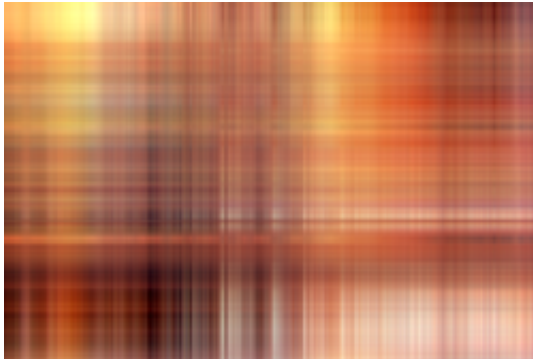


# Principal component analysis



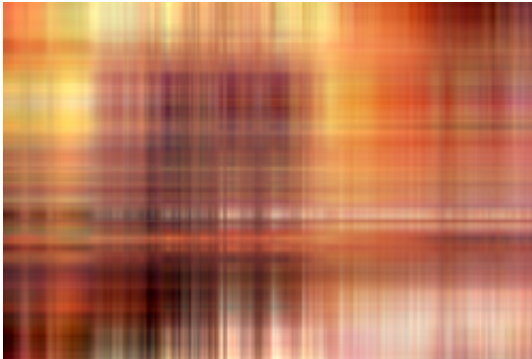
Ingo Blechschmidt  
December 17th, 2014

# Principal component analysis



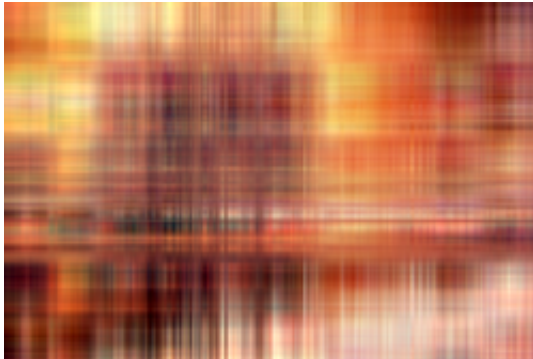
Ingo Blechschmidt  
December 17th, 2014

# Principal component analysis



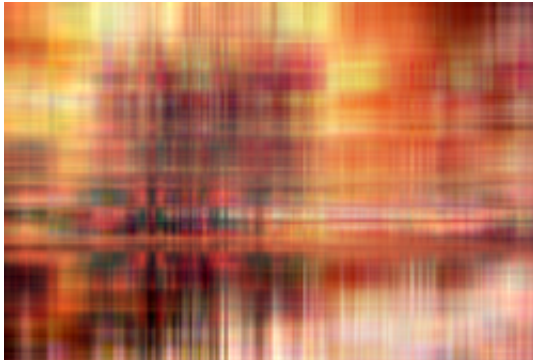
Ingo Blechschmidt  
December 17th, 2014

# Principal component analysis



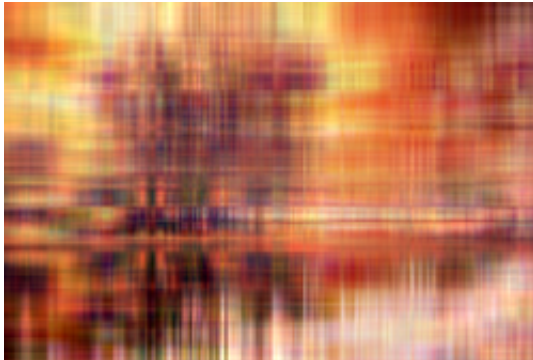
Ingo Blechschmidt  
December 17th, 2014

# Principal component analysis



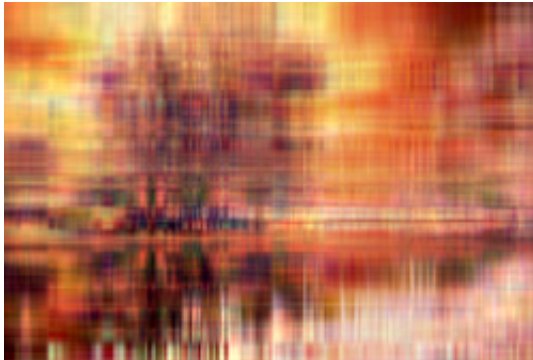
Ingo Blechschmidt  
December 17th, 2014

# Principal component analysis



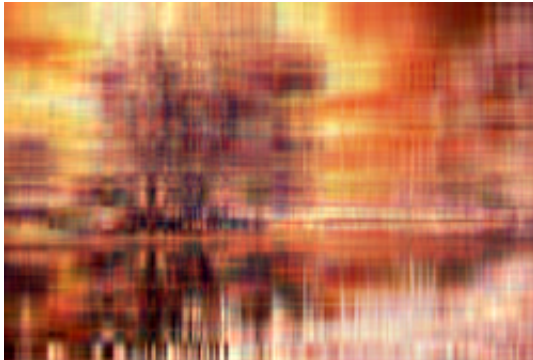
Ingo Blechschmidt  
December 17th, 2014

# Principal component analysis



Ingo Blechschmidt  
December 17th, 2014

# Principal component analysis



Ingo Blechschmidt  
December 17th, 2014



# Principal component analysis



Ingo Blechschmidt  
December 17th, 2014

# Principal component analysis



Ingo Blechschmidt  
December 17th, 2014

# Principal component analysis



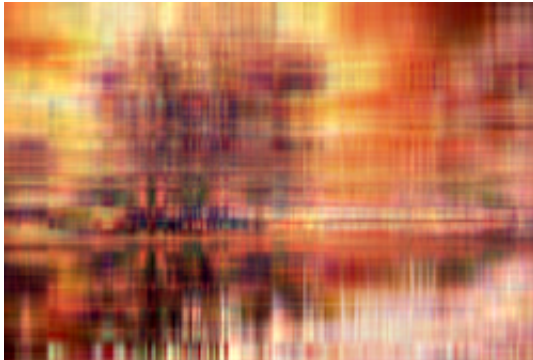
Ingo Blechschmidt  
December 17th, 2014

# Principal component analysis



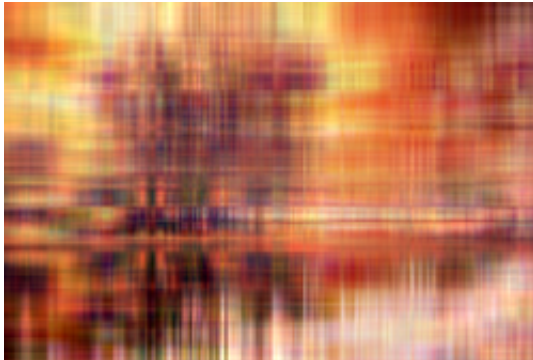
Ingo Blechschmidt  
December 17th, 2014

# Principal component analysis



Ingo Blechschmidt  
December 17th, 2014

# Principal component analysis



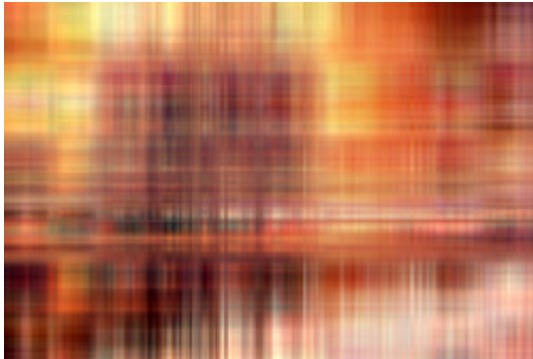
Ingo Blechschmidt  
December 17th, 2014

# Principal component analysis



Ingo Blechschmidt  
