009).

j. Finally, $\|x_{ij} - \bar{x}_j\|$ and $\|y_{ij} - \bar{y}_j\|$ are the norm of centred x_{ij} and y_{ij}.

This process normalizes the covariance matrix. Since the correlation matrix is the normalized covariance, the correlation matrix is computed in order to find the patterns of relationship between **X** and **Y.** Assuming that both **X** and **Y** are centred and normalized, the correlation matrix of **X** and **Y** is computed:

$$\mathbf{R} = \mathbf{Y}^{\mathbf{T}}\mathbf{X}$$

By decomposing **R** using singular value decomposition method, the following equation is obtained:

$$\mathbf{R} = \mathbf{U}\Delta\mathbf{V}^{\mathbf{T}}$$

where **U** = matrix of normalized eigenvectors of $\mathbf{RR}^{\mathrm{T}}$, **V** = matrix of normalized eigenvectors of $\mathbf{R}^{\mathrm{T}}\mathbf{R}$, **Δ**= diagonal matrix with square root of eigenvalues of $\mathbf{RR}^{\mathrm{T}}$ or $\mathbf{R}^{\mathrm{T}}\mathbf{R}$.

U is the matrix of the left singular vectors and **V** is the matrix of the right singular vectors of **R**. Both **U** and **V** are orthonormal (orthogonal and normalized at the same time) that means $\mathbf{UU}^{\mathbf{T}} = \mathbf{I}$ and $\mathbf{VV}^{\mathbf{T}} = \mathbf{I}$. These are known as saliences (i.e. the most noticeable or important in relation to its neighbouring values).

As Svante Wold described, the latent variables are projections of the original matrices onto their respective saliences; they are a linear combination of the original variables and explain the largest portion, in general 80 to 95 percent, of the covariance between the two matrices. The number of saliences is equal to the rank of **R** (Krishnan, Williams, McIntosh, & Abdi, 2011). The benefit of using latent variables is that it reduces the dimensionality of the data.

In other words, the latent variables of **X** and **Y** ($\mathbf{L}_x$ and $\mathbf{L}_y$) are obtained by projecting the brain activity and behavioural data, **X** and **Y,** onto their respective saliences, which are the singular vectors **V** and **U,** as follow:

$$\mathbf{L}_x = \mathbf{XV}$$

$$\mathbf{L}_y = \mathbf{YU}$$

X Latent variables, $\mathbf{L}_x$, gives us the brain scores and **Y** latent variables, $\mathbf{L}_y$, gives us behaviour scores.

Arranging the PLSC data. All the data are arranged in one block (See Figure 4) and then concatenated.

In order to compute the latent variables, the saliences are obtained by decomposing the correlation matrix. Saliences are similar to the *loadings* in principal component analysis (PCA) and latent variables are similar to PCA components. For the examples in this tutorial, neuroimaging data are the brain activity data of participants in three university degree choices, English Major (EM), Mathematics Major (MM), and No Major (NM). Behavioural data are the scores of participants in a memory task. The task comprised of two scores, Words Recalled (WR) and Reaction Time (RT).

The PLS Correlation Procedure. Algorithm 1 is used to conduct the PLS correlation between the brain activity data, **X**, and the behavioural data, **Y**.

Algorithm 1 PLSC for **X** and **Y**

Input: $\mathbf{X} \in R^{n \times m}$ and $\mathbf{Y} \in R^{n \times k}$

Output: **U**; **V**; **Lx**; **Ly**
X ← normalize(**X**)
Y ← normalize(**Y**)
R ← $\mathbf{Y}^{\mathrm{T}}$.**X**
[**U**;**W**;**V**] ← SVD[**R**]
Lx ← **X**.**V**
Ly ← **Y**.**U**

Brain activity data and behavioural data are first normalized for each group and then the correlation matrix is formed for each group individually. Group correlation matrices are joined to get the correlation matrix **R**. Matrix **R** is then decomposed into three matrices, **U**, **W**, and **V**. Finally, latent vectors for **X** and **Y** are computed by projecting **X** and **Y** data to the two first vectors of **V** and **U** respectively.

Mathematica notebook for PLS correlation. A PLSC toolbox in MATLAB is implemented for neuroimaging by McIntosh, Chau, Lobaugh, & Shen, (2013). In this article, a program called "PLSCexample.nb" (implemented in Mathematica v. 8.0.1, Champaign, IL, USA) is available to download on the TQMP website (http://www.tqmp.org/). All sections and subsections are carefully labelled. Section 1 describes data entry for each condition and centres and normalizes the data. Section 2 computes the correlation matrix for each condition then "joins" all the matrices to obtain the correlation matrix "Rb". The correlation matrix is then decomposed using SVD. Section 3 visualizes the first and second behavioural saliences. Section 4 computes the latent variables for behavioural and neuroimaging data. Section 5 plots brain scores and behaviour scores on the first two latent variables respectively.

PLS methods – PLS regression

PLS regression was originally developed for econometrics to deal with collinear predictor variables. More recently this method has been applied to the analysis of brain imaging data (e.g., ERPs, functional magnetic resonance imaging, and magnetoencephalography).

