DEPARTMENT OF STATISTICS, THE UNIVERSITY OF BRITISH COLUMBIA

STAT406

PCA + alternating regression

Matías Salibián Barrera

PCA + ALTERNATING REGRESSION

Let $\mathbf{X}_1, \ldots, \mathbf{X}_n \in \mathbb{R}^p$ be the observations for which we want to compute the corresponding PCA. Without loss of generality we can always assume that

$$\frac{1}{n} \sum_{i=1}^{n} \mathbf{X}_i \; = \; \mathbf{0} \, ,$$

so that the sample covariance matrix $\mathbf{S}_n$ is

$$\mathbf{S}_n \; = \; \frac{1}{n-1} \sum_{i=1}^{n} \mathbf{X}_i \, \mathbf{X}_i^\top \, .$$

We saw in class that if $\mathbf{B} \in \mathbb{R}^{p \times k}$ has in its columns the eigenvectors of $\mathbf{S}_n$ associated with its k largest eigenvalues, then

$$\frac{1}{n} \sum_{i=1}^{n} \| \mathbf{X}_i - P(\mathbf{L}_{\mathbf{B}}, \mathbf{X}_i) \|^2 \; \le \; \frac{1}{n} \sum_{i=1}^{n} \| \mathbf{X}_i - P(\mathbf{L}, \mathbf{X}_i) \|^2 \, ,$$

for any k-dimensional linear subspace $\mathbf{L} \subset \mathbb{R}^p$ where $P(\mathbf{L}, \mathbf{X})$ denotes the orthogonal projection of $\mathbf{X}$ onto the subspace $\mathbf{L}$, $P(\mathbf{L}_{\mathbf{B}}, \mathbf{X}) = \mathbf{B}\mathbf{B}^\top \mathbf{X}$ (whenever $\mathbf{B}$ is chosen so that $\mathbf{B}^\top \mathbf{B} = \mathbf{I}$) and $\mathbf{L}_{\mathbf{B}}$ denotes the subspace spanned by the columns of $\mathbf{B}$.

We will show now that, instead of finding the spectral decomposition of $\mathbf{S}_n$, principal components can also be computed via a sequence of "alternating least squares" problems. To fix ideas we will consider the case $k = 1$, but the method is trivially extended to arbitrary values of k.

When $k = 1$ we need to solve the following problem

$$\min_{\|\mathbf{a}\|=1, \mathbf{v}\in\mathbb{R}^n} \sum_{i=1}^{n} \|\mathbf{X}_i - \mathbf{a}\, v_i\|^2 \tag{0.1}$$

where $\mathbf{v} = (v_1, \ldots v_n)^\top$ (in general, for any k we have

$$\min_{\mathbf{A}^\top\mathbf{A}=\mathbf{I}, \mathbf{v}_1,\ldots,\mathbf{v}_n\in\mathbb{R}^k} \sum_{i=1}^{n} \|\mathbf{X}_i - \mathbf{A}\mathbf{v}_i\|^2)$$

The objective function in (0.1) can also be written as

$$\sum_{i=1}^{n} \sum_{j=1}^{p} \left(\mathbf{X}_{i,j} - a_j\, v_i\right)^2 , \tag{0.2}$$

and hence, for a given vector $\mathbf{a}$, the minimizing values of $v_1, \ldots, v_n$ in (0.2) can be found solving n separate least squares problems:

$$v_\ell = \arg\min_{d\in\mathbb{R}} \sum_{j=1}^{p} \left(\mathbf{X}_{\ell,j} - a_j\, d\right)^2 , \qquad \ell = 1, \ldots, n .$$

Similarly, for a given set $v_1, \ldots, v_n$ the entries of $\mathbf{a}$ can be found solving p separate least squares problems:

$$a_r = \arg\min_{d\in\mathbb{R}} \sum_{i=1}^{n} \left(\mathbf{X}_{i,r} - d\, v_i\right)^2 , \qquad r = 1, \ldots, p .$$

We can then set $\mathbf{a} \leftarrow \mathbf{a}/\|\mathbf{a}\|$ and iterate to find new v's, then a new $\mathbf{a}$, etc.
Below is a simple implementation of this algorithm in `R`, and two simple examples.

```
alter.pca.k1 <- function(x, max.it = 500, eps=1e-10) {
      n2 <- function(a) sum(a^2)
      p <- dim(x)[2]
      x <- scale(x, scale=FALSE)
      it <- 0
      old.a <- c(1, rep(0, p-1))
      err <- 10*eps
      while( ((it <- it + 1) < max.it) & (abs(err) > eps) ) {
             b <- as.vector( x %*% old.a ) / n2(old.a)
             a <- as.vector( t(x) %*% b ) / n2(b)
             a <- a / sqrt(n2(a))
             err <- sqrt(n2(a - old.a))
             old.a <- a
      }
      conv <- (it < max.it)
      return(list(a=a, b=b, conv=conv))
}
```

and

```
> set.seed(678)
> n <- 20
> p <- 5
> x <- matrix(rt(n*p, df=2), n, p)
>
> alter.pca.k1(x)$a
[1]  0.04594657  0.00282812  0.01926534  0.02993064 -0.99830552
> svd(cov(x))$u[,1]
[1] -0.04594657 -0.00282812 -0.01926534 -0.02993064  0.99830552
>
> n <- 2000
> p <- 500
> x <- matrix(rt(n*p, df=2), n, p)
>
> system.time( tmp <- alter.pca.k1(x) )
   user  system elapsed
   0.43    0.09    0.53
> a1 <- tmp$a
> system.time( e1 <- svd(cov(x))$u[,1] )
   user  system elapsed
   2.64    0.02    2.62
> a1 <- a1 * sign(e1[1]*a1[1])
> summary( abs(e1 - a1) )
     Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
2.000e-18 5.677e-15 1.198e-14 4.538e-14 2.262e-14 1.102e-11
```