and 11
Introduction to Machine Learning
Fall Semester, 2013
Recitation 10 and 11: December 22, 29
Lecturer: Mariano Schain
Scribe: ym

### 10.1 Project

The project was described. More details in the web site.

### 10.2 PCA

### 10.2.1 SVD properties

We have a data matrix $X$ that has all the $m$ data points as columns. The dimensions of $X$ is $N \times m$, where $N$ is the number of attributes of a data point. We assume that the average of $X$ (column-wise. that is, for each attribute) is $0^{1}$. The rank of the matrix $X$ is $r \leq \min \{m, N\}$.

$$
X=\left(\begin{array}{ccc}
\vdots & & \vdots \\
x_{1} & \cdots & x_{m} \\
\vdots & & \vdots
\end{array}\right)
$$

The SVD of $X$ has the form

$$
X=U_{x} \Sigma_{x} V_{x}^{t}
$$

where $\Sigma$ is a diagonal matrix,

$$
\Sigma_{x}=\operatorname{diag}\left(\sigma_{1} \geq \sigma_{2} \geq \cdots \geq \sigma_{r}>0\right)
$$

$U_{x}$ is an $N \times r$ orthonormal matrix, i.e., $u_{i}^{t} u_{j}=\delta_{i, j}$ where $\delta_{i, j}=1$ if $i=j$ and $\delta_{i, j}=0$ if $i \neq j$.

$$
U_{x}=\left(\begin{array}{ccc}
\vdots & & \vdots \\
u_{1} & \cdots & u_{r} \\
\vdots & & \vdots
\end{array}\right)
$$

[^0]Similarly, $V_{x}$ is an $r \times m$ orthonormal matrix,

$$
V_{x}=\left(\begin{array}{ccc}
\cdots & v_{1}^{t} & \cdots \\
& \vdots & \\
\cdots & v_{r}^{t} & \cdots
\end{array}\right)
$$

Another property of the SVD is that: $u_{j}^{t} X=\sigma_{j} v_{j}^{t}$ and $X v_{j}=\sigma_{j} u_{j}$.
Consider now the covariance matrix of $X$ :

$$
\begin{aligned}
X X^{t} & =\left(U_{x} \Sigma_{x} V_{x}^{t}\right)\left(U_{x} \Sigma_{x} V_{x}^{t}\right)^{t} \\
& =U_{x} \Sigma_{x} V_{x}^{t} V_{x} \Sigma_{x} U_{x}^{t} \\
& =U_{x} \Sigma_{x}^{2} U_{x}^{t}
\end{aligned}
$$

This gives an alternative way to get the elements of the SVD of $X$ - by finding the eigenvectors $U_{x}$ and corresponding eigenvalues $\Sigma_{x}$ of the symmetric matrix $X X^{t}$ ( $V_{x}$ may then be computed using the relation $V_{x}=X^{t} U_{x} \Sigma^{-1}$. Also, a similar derivation as above shows that $V_{x}$ are the eigenvectors of $X^{t} X$ with the same eigenvalues $\Sigma_{x}^{2}$ ).

### 10.2.2 Mapping the data to a lower dimension feature space

Let $U_{k}$ be the orthonormal matrix $U$ with rank $k \leq r$ :

$$
U_{k}=\left(\begin{array}{ccc}
\vdots & & \vdots \\
u_{1} & \cdots & u_{k} \\
\vdots & & \vdots
\end{array}\right)
$$

We define $P_{k}=U_{k} U_{k}^{t}$. We like to see what happens when we apply $P_{k}$ to a data point $x_{i}$ :

$$
P_{k} x_{i}=U_{k} U_{k}^{t} x_{i}=U_{k}\left(\begin{array}{c}
u_{1}^{t} x_{i} \\
\vdots \\
u_{k}^{t} x_{i}
\end{array}\right)=\sum_{j=1}^{k}\left(u_{j}^{t} x_{i}\right) u_{j}
$$

so it turns out that the result is a linear combination of the columns of $U_{k}$. Therefore $P_{k}$ can be viewed as a projection of the data to a $k$ dimension subspace of $\mathbb{R}^{N}$. It is easy to verify the following two properties: (1) $P_{k}^{t}=P_{k}$, and (2) $P_{k}^{2}=P_{k}$.