Why and when is PLSR applied? PLS Regression is used to predict relationships between two datasets and is very useful for datasets with missing values, collinear or noisy independent variables (indicators). PLSR is used when the number of predictors is large compared to the number of observations and when the regression is not feasible because of multicollinearity (i.e., predictors are highly correlated and linearly dependent). In some statistical packages, if the researcher encounters missing data, the participant is sometimes completely removed from the analysis. With PLSR, missing data are estimated from the principle factors and principle components.

How does it work? Assume **X** (predictors, independent variables) is an *n*x*m* matrix and **Y** (response, dependent variables) is an *n*x*p* matrix. This can be formulated as:

$$\mathbf{X}=\mathbf{TP}^{\mathrm{T}}+\mathbf{E} = \sum_{i=1}^{a} \boldsymbol{t}_i \boldsymbol{p}_i^T + \mathbf{E}$$

$$\mathbf{Y}=\mathbf{UQ}^{\mathrm{T}}+\mathbf{F} = \sum_{i=1}^{a} \boldsymbol{t}_i \boldsymbol{c}_i^T + \mathbf{F}$$

$$\mathbf{U}=\mathbf{TD}$$

where **T** and **U**, with dimension *n*x*a*, are X-scores and Y-scores, **P** and **Q** are X-loadings and Y-loadings, **E** and **F** are X-residuals and Y-residuals, respectively and **D** is a diagonal matrix with $d_i i = (u_i^T t_i)/(t_i^T t_i)$

Latent variables are also called latent vectors (Zhao et al., 2013). **T** consists of extracted **X** latent variables. **T** is orthonormal which means $\mathbf{T}^{\mathrm{T}}\mathbf{T}= \mathbf{I}$. **U** consists of **Y** latent variables. **U** has maximum covariance with **T** column-wise. In order to find the latent variables, the sets of weights, **w**, **c** need to be optimized to satisfy,

$$\mathbf{max}\,[\mathbf{t}^{\mathrm{T}}\mathbf{u}] = \max_{\{w,c\}} [\mathbf{w}^T\mathbf{X}^T\mathbf{Y}\mathbf{c}] \text{ such that } \mathbf{w}^{\mathrm{T}}\mathbf{w}=1\,,\ \mathbf{c}^{\mathrm{T}}\mathbf{c}=1$$

The latent vector is then estimated as $\mathbf{t}=\mathbf{Xw}$. Based on the assumption of the linear inner relation of $\mathbf{u}=\mathbf{Dt}$, the predicted **Y** is obtained from:

$$\hat{\mathbf{Y}} = \mathbf{TDQ}^{\mathrm{T}} = \mathbf{XB}$$

where **B** is an *m*x*p* matrix of regression coefficients of the model.

This implies finding common latent vectors, $\mathbf{t}_i$, that explain the variances of both **X** and **Y** (See Figure 5).

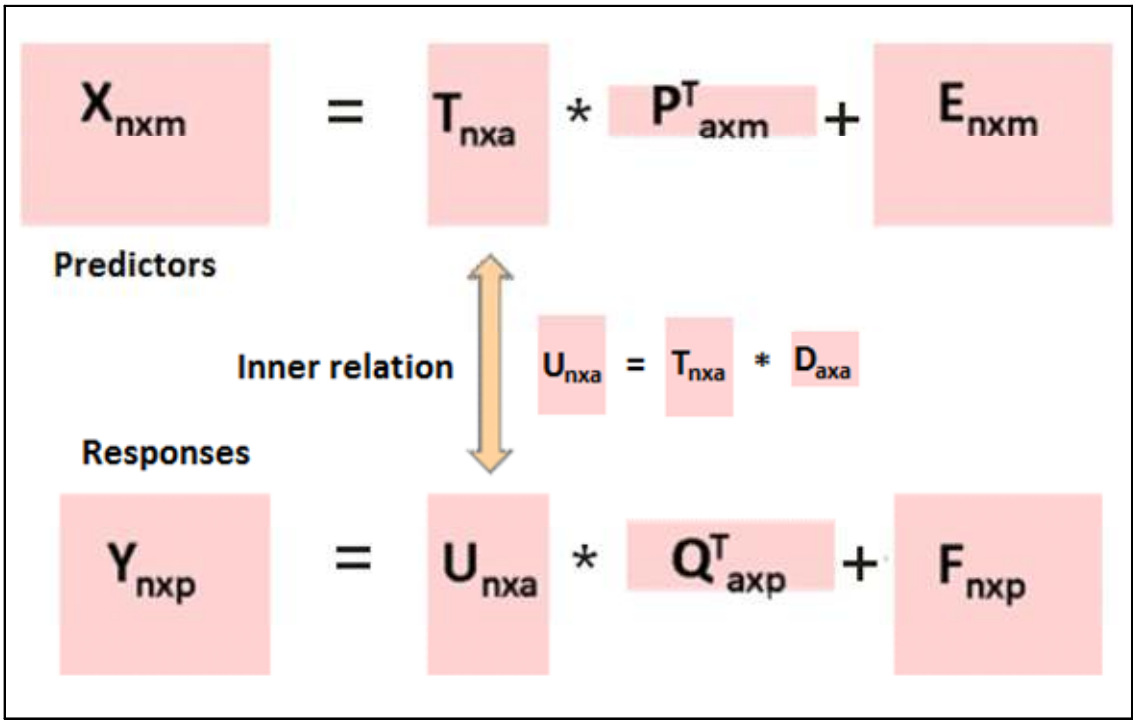


Figure 5 ■ PLSR decomposes **X** and **Y** data into orthogonal sets of scores (**T**, **U**), loadings (**P**, **Q**), and weights (w, c) which are evaluated to maximize the covariance between **T** and **U**. The central inner PLS relation is made up of a standard univariate regression of **U** upon **T**. In the PLSR model, this is called the operative **X-Y** Link. The weights are used to compute the regression coefficients of PLS, $\mathbf{B} = \mathbf{W}(\mathbf{P}^{T}\mathbf{W})^{-1}\mathbf{D}\mathbf{Q}^{T}$ (adapted from Zhao et al., 2013).

