Module 6: Principal components regression

6.1 The SVD factorization

You're saying this only to make me go. [Ilsa Lund Laszlo, *Casablanca,* 1942]

In this module we turn to the Principal Components Regression (PCR) method, in which the PCA (Principal Components Analysis) method from the previous module is put to work in regression. To this end we consider the principal components of $\boldsymbol{X}^\top \boldsymbol{X}$, where $\boldsymbol{X}$ is a centered $n \times k$ data matrix.

There are several ways of finding the principal components of the $\boldsymbol{X}^\top \boldsymbol{X}$ matrix. One possibility is to apply the SVD method to $\boldsymbol{X}$, writing the reduced form of SVD as follows:

$$\boldsymbol{X} = \boldsymbol{U}\boldsymbol{D}\boldsymbol{P}^\top,$$

where $\boldsymbol{U}$ $(n \times r)$ and $\boldsymbol{P}$ $(k \times r)$ are orthogonal matrices corresponding to r singular values, in the notation of Module 5.

Let the scores matrix be defined by

$$\boldsymbol{T} = \boldsymbol{U}\boldsymbol{D},$$

a matrix with orthogonal, but not necessarily orthonormal columns. In fact

$$\begin{aligned} \boldsymbol{T}^\top \boldsymbol{T} &= \boldsymbol{D}\boldsymbol{U}^\top \boldsymbol{U}\boldsymbol{D} \\ &= \boldsymbol{D}^2 \\ &= \boldsymbol{\Lambda}_r, \end{aligned}$$

where $\boldsymbol{\Lambda}_r = \text{diag}\{\lambda_1, \ldots, \lambda_r\}$ contains the non-zero eigenvalues of $\boldsymbol{X}^\top \boldsymbol{X}$ in its diagonal. We assume that the eigenvalues are in decreasing order, $\lambda_1 \geq \cdots \geq \lambda_r > 0$.

Since

$$\boldsymbol{X} = \boldsymbol{T}\boldsymbol{P}^\top \tag{6.1}$$

we find that

$$\begin{aligned} \boldsymbol{X}^\top \boldsymbol{X} &= \boldsymbol{P}\boldsymbol{T}^\top \boldsymbol{T}\boldsymbol{P}^\top \\ &= \boldsymbol{P}\boldsymbol{\Lambda}_r \boldsymbol{P}^\top, \end{aligned}$$

which is the spectral decomposition for $\boldsymbol{X}^\top \boldsymbol{X}$, except that columns of $\boldsymbol{P}$ corresponding to zero eigenvalues have been left out. By using that $\boldsymbol{P}$ is orthogonal, we may also write (6.1) as follows:

$$\boldsymbol{T} = \boldsymbol{X}\boldsymbol{P}, \tag{6.2}$$

which follows by noting that $\boldsymbol{X}\boldsymbol{P} = \boldsymbol{T}\boldsymbol{P}^\top \boldsymbol{P} = \boldsymbol{T}$. Recall from Module 5 that the columns of $\boldsymbol{T}$ are known as *scores*, and those of $\boldsymbol{P}$ as *loadings*.

6.2 The NIPALS algorithm for PCA

Now we consider the NIPALS (Nonlinear Iterative Partial Least Squares) algorithm for finding the principal components of $\boldsymbol{X}^\top \boldsymbol{X}$. We want to find the first g principal components of $\boldsymbol{X}^\top \boldsymbol{X}$, starting with the largest eigenvalue λ_1 and down. g must be less than or equal to r.

6.2.1 Iterations

The NIPALS algorithm starts with the initialization $j = 1$ and $\boldsymbol{X}_1 = \boldsymbol{X}$. The algorithm then iterates through the following steps:

1. Choose $\boldsymbol{t}_j$ as any column of $\boldsymbol{X}_j$.
2. Let $\boldsymbol{p}_j = \boldsymbol{X}_j^\top \boldsymbol{t}_j \,/\, \left\| \boldsymbol{X}_j^\top \boldsymbol{t}_j \right\|$.
3. Let $\boldsymbol{t}_j = \boldsymbol{X}_j \boldsymbol{p}_j$.
4. If $\boldsymbol{t}_j$ is unchanged continue; otherwise return to Step 2.
5. Let $\boldsymbol{X}_{j+1} = \boldsymbol{X}_j - \boldsymbol{t}_j \boldsymbol{p}_j^\top$.
6. Stop if $j = g$; otherwise let $j = j + 1$ and return to Step 1.

Assume first that $g = r$, so we have found all the principal components. Now form the matrices $\boldsymbol{T}$ and $\boldsymbol{P}$ with columns $\boldsymbol{t}_j$ and $\boldsymbol{p}_j$, respectively; these matrices now satisfy (6.1).

It is possible to modify the NIPALS algorithm to take missing data into account, see Bro (1996), pp. 43–44.

6.2.2 Properties

Let us consider some properties of the NIPALS algorithm, which also help understand the PCA method.

