Xihao Zhu
Xz36
Comp502 midterm exam
Honor code:

COMP / ELEC / STAT 502 Exam 1, Spring 2014
Handed out March 13, 2014, due 11:00pm, March 17, 2014

NAME:

DATE and TIME of SITTING FOR the NON-SIMULATION PROBLEM:

You are not allowed to give or receive help or discuss this exam with anyone. You may use your class notes, anything that was handed out or posted for this class in Owl Space, including homework problems and solutions (including your own codes), a calculator and MATLAB. For those who prefer to write code in C or Fortran or R, those compilers and appropriate graphics packages for visualizing and displaying results are also allowed. Do not use other books, materials, or references (including web browsers, codes available on-line, codes from other students, etc.). A web browser can be used only for accessing data and code in http://terra.ece.rice.edu/ANNclass502/ and the materials posted at the course web site, and for MATLAB (R, C, Fortran) help. All problems are of equal weight, 33 points maximum each. 90 points = 100 %.

For the non-simulation problem (3) the time limit is 2 hours from the time of opening this envelope, and you have to do it in one sitting before you start working on simulations.

For simulation problems (1, 2) you have no other time constraint than the submission deadline, and you can do them in multiple sittings.

When finished, please write and sign the honor pledge in the space below. Scan and include it with your solutions. Submit your Exam through Owl Space (under Exam 1) by the deadline, as instructed on the envelope. The exam should consist of two files: a pdf with your write-up, and a separate text file containing all your code. Please find a few more reminders overleaf.

PLEDGE:

On my honor, I have neither given nor received any unauthorized aid on this exam.

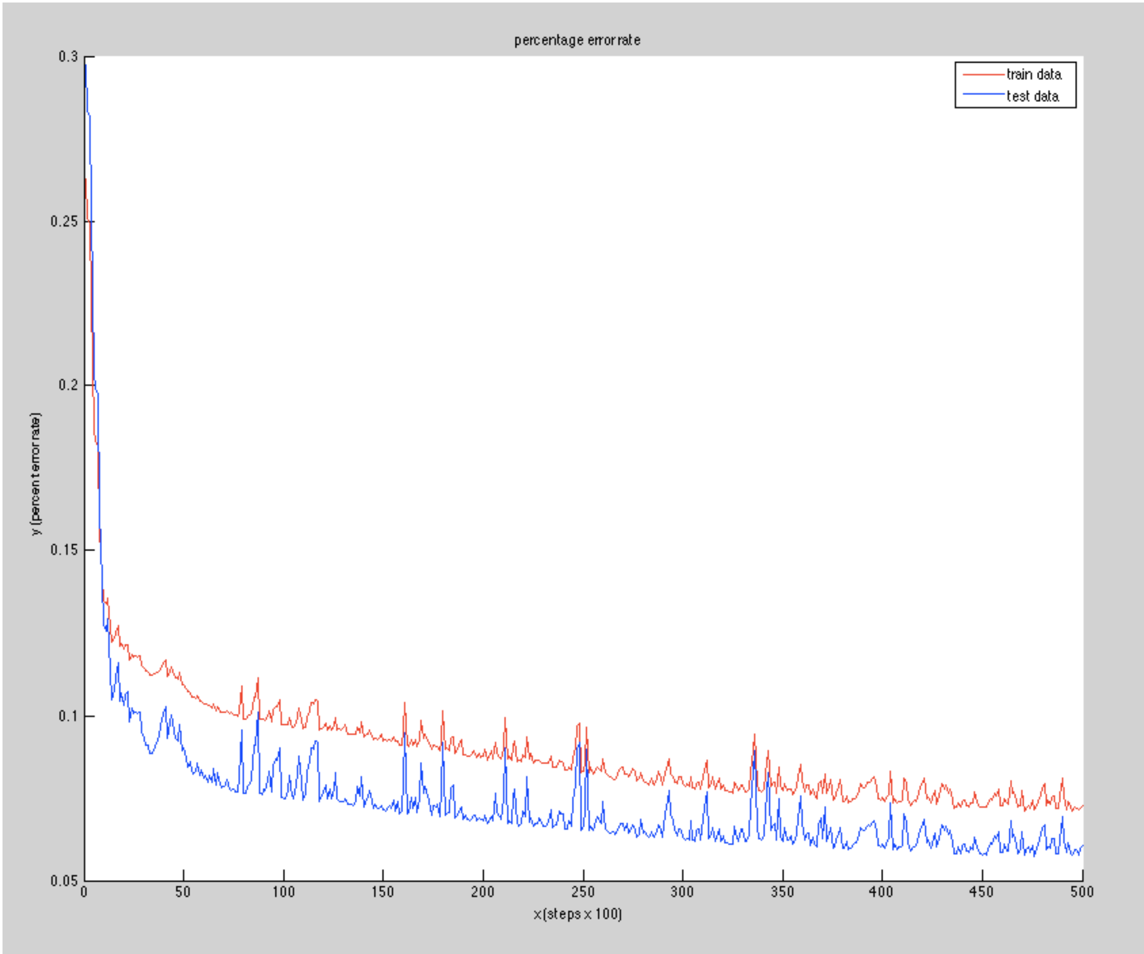
Xihao Zhu
[illegible]

