Dimension reduction 2: Principal component analysis (continued)

Ryan Tibshirani
Data Mining: 36-462/36-662

February 7 2012

Optional reading: ISL 10.2, ESL 14.5
Reminder: projections onto unit vectors

The projection of $x \in \mathbb{R}^n$ onto a unit vector $v \in \mathbb{R}^n$ is given by $(x^Tv)v \in \mathbb{R}^n$. The score from this projection is $x^Tv \in \mathbb{R}^n$.

The projections of the rows of $X \in \mathbb{R}^{n \times p}$ onto unit vector $v \in \mathbb{R}^p$ are given by rows of $Xvv^T \in \mathbb{R}^{n \times p}$. The scores are the entries of $Xv \in \mathbb{R}^n$.

Example from last time: $X \in \mathbb{R}^{50 \times 2}$, $v_1, v_2 \in \mathbb{R}^2$. 
Reminder: first principal component direction and score

Recall: given data matrix \( X \in \mathbb{R}^{n \times p} \) (\( n \) observations, \( p \) features), with centered its columns removed column-wise means.

The first principal component direction of \( X \) is the unit vector \( v_1 \in \mathbb{R}^p \) such that \( Xv_1 \) has the highest sample variance compared to all other unit vectors, i.e.,

\[
v_1 = \arg \max_{\|v\|_2=1} (Xv)^T(Xv) \quad \text{sample variance of } Xv
\]

The vector \( Xv_1 \in \mathbb{R}^n \) is called the first principal component score of \( X \), and \( u_1 = (Xv_1)/d_1 \in \mathbb{R}^n \) is the normalized first principal component score, where \( d_1 = \sqrt{(Xv_1)^T(Xv_1)} \). The quantity \( d_1^2/n \) is the amount of variance explained by \( v_1 \).

The entries of \( Xv_1 = d_1 u_1 \) are the scores from projecting \( X \) onto \( v_1 \), and the rows of \( Xv_1v_1^T = d_1 u_1 v_1^T \) are the projected vectors.
Example: first principal component direction and score

Example from last time: $X \in \mathbb{R}^{50 \times 2}$

![Diagram of first principal component direction]

![Diagram of first principal component score]

$X v_1 = d_1 u_1$
Reminder: projections onto orthonormal sets

Vectors $v_1, \ldots v_k \in \mathbb{R}^p$ are called orthonormal if each pair $v_i, v_j$ is orthogonal, $v_i^T v_j = 0$, and each $v_j$ has unit norm.

The projection of $x \in \mathbb{R}^p$ onto an orthonormal set $v_1, \ldots v_k \in \mathbb{R}^p$ is $\sum_{i=j}^k (x^T v_j) v_j \in \mathbb{R}^p$. The score along $v_j$ is $x^T v_j$. Project $v_i$ onto $v_1, \ldots v_k$?

The projections of rows of $X \in \mathbb{R}^{n \times p}$ onto orthonormal columns of $V \in \mathbb{R}^{p \times k}$ are given by rows of $XVV^T \in \mathbb{R}^{n \times p}$. The scores are columns of $XV \in \mathbb{R}^{n \times k}$, i.e., the scores along $v_j$ are given by the entries of $Xv_j \in \mathbb{R}^{n}$.

$\sqrt{V = [v_1, \ldots , v_k]} \in \mathbb{R}^{p \times k}$

Example from last time: $X \in \mathbb{R}^{2000 \times 3}$.
Further principal component directions and scores

Given first \( k - 1 \) principal component directions \( v_1, \ldots, v_{k-1} \in \mathbb{R}^p \) (these are orthonormal), the \( k \)th principal component direction \( \mathbf{v}_k \in \mathbb{R}^p \) is the unit vector such that \( \mathbf{X} \mathbf{v}_k \) has the highest sample variance over all directions orthogonal to \( \mathbf{v}_1, \ldots, \mathbf{v}_{k-1} \), i.e.,

\[
\mathbf{v}_k = \arg\max_{\|\mathbf{v}\|_2 = 1} (\mathbf{Xv})^T(\mathbf{Xv}) \quad \text{sample}
\]

\[
\mathbf{v}_k^T \mathbf{v}_j = 0, \ j = 1, \ldots, k-1
\]

