CS 5785 – Applied Machine Learning – Lec. 13

Prof. Serge Belongie, Cornell Tech
Scribe: Sonia Sen, Inna Shteinbuk, Vivek Sudarsan

Oct. 17, 2017 (Under construction)

1   Principal Components Analysis (PCA)

1.1   Recap

PCA is an unsupervised dimensionality reduction technique that stacks all the feature vectors x_i into a matrix $\mathbf{X} \in \mathbb{R}^{N \times p}$. PCA is used to discover axes of variation in a dataset. We either assume that the data is centered, i.e., $\bar{x} = 0$, or we center the data by setting $x_i = x_i - \bar{x}$, and we get:

$$\mathbf{S} = \frac{1}{N}\mathbf{X}^\top\mathbf{X} \in \mathbb{R}^{p \times p}$$

We compute the SVD, $X = \mathbf{UDV}^\top$. The columns of $\mathbf{UD}$ are the principal components of $\mathbf{X}$. The singular values $d_1 \geq \ldots \geq d_p \geq 0$ capture the variance along each principal axis.

Figure 1 is an example of a dimensionality reduction where $p = 2$ and $q = 1$. In Figure 2, $p = 3$ and $q = 2$. For simplicity, assume we have a *centered* set of data, i.e $\bar{x} = 0$.

As an aside of "punk stats", it may be helpful to think of 2 different types of datasets: clumpy and manifold-y. Clumpy datasets would use methods like k-means, GMM with EM algorithm, etc and are well represented by tokens/cluster centers. Manifold-y datasets, where instead of clusters there are curves and low dimensional coordinate systems, methods like PCA would be used with these datasets if the manifold is known.

1.2   MNIST Digits

We can think of images of handwritten digits as living on a manifold since it's plausible that we could express one instance as a linear (weighted) combination of other instances. If we look at Figure 3 we can see a collection of instances of the digit 3 drawn by different people. Like K-means, PCA operates on a set of vectors, which means we need to transform the digit images, each of which are 28×28 pixel images, into to vectors. The way we do that is by concatenating the rows or columns to create a single vector of length 784. When we run either K-means or PCA on these vectors we obtain vectors as a result which we can then reschape or "un-concatenate" for purposes of visualization.

Given a set of digits (such as those in Figure 3), each digit can be represented by a linear combination of the mean of all the digit examples plus a constant
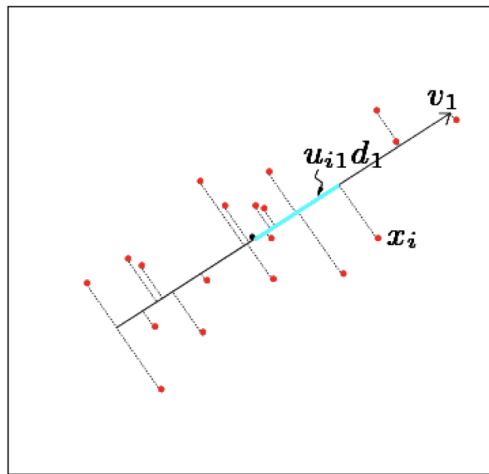
FIGURE 14.20. *The first linear principal component of a set of data. The line minimizes the total squared distance from each point to its orthogonal projection onto the line.*

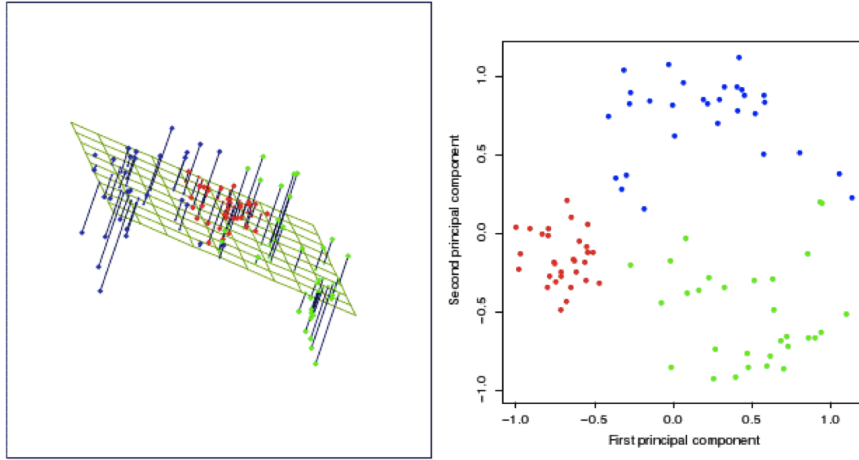Figure 1: Dimensionality reduction from 2 dimensions to 1 dimension