That the NIPALS algorithm gives PCA may be seen as follows. Let $\lambda_j = \left\|\boldsymbol{X}^\top \boldsymbol{t}_j\right\|$ and write Step 2 as follows:

$$\boldsymbol{X}^\top \boldsymbol{t}_j = \lambda_j \boldsymbol{p}_j.$$

Now insert $\boldsymbol{t}_j = \boldsymbol{X}\boldsymbol{p}_j$ from Step 3, giving

$$\boldsymbol{X}^\top \boldsymbol{X}\boldsymbol{p}_j = \lambda_j \boldsymbol{p}_j. \tag{6.3}$$

This equation is satisfied upon convergence of the loop 2–4. This shows that λ_j and $\boldsymbol{p}_j$ are an eigenvalue and eigenvector of $\boldsymbol{X}^\top \boldsymbol{X}$, respectively. Also note that using $\boldsymbol{t}_j = \boldsymbol{X}\boldsymbol{p}_j$ and (6.3) we obtain

$$\begin{aligned} \boldsymbol{t}_j^\top \boldsymbol{t}_j &= \boldsymbol{p}_j^\top \boldsymbol{X}^\top \boldsymbol{X}\boldsymbol{p}_j \\ &= \boldsymbol{p}_j^\top \left(\boldsymbol{X}^\top \boldsymbol{X}\boldsymbol{p}_j\right) \\ &= \lambda_j \boldsymbol{p}_j^\top \boldsymbol{p}_j \\ &= \lambda_j, \end{aligned} \tag{6.4}$$

where in the last step we have used the fact that $\boldsymbol{p}_j$ is a unit vector (see Step 2).

After the first run through the loop 1–5, Step 5 with $j = 1$ gives that

$$\boldsymbol{X} = \boldsymbol{t}_1 \boldsymbol{p}_1^\top + \boldsymbol{X}_2. \tag{6.5}$$

Let us show that $\boldsymbol{t}_1$ and $\boldsymbol{X}_2 = \boldsymbol{X} - \boldsymbol{t}_1 \boldsymbol{p}_1^\top$ are orthogonal. In fact

$$\begin{aligned} \left(\boldsymbol{X} - \boldsymbol{t}_1 \boldsymbol{p}_1^\top\right)^\top \boldsymbol{t}_1 &= \boldsymbol{X}^\top \boldsymbol{t}_1 - \boldsymbol{p}_1 \, \boldsymbol{t}_1^\top \boldsymbol{t}_1 \\ &= \boldsymbol{X}^\top \boldsymbol{X}\boldsymbol{p}_1 - \boldsymbol{p}_1 \, \lambda_1 \\ &= \boldsymbol{0}, \end{aligned}$$

as seen from (6.3) with $j = 1$. Since $\boldsymbol{t}_2$ was initially picked as a column of $\boldsymbol{X}_2$, it is hence orthogonal to $\boldsymbol{t}_1$ and remains so to the end of the loop.

After the second run through the loop 1–5, we obtain

$$\boldsymbol{X} = \boldsymbol{t}_1 \boldsymbol{p}_1^\top + \boldsymbol{t}_2 \boldsymbol{p}_2^\top + \boldsymbol{X}_3.$$

After g runs through the loop, we similarly have

$$\boldsymbol{X} = \boldsymbol{t}_1 \boldsymbol{p}_1^\top + \boldsymbol{t}_2 \boldsymbol{p}_2^\top + \cdots + \boldsymbol{t}_g \boldsymbol{p}_g^\top + \boldsymbol{X}_{g+1}, \tag{6.6}$$

where $\boldsymbol{X}_{g+1} = \boldsymbol{0}$ in the case $g = r$ (compare with (6.1)).

6.3 How many components?

As the example suggests, the essence of the PCA method is to decompose the $\boldsymbol{X}$ matrix as in (6.6),

$$\begin{aligned} \boldsymbol{X} &= \boldsymbol{t}_1\boldsymbol{p}_1^\top + \boldsymbol{t}_2\boldsymbol{p}_2^\top + \cdots + \boldsymbol{t}_g\boldsymbol{p}_g^\top + \boldsymbol{X}_{g+1} \\ &= \boldsymbol{T}_g\boldsymbol{P}_g^\top + \boldsymbol{X}_{g+1}, \end{aligned} \tag{6.7}$$

say, where $\boldsymbol{T}_g$ and $\boldsymbol{P}_g$ contain the first g columns of $\boldsymbol{T}$ and $\boldsymbol{P}$, respectively. We want to choose g in such a way that $\boldsymbol{X}_{g+1}$ is small and represents only noise, while the term $\boldsymbol{T}_g\boldsymbol{P}_g^\top$ represents the salient features of $\boldsymbol{X}$. In order to accomplish this, g must be chosen in such a way that the $r-g$ terms that are ignored correspond to zero or negligible eigenvalues.

In order to help rationalize the choice of g, the relative size of the eigenvalues are expressed as a percentage of the sum of all eigenvalues,

$$\frac{\lambda_1}{\lambda_1 + \cdots + \lambda_r} \times 100$$

and this percentage is interpreted as the *percent variation explained* by the corresponding principal component. Often, the cumulated percentages are used, so that the percent variation explained by the first g components is

$$\frac{\lambda_1 + \cdots + \lambda_g}{\lambda_1 + \cdots + \lambda_r} \times 100$$

As a rule, g should be chosen so that at least about 80–90 percent of the variation is explained.