December 17th, 2014

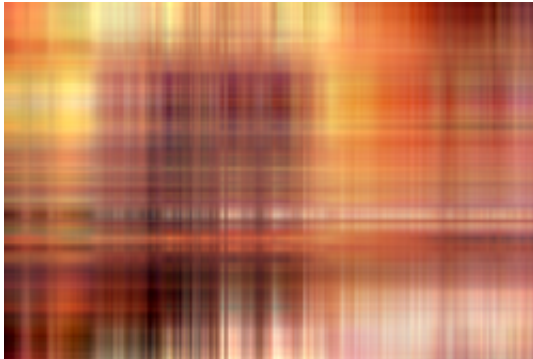
# Principal component analysis



Ingo Blechschmidt  
December 17th, 2014

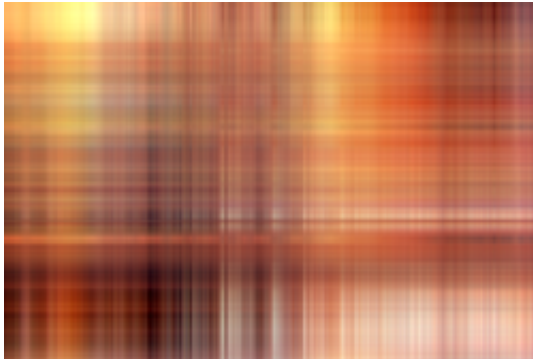


# Principal component analysis



Ingo Blechschmidt  
December 17th, 2014

# Principal component analysis



Ingo Blechschmidt  
December 17th, 2014

# Principal component analysis



Ingo Blechschmidt  
December 17th, 2014

# Principal component analysis



Ingo Blechschmidt  
December 17th, 2014

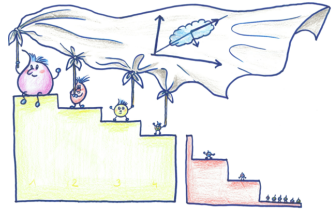
# Outline

## 1 Theory

- Singular value decomposition
- Pseudoinverses
- Low-rank approximation

## 2 Applications

- Image compression
- Proper orthogonal decomposition
- Principal component analysis
- Eigenfaces
- Digit recognition



# Singular value decomposition

Let  $A \in \mathbb{R}^{n \times m}$ . Then there exist

- numbers  $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_m \geq 0$ ,
- an orthonormal basis  $\mathbf{v}_1, \dots, \mathbf{v}_m$  of  $\mathbb{R}^m$ , and
- an orthonormal basis  $\mathbf{w}_1, \dots, \mathbf{w}_n$  of  $\mathbb{R}^n$ ,

such that

$$A\mathbf{v}_i = \sigma_i\mathbf{w}_i, \quad i = 1, \dots, m.$$

In matrix language:

$$A = W\Sigma V^t,$$

where  $V = (\mathbf{v}_1 | \dots | \mathbf{v}_m) \in \mathbb{R}^{m \times m}$  orthogonal,

$W = (\mathbf{w}_1 | \dots | \mathbf{w}_n) \in \mathbb{R}^{n \times n}$  orthogonal,

$$\Sigma = \text{diag}(\sigma_1, \dots, \sigma_m) \in \mathbb{R}^{n \times m}.$$