FIGURE 14.21. *The best rank-two linear approximation to the half-sphere data. The right panel shows the projected points with coordinates given by $\mathbf{U}_2\mathbf{D}_2$, the first two principal components of the data.*

Figure 2: Dimensionality reduction from 3 dimensions to 2 dimension

times the first principal component and the same for the second principal component and so on. Figure 4 shows an example of a digit being represented by the mean, $\bar{x}$, and a linear combination of the two leading principal components, v_1 and v_2. Each principal component represents slight changes relative to the mean (like stretching, rotation or stroke dilation).

Figure 5 shows the result of performing PCA on a set of thousands of digit 3's and keeping only the 2 strongest principal components. The figure also shows how picking 15 examples scattered on this 2 dimensional space can inform us of the type of linear change that each principal component captures. Each red dot on the graph corresponds to the relative digit depicted on the right. The digits are reconstructed from the principal components as specified in Figure 4. A closer look at the digits reveals that the principal component represented by the y axis captures the stroke strength and the one represented by the x axis captures the curvature of the digit.

1.2.1 Eigenfaces

At the bottom of Figure 6 we can see an example of PCA used for depicting faces. The top left part of the matrix on the left represents the mean face. Each of the other sections of this matrix represents a prinicipal component (where the black ranges represent positive values and the red ranges represent negative values). The 2-D array we use to multiply the matrix by shows the constants we multiply each principal component by to get the 48 dimension approximation to the origianl face - The PCA reconstruction of this face. If we look closely we
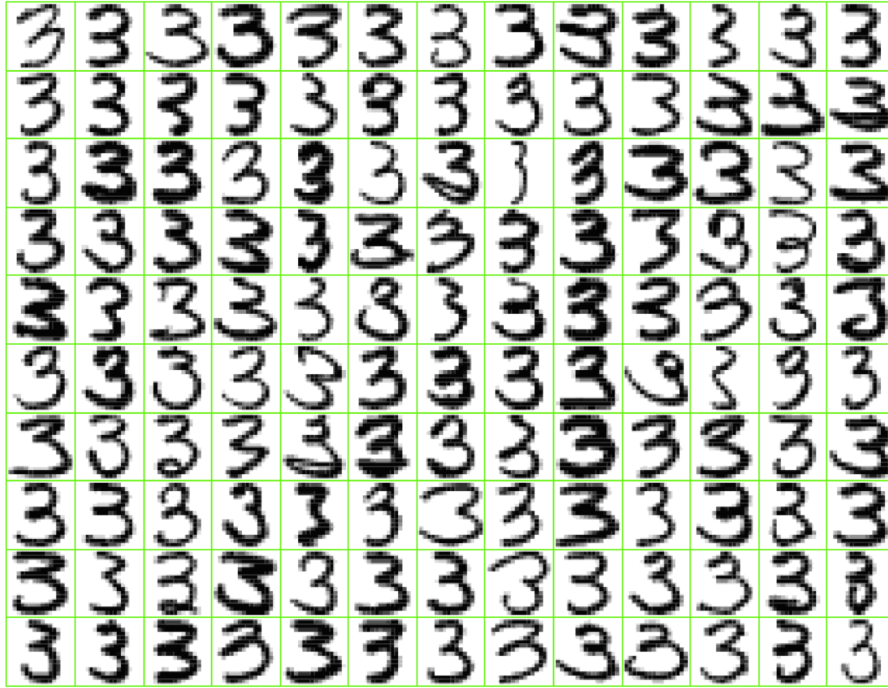
3