Arranging the PLSR data. Data arrangement is the same as shown for PLSC.

The PLS regression procedure. PLSR successively extracts latent vectors from both **X** and **Y** such that the covariance between extracted latent vectors is maximal. The extraction can be done by iterative algorithms such as Non-linear Iterative PArtial Least Squares (NIPALS; Abdi, 2010, 2012). NIPALS algorithm is used to calculate weights (**w**,**c**), loadings (**P**,**Q**) and scores (**T**,**U**) in this tutorial.

Algorithm 2 PLSR for **X** and **Y**

Input: $\boldsymbol{X} \in R^{n\times m}$ and $\boldsymbol{Y} \in R^{n\times p}$
The Number of latent vectors to be extracted is *nfactor*
Output: P; **U**; **B**; **T**

Initialization: E ← **X, F** ← **Y**, **t1** ← **F1**
For i = 1 **to** *nfactor* **do**
  While ($\|\boldsymbol{t}_0 - \boldsymbol{t}_i\| > \epsilon/2$)
    t0 ← **t**i
    wi ← $\boldsymbol{E}^{T}.\boldsymbol{u}_i / \|\boldsymbol{E}^{T}.\boldsymbol{u}_i\|$
    ti ← $\boldsymbol{E}.\boldsymbol{w}_i / \|\boldsymbol{E}.\boldsymbol{w}_i\|$
    ci ← $\boldsymbol{F}^{T}.\boldsymbol{t}_i / \|\boldsymbol{F}^{T}.\boldsymbol{t}_i\|$
    ui ← $\boldsymbol{F}.\boldsymbol{c}_i$
  end while
  pi ← $\boldsymbol{E}^{T}.\boldsymbol{t}_i$
  bi ← $\boldsymbol{u}_i^{T}.\boldsymbol{t}_i$
  Deflation:
  $\boldsymbol{E} \leftarrow \boldsymbol{E}$ - $\boldsymbol{t}_i.\boldsymbol{p}_i^{T}$
  $\boldsymbol{F} \leftarrow \boldsymbol{F}$ - **b**i x $\boldsymbol{t}_i.\boldsymbol{c}_i^{T}$
end for

E (residual of **X**) and **F** (residual of **Y**) are initialized with **X** and **Y** respectively. After the initialization process, which includes standardization of **X** and **Y** and resetting the variables used in the computation, the recursive process is performed on **w**, **c**, **u**, and **t** while the covariance between **u** and **t** vectors is examined in each iteration. The **u** and **t** are found and extracted when the covariance between them is maximal.

After extracting the latent vector, this vector will be removed from the **X** and **Y** before extracting the next latent vector and then this procedure is repeated. This

process is referred to as deflation of **X** and **Y**.

Mathematica notebook for PLS regression. The Mathematica notebook "PLSRexample.nb" is available to download from the TQMP website (http://www.tqmp.org/). Each section is clearly labeled. Some sections have subsections. Section 1 contains the modules for centring and normalizing data that will be used later. In Section 2, the neuroimaging and behavioural data are entered. The number of latent vectors to be extracted is also entered here. Section 3 involves standardizing the input data (neuroimaging and behavioural data) and defining and initializing the required matrices and vectors. Section 4 finds and extracts the latent vectors using the NIPALS algorithm. Latent vector number and iteration are printed here. Section 5 depicts the table of variances which displays the variability accounted for by each latent vector of the predictor and response data. Section 6 outputs weight matrix **C**, score matrices **T** and **U**, and graphs score plots, which are the projection of predictor and response data onto their first two latent vectors. Section 7 reconstitutes the predictor and response data and brings them back to their original units. A table for the predicted response (behavioural) data, $\widehat{\boldsymbol{Y}}$, and a table of regression coefficients are output. PLSR method is available in MATLAB as well. The program in MATLAB has been written by Hervé Abdi and can be found (www.utdallas.edu/~herve, article A76).

Examples

Data used for the PLSC and PLSR examples are behavioural and neuroimaging data for three groups of participants with three participants in each group (Krishnan et al., 2011). Matrix **X** stores neuroimaging or brain activity data (i.e. amplitudes across time for the vertex electrode, Cz) and matrix **Y** stores the behavioural data from a memory task. The brain activity data of participants are organized into three university degree choices, English Major (EM), Mathematics Major (MM), and No Major (NM). Behavioural data are the scores of participants in a memory task. The task comprised two scores, Words Recalled (WR) and Reaction Time (RT).

