Lecture eleven

April 28, 2016

1 Final - non negative matrix factorization

Data needs to be vectorized. The W is the code book (basis). There is the code h. We need to develop code books that will describe good features.

From the images we need to generate the code book as well as the code. This method is called a factorization of a vector into two matrices. The code book should extract different features.

In the real-world case we turn many images into a codebook and many codes for each image. Get a factorization of the dataset into a code book and a matirx of codes.

2 methods

- vector quantization
  - clustering - similar to Expectation maximization algorithm.
  - Linde-buzo-gray
  - nearest neighboars

  With this you can generate basis vectors in the data set. You can then reconstruct the image from the basis vectors. This is one solution to the code book problem.

- Principle components: [eigen faces]
  - Eigenvectors of the covariance matrix.
  - Eigenfaces: since the eigen vectors are orthoganol you get a new basis from the eigenvectors.
  - You don't need all of the eigenvectors to get good vectors. you want the eigenvectors with the highest eigen value.

  You can then generate a code book from the eigen vectors to reconstruct the original images.

- non-negative Matrix factorization

- basis W and code H are positive or zero
- error V-V' is minimized
- Algorithm:
  - Ezpectatioin-maximization (EM)-algorithm
  - W and H are optimized alternatively
  - Error function: distance

$$||V - V'||^2 = \sum_{ij} (v_{ij} - v'_{ij})^2 \tag{1}$$

Kullback-leiber difference:

$$D(V||V') = \sum_{ij} \left(v_{ij} log(\frac{v_{ij}}{v'_{ij}}) - v_{ij} + v'_{ij} \right) \tag{2}$$

  - Algorithm:

$$W_{ia} \leftarrow W_{ia} \sum_{\mu} \frac{V_{i\mu}}{V'_{i\mu}} h_{a\mu} \tag{3}$$

$$W_{ia} \leftarrow \frac{W_{ia}}{\sum_j W_{ja}} \tag{4}$$

$$h_{a\mu} \leftarrow h_{a\mu} \sum_i W_{ia} \frac{V_{i\mu}}{V'_{i\mu}} \tag{5}$$

i is the row and a is the column in W. μ is the column of H and V/V'. μ is running over the training set. a is the row in H.

- Gradient Descent:

$$\frac{D(v||v')}{\partial v'} = -\frac{v}{v'} + 1 \tag{6}$$

This is a method for converting gradient descent into a multiplication.

$$+ \sum_{ijk} W_{ik} W_{ij} + \sum_{ij} \tag{7}$$

missing things.

The use of code books can reduce the dimensionality of the features in the classification problem. By doing the dimensionality reduction the classifier is given a much better feature to classify.

3 Literature

Daniel lee and sebastian Seung: learning the parts of objects by non-negative matrix factorization

Daniel Lee and Sebastian Seung: algorithms for Non-negative Matrix Factorization.

Yuan Wang et al.: fisher Non-negative Matrix Factorization for learning Local Features.

4 Second approach to building features

4.1 Fourier Transform

A spectrogram is a display of the frequencies in time with respect to a delta t.

5 Hadamard Transform

$$H_2 = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \tag{8}$$

$$\vec{b} = H_2 \vec{a} \tag{9}$$

$$H_4 = \frac{1}{\sqrt{2}} \begin{bmatrix} H_2 & H_2 \\ H_2 & -H_2 \end{bmatrix} \tag{10}$$

$$b_{2^n} = H_{2^n} a_{2^n} \tag{11}$$

$$\begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 1 & 1 & 1 & 1 \\ -1 & 1 & -1 & 1 \\ -1 & -1 & 1 & 1 \\ 1 & -1 & -1 & 1 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \end{bmatrix} \tag{12}$$

$$P' = H_{2^n} P H_{2^n} \tag{13}$$

$$P = H_{2^n} P' H_{2^n} \tag{14}$$

You can perform a low pass filter of an image by removing the high frequency terms. This is accomplished by setting the high frequency features to zero.

5.1 FHT

$$\frac{1}{\sqrt{2}} \begin{bmatrix} H_2 & H_2 \\ H_2 & -H_2 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \end{bmatrix} = \frac{1}{\sqrt{2}} \begin{bmatrix} H_2 a_u + H_2 a_l \\ H_2 a|u - H_2 a_l \end{bmatrix} \tag{15}$$

5.2 FT

Roots of unity: $X^n = 1$. For the case of $n = 2$ there are two roots. For the case when $x^4 = 1$ there are 4 roots. $x_1 = 1, x_2 = -i, x_3 = -1, x + 4 = -i$ are the four roots of unity.

$$F_8 = \frac{1}{\sqrt{n}} \begin{bmatrix} \omega_8^0 & \omega_8^0 & \omega_8^0 & \dots & \omega_8^0 \\ \omega_8^0 & \omega_8^1 & \omega_8^2 & \dots & \omega_8^7 \\ \omega_8^0 & \omega_8^2 & \omega_8^4 & \dots & \omega_8^14 \\ \omega_8^0 & \omega_8^3 & \omega_8^6 & \dots & \omega_8^21 \\ \omega_8^0 & \omega_8^4 & \omega_8^8 & \dots & \omega_8^28 \\ \omega_8^0 & \omega_8^5 & \omega_8^10 & \dots & \omega_8^35 \\ \omega_8^0 & \omega_8^6 & \omega_8^12 & \dots & \omega_8^42 \\ \omega_8^0 & \omega_8^7 & \omega_8^14 & \dots & \omega_8^49 \end{bmatrix} \tag{16}$$

$$B = FAF \tag{17}$$

$$||B|| \leftarrow \text{power spectrum} \tag{18}$$

The power spectrum is spatially invariant in the frequency domain. The convolutional property is valid for a torus. Which is what makes the power spectrum spatially invariant, but only when objects do no lie on the edge.

6 Final

write a report. The report is going to be about Deep learning.

1. Read about deep learning.
2. Describe an example of a deep learning network
3. Look at the Activation functions?
4. Look at the topology of the network:
   - pyramids
   - convolution
5. Applications

Send PDF, May 7th, Sat Midnight: rojas@inf.fu-berlin.de
Don't write more than 10 pages.