FIGURE 14.22. *A sample of 130 handwritten 3's shows a variety of writing styles.*

Figure 3:

$$
\begin{aligned}
\hat{f}(\lambda) &= \bar{x} + \lambda_1 v_1 + \lambda_2 v_2 \\
&= \boxed{3} + \lambda_1 \cdot \boxed{3} + \lambda_2 \cdot \boxed{3}. \quad (14.55)
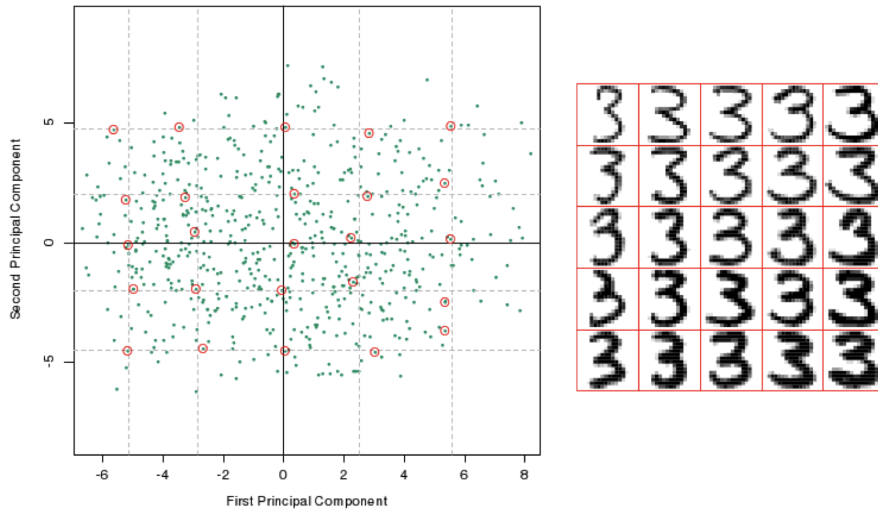\end{aligned}
$$

Figure 4:

4

Figure 5: Dimensionality reduction from 784 dimensions to 2 dimensions using PCA.

notice that the top left constant in this 2-D array is black, meaning 1, which will always be true because as we saw in Figure 4 the mean appears once in every reconstructed sample.

1.2.2 Uses of PCA

There are a few possible uses for PCA that take advantage of the approximated representation of each sample as a q sized vector:

- **Compression** We use less space to store the samples, since each reconstructed sample is shorter than the original. In Figure 5 we have examples of digits that originally were represented by vectors of length 784 and now are represented by vectors of length 2.

- **Recognition** After running PCA and keeping only the most meaningful components we have a shorter representation for each face. Two face images that have relatively close principal components might belong to the same person. In other words the *dot product* of the 2 samples with the principal components would result in similar vectors for samples belonging to the same person.

- **Denoising** The smaller principal components that were removed during the dimensionality reduction might contain noise instead of meaningful, useful and interesting information. By removing them we denoise the given sample set.