- The singular value decomposition (SVD) exists for any real matrix, even rectangular ones.
- The singular values  $\sigma_i$  are unique.
- The basis vectors are not unique.
- If  $A$  is orthogonally diagonalizable with eigenvalues  $\lambda_i$  (for instance, if  $A$  is symmetric), then  $\sigma_i = |\lambda_i|$ .
- $\|A\|_{\text{Frobenius}} = \sqrt{\sum_{ij} A_{ij}^2} = \sqrt{\text{tr}(A^t A)} = \sqrt{\sum_i \sigma_i^2}$ .
- There exists a generalization to complex matrices. In this case, the matrix  $A$  can be decomposed as  $W\Sigma V^*$ , where  $V^*$  is the complex conjugate of  $V^t$  and  $W$  and  $V$  are unitary matrices.
- The singular value decomposition can also be formulated in a basis-free manner as a result about linear maps between finite-dimensional Hilbert spaces.

Existence proof (sketch):

1. Consider the eigenvalue decomposition of the symmetric and positive-semidefinite matrix  $A^t A$ : We have an orthonormal basis  $\mathbf{v}_i$  of eigenvectors corresponding to eigenvalues  $\lambda_i$ .
2. Set  $\sigma_i := \sqrt{\lambda_i}$ .
3. Set  $\mathbf{w}_i := \frac{1}{\sigma_i} A \mathbf{v}_i$  (for those  $i$  with  $\lambda_i \neq 0$ ).
4. Then  $A \mathbf{v}_i = \sigma_i \mathbf{w}_i$  holds trivially.
5. The  $\mathbf{w}_i$  are orthonormal:  $(\mathbf{w}_i, \mathbf{w}_j) = \frac{1}{\sigma_i \sigma_j} (A^t A \mathbf{v}_i, \mathbf{v}_j) = \frac{\lambda_i \delta_{ij}}{\sigma_i \sigma_j}$ .
6. If necessary, extend the  $\mathbf{w}_i$  to an orthonormal basis.