Good luck!

P1.
(1.1)
For my parameters, I use 50000 steps and set learning rate as 0.02. A momentum term constant is 0.3; stopping criteria is stop after 50000 steps, 15 hidden layers and 100 as epoch size.

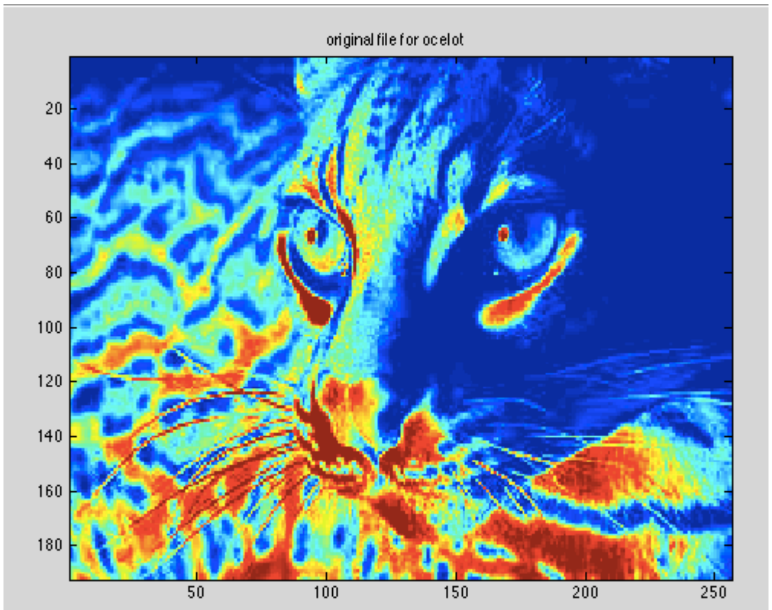
	value
learning steps	50000
learning rate	0.02
momentum term constant	0.3
layer #	15

For error rate calculation, I use average percent error. The formula is to get sum of absolute value of every network's output minus desired output. Then average the sum of error to be error per pixel. That is, once I got sum of errors(S) of my network output from every input, I calculate average error like this: E=S/(192*256). (there are 192*256 pixels as I understand)Then since data range is -1 to 1, so percent error Percent E=E/2.
Here is the plot(y is percent average error)

