Handout 14

Partial Least Squares

Partial least squares (PLS) is a method that is quite popular among many scientists. Like the ridge regression, this method is useful when there is a substantial amount of collinearity among the independent variables. It is important to point out that the PLS method is statisticians' renaming of what is known as the "conjugate gradient" method of optimization. Since the conjugate gradient method can be applied to many different problems, there are versions of the PLS method for statistical procedures such as regression, classification etc. This note is concerned with regression. We will assume that all the variables (Y and the X's) have been standardized.

This method can be described as follows. Suppose $(Y_i, x_i), i = 1, \ldots, n$ are observations where x_i's are p dimensional, i.e., there are p independent variables. We will assume that all the variables have been standardized. Denote $b = \sum x_i Y_i$ and $S = \sum x_i x_i^T$. This method creates univariate (independent) mutually uncorrelated variables of the form $\{z_{i1} = f_1^T x_i\}$, $\{z_{i2} = f_2^T x_i\}, \{z_{i3} = f_3^T x_i\}$,... by finding f_1, f_2, etc. Once these independent variables have been created, then it is fairly easy to run a linear regression of Y on $z_1, ..., z_k$. Principal components regression (PCR) looks similar to the PLS method since both methods generate independent variables $z_{i1}, z_{i2}, ...$ etc., except that the generated independent variables are of different nature for the two methods There is reason to believe that the PLS regression is superior to PCR. We will discuss only the PLS regression method here.

If $k = p$, then PLS method is equivalent to a multiple regression of Y on x. In practice, one needs to deal with two issues:

(i) how to construct the variables z_1, z_2, etc., i.e., how to find the vectors f_1, f_2, etc.

(ii) how to choose k?

Regression of Y on $z_1, ..., z_k$.

Since the independent variables $z_1, ..., z_k$ are uncorrelated, if we run a least squares we need to minimize

$$\begin{aligned}&\sum (Y_i - \alpha_1 z_{i1} - \cdots - \alpha_k z_{ik})^2 \\ &= \sum Y_i^2 + \alpha_1^2 \sum z_{i1}^2 + \cdots + \alpha_k^2 \sum z_{ik}^2 - 2\alpha_1 \sum Y_i z_{i1} - \cdots - 2\alpha_k \sum Y_i z_{ik}.\end{aligned}$$

Minimization of the above quantity leads to

$$\hat{\alpha}_1 = \sum Y_i z_{i1} \Big/ \sum z_{i1}^2 \ , \ldots, \hat{\alpha}_k = \sum Y_i z_{ik} \Big/ \sum z_{ik}^2 \ .$$

Thus the fitted regression is of the form

$$\begin{aligned}\hat{\mu}_i(k) &= \hat{\alpha}_1 z_{i1} + \cdots + \hat{\alpha}_k z_{ik} = \hat{\beta}(k)^T x_i, \text{ where} \\ \hat{\beta}(k) &= \hat{\alpha}_1 f_1 + \cdots + \hat{\alpha}_k f_k.\end{aligned}$$

Analysis of cars93 data.

Ridge regression was used to analyze cars93 data. Recall that it has substantial amount of multicollinearity. The PLS method is employed here using the R-package 'pls'. The data has been transformed and standardized. We fitted a PLS model using $k = 6$ components using the R command

cars=plsr(y~0+x1+x2+x3+x4+x5+x6,6,validation="CV")

The second entry is for number of components (the number of components could be anywhere between 1 and 6).

A plot of the first three component scores is given [R command: plot(cars,plottype="scores",comps=1:3)]

Here is the output when using the command summary(cars).

Data: X dimension: 93 6

Y dimension: 93 1

Fit method: kernelpls

Number of components considered: 6

VALIDATION: RMSEP

Cross-validated using 10 random segments.