2 Non-Negative Matrix Factorization (NMF)

One aspect of PCA that may be undesireable in some cases is that it leaves the basis functions and coefficients free to swing into negative values, as seen
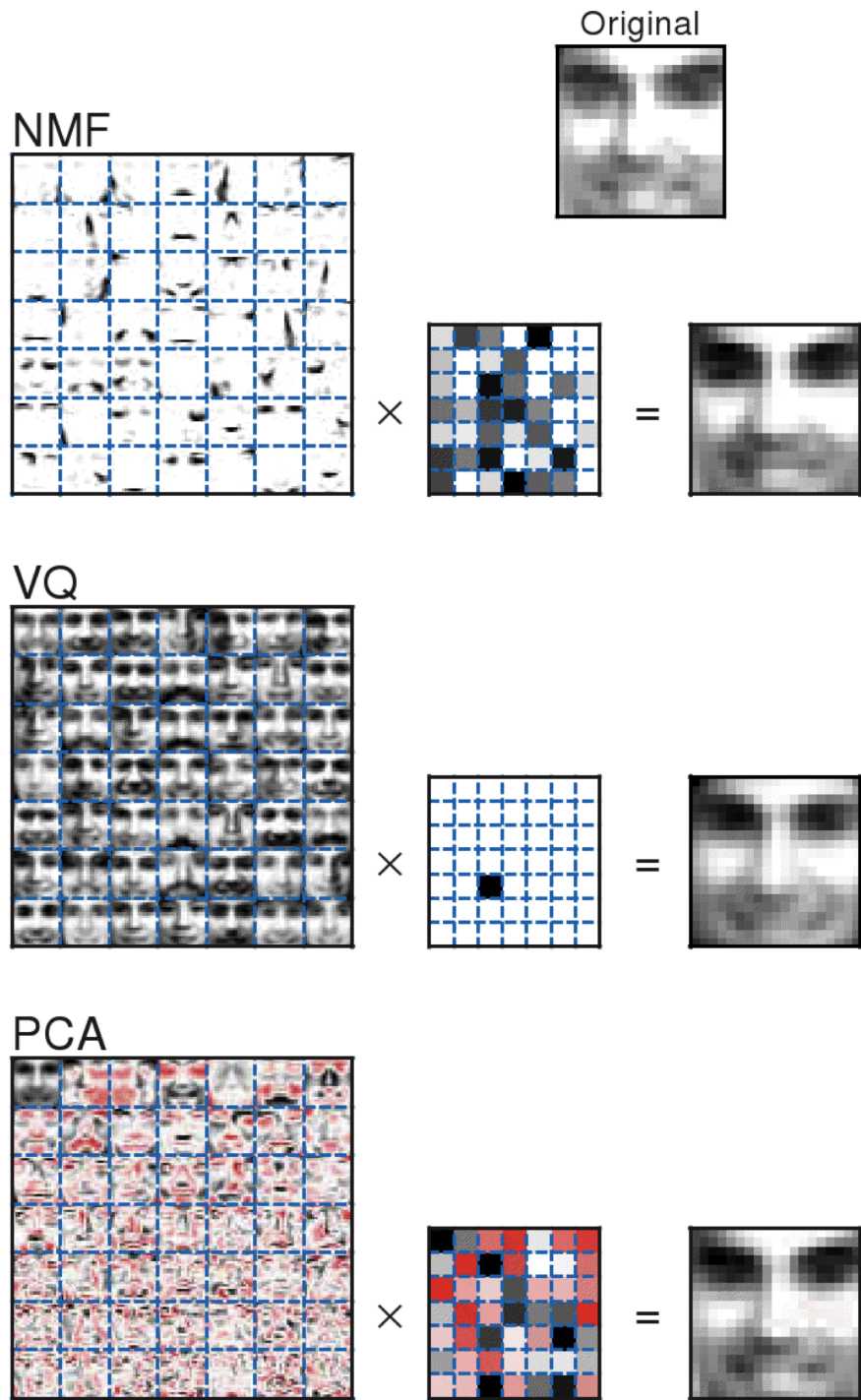
5

Figure 6: Representing faces using dimensionality reduction techniques. Each sample face is a 19×19 array, and the algorithms used $N = 2,429$ training examples. In the PCA example, negative values are shown in red/pink.

by the amount of pink in Figure 6. This means that when we reconstruct samples using PCA some of the principal components cancel out part of other principal components, in a manner analogous to Fourier Series. A lot of the measurements of practical interest to us are non-negative. Nonnegative Matrix Factorization (NMF) takes this into consideration and uses this assumption to create a different method of dimensionality reduction (Lee & Seung, 1999). Given an $N \times p$ matrix, where each row represents a sample in our sample set, we use the approximation $\mathbf{X} \approx \mathbf{WH}$, where:

- $\mathbf{W} \in \mathbf{R}^{N \times r}$

- $\mathbf{H} \in \mathbf{R}^{r \times p}$

- $r \leq \max(N, p)$

- Assume $x_{ij}, w_{ik}, h_{kj} \geq 0$, i.e., everything is nonnegative.