$$\boldsymbol{X} = \begin{bmatrix} 2 & 5 & 6 & 1 & 9 & 1 & 7 & 6 & 2 & 1 & 7 & 3 \\ 4 & 1 & 5 & 8 & 8 & 7 & 2 & 8 & 6 & 4 & 8 & 2 \\ 5 & 8 & 7 & 3 & 7 & 1 & 7 & 4 & 5 & 1 & 4 & 3 \\ 3 & 3 & 7 & 6 & 1 & 1 & 10 & 2 & 2 & 1 & 7 & 4 \\ 2 & 3 & 8 & 7 & 1 & 6 & 9 & 1 & 8 & 8 & 1 & 6 \\ 1 & 7 & 3 & 1 & 1 & 3 & 1 & 8 & 1 & 3 & 9 & 5 \\ 9 & 0 & 7 & 1 & 8 & 7 & 4 & 2 & 3 & 6 & 2 & 7 \\ 8 & 0 & 6 & 5 & 9 & 7 & 4 & 4 & 2 & 10 & 3 & 8 \\ 7 & 7 & 4 & 5 & 7 & 6 & 7 & 6 & 5 & 4 & 8 & 8 \end{bmatrix}$$

$$\boldsymbol{Y} = \begin{bmatrix} 15 & 600 \\ 19 & 520 \\ 18 & 545 \\ 22 & 426 \\ 21 & 404 \\ 23 & 411 \\ 29 & 326 \\ 30 & 309 \\ 30 & 303 \end{bmatrix}$$

Each row in **X** matrix shows the neuroimaging data for each participant. Each row in matrix **Y** shows the number of words each participant recalled (words Recalled) and the average time he/she took to recall the words, Reaction Time (RT).

PLS Correlation Example

In this example, the experiment was looking at degree choice and relating it to scores in a memory task for Words Recalled and Reaction Time. In PLSCexample.nb notebook, upon activation of sections 1 and 2, matrices of centered and normalized data and the matrix of correlation will be output. Activating cells in the first subsection of section 3 produces a bar graph that shows the first behaviour salience for each behavioural measurement, words recalled and RT.

Figure 6 shows that the first behaviour salience differentiates the EM group from two other groups in the Words Recalled behavioural measurement. This salience also differentiates the NM groups from two other groups in the Reaction Time behavioural measurement.

Activating cells in the second subsection of section 3 produces a second behaviour salience graph (See Figure 7). The second behaviour salience differentiates the MM group from two other groups in words recalled.

Activating cells in subsections of section 4 computes latent variables for neuroimaging (brain activity) and behavioural data and are computed from the saliences **V** and **U** respectively.

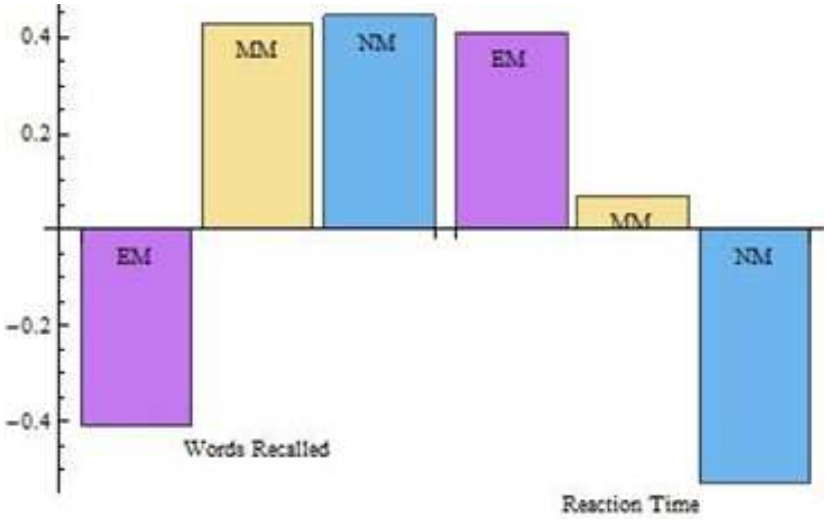


Figure 6 ■ First behaviour salience. The EMs differs from MMs and NMs in Words Recalled. The Reaction Time (RT) NM students differs from EM and MM.

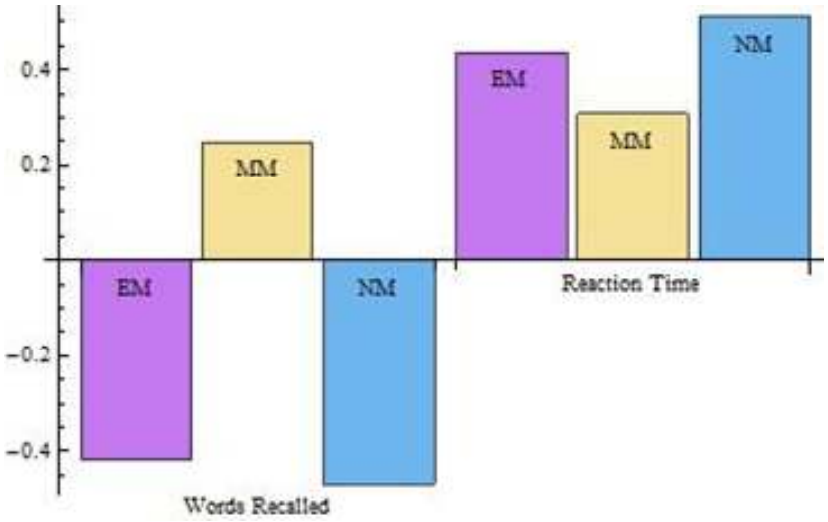


Figure 7 ■ Second behaviour salience. The second salience shows MM to be quite different from EM and NM students for Words Recalled. The salience does not differentiate the RT between the groups.