This proof gives rise to an algorithm for calculating the SVD, but unless  $A^t A$  is small, it has undesirable numerical properties. (But note that one can also use  $AA^t$ !) Since the 1960ies, there exists a stable iterative algorithm by Golub and van Loan.



# The pseudoinverse of a matrix

Let  $A \in \mathbb{R}^{n \times m}$  and  $\mathbf{b} \in \mathbb{R}^n$ . Then the solutions to the optimization problem

$$\|A\mathbf{x} - \mathbf{b}\|_2 \longrightarrow \min$$

under  $\mathbf{x} \in \mathbb{R}^m$  are given by

$$\mathbf{x} = A^+ \mathbf{b} + V \begin{pmatrix} 0 \\ \star \end{pmatrix},$$

where  $A = W\Sigma V^t$  is the SVD and

$$\begin{aligned} A^+ &= W\Sigma^+ V^t, \\ \Sigma^+ &= \text{diag}(\sigma_1^{-1}, \dots, \sigma_m^{-1}). \end{aligned}$$

- In the formula for  $\Sigma^+$ , set  $0^{-1} := 0$ .
- If  $A$  happens to be invertible, then  $A^+ = A^{-1}$ .
- The pseudoinverse can be used for polynomial approximation: Let data points  $(x_i, y_i) \in \mathbb{R}^2$ ,  $1 \leq i \leq N$ , be given. Want to find a polynomial  $p(z) = \sum_{k=0}^n \alpha_k z^k$ ,  $n \ll N$ , such that

$$\sum_{i=1}^N |p(x_i) - y_i|^2 \longrightarrow \min.$$

In matrix language, this problem is written

$$\|A\mathbf{u} - \mathbf{y}\|_2 \longrightarrow \min$$

where  $\mathbf{u} = (\alpha_0, \dots, \alpha_N)^T \in \mathbb{R}^{n+1}$  and

$$A = \begin{pmatrix} 1 & x_1 & x_1^2 & \cdots & x_1^n \\ 1 & x_2 & x_2^2 & \cdots & x_2^n \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_N & x_N^2 & \cdots & x_N^n \end{pmatrix} \in \mathbb{R}^{N \times (n+1)}, \quad \mathbf{y} = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{pmatrix} \in \mathbb{R}^N.$$

# Low-rank approximation

Let  $A = W\Sigma V^t \in \mathbb{R}^{n \times m}$  and  $1 \leq r \leq n, m$ . Then a solution to the optimization problem

$$\|A - M\|_{\text{Frobenius}} \longrightarrow \min$$

under all matrices  $M$  with  $\text{rank } M \leq r$  is given by

$$M = W\Sigma_r V^t,$$

$$\text{where } \Sigma_r = \text{diag}(\sigma_1, \dots, \sigma_r, 0, \dots, 0).$$

The approximation error is

$$\|A - W\Sigma_r V^t\|_F = \sqrt{\sigma_{r+1}^2 + \dots + \sigma_m^2}.$$

- This is the Eckart–Young(–Mirsky) theorem.
- Beware of false and incomplete proofs in the literature!

# Image compression

- Think of images as matrices.
- Substitute a matrix  $W\Sigma V^t$  by  $W\Sigma_r V^t$  with  $r$  small.
- To reconstruct  $W\Sigma_r V^t$ , only need to know
  - the  $r$  singular values  $\sigma_1, \dots, \sigma_r$ ,  $r$
  - the first  $r$  columns of  $W$ , and height  $\cdot r$
  - the top  $r$  rows of  $V^t$ . width  $\cdot r$
- Total amount:  
 $r \cdot (1 + \text{height} + \text{width}) \ll \text{height} \cdot \text{width}$

- Image compression by singular value decomposition is mostly of academic interest only.
- This might be for the following reasons: other compression algorithms have more efficient implementations; other algorithms tailor to the specific properties of human vision; the basis vectors of other approaches (for instance, DCT) are similar to the most important singular basis vectors of a sufficiently large corpus of images.

# Proper orthogonal decomposition

Given data points  $\mathbf{x}_i \in \mathbb{R}^N$ , want to find a low-dimensional linear subspace which **approximately contains** the  $\mathbf{x}_i$ .

Minimize

$$J(U) := \sum_i \|\mathbf{x}_i - P_U(\mathbf{x}_i)\|^2$$

under all  $r$ -dimensional subspaces  $U \subseteq \mathbb{R}^N$ ,  $r \ll N$ ,  
where  $P_U : \mathbb{R}^N \rightarrow \mathbb{R}^N$  is the orthogonal projection onto  $U$ .