This can be viewed as a mapping of the $N$ dimensional data to a $k$ dimensional space (still embedded in $\mathbb{R}^{N}$ ), where the basis elements (the axis) are the $u_{i}$. With that interpretation, the data point $x_{i}$ is mapped to the vector $\left(\left(u_{1}^{t} x_{i}\right), \ldots,\left(u_{k}^{t} x_{i}\right)\right)$.

To truly get a mapping to $\mathbb{R}^{k}$ we use the mapping $Y=U_{k}^{t} X$. this defines the $k \times m$ mapped data matrix $Y$

$$
Y=U_{k}^{t} X=\left(\begin{array}{ccc}
\vdots & & \vdots \\
U_{k}^{t} x_{1} & \cdots & U_{k}^{t} x_{m} \\
\vdots & & \vdots
\end{array}\right)
$$

The columns of $Y$ (each $y_{i}=U_{k}^{t} x_{i}$ called the score of data point $x_{i}$ ) are the projection of the original data to $\mathbb{R}^{k}$. Those scores, in a $k$ dimensional space (where the axes - sometimes called the loading - are $\left\{u_{1}, \ldots, u_{k}\right\}$ ) may now be process by other machine learning algorithms (e.g. SVM, clustering, etc') that can now be executed faster (due to fewer dimensions) and with better results (due to the elimination of irrelevant/noise attributes).

### 10.2.3 Optimality of the mapping

To justify the choice of the transformation $P_{k} X$ (using $U_{k}$, the loadings related to the $k$ highest eigenvectors of the SVD) we consider the norm of the residual,

$$
\left\|P_{k} X-X\right\|_{F}^{2}
$$

where the $F$ stands for the Frobenious norm, which is the sum of the squares of the entries. By the properties of the projection $P$ (i.e. $P^{2}=P$ ) and linearity of the trace operator we have

$$
\begin{equation*}
\left\|P_{k} X-X\right\|_{F}^{2}=\operatorname{Tr}\left(\left(P_{k} X-X\right)^{t}\left(P_{k} X-X\right)\right)=-\operatorname{Tr}\left(X^{t} P X\right)+\operatorname{Tr}\left(X^{t} X\right) \tag{10.1}
\end{equation*}
$$

Therefore, minimizing the Frobenious norm is equivalent to maximizing the trace, and
we get (using $\operatorname{Tr}(A B)=\operatorname{Tr}(B A)$ in the one-before-last equality)

$$
\begin{aligned}
\min _{P_{k}}\left\|P_{k} X-X\right\|_{F}^{2} & =\max _{P_{k}} \operatorname{Tr}\left(X^{t} P X\right) \\
& =\max _{P_{k}} \operatorname{Tr}\left(X^{t} U_{k} U_{k}^{t} X\right) \\
& =\max _{P_{k}} \operatorname{Tr}\left(\left(X^{t} U_{k}\right)\left(X^{t} U_{k}\right)^{t}\right) \\
& =\max _{P_{k}} \operatorname{Tr}\left(\left(\begin{array}{ccc}
\vdots & & \vdots \\
\sigma_{1} u_{1} & \cdots & \sigma_{k} u_{k} \\
\vdots & & \vdots
\end{array}\right)\left(\begin{array}{ccc}
\cdots & \sigma_{1} v_{1}^{t} & \cdots \\
& \vdots & \\
\cdots & \sigma_{k} v_{k}^{t} & \cdots
\end{array}\right)\right) \\
& =\max _{P_{k}} \operatorname{Tr}\left(\left(\begin{array}{ccc}
\cdots & \sigma_{1} v_{1}^{t} & \cdots \\
& \vdots & \\
\cdots & \sigma_{k} v_{k}^{t} & \cdots
\end{array}\right)\left(\begin{array}{ccc}
\vdots & & \vdots \\
\sigma_{1} u_{1} & \cdots & \sigma_{k} u_{k} \\
\vdots & & \vdots
\end{array}\right)\right) \\
& =\sum_{j=1}^{k} \sigma_{j}^{2}
\end{aligned}
$$

We conclude that the projection that takes the loadings corresponding to the $k$ highest eigenvalues of the SVD of the original data $X$ is optimal in the sense of preserving the information in the original data (as measured by the Frobenious norm).