The justification for the above terminology is that the variance of the score vector $\boldsymbol{t}_j$ is

$$\begin{aligned} s^2(\boldsymbol{t}_j) &= \frac{1}{n-1}\|\boldsymbol{t}_j\|^2 \\ &= \frac{1}{n-1}\boldsymbol{t}_j^\top\boldsymbol{t}_j \\ &= \frac{1}{n-1}\lambda_j, \end{aligned}$$

so that λ_j is proportional to the variance of the corresponding score. In particular, all components with $\lambda_j = 0$ should be left out. Also, since the covariance matrix of $\boldsymbol{X}$ is

$$\begin{aligned} \boldsymbol{V}_X &= \frac{1}{n-1}\boldsymbol{X}^\top\boldsymbol{X} \\ &= \frac{1}{n-1}\sum_{j=1}^{k}\lambda_j\boldsymbol{p}_j\boldsymbol{p}_j^\top, \end{aligned}$$

we may interpret λ_j as the contribution of $\boldsymbol{t}_j$ to the total (co-)variance for $\boldsymbol{X}$. Note also that the sum of the eigenvalues is equal to

$$\mathrm{tr}\{\boldsymbol{X}^\top\boldsymbol{X}\} = (n-1)\left\{s^2(\boldsymbol{x}_1) + \cdots + s^2(\boldsymbol{x}_k)\right\},$$

which is interpreted as the total variance in $\boldsymbol{X}$.

6.4 Principal components regression

The basic idea in Principal Components Regression (PCR) is that after choosing a suitable value for g in (6.7), the important features of $\boldsymbol{X}$ have been retained by $\boldsymbol{T}_g$. We then perform the MLR with $\boldsymbol{T}_g$ in place of $\boldsymbol{X}$ for an $n \times m$ calibration data matrix $\boldsymbol{Y}$,

$$\boldsymbol{Y} = \boldsymbol{T}_g \boldsymbol{C} + \boldsymbol{F}. \tag{6.8}$$

The least squares method then gives

$$\widehat{\boldsymbol{C}} = \left(\boldsymbol{T}_g^\top \boldsymbol{T}_g\right)^{-1} \boldsymbol{T}_g^\top \boldsymbol{Y},$$

where $\boldsymbol{T}_g^\top \boldsymbol{T}_g$, being diagonal, is easy to invert. The fact that we have left out the loadings matrix $\boldsymbol{P}_g$ in (6.8) is of no consequence for prediction, because the scores $\boldsymbol{t}_j$ are linear combinations of the columns of $\boldsymbol{X}$, and the PCR method amounts to singling out those linear combinations that are best for predicting $\boldsymbol{Y}$.

6.4.1 Prediction

For prediction with PCR, it is necessary to turn to $\boldsymbol{X}$ again, and using (6.2) we may write the regression equation as follows:

$$\begin{aligned} \boldsymbol{Y} &= \boldsymbol{T}_g \boldsymbol{C} + \boldsymbol{F}. \\ &= \boldsymbol{X} \boldsymbol{P}_g \boldsymbol{C} + \boldsymbol{F}. \end{aligned} \tag{6.9}$$

Consider a new sample spectrum $\boldsymbol{z}$ and predicted value $\widehat{\boldsymbol{y}}$ (both uncentered), and let $\overline{\boldsymbol{x}}$ and $\overline{\boldsymbol{y}}$ be the calibration sample averages. Then the prediction takes the form

$$\widehat{\boldsymbol{y}} = \overline{\boldsymbol{y}} + (\boldsymbol{z} - \overline{\boldsymbol{x}})\, \boldsymbol{P}_g \widehat{\boldsymbol{C}}.$$

The matrix $\boldsymbol{P}_g \widehat{\boldsymbol{C}}$ is called the *regression matrix*, and may be compared with the $\widehat{\boldsymbol{B}}$ matrix of MLR.

6.4.2 PCR and MLR

Just as in MLR, the same prediction would be obtained if the columns of $\boldsymbol{Y}$ were considered separately. The fact that $\boldsymbol{P}_g$ appears in the PCR Equation (6.9) may be seen as compensating for the fact that we left out $\boldsymbol{P}_g$ in (6.8). When comparing with MLR, the role of the $\boldsymbol{B}$ matrix is now played by $\boldsymbol{P}_g \boldsymbol{C}$.

In the case where $\boldsymbol{X}$ has rank k and $g = k$, the two methods will give identical results. When $g < k$, and still $\boldsymbol{X}$ has rank k, the results of PCR may differ somewhat from those of MLR, depending on the number and sizes of the components left out.

However, PCR has some major advantages over MLR, in that $\boldsymbol{X}$ may be singular, and the case $k \geq n$ may be dealt with. Note, however, that in the latter case $\lambda_n = \cdots = \lambda_k = 0$, and so at most $g = n - 1$ components may be included.