Activation of the first subsection of section 5 will give the score plot (PCA style) for the neuroimaging (brain activity) data as shown in figure 8. Finally, activating the last cell in section 5 will print the behaviour score plot, as in Figure 9. Only two latent variables were extracted because these two latent variables explain the highest percentage of covariance that describes the correlation matrix between the two datasets (e.g., neuroimaging and behaviour).

In short, the main finding is that NM students were negatively correlated in brain and behaviour scores with EMs and MMs. This means that the brain response (i.e., in terms of ERP averages, latencies, or both) in conjunction with the behaviour scores for the No Major students are decreasing as the scores for the Mathematics Majors and English Majors increase. This means for the No Major students, there is lower word recall and slower reaction times as compared with English or Mathematics Majors.

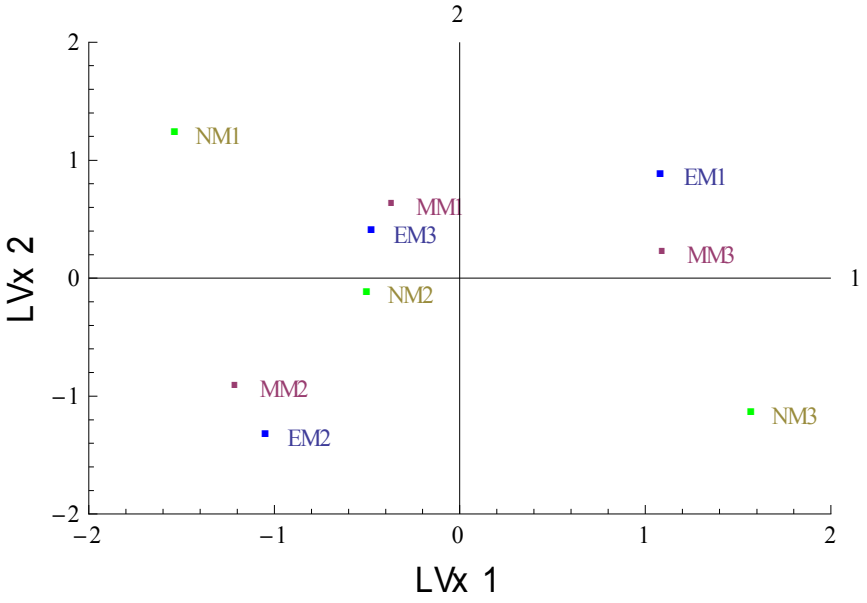


Figure 8 ■ Plot of latent variables to show the relationship of the covariances of the brain scores. Distance on the plot will directly reflect the amount of explained covariances of Rb (i.e. the correlation matrix). EM1, MM3 and NM3 are separated from all other scores; NM2, EM2, MM2, and NM3 are also separated from all other scores. NM is negatively correlated with the other two groups. The first LVx separates NM3, MM3, and EM1 from the other participants.

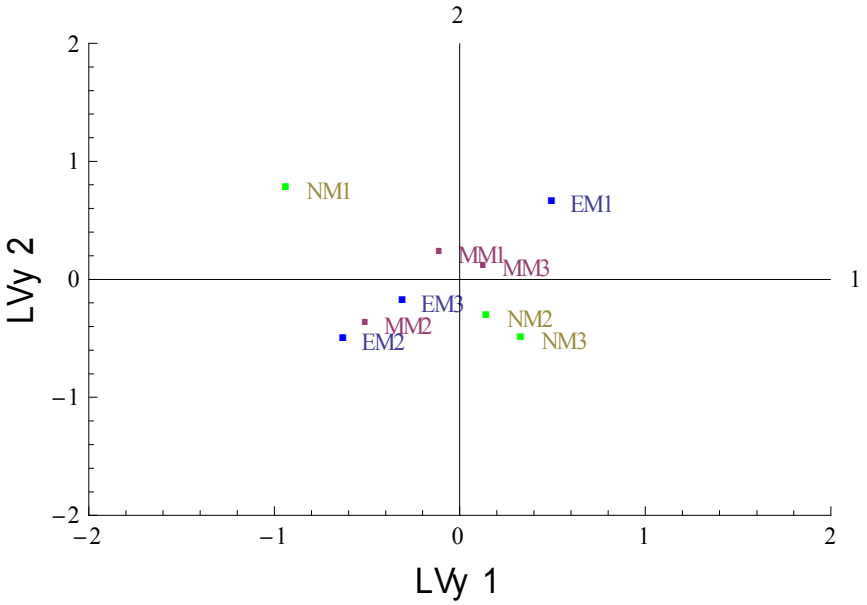


Figure 9 ■ Plot of the latent variables to show the relationship of the covariances of the behaviour scores (i.e. WR and RT). The NM group is negatively correlated with the EM and MM groups. The first latent variable, LVy, separates EM1, MM3, NM3 and NM2 participants from the others.

For more detailed information regarding the relationships, it is highly recommended to add bootstrapping techniques or a "constrained" PLS solution with *a priori* contrasts. The analyzed data in the correlation can then be plotted on an overhead view of the electrodes showing the brain scores with design scores for all recording electrodes. Other possible displays include design LV bar plots, and bar plots of the singular values and permutation test results. These plots are used to more easily view and interpret the resulting patterns' similarities and differences in the brain-behaviour relationships.