### 10.2.4 Mapped samples $Y$ are uncorrelated

Recall that the dimension of $X$ is $N \times m$, having each example as a column. The dimension of $X^{t} X$ is $m \times m$ and of $X X^{t}$ is $N \times N$. The eigenvectors of $X X^{t}$ are the columns of $U$ and the eigenvectors of $X^{t} X$ are the columns of $V$. Recall also that $Y=U_{k}^{t} X$ (taht is, the data mapped to a $k$ dimensional feature space using the $k$ first principal components $\left.U_{k}\right)$. Therefore we can write $Y Y^{t}=U_{k}^{t} X X^{t} U_{k}=\operatorname{diag}\left(\sigma_{1}^{2}, \ldots, \sigma_{r}^{2}\right)$. On the other hand $Y Y^{t}=\sum_{i} y_{i} y_{i}^{t}$. This implies that the resulting samples $Y$ in $k$ dimensions are not correlated.

### 10.2.5 Example - Eigenfaces

## Setting

Given a set of pictures (portraits) of world leaders, each picture is 50 by 35 gray-scale pixels, hence represented by a $1850 \times 1$ column vector. PCA with $k=6,12,120$ was done on a subset $X$ of $\sim 1000$ pictures of the six most frequent leaders appearing in the pictures (each leader having at least $\sim 90$ different pictures).

## bi-plots

For the PCA with $k=6$, the mapped pictures $Y$ are the resulting 6 -dimentional representation of each of the original pictures. Each $y_{i}$ may now be plotted in a 6 dimensional plot where the axes are the columns of $U_{6}$ (the loadings). The coordinates of $y_{i}$ are the scores. This plot is a representation of what may be used to further apply machine learning (e.g. for clustering, classification, etc'). The same $k=6$ axes may be used to plot (overlay the plots of $Y$, hence the name $b i$-plot) a point for each original feature (each pixel in our case) - each feature $j$ of the original $N=1850$ features is now represented as a 6 -long vector $\left(u_{1 j}, \ldots, u_{6 j}\right)$.

## Results and Interpretations

Each principal component $u_{j}$ is a 1850 long vector and hence also a picture (an eigenface). Figure 10.1 shows the principal eigenfaces for $k=12$.


Figure 10.1: Eigenfaces

To illustrate the possible semantics of each principal component we did the following $(k=6)$ : For each principal component $u_{j}$ twelve representative (original) pictures were chosen, with scores evenly spread accross the range of the $u_{j}$ coordinate. Figure 10.2 shows the representative pictures chosen (a row, top to bottom, for each principal component). It is easy to see that the first two principal components capture (respectively) the illumination level and direction.


Figure 10.2: Represantative (original) pictures, for each principal component. In each row (representing one component) the pictures are ordered left to right with increasing score.

### 10.3 Kernel PCA

### 10.3.1 The case $N \gg m$

Consider the case that we have much more features than examples $N \gg m$. We like to consider $k \leq r$ and have,

$$
V_{k}=\left(\begin{array}{ccc}
\vdots & & \vdots \\
v_{1} & \cdots & v_{k} \\
\vdots & & \vdots
\end{array}\right)
$$

Those are the first $k$ eigenvectors of $X^{t} X$ with orresponding $\sigma_{1}^{2} \geq \sigma_{2}^{2} \geq \cdots \geq \sigma_{k}^{2}>0$ eigenvalues.

