Olympic Medal Ranking and Principal Component Analysis (PCA)

Let X be an m by k spreadsheet/matrix/data set. For instance, when $k = 3$ the three columns may be the numbers of gold, silver and bronze medals, respectively. Each row represents a country and there are m countries. An interesting question is how to rank those countries based on medal counts.

This problem has no unique answer. The ranking method used by NBC TV network is looking at the total number of medals for each country, which essentially treats gold and bronze medals equally. This note introduces a technique called Principal Component Analysis (PCA) that generates a specific linear combination, or loosely speaking, a weighted average of medals. For instance, one weighting vector may be $(3, 2, 1)$, which implies a gold medal weighs as much as three bronze medals. Next, a country can be assigned a score or principal component value

$$score\ or\ PC\ value = 3 * gold + 2 * silver + 1 * bronze$$

The country with the highest score may be ranked as the first.

To see how PCA works, let c be a k by one column vector. Usually we normalize c so that $c'c = 1$. The matrix product Xc is a linear combination of columns of X.

$$Xc = \begin{pmatrix} x_1 & x_2 & \dots & x_k \end{pmatrix} \begin{pmatrix} c_1 \\ c_2 \\ \vdots \\ c_k \end{pmatrix} = c_1x_1 + c_2x_2 + \dots c_kx_k$$

where x_i is the i-th column of X, and c_i is the i-th entity of c. Note Xc is m by one, and its entity is the score for each country.

Intuitively we hope Xc preserves the information in X to the largest extent possible. Thus the goal of PCA is to find an optimal c that maximizes the variation of Xc :

$$\max\ (Xc)'(Xc) = \max\ c'X'Xc \quad (subject\ to\ c'c = 1) \tag{1}$$

Note that the k by k square matrix $X'X$ is symmetric. As a result, we can apply a special spectral decomposition

$$X'X = V\Lambda V^{-1} = V\Lambda V' \tag{2}$$

where Λ is the diagonal matrix of eigenvalues, and the columns of V are normalized eigenvectors (i.e., $v_i'v_i = 1, \forall i = 1, \ldots, k$). We can show $V^{-1} = V'$ because the eigenvectors of a symmetric matrix are orthogonal $v_i'v_j = 0, (\forall i \neq j)$. . It follows that

$$c'X'Xc = c'V\Lambda V'c = (c'v_1, \ldots c'v_k) \begin{pmatrix} \lambda_1 & ... & 0 \\ 0 & ... & 0 \\ 0 & ... & \lambda_k \end{pmatrix} \begin{pmatrix} v_1'c \\ ... \\ v_k'c \end{pmatrix} = \lambda_1 c'v_1v_1'c + ... + \lambda_k c'v_kv_k'c \tag{3}$$

Suppose the eigenvalues are sorted in descending order:

$$\lambda_1 \geq \lambda_2 ... \geq \lambda_k$$

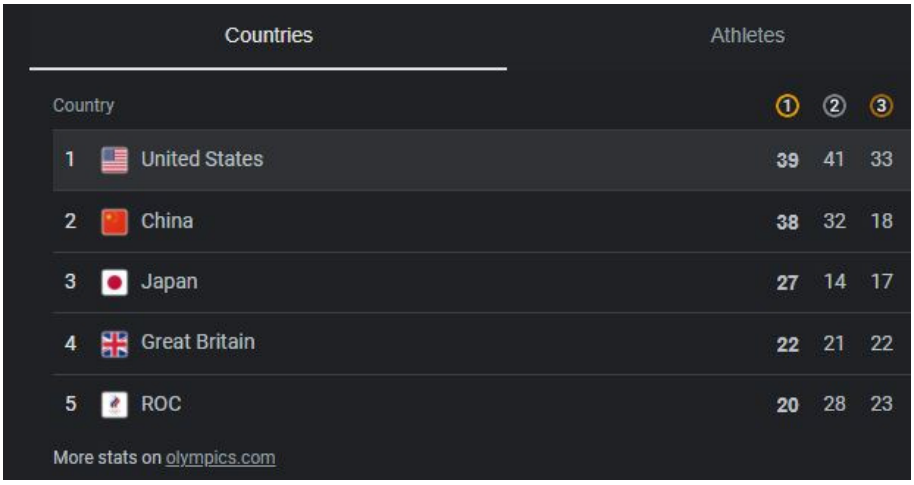
Because each $c'v_iv_i'c = (c'v_i)^2$ is a square term and non-negative, it follows that