Table 1 ■ Explained variability of X and Y by each latent vector (in percentage).

Latent Vector	% explained Variance for X	% explained Variance for Y
1	31.7858	71.6899
2	18.8737	18.9773
3	18.5688	5.58335
4	6.5618	2.33263
5	12.0726	0.704142
6	5.78424	0.556225
7	5.25903	0.119044
8	1.09411	0.0374037

Table2 ■ Matrix C weights for response variables (behavioural measurements)

	y1	y2
c1	0.713805	-0.700345
c2	0.727437	-0.686175
c3	0.369485	-0.929237
c4	0.970131	-0.242582
c5	0.344749	-0.938695
c6	0.708496	-0.705715
c7	0.8071	-0.590415
c8	0.933474	0.358646

PLS Regression Example

Developing a regression model to predict the behavioural data from brain activity data are the focus for PLS Regression. Brain activity data, matrix **X**, are the predictor in this example. The behaviour data, matrix **Y**, are the response data. The results of this section can be obtained using the Mathematica notebook "PLSRexample.nb". Section 1 contains standardization and normalization modules that will be used later. Activating cells in Section 2 will input data and this is where the number of latent vectors for extraction may be changed by replacing the value of the "nfactor" variable with a desired value. The latent vectors to be extracted are obtained in sections 3 and 4. Section 5 will produce Table1 which shows the percentage of variances for **X** and **Y** accounted for by each latent vector. In this example, the first two latent vectors account for more than 90 percent of the variability of **Y.** This means the optimum prediction can be reached by a prediction model using only two latent vectors.

Activation of cells in Section 6 outputs score matrices **T** and **U** and weight matrix **C** (Table 2) and produces a score plot that shows the projection of participants onto the two first latent vectors of **X** (Figure 10) and a score plot that shows the projection of participants onto the two first vectors of **Y** (Figure 11.3).

In Figure 11, the first latent vector separates the NM group from two other groups and the second latent vector separates the EM group from the other groups. The plot explains the same scenario as Figure 10. This shows the accuracy of the prediction.

By activating the cells in Section 7, predictor and response data are reconstituted and the predicted response data (behavioural measurement), $\hat{\boldsymbol{Y}}$ is computed and printed out as, Table 3. The regression coefficient table is also printed out as Table 4. The predicted **Y**, $\hat{\boldsymbol{Y}}$, is exactly the same as $\boldsymbol{Y}$, the behavioural data, with eight extracted latent vectors.

Different values for *nfactor* may be entered to see how the number of extracted latent vectors affects the regression model.

One way to predict **Y** from new **X** data is to multiply **X** with the regression coefficients matrix. Therefore, multiplying new brain activity data by the regression coefficient matrix of the PLSR model will yield predicted behaviour data. This is very useful especially when only neuroimaging data are available. Studying the behaviour effect of drugs from neuroimaging data can be a possible application of PLSR.

Discussion

Partial Least Squares (PLS) has many advantages over simple regression or multiple linear regressions. For example, PLS is able to handle (i) more descriptor variables than compounds, (ii) non-orthogonal descriptors, and (iii) multiple biological results. In addition, PLS has (i) more predictive accuracy, and (ii) a lower risk of chance correlation.

PLSC looks at the 'shared' information between the variables whereas PLSR looks at the directionality and predictability of the DVs from a set of IVs.

Some of the major limitations of PLS include (i) a higher risk of overlooking "real" correlations, and (ii) sensitivity to the relative scaling of the descriptor variables. In regard to the data analyzed in this article, no evaluation is performed due to the size of the dataset. This artificial dataset was used only as an example of procedures using Mathematica. However, the data are not generalizable to the population due to the low N values used. The code can be further

Table 3 ■ Matrix $\hat{Y}$ when 8 latent vectors are used.

	Words Recalled	RT
Participant1	15	600
Participant2	19	520
Participant3	18	545
Participant4	22	426
Participant5	21	404
Participant6	23	411
Participant7	29	326
Participant8	30	309
Participant9	30	303

Table 4 ■ Regression coefficients for 8 latent vectors model

Words Recalled	RT
0.579289	-0.42948
0.033262	-0.00023
-0.2082	0.20906
0.114298	-0.08455
-0.26369	0.404358
0.172974	-0.22565
-0.06364	0.020402
-0.18022	0.218192
-0.17349	0.120258
-0.01139	-0.02187
0.10998	-0.10843
0.452123	-0.49455

automated using modules. If there are a sufficient number of cases and a robust model is required, the model should be validated using bootstrap or k-folds techniques.

How the Performance of PLS Correlation Model is Evaluated?

Bootstrap sampling is used to estimate the standard error in PLS correlation (Krishnan et al., 2011). Using the dataset, a bootstrap sample is created by repeatedly randomly sampling with replacement. Error is estimated by applying PLSC to this bootstrap sample. Permutation tests are generally employed to perform hypothesis tests. In permutation tests, Student and Fisher's nonparametric estimation of sampling distributions randomly rearranges rows of each matrix and then re-applies PLSC. This process is repeated many times in order to estimate the probability distribution of singular values under the null hypothesis.

How the Performance of PLS Regression Model is Evaluated?