	(Intercept)	1 comps	2 comps	3 comps	4 comps	5 comps	6 comps
CV	1.005	0.5833	0.5354	0.5343	0.5492	0.5457	0.5452
adjCV	1.005	0.5821	0.5338	0.5325	0.5458	0.5428	0.5424

TRAINING: % variance explained

	1 comps	2 comps	3 comps	4 comps	5 comp	6 comps
X	84.17	90.32	95.08	95.93	98.66	100.00
Y	68.17	74.62	75.12	75.45	75.45	75.45

The function "plsr" uses a multifold cross-validation procedure (often 10-fold CV) or it makes an adjustment to the cross-validation. It looks as if the CV criterion asks us to choose a PLS model with three components i.e., $\hat{k} = 3$. The R^2 values are given. With $k = 1$ component, $R = 0.6817$; with $k = 2$, $R_2^2 = 0.7462$ etc. We may also calculate;ate the adjusted R^2 values noting that

$$R^2_{adj,k} = 1 - \frac{n-1}{n-k-1}(1 - R^2_k).$$

Thus the adjusted $R^2_{adj,k}$ values are given in the table below along with some F-statistics.

We may also examine the results using a hypothesis testing framework to decide if three components are really needed. Note that

$$SSR(k) = R^2_k \times SSTO,\ SSE(k) = SSTO - SSR(k) = (1 - R^2_k)SSTO$$

So if we want to test if the k^{th} component is better than the model with $k-1$ components, the F-statistic

is

$$\begin{aligned} F_k &= [SSR(k) - SSR(k-1)]/MSE(k) \\ &= [R_k^2 \times SSTO - R_{k-1}^2 \times SSTO]/[(1-R_k^2)SSTO/(n-k-1)] \\ &= (n-k-1)[R_k^2 - R_{k-1}^2]/(1-R_k^2). \end{aligned}$$

Noting that $R_0^2 = 0$, we therefore have

k	1	2	3	4	5	6
$R^2_{adj,k}$	0.6782	0.7406	0.7428	0.7433	0.7404	0.7373
F_k	194.92	22.87	1.79	1.18	< 0.5	< 0.5

These F-values clearly indicate that $k = 2$ components is enough. For $k = 2$, the regression coefficients $\hat{\beta}(2)$ are (need to rerun "pls" with number of components equal to 2):

$\hat{\beta}_1 = -0.1085, \hat{\beta}_2 = 0.0225, \hat{\beta}_3 = 0.1154, \hat{\beta}_4 = 0.5537, \hat{\beta}_5 = 0.0618, \hat{\beta}_6 = 0.1067.$

The vectors f_1, f_2 and f_3 are given below(these are also called loadings): [R command "loadings(object)"]

Loadings:

Comp 1	Comp 2	Comp 3
-0.425	0.262	-0.360
-0.398	0.481	-0.602
0.405	0	-0.719
0.385	0.837	0.170
0.414	-0.188	-0.122
0.429	-0.120	-0.272

How are the vectors $f_1, \ldots, f_k$ constructed?

Construction of $f_1, f_2, \ldots$ are done iteratively and the method automatically finds $\hat{\alpha}$'s and $\hat{\beta}(k)$'s as part of the construction.

Step 1. Take $f_1 = b$ where $(b = \sum x_i Y_i)$ and $r_0 = b$. Then

$$z_{i1} = b^T x_i, \ \hat{\alpha}_1 = \sum Y_i z_{i1} \Big/ \sum z_{i1}^2 \ = f_1^T b / f_1^T S f_1$$

and $\hat{\beta}(1) = \hat{\alpha}_1 f_1$. Create a remainder vector $r_1 = b - S\hat{\beta}(1)$.

Step 2. Take $f_2 = r_1 + \gamma f_1$ so that $\{z_{i2} = f_2^T x_i\}$ is uncorrelated with $\{ \ z_{i1} = f_1^T x_i\}$. This requires choosing γ properly. Note that the empirical covariance of $\{z_{i2}\}$ and $\{z_{i1}\}$ is n times $\sum z_{i1} z_{i2} = f_1^T S f_2$. So equating $f_1^T S f_2$ to zero we get $\gamma_1 = -f_1^T S r_1 / f_1^T S f_1$. Thus if we take $f_2 = r_1 + \gamma_1 f_1$, then $\{z_{i2}\}$ and $\{z_{i1}\}$ are uncorrelated. Now regress Y on z_{i1} and z_{i2}. Since $\{z_{i2}\}$ and $\{z_{i1}\}$ are uncorrelated we get

$$\hat{\alpha}_1 = \sum Y_i z_{i1} \Big/ \sum z_{i1}^2 \ = f_1^T r_0 / f_1^T S f_1, \ \hat{\alpha}_2 = \sum Y_i z_{i2} \Big/ \sum z_{i2}^2 \ . = f_2^T b / f_2^T S f_2.$$