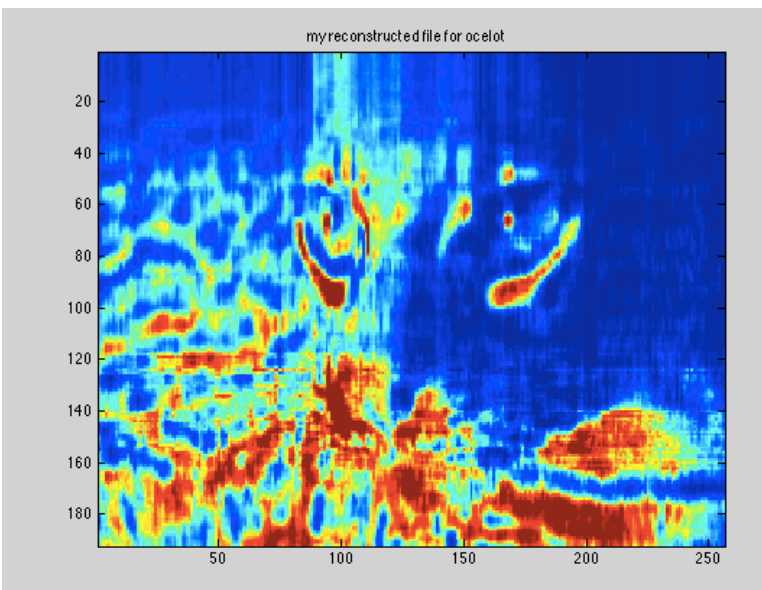


So finally I got average percent error rate for train data(ocelot) to be 8%, while that for test data(fruitfill) is 6%. I think this result is good.

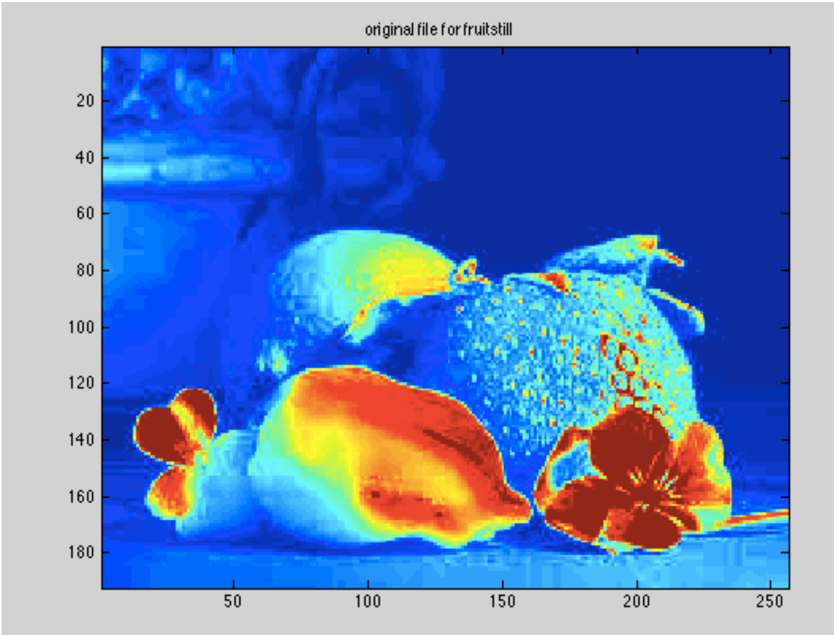
For Ocelot picture, here is original file:



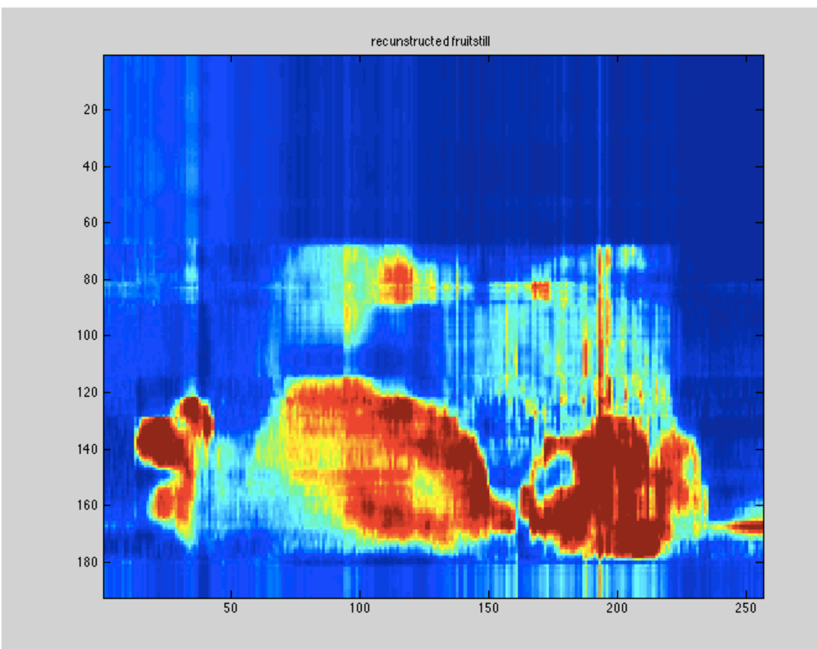
here is my reconstructed file:



For fruitfill:
Here is original file



Here is reconstructed file:



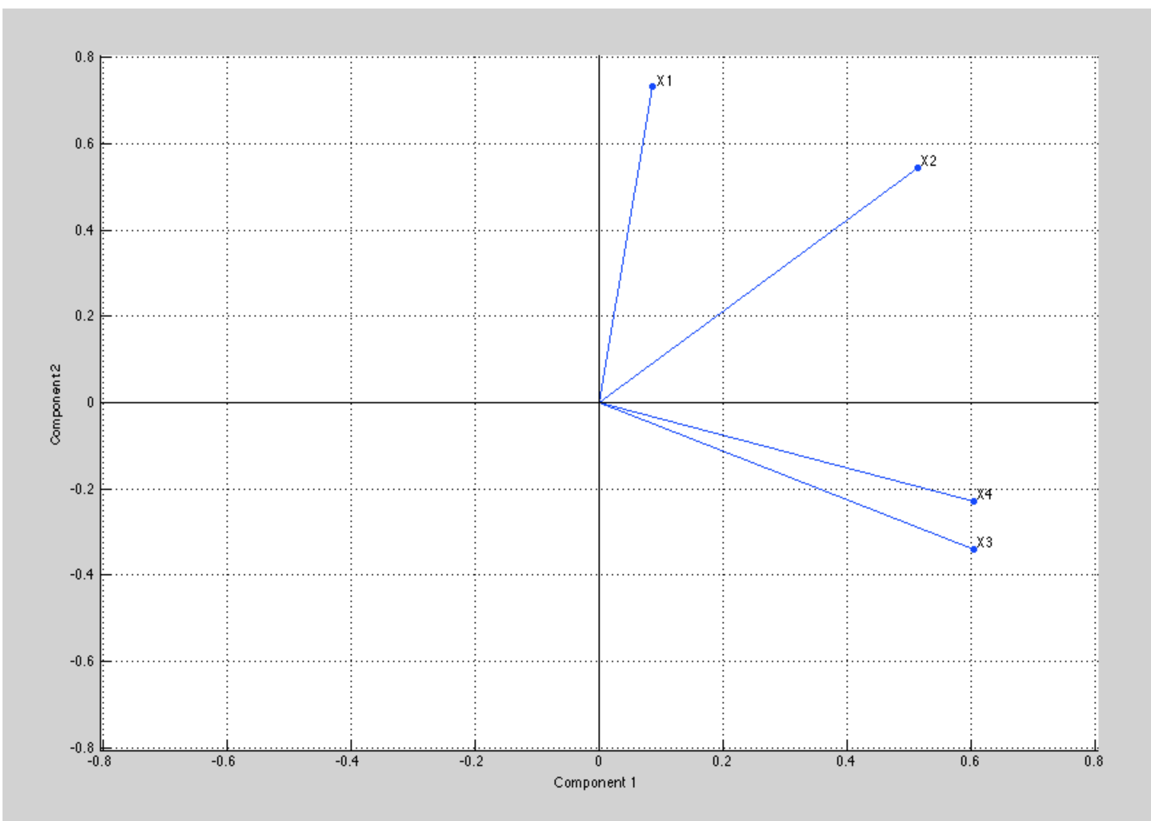
P2.
(2.1)
So I use x2 to compute the eigenvectors for the left quarter of this image.
Eigenvector with x2 preprocessed to have 0 means in columns.
Eigenvectors =
[