The prediction performance of the PLSR model can be examined using cross-validation techniques such as k-fold (Zhao et al., 2013). With *k* fold cross-validation, **X** and **Y** data are randomly partitioned into approximately equal *k* size subsamples of observations. From the *k* subsamples, *k*-1 subsamples are used as training data and the remaining subsample is used as validation data. A PLSR model is developed on the training data and then tested with the validation data. This is repeated *k* times with each of the individual subsamples being used only once for the validation data. The repetitions produce *k* models with *k* results. The estimated prediction error will be obtained from discrepancies between the predicted response data or results, $\hat{\boldsymbol{Y}}$ and the observed response data, **Y**. The smaller the prediction error becomes, the better the prediction. The root mean square error (RMSE) of prediction can be used to measure the prediction performance of the model.

Acknowledgements

We thank Trista Takacs for comments on earlier drafts of this article.

(article continues on next page)

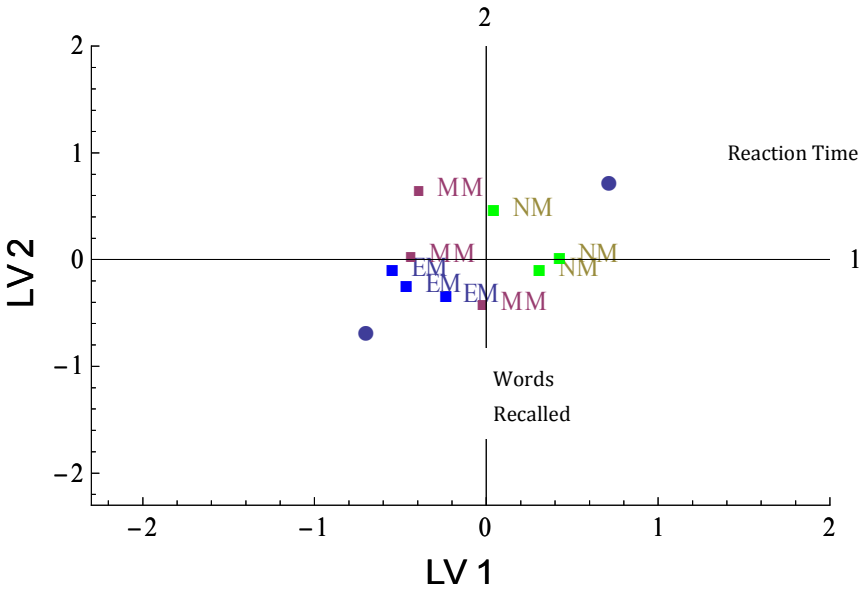


Figure 10 ■ Score plot shows the projection of participants on the first two latent vectors of X. The first latent vector separates the NM group from the other groups in terms of brain activity and the second latent vector separates the EM group from other two groups. The plot also shows how brain activity predicts behavioural data. The EM group shows the lowest brain activity that predicts the highest words recalled with the lowest reaction time and the NM group shows highest brain activity that predicts the lowest word recalled with the highest reaction time. This means that EMs can recall words with less effort and the NM group has difficulty recalling words.

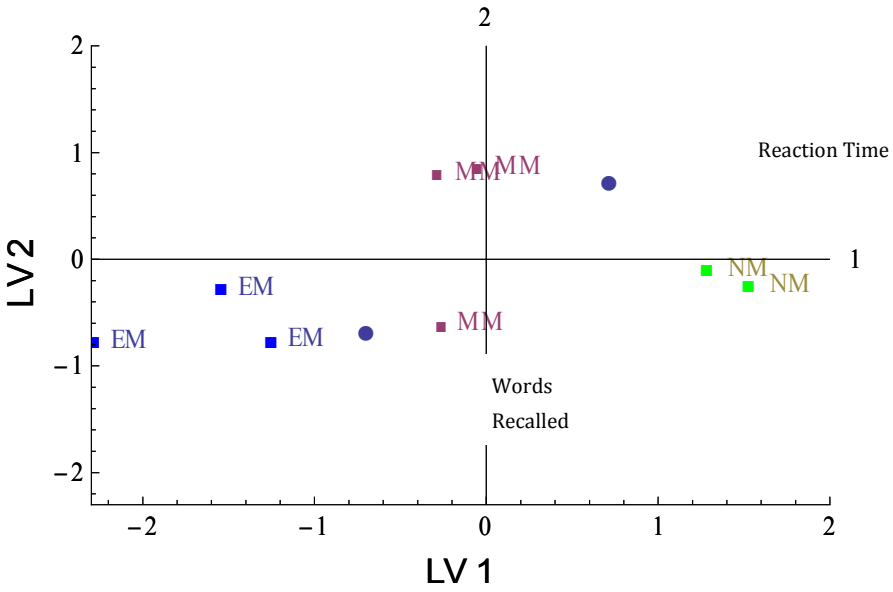


Figure 11 ■ The score plot shows the projection of participants onto the first two latent vectors of Y. The first latent vector separates the NM group from the other groups in terms of brain activity and the second latent vector separates the EM group from the other two groups. The plot also shows how brain activity predicts behavioural data. The EM group shows the lowest brain activity that predicts the highest words recalled with the lowest reaction time and the NM group shows the highest brain activity that predicts the lowest word recalled with the highest reaction time. This means that EMs can recall words with less effort and the NM group has difficulty recalling words.