$$\lambda_1 c'v_1v_1'c + ... + \lambda_k c'v_kv_k'c \leq \lambda_1(c'v_1v_1'c + ... + c'v_kv_k'c') = \lambda_1 c'VV'c = \lambda_1 c'VV^{-1}c = \lambda_1 c'c = \lambda_1 \tag{4}$$

The equality in (4) holds only when $c = v_1$.

To summarize, the first principal component is given by Xv_1, where the weighting vector v_1 is the eigenvector belonging to the largest eigenvalue λ_1 of $X'X$. The variation of the first principal component is λ_1.

For instance, consider the medal counts for US, China, Japan, UK and Russia in 2020 Tokyo summer Olympics game.



The R codes and result for obtaining the first principal component and ranking those countries are

```
> # PCA applied to medal counts in 2020 Summer Olympics Game
> X = matrix(c(39,41,33,38,32,18,27,14,17,22,21,22,20,28,23),nrow=5,byrow=T)
> res = eigen(t(X)%*%X)
> res
eigen() decomposition
$values
[1] 11182.37885   167.57303    69.04812

$vectors
          [,1]       [,2]       [,3]
[1,] 0.6329206  0.7611479  0.1416525
[2,] 0.6031903 -0.3700835 -0.7065407
[3,] 0.4853587 -0.5326275  0.6933505
```

Thus the weight vector c is $(0.6329206, 0.6031903, 0.4853587)$. Those weights are data-driven. Using this vector we can derive the vector of scores for each countries Xc as

```
> # Ranking by first principal component
> PC1 = X%*%res$vectors[,1]
> PC1
         [,1]
```

```
[1,] 65.43154
[2,] 52.08953
[3,] 33.78462
[4,] 37.26914
[5,] 40.71099
>
> # US score
> X[1,]%*%res$vectors[,1]
[,1]
[1,] 65.43154

> # variation of first principal component
> t(PC1)%*%PC1
[,1]
[1,] 11182.38
>
> # 1st and 2nd principal components are orthogonal
> PC2 = X%*%res$vectors[,2]
> t(PC1)%*%PC2
[,1]
[1,] 1.421085e-12
```

So US has a score or PC value of 65.43154, followed by China 52.08953 and Russia 40.71099. As expected, the variation of PC1 is the same as the largest eigenvalue 11182.37885. By contrast Russia is ranked as number 5 based on the number of gold medals.

```
> # Ranking by total count
> X%*%matrix(c(1,1,1),nrow=3)
     [,1]
[1,]  113
[2,]   88
[3,]   58
[4,]   65
[5,]   71

> # Ranking by gold count
```

```
> X%*%matrix(c(1,0,0),nrow=3)
     [,1]
[1,]   39
[2,]   38
[3,]   27
[4,]   22
[5,]   20
```

The R canned command prcomp can be used to obtain the PCs.

```
> X.pca = prcomp(X, center = F, scale= F)
> X.pca$x
           PC1       PC2        PC3
[1,] -65.43154 -3.065364  0.5631533
[2,] -52.08953  7.493653  4.7461968
[3,] -33.78462  6.315156 -5.7200068
[4,] -37.26914 -2.744305 -3.5327119
[5,] -40.71099 -7.389813  1.0030274
```

Other applications of PCA may be ranking the safety of cities based on numbers of murders, rapes, and robberies, or ranking schools based on several metrics. In short PCA is a data-reduction method that reduces an m by k matrix into a m by one vector.