Note that $\hat{\alpha}_1$, the estimated regression coefficient for z_1, is the same as in Step 1. Then the estimated regression is

$$\hat{\mu}_i(2) = \hat{\alpha}_1 z_{i1} + \hat{\alpha}_2 z_{ik} = \hat{\beta}(2)^T x_i, \text{ where } \hat{\beta}(2) = \hat{\alpha}_1 f_1 + \hat{\alpha}_2 f_2 = \hat{\beta}(1) + \hat{\alpha}_2 f_2.$$

Create the remainder $r_2 = b - S\hat{\beta}(2)$.

We can continue in this manner. Let us now describe Step k assuming that we have done Step $k-1$.
Step k. Take $f_k = r_{k-1} + \gamma f_{k-1}$ so that uncorrelatedness of $\{z_{ik} = f_k^T x_i\}$ and $\{z_{i,k-1} = f_{k-1}^T x_i\}$ is guaranteed and it happens if γ is taken to be $\gamma_k = -f_{k-1}^T S r_{k-1} / f_{k-1}^T S f_{k-1}$. Thus $f_k = r_{k-1} + \gamma_k f_{k-1}$. A not-so-obvious mathematical result says that $\{z_{i1} = f_1^T x_i\}, \ldots, \{z_{ik} = f_k^T x_i\}$ are mutually uncorrelated. If Y is regressed on $z_1, \ldots, z_k$, then the estimated regression coefficients and the fitted values are

$$\begin{aligned}
\hat{\alpha}_1 = \sum Y_i z_{i1} \Big/ \sum z_{i1}^2 &= f_1^T b / f_1^T S f_1, \ldots, \hat{\alpha}_k = \sum Y_i z_{ik} \Big/ \sum z_{ik}^2 = f_k^T b / f_k^T S f_k, \\
\hat{\mu}_i(k) &= \hat{\alpha}_1 z_{i1} + \cdots + \hat{\alpha}_k z_{ik} = \hat{\beta}(k)^T x_i, \text{ where} \\
\hat{\beta}(k) &= \hat{\alpha}_1 f_1 + \cdots + \hat{\alpha}_k f_k = \hat{\beta}(k-1) + \hat{\alpha}_k f_k, \\
r_k &= b - S\hat{\beta}(k).
\end{aligned}$$

Algorithmic steps.

Step 1. (i) Let $f_1 = b, \hat{\alpha}_1 = f_1^T b / f_1^T S f_1$,(ii) $\hat{\beta}(1) = \hat{\alpha}_1 f_1, \hat{\alpha}_1 = f_1^T b / f_1^T S f_1$(iii) $r_1 = b - S\hat{\beta}(1)$
Step 2. (i) $f_2 = r_1 + \gamma_1 f_1, \gamma_1 = -f_1^T S r_1 / f_1^T S f_1$,
(ii) $\hat{\beta}(2) = \hat{\beta}(1) + \hat{\alpha}_2 f_2$, $\hat{\alpha}_2 = f_2^T b / f_2^T S f_2$, (iii) $r_2 = b - S\hat{\beta}(2)$,
Step 3. (i) $f_3 = r_2 + \gamma_2 f_2, \gamma_2 = -f_2^T S r_2 / f_2^T S f_2$,
(ii) $\hat{\beta}(3) = \hat{\beta}(2) + \hat{\alpha}_3 f_3$, $\hat{\alpha}_3 = f_3^T b / f_3^T S f_3$(iii) $r_3 = b - S\hat{\beta}(3)$,
$\vdots$
Step k. (i) $f_k = r_{k-1} + \gamma_k f_{k-1}, \gamma_k = -f_{k-1}^T S r_{k-1} / f_{k-1}^T S f_{k-1}$,
(ii) $\hat{\beta}(k) = \hat{\beta}(k-1) + \hat{\alpha}_k f_k$, $\hat{\alpha}_k = f_k^T b / f_k^T S f_k$, (iii) $r_k = b - S\hat{\beta}(k)$.