The vector \( \mathbf{X} \mathbf{v}_k \in \mathbb{R}^n \) is called the \( k \)th principal component score of \( \mathbf{X} \), and \( \mathbf{u}_k = (\mathbf{X} \mathbf{v}_k)/d_k \in \mathbb{R}^n \) is the normalized \( k \)th principal component score, where \( d_k = \sqrt{(\mathbf{Xv}_k)^T(\mathbf{Xv}_k)} \). The quantity \( d_k^2/n \) is the amount of variance explained by \( \mathbf{v}_k \).

The entries of \( \mathbf{X} \mathbf{v}_k = d_k \mathbf{u}_k \) are the scores from projecting \( \mathbf{X} \) onto \( \mathbf{v}_k \), and the rows of \( \mathbf{X} \mathbf{v}_k \mathbf{v}_k^T = d_k \mathbf{u}_k \mathbf{v}_k^T \) are the projected vectors.
Example: second principal component direction and score

Same example as before: $X \in \mathbb{R}^{50 \times 2}$
Example: principal component analysis in $\mathbb{R}^3$

Example from last time: $X \in \mathbb{R}^{2000 \times 3}$. Shown are the first three principal component directions $v_1, v_2, v_3 \in \mathbb{R}^3$, and the scores from projecting onto the first two directions.
Example: projecting onto principal component directions

Same example. What happens if replace $X$ by its projection onto $v_1$? Onto $v_1, v_2$? Onto $v_1, v_2, v_3$?

The third plot looks exactly the same as the original data. Is this a coincidence? No! (Why not?)

$$V_k = [v_1 \ldots v_k] \in \mathbb{R}^{p \times k}$$

Projection onto $k$ p.c. directions:

$$XV_kV_k^T = X$$

$$= I$$

$$V_pV_p^T = I$$

Fact: $V_p$ is square
Example: principal component analysis in $\mathbb{R}^{12}$

Example: data from 2012 Cadillac Championship, professional golf tournament. Here $X \in \mathbb{R}^{72 \times 12}$, 72 golfers with 12 features:

eagles
birdies
pars
bogeys
double.bogeys
driving.accuracy
driving.distance
strokes.gained.putting
putts.per.round
putts.per.gir
greens.in.reg
sand.saves

These are average measurements over the 4 day tournament
The first two principal component directions $v_1, v_2 \in \mathbb{R}^{12}$ are:

<table>
<thead>
<tr>
<th></th>
<th>PC1</th>
<th>PC2</th>
</tr>
</thead>
<tbody>
<tr>
<td>eagles</td>
<td>-0.139</td>
<td>0.208</td>
</tr>
<tr>
<td>birdies</td>
<td>-0.463</td>
<td>0.185</td>
</tr>
<tr>
<td>pars</td>
<td>0.168</td>
<td>-0.582</td>
</tr>
<tr>
<td>bogeys</td>
<td>0.303</td>
<td>0.420</td>
</tr>
<tr>
<td>double.bogeys</td>
<td>0.062</td>
<td>0.181</td>
</tr>
<tr>
<td>driving.accuracy</td>
<td>-0.128</td>
<td>-0.241</td>
</tr>
<tr>
<td>driving.distance</td>
<td>-0.036</td>
<td>0.430</td>
</tr>
<tr>
<td>strokes.gained.putting</td>
<td>-0.438</td>
<td>-0.091</td>
</tr>
<tr>
<td>putts.per.round</td>
<td>0.325</td>
<td>0.026</td>
</tr>
<tr>
<td>putts.per.gir</td>
<td>0.491</td>
<td>-0.158</td>
</tr>
<tr>
<td>greens.in.reg</td>
<td>-0.171</td>
<td>-0.099</td>
</tr>
<tr>
<td>sand.saves</td>
<td>-0.238</td>
<td>-0.296</td>
</tr>
</tbody>
</table>