# Proper orthogonal decomposition

Given data points  $\mathbf{x}_i \in \mathbb{R}^N$ , want to find a low-dimensional linear subspace which **approximately contains** the  $\mathbf{x}_i$ .

Minimize

$$J(U) := \sum_i \|\mathbf{x}_i - P_U(\mathbf{x}_i)\|^2$$

under all  $r$ -dimensional subspaces  $U \subseteq \mathbb{R}^N$ ,  $r \ll N$ ,  
where  $P_U : \mathbb{R}^N \rightarrow \mathbb{R}^N$  is the orthogonal projection onto  $U$ .

More concrete formulation: Minimize

$$J(\mathbf{u}_1, \dots, \mathbf{u}_r) := \sum_i \left\| \mathbf{x}_i - \sum_{j=1}^r \langle \mathbf{x}_i, \mathbf{u}_j \rangle \mathbf{u}_j \right\|^2,$$

where  $\mathbf{u}_1, \dots, \mathbf{u}_r \in \mathbb{R}^N$ ,  $\langle \mathbf{u}_j, \mathbf{u}_k \rangle = \delta_{jk}$ .



Collect the data points  $\mathbf{x}_i$  as columns of a matrix

$$X = (\mathbf{x}_1 | \cdots | \mathbf{x}_\ell) \in \mathbb{R}^{N \times \ell}$$

and consider its singular value decomposition

$$X = W\Sigma V^t.$$

Then a solution to the minimization problem is given by the first  $r$  columns of  $W$ , with approximation error

$$J = \sum_i \|\mathbf{x}_i\|^2 - \sum_{j=1}^r \sigma_j^2.$$

- Proper orthogonal decomposition (POD) cannot be used to find low-dimensional *submanifolds* which approximately contain given data points. But check out *kernel principal component analysis*.
- Also, POD does not work well with *affine* subspaces. But in this case, the fix is easy: Simply shift the data points so that their mean is zero.
- POD is a general method for *dimension reduction* and can be used as a kind of “preconditioner” for many other algorithms: Simply substitute the given points  $\mathbf{x}_i$  by their projections  $P_U(\mathbf{x}_i)$ .
- The optimal subspace  $U$  is, in general, not unique. The POD basis is never unique, since  $J$  is invariant under the action of  $O(r)$ .

# Principal component analysis

Given observations  $x_i^{(k)}$  of random variables  $X^{(k)}$ , want to find **linearly uncorrelated** principal components.

Write  $X = (\mathbf{x}_1 | \cdots | \mathbf{x}_\ell) \in \mathbb{R}^{N \times \ell}$ . Calculate  $X = W \Sigma V^t$ .  
Then the principal components are the variables

$$Y^{(j)} = \sum_k W_{kj} X^{(k)}.$$

Most of the variance is captured by  $Y^{(1)}$ ; second to most is captured by  $Y^{(2)}$ ; and so on.

- For instance, in a study about circles, the variables *radius*, *diameter*, and *circumference* are linearly correlated.
- Principal component analysis (PCA) would automatically pick one of these attributes as a principal component.
- We have to normalize the data to have zero empirical mean first. Then  $XX^t$  is the empirical covariance matrix.
- Note that, in the given sample, the  $Y^{(j)}$  are indeed uncorrelated:

$$\begin{aligned}
 E(Y^{(j)}XX^tY^{(k)}) &= (W\mathbf{e}_j)^tXX^t(W\mathbf{e}_k) \\
 &= \mathbf{e}_j^tW^tW\Sigma V^tV\Sigma^tW^tW\mathbf{e}_k \\
 &= \Sigma\Sigma^t.
 \end{aligned}$$

- Beware that PCA cannot resolve nonlinear correlation.
- Also note that PCA is sensitive to outliers and not scaling-independent.