References

Abdi, H. (2010). Partial least squares regression and projection on latent structure regression (PLS Regression). *Wiley Interdisciplinary Reviews: Computational Statistics*, *2*(1), 97–106. doi:10.1002/wics.51

Abdi, H., & Williams, L. J. (2013). Partial least squares methods: Partial least squares correlation and partial least square regression. In B. Reisfeld & A. N. Mayeno (Eds.), *Computational Toxicology, Volume II, Methods in Molecular Biology* (Vol. 930, pp. 549–578). Totowa, NJ: Humana Press. doi:10.1007/978-1-62703-059-5

Beck, M., & Geoghegan, R. (2010). *The Art of Proof*. *Undergraduate Texts in Mathematics* (pp. 7023–7027). New York, NY: Springer New York. doi:10.1007/978-1-4419-7023-7

Bóna, M. (2011). *A Walk Through Combinatorics* (3rd ed., p. 568). Hackensack, NJ: World Scientific Publishing Co. Pte. Ltd.

Brown, J. D. (2009). Principal components analysis and exploratory factor analysis – Definitions, differences and choices. *Shiken: JALT Testing & Evaluation, SIG Newsletter*, *13*(1), 26–30.

Esposito Vinzi, V., Trinchera, L., & Amato, S. (2010). *Handbook of Partial Least Squares: Concepts, Methods and Applications*. (V. Esposito Vinzi, W. W. Chin, J. Henseler, & H. Wang, Eds.) (pp. 47–83). Berlin, Heidelberg: Springer Berlin Heidelberg. doi:10.1007/978-3-540-32827-8

Ghazy, R. A., Hadhoud, M. M., Dessouky, M. I., El-Fishawy, N. A., & Abd El-Samie, F. E. (2008). Performance evaluation of block based SVD image watermarking. *Progress in Electromagnetics Research B*, *8*, 147–159.

Haenlein, M., & Kaplan, A. M. (2004). A Beginner's Guide to Partial Least Squares Analysis. *Understanding Statistics*, *3*(4), 283–297. doi:10.1207/s15328031us0304_4

Haykin, S. (1991). *Adaptive Filter Theory* (2nd ed.). Upper Saddle River, NJ: Prentice Hall.

Henning, G. (1989). Meanings and implications of the principle of local independence. *Language Testing*, *6*(1), 95–108. doi:10.1177/026553228900600108

Krishnan, A., Williams, L. J., McIntosh, A. R., & Abdi, H. (2011). Partial Least Squares (PLS) methods for neuroimaging: a tutorial and review. *NeuroImage*, *56*(2), 455–475. doi:10.1016/j.neuroimage.2010.07.034

Legendre, A. M. (1806). *Nouvelles méthodes pour la détermination des orbites des comètes : avec un supplément contenant divers perfectionnemens [perfectionnements] de ces méthodes et leur application aux deux comètes de 1805* (p. 156). Paris, France: Courcier, Imprimeur-Libraire pour les Mathematiques.

McIntosh, A. R., Bookstein, F. L., Haxby, J. V., & Grady, C. L. (1996). Spatial pattern analysis of functional brain images using partial least squares. *NeuroImage*, *3*(3 Pt 1), 143–157. doi:10.1006/nimg.1996.0016

McIntosh, A. R., Chau, W., Lobaugh, N., & Shen, J. (2013). Partial Least Squares GUI for PET, fMRI & EEG/MEG. Toronto, ON, Canada. Retrieved from http://www.cma.mgh.harvard.edu/iatr/display.php?spec=id&ids=305

Picton, T. W., Lins, O. G., & Scherg, M. (1995). The recording and analysis of event-related potentials. In F. Boller & J. Grafman (Eds.), *Handbook of Neuropsychology* (Vol. 10, pp. 3–73). Amsterdam: Elseviier Science B.V.

Sigler, L. E. (2002). *Fibonacci's Liber Abaci* (p. 636). Berlin, Germany: Springer-Verlag.

Tenenhaus, M., Esposito Vinzi, V., Chatelin, Y.-M., & Lauro, C. (2005). PLS path modeling. *Computational Statistics & Data Analysis*, *48*(1), 159–205. doi:10.1016/j.csda.2004.03.005

Tong, H., Papadimitriou, S., Yu, P. S., & Faloutsos, C. (2008). Proximity tracking on time-evolving bipartite graphs. *SDM*. Retrieved from http://www.cs.cmu.edu/~htong/pdfs/kdd08_tong.ppt

Vallesi, A. (2009). Effects of aging on selecting not to respond: A cross-sectional ERP study. *Rotman Research Institute, ERP lab meeting Powerpoint presentation*.

Wold, H. (1964). *Econometric Model Building: Essays on the causal chain approach*. Amsterdam: North-Holland.

Zhao, Q., Caiafa, C. F., Mandic, D. P., Chao, Z. C., Nagasaka, Y., Fujii, N., … Cichocki, A. (2013). Higher order partial least squares (HOPLS): a generalized multilinear regression method. *IEEE transactions on pattern analysis and machine intelligence*, *35*(7), 1660–1673. doi:10.1109/TPAMI.2012.254

Citation

Van Roon, P., Zakizadeh, J., & Chartier, S. (2014). Partial Least Squares Tutorial for Analyzing Neuroimaging Data. *The Quantitative Methods for Psychology*, *10* (2), 200-215.

Received: 22/06/13 ~ Accepted: 24/09/13