Find $\mathbf{W}$ and $\mathbf{H}$ by maximizing the log likelihood for a model in which x_{ij} has a Poisson distribution with mean $(\mathbf{WH})_{ij}$:

$$L(\mathbf{W}, \mathbf{H}) = \sum_{i=1}^{N} \sum_{j=1}^{p} \left[x_{ij} \log (\mathbf{WH})_{ij} - (\mathbf{WH})_{ij} \right]$$

We will skip the discussion about how they arrived at this objective function. They use the following alternating algorithm to find a local maximum of $L(\mathbf{W}, \mathbf{H})$:

$$w_{ik} \quad \leftarrow \quad w_{ik} \frac{\sum\limits_{j=1}^{p} h_{kj} x_{ij} / (\mathbf{WH})_{ij}}{\sum\limits_{j=1}^{p} h_{kj}}$$

$$h_{kj} \quad \leftarrow \quad h_{kj} \frac{\sum\limits_{i=1}^{N} w_{ik} x_{ij} / (\mathbf{WH})_{ij}}{\sum\limits_{i=1}^{N} w_{ik}}$$

Let us look at Figure 6 again. It is clearly noticeable that in NMF all the basis images (on the left) and coefficients (on the right) are non-negative, since there are no shades of red. Each basis image represents part of a face, a feature, like glasses, beards, higher cheek bones, etc.

2.1 Topic Modeling

One application of NMF is *topic modeling* in a collection of documents. Intuitively, given that a document is about a particular topic, one would expect particular words to appear in the document more or less frequently. In NMF topic modeling each document is represented by a *bag of words*, a vector where each index correxponds to a word and the value to the number of times the words appear in the document. Since the vectors, which are just word-count histograms, are counting word occurences, they are non-negative by definition.

Each topic is represented by a vector, similar to the basis images in the example shown in Figure 6, and each document can be described as a superposition of these components, all with nonnegative weights. In practice, Latent Dirichlet Allocation is a more commonly used technique for document topic modeling.

2.2 Sparsity

If we look again at Figure 6 we can see that representing a sample from the sample set using VQ is clearly cheaper in space than using the other methods, since it uses 1 value only. We call this a *sparse* representation. VQ is the sparsest method but also the least accurate. If we regard the example in Figure 6 as a way for Cornell campus security to cheaply represent the pictures of each student that using VQ each new student would be represented by the picture of the closest matched basis image from last year, clearly not a good enough estimation. In the other two cases, a new student would be represented by the mean face from last year (which given a large enough sample set shouldn't change much) and a linear combination of the basis images/principal components which is relatively unique to him.

In our case, sparsity means the number of zeros in the linear combination representing each sample. As stated before VQ is clearly the sparsest between the three, but between NMF and PCA, NMF is sparser. This because as we explained earlier, PCA hinges on the fact that the different principal components rely in part on cancellation effects. As such the linear combination is full of positive and negative values that balance to the final result. NMF on the other hand adds non-negative features only in the cases where they are relevant and there is no mutual cancellations so the resulting vector is sparser.

2.2.1 Cleve Moler's example of Sparsity

"I'm thinking of two numbers whose average is 3. What are the numbers?" Clearly, there is not enough information to answer the question definitively, but most people will answer "2 and 4." There are, of course, an infinite number of possible answers, the fact that most people would still give the same answer implies that there is some kind of *regularization* on the solution that people perform when answering the question, in particular, one that picks a pair of numbers equidistant and not too far from the average.

The problem has the form $\mathbf{A}y = b$ with $\mathbf{A} = [0.5, 0.5]$ and $b = 3$. Possible solutions are: $y = [2, 4]$, $y = [1, 5]$, $y = [3, 3]$ and $y = [0, 6]$. The solution $y = [0, 6]$ is the sparsest one, since half of its values are zero.

2.2.2 Compressive Sensing

Compressive Sensing approaches seek the solution with minimum ℓ_0 norm, which counts the non zero elements in a vector. The ℓ_0 norm leads to very difficult combinatorial optimization problems. Donoho, Candès and Tao have shown that one can use the ℓ_1 norm, the sum of absolute values, instead of the ℓ_0 norm and get basically the same solution. This permits a solution via linear programming. What we gain by choosing sparser solutions is the ability to recreate samples accurately using less information.