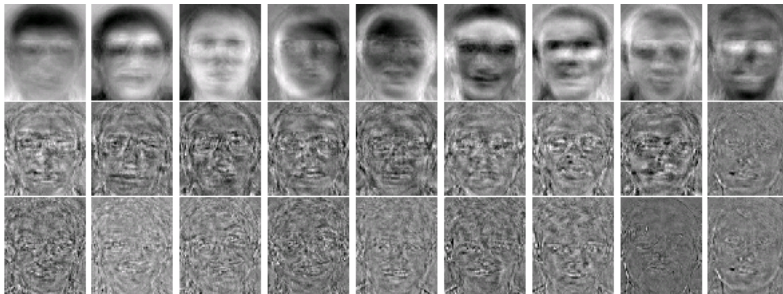
# Eigenfaces

- Record sample faces  
 $\mathbf{x}_1, \dots, \mathbf{x}_N \in \mathbb{R}^{\text{width} \cdot \text{height}}$ .
- Calculate a POD basis of **eigenfaces**.
- Recognize faces by looking at the coefficients of the most important eigenfaces.



Eigenfaces resemble faces.

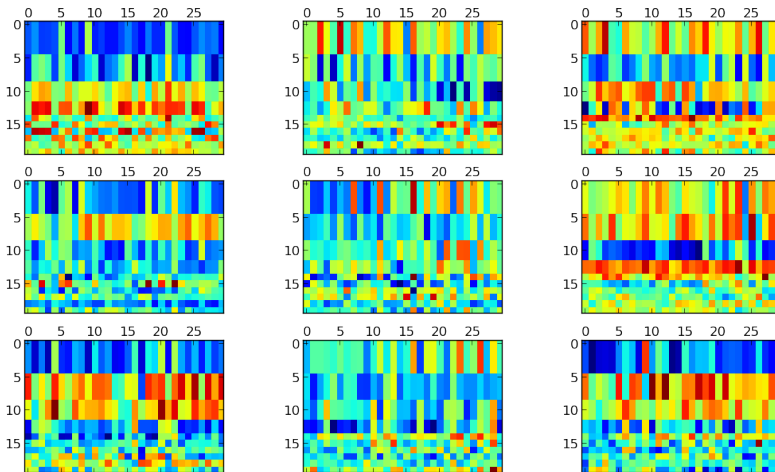
# More eigenfaces



- A naive approach is very sensitive to lighting, scale and translation.
- But extensions are possible, for instance considering the eyes, the nose, and the mouth separately; this leads to eigeneyes, eigennoses, and eigenmouths.
- Image credit:  
<http://upload.wikimedia.org/wikipedia/commons/6/67/Eigenfaces.png>  
<http://www.cenparmi.concordia.ca/~jdong/eigenface.gif>
- Live demo:  
<http://cognitron.psych.indiana.edu/nsfgrant/FaceMachine/faceMachine.html>
- Examples:  
<http://www.cs.princeton.edu/~cdecoro/eigenfaces/>

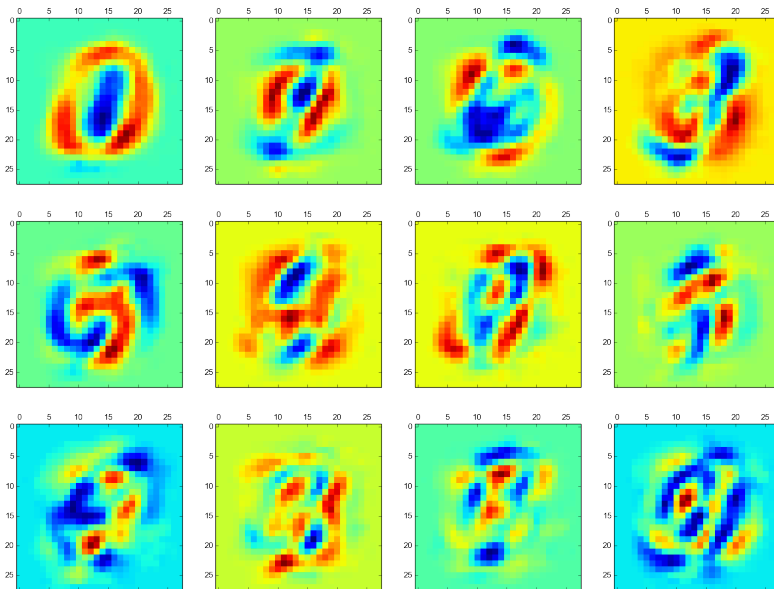
# Digit recognition

Apply POD for dimension reduction, then use some similarity measure or clustering technique. Results:





# Eigendigits



- This naive approach does not work very well, but refined techniques are discussed in the literature.

**I don't always do  
model reduction**

**but when I do I use  
singular value decomposition**