Statistical reasonings (covariance maximization interpretation).

Even though PLS is a renaming of the conjugate gradient method, one can provide some statistical insights. Below is a statistical intuition for PLS which creates $f_1, f_2, \ldots$ iteratively. In this interpretation $f_1, f_2, \ldots$ are of unit length, but that does not change the actual procedure, since in the algorithmic steps given above, in each substep (i) we may simply normalize each f_j to unit length. Neither the values of$\hat{\beta}(k)$'s nor the fitted Y values are changed due to this normalization. Only thing that change that are values of $\hat{\alpha}_k$'s as they get rescaled.

Suppose that we are trying to regress Y_i on $f^T x_i$ where f is a given vector of unit length in R^p. So we

can minimize $\sum(Y_i - cf^T x_i)^2$ over c and this leads to $\hat{c} = \sum Y_i f^T x_i / \sum (f^T x_i)^2 = f^T b / f^T S f$. Thus we have

$$\begin{aligned}
&\min_c \sum (Y_i - cf^T x_i)^2 \\
&= \sum (Y_i - \hat{c} f^T x_i)^2 = \sum Y_i^2 - \hat{c}^2 \sum (f^T x_i)^2 \\
&= \sum Y_i^2 - [\sum Y_i f^T x_i]^2 / [\sum (f^T x_i)^2] \\
&= \sum Y_i^2 - Cov(Y, f^T x)^2 / Var(f^T x).
\end{aligned}$$

If we want to choose the best f which minimizes the residual sum of squares, it is equivalent to maximizing $Cov(Y, f^T x)^2 / Var(f^T x)$ or maximizing

$$Cov(Y, f^T x)^2 / [Var(Y) Var(f^T x)] = Corr(Y, f^T x)^2.$$

It is not difficult to see that the minimization occurs when

$$f \propto S^{-1} b, \quad \text{or } f = S^{-1} b / ||S^{-1} b||.$$

In other words, the maximal correlation between Y and $f^T x$ is achieved when f is the least squares estimate (or proportional to it). Now suppose instead of maximizing the quantity $Cov(Y, f^T x)^2 / Var(f^T x)$ over f which is equivalent to maximizing $Corr(Y, f^T x)^2$, we choose to maximizing only $Cov(Y, f^T x)^2$ over f of unit length. Since $Cov(Y, f^T x)^2 = (f^T b)^2$, this is equivalent to maximizing $(f^T b)^2$ over f with $||f|| = 1$. The solution of this is clearly, $f_1 = b/||b||$. Thus the first created independent variable is $z_{i1} = f_1^T x_i$. Thus the fitted values are $\hat{\mu}_{i,1} = \hat{\alpha}_1 f_1^T x_i$, where $\hat{\alpha}_1 = \hat{\alpha}_1 = f_1^T b / f_1^T S f_1$ and the residuals are $e_{i,1} = Y_i - \hat{\mu}_{i,1}$. Now suppose that we want to regress Y on z_{i1} and $f^T x_i$ for some given f where z_{i1} and $f^T x_i$ are uncorrelated, then we will find α_1 and α so that sum of squared deviations of $Y_i - \alpha_1 z_{i1} - \alpha f^T x_i$ are minimized, i.e., the quantity

$$\begin{aligned}
&\sum (Y_i - \alpha_1 z_{i1} - \alpha f^T x_i)^2 \\
&= \sum Y_i^2 + \alpha_1^2 \sum z_{i1}^2 + \alpha^2 \sum (f^T x_i)^2 - 2\alpha_1 \sum Y_i z_{i1} - 2\alpha \sum Y_i f^T x_i \\
&= \sum Y_i^2 + \alpha_1^2 f_1^T S f_1 + \alpha^2 f^T S f - 2\alpha_1 f_1^T b - 2\alpha f^T b
\end{aligned}$$

is minimized with respect to α_1 and α. These minimums occur at

$$\hat{\alpha}_1 = f_1^T b / f_1^T S f_1, \ \hat{\alpha} = f^T b / f^T S f.$$

Thus we have

$$\min_{\alpha_1, \alpha} \sum (Y_i - \alpha_1 z_{i1} - \alpha f^T x_i)^2 = \sum Y_i^2 - (f_1^T b)^2 / (f_1^T S f_1) - (f^T b)^2 / f^T S f.$$