## Claim 10.1

$$
\begin{equation*}
U_{k}=X V_{k} \Sigma_{k}^{-1} \tag{10.2}
\end{equation*}
$$

where $\Sigma_{k}^{2}=\operatorname{diag}\left(\sigma_{1}^{2}, \ldots, \sigma_{k}^{2}\right)$.
Proof: Recall that $X^{t} X V_{k}=V_{k} \Sigma_{k}^{2}$, since the columns of $V_{k}$ are the eigenvectors of $X^{t} X$. We can now multiply by $X$ from the left and have $X X^{t}\left(X V_{k}\right)=\left(X V_{k}\right) \Sigma_{k}^{2}$. This implies that the columns of $X V_{k}$ are the eigenvectors of $X X^{t}$. Since the first $k$ eigenvectors of $X X^{t}$
are the columns of $U_{k}$ we are almost done, the only issue is that the eigenvectors are not normalized.

To normalize, we multiply the eigenvectors $X V_{k}$ by $\Sigma_{k}^{-1}$. Set $U_{k}=X V_{k} \Sigma_{k}^{-1}$. We now have that

$$
U_{k}^{t} U_{k}=\Sigma_{k}^{-1} V_{k}^{t} X^{t} X V_{k} \Sigma_{k}^{-1}=\Sigma_{k}^{-1} V_{k}^{t}\left(V_{k} \Sigma_{k}^{2}\right) \Sigma_{k}^{-1}=\Sigma_{k}^{-1} \Sigma_{k}^{2} \Sigma_{k}^{-1}=I
$$

This allows us to do PCA by first computing $V, \Sigma^{2}$ - the eigenvectors and eigenvalues of $X^{t} X$ (a matrix $m \times m$ ), an operation of complexity $O\left(m^{3}\right)$, and recover $U$ using (10.2) ${ }^{2}$.

### 10.3.2 The kernel trick

We can now discuss the Kernel PCA. Consider a mapping $x \rightarrow \phi(x)$ (where $x$ is $n \times 1$ and $\phi(x)$ is $N \times 1$ ) induced by a proper kernel $K(\cdot, \cdot)$. Recal the definition of the kernel data matrix $\overline{\bar{K}}=\overline{\bar{K}}(X) \triangleq X^{t} X$ (That is, $\left.\overline{\bar{K}}_{i, j}=\left(\phi\left(x_{i}\right)\right)^{t} \phi\left(x_{j}\right)=K\left(x_{i}, x_{j}\right)\right)$.

If we try to do PCA in a straightforward way, we need to compute the matrix $U$ whose dimension depend on $N$ which might be huge or infinite. The main observation is that we do no need to compute $U$, we are rather interested only in $Y$ - the mapping to the $k$ dimensional subspace. Using (10.2) above, we have that

$$
Y=U_{k}^{t} X=\Sigma_{k}^{-1} V_{k}^{t} X^{t} X=\Sigma_{k}^{-1} V_{k}^{t} \overline{\bar{K}}
$$

Therefore, in order to compute $Y$ we do not have to compute $U$, but rather we can compute $Y$ using $\overline{\bar{K}}$ and $V_{k}$. Furthermore, the mapping of an arbitrary $x$ (in the original feature space, before the mapping induced by the kernel $K$ ) to the $k$ dimensional space spanned by the columns of $U_{k}$ can be similarly computed:

$$
y=U_{k}^{t} \phi(x)=\Sigma_{k}^{-1} V_{k}^{t} X^{t} \phi(x)=\Sigma_{k}^{-1} V_{k}^{t} K(X, x)
$$

Where $K(X, x)$ is a column vector with $K\left(x_{i}, x\right)$ at the $i^{t h}$ position.

[^1]
[^0]:    ${ }^{1}$ Question: cosidering the material discussed later, is it important to normalize the attributes such that all have the same variance?

[^1]:    ${ }^{2}$ The alternative, finding $U$ first, is significantly less efficient in the case $N \gg m$.