For each direction, look at the signs ... what do you notice here?
Scores from projecting onto $\mathbf{v}_1, \mathbf{v}_2 \in \mathbb{R}^{12}$:

First two principal component scores

- Watson 2
- Garcia T60
- McIlroy 3
- Colgate T35
- Takayama T35
- Simpson T35
- Watney T17
- Kaymer T38
- Larrazabal T66
- Noren 69
- Singh T66
- Kruger T57
- Oosthuizen T60
- Cabrera Bello 65
- Jacobson 88
- Hiratsuka 70
- Bae 71
- Bjerregaard 71
- Scott 73
- Langer T17
- Scott T35
- Byrd T35
- Kuchar T8
- Stricker T8
- Kim T51
- Donald T6
- Bjorn T35
- Wagner T13
- Padley T49
- Snedeker T45
- Fowler T45
- McIlroy T38
- Proctor T35
- Rose 1
- Vanderpool T13
- Mollnari T13
Dimension reduction via the principal component scores

As we’ve seen in the examples, dimension reduction via principal component analysis can be achieved by taking the first $k$ principal component scores $X v_1, \ldots, X v_k \in \mathbb{R}^n$.\[ X V_k \in \mathbb{R}^{n \times k} \]

We can think of $X v_1, \ldots, X v_k$ as our new feature vectors, which is a big savings if $k \ll p$ (e.g. $k = 2$ or $3$)

An important question: how good are these features at capturing the structure of our old features? Broken up into two questions:

1. How good are they, for a fixed $k$?
2. What exactly do we gain by increasing $k$?

Recall that the second question can be addressed by looking at the proportion of variance explained as a function of $k$
Example: proportion of variance explained

For the golf data set:

\[ PVE(k) = \frac{\sum_{i=1}^{k} d_i^2}{\sum_{i=1}^{\infty} d_i^2} \]

Proportion of variance explained

Number of component directions
Approximation by projection

As for the first question, think about approximating $X$ by $XV_k V_k^T$, the projection of $X$ onto the first $k$ principal component directions.

An important alternate characterization of the principal component directions: given centered $X \in \mathbb{R}^{n \times p}$, if $V_k = [v_1 \ldots v_k] \in \mathbb{R}^{p \times k}$ is the matrix whose columns contain the first $k$ principal component directions of $X$, then

$$V_k = \begin{bmatrix} v_1 & \cdots & v_k \end{bmatrix}$$

$$XV_k V_k^T = \underset{\text{rank}(A)=k}{\text{argmin}} \| X - A \|_F^2 = \underset{\text{rank}(A)=k}{\text{argmin}} \sum_{i=1}^{n} \sum_{j=1}^{p} (X_{ij} - A_{ij})^2$$

In other words, $XV_k V_k^T$ is the best rank $k$ approximation to $X$.

(Aside: the above problem is nonconvex, and would be very hard to solve in general!)
Scaling the features

We always center the columns of $X$ before computing the principal component directions. Another common pre-processing step is to scale the columns of $X$, i.e., to divide each feature by its sample variance, so that each feature in our new $X$ has a sample variance of one

Why? Look at the principal component of golf data, without scaling:

<table>
<thead>
<tr>
<th>Feature</th>
<th>Value</th>
<th>Feature</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>eagles</td>
<td>-0.001</td>
<td>birdies</td>
<td>0.007</td>
</tr>
<tr>
<td>bogeys</td>
<td>-0.015</td>
<td>double.bogeys</td>
<td>0.002</td>
</tr>
<tr>
<td>driving.distance</td>
<td>-0.122</td>
<td>strokes.gained.putting</td>
<td>0.015</td>
</tr>
<tr>
<td>putts.per.gir</td>
<td>-0.001</td>
<td>greens.in.reg</td>
<td>-0.004</td>
</tr>
<tr>
<td>driving.accuracy</td>
<td>0.007</td>
<td>putts.per.round</td>
<td>-0.016</td>
</tr>
<tr>
<td>sand.saves</td>
<td>0.990</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
And note that the golf features have sample variance:

<table>
<thead>
<tr>
<th>Eagles</th>
<th>Birdies</th>
<th>Pars</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.033</td>
<td>0.685</td>
<td>0.965</td>
</tr>
<tr>
<td>Bogey</td>
<td>Double Bogey</td>
<td>Driving Accuracy</td>
</tr>
<tr>
<td>0.561</td>
<td>0.095</td>
<td>59.837</td>
</tr>
<tr>
<td>Driving Distance</td>
<td>Strokes Gained Putting</td>
<td>Putts Per Round</td>
</tr>
<tr>
<td>100.702</td>
<td>0.739</td>
<td>1.263</td>
</tr>
<tr>
<td>Putts Per GIR</td>
<td>Greens In Reg</td>
<td>Sand Saves</td>
</tr>
<tr>
<td>0.006</td>
<td>54.162</td>
<td>423.474</td>
</tr>
</tbody>
</table>

But sometimes scaling is not appropriate (e.g., when you know the variables are all on the same scale to begin with)
Computing principal component directions

There are various ways to compute principal component directions. We'll consider computation via the singular value decomposition (SVD) of \( X \):

\[
X = U D V^T \quad \leftarrow \quad n \times p \quad n \times p \quad p \times p \quad p \times p
\]

Here \( D = \text{diag}(d_1, \ldots, d_p) \) is diagonal with \( d_1 \geq \ldots \geq d_p \geq 0 \), and \( U, V \) both have orthonormal columns. This gives us everything:

- columns of \( V \), \( v_1, \ldots, v_p \in \mathbb{R}^p \), are the principal component directions

- columns of \( U \), \( u_1, \ldots, u_p \in \mathbb{R}^n \), are the normalized principal component scores

- squaring the \( j \)th diagonal element of \( D \) and dividing by \( n \),
  \[ d_j^2 / n \]
  gives the variance explained by \( v_j \)

(Don't forget that we must first center the columns of \( X \)!)
Note that

\[ XV = UDV^TV = UD \]

because \( V^TV = I \). This means that

\[ Xv_j = d_ju_j, \quad j = 1, \ldots, p \]

two ways of representing principal component scores, as expected.

Note also that

\[ X^TX = VD^2V^T \]

and so \( v_1, \ldots v_p \) are eigenvectors of \( X^TX \). (Check?)
Recap: principal component analysis

We reviewed the principal component directions $v_1, \ldots v_p \in \mathbb{R}^p$
and scores $Xv_1, \ldots Xv_p \in \mathbb{R}^n$ of a centered matrix $X \in \mathbb{R}^{n \times p}$

The matrix $XV_k \in \mathbb{R}^{n \times k}$ (where $V_k$ contains the first $k$ principal
component directions) can be thought of as a reduced dimension version of $X$

The matrix $XV_kV_k^T \in \mathbb{R}^{n \times p}$ (projecting $X$ onto its first $k$ principal
component directions) can be thought of as an approximation to $X$ in the original feature space. For a fixed $k$ this approximation is the best we can do across rank $k$ matrices (measured by Frobenius distance to $X$)

Computation can be done via the singular value decomposition

Scaling the variables can crucial, especially if they are on different numeric scales
Next time: nonlinear dimension reduction

The famous "swiss roll" data set ...

(From Roweis et al. (2000), "Nonlinear dimensionality reduction by locally linear embedding")