In order to find the best f if we minimize $(f^T b)^2 = Cov(Y, f^T x)$ with respect to f subject to the constraints that $||f|| = 1$ and $Cov(f_1 x, f^T x) = f_1^T S f = 0$, then it can be shown that the best f is

$$f_2 \propto r_1 + \gamma_1 f_1, \text{ with } \gamma_1 = -f_1^T S r_1 / f_1^T S f_1,$$

where $r_1 = b - S\hat{\beta}(1) = \sum(Y_i - \hat{\alpha}_1 f_1^T x_i)x_i$ and $\hat{\beta}(1) = \hat{\alpha}_1 f_1$. This the gives us the second independent variable $z_{i2} = f_2^T x_i$.

Similar interpretations hold for f_3, f_4 etc.

How to select k.

Usually k is selected by a cross-validation criterion. Suppose that the $z_1, .., z_k$ have been created by deleting the i^{th} case and the resulting estimated beta parameter is denoted by $\hat{\beta}^{(-i)}(k)$, then a cross-validation criterion can be written as

$$CV(k) = n^{-1} \sum (Y_i - \hat{\beta}^{(-i)}(k)^T x_i)^2.$$

One can minimize this criterion over k and if the minimum is attained at $k = \hat{k}$, then we create $\hat{k}$ z-variables and run a PLS regression.

Can we avoid calculating $\hat{\beta}^{(-i)}(k)$ n-times as in the ridge regression case and provide a GCV type criterion? The answer is unknown. Nor is it known how to obtain GCV type criterion. It may not be possible avoid calculating the beta parameters n times. Thus the ordinary cross-validation (also called leave-one-out cross validation) may turn out to be computationally expensive for large data sets. However, one may use a modification of the cross-validation criterion by using what is known as multifold cross-validation. The method is described below.

Randomly partition the n observations in v groups of roughly equal size, say $v = 10$. Assume that the indices of the groups are $I_1, ..., I_v$.The number of elements is in group j is $n_j \approx n/v$. Call this group the test set and the remaining $v - 1$ groups (put together) as the learning set. The idea to obtain parameter estimates based on the observations in the learning set and then use these parameter estimates to predict in the test set. For a each j, run a PLS regression on the basis of the observation the learning set, i.e., in $I_1, .., I_{j-1}, I_{j+1}, ..., I_v$, and denote the resulting estimate of β as $\hat{\beta}^{(-j)}$. Note that $\hat{\beta}^{(-j)}$ has been estimated on the basis of $n - n_j \approx n - n/v$ observations. Now one can find out how well the regression with the beta parameters $\hat{\beta}^{(-j)}$ predict the observation in the test set whose indices are in I_j. We can obtain an estimate of the prediction error

$$n_j^{-1} \sum_{i \in I_j} (Y_i - \hat{\beta}^{(-j)T} x_i)^2.$$

Now we may do this procedure for each $j = 1, ..., v$ and obtain v estimates of the prediction error. Their weighted average will lead to an estimate of the prediction error

$$CV^{(v)}(k) = \sum_{j=1}^{v} (n_j/n) n_j^{-1} \sum_{i \in I_j} (Y_i - \hat{\beta}^{(-j)T} x_i)^2 = n^{-1}. \sum_{j=1}^{v} \sum_{i \in I_j} (Y_i - \hat{\beta}^{(-j)T} x_i)^2$$

Remarks: 1. When $v = n$, the v-fold cross-validation is the same as ordinary cross-validation.
2. It is not advisable to take v to be small such 2 or 3. Ideally one should at least a 8-fold cross-validation. When v is small such as 2 or 3, a correction factor has to added to $CV^{(v)}(k)$ in order to make it a good approximation to the true prediction error.

Figure 1: PLS Analysis of cars93 data

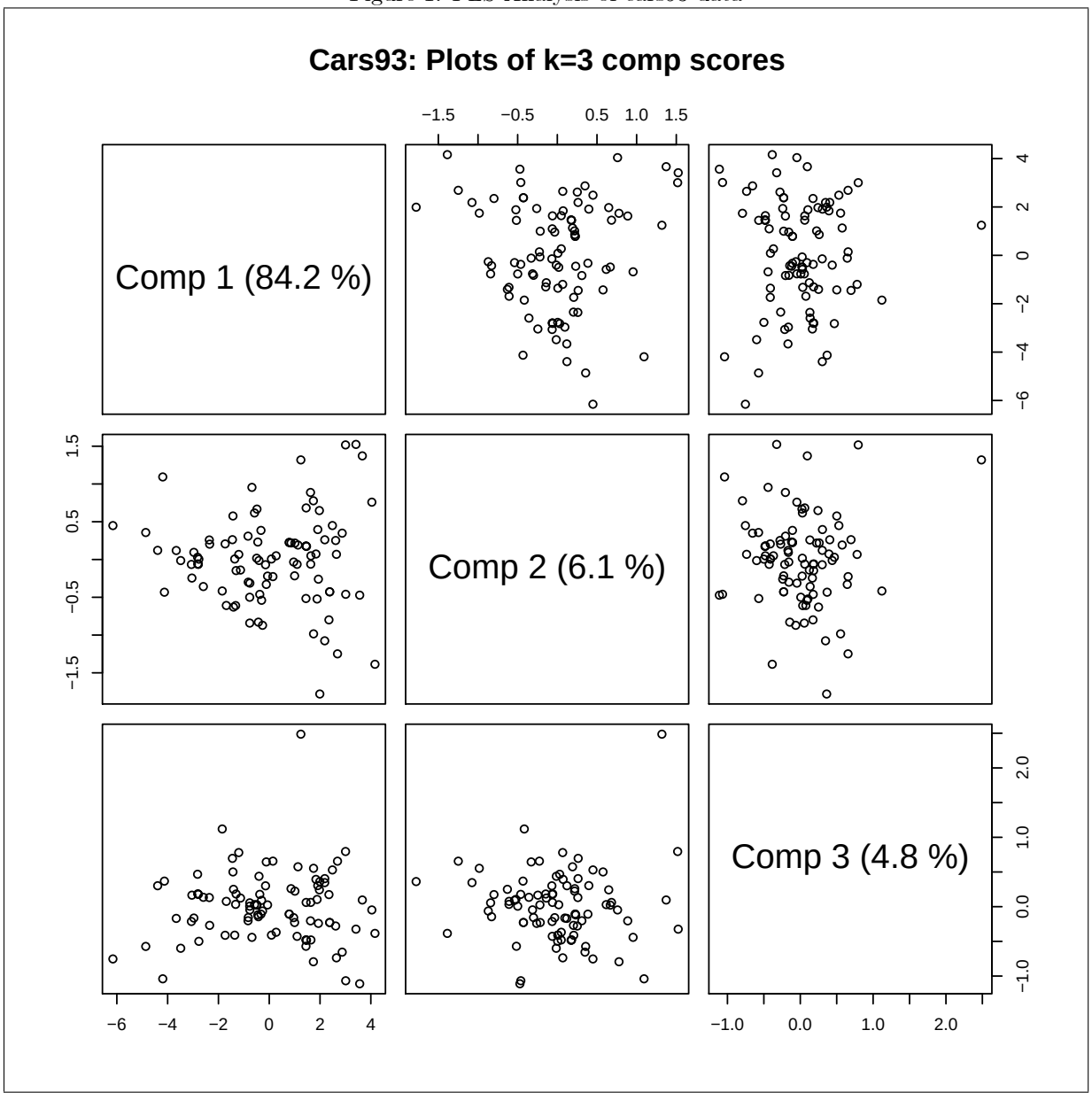

Figure 2: PLS Analysis of cars93 data