0.0869	0.7317	0.6212	0.2669
0.5129	0.5450	-0.5633	-0.3500
0.6041	-0.3392	0.5270	-0.4912
0.6031	-0.2275	-0.1381	0.7510

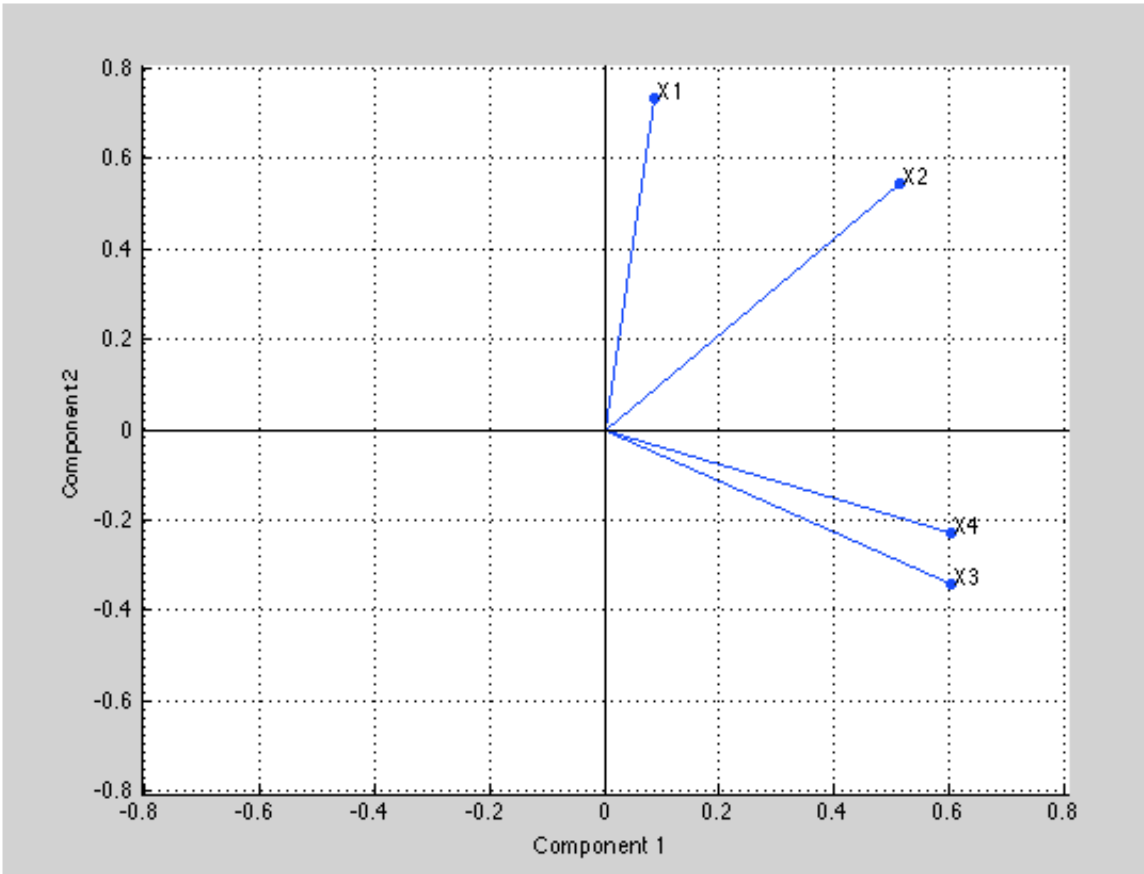
]

(2.2)
PCA analysis: 1st component:

0.0869	0.7317
0.5129	0.5450
0.6041	-0.3397
0.6037	-0.2284



Matlab's 1st PC



P3.

since Oja's algorithm is like this:

$$w_{ij}(t+1) = w_{ij}(t) + u[y_i(t)x_j(t) - y_i^2(t)w_{ij}(t)]$$

in P_1's configuration, using Back propagation

feed forward out

$$y = xw_1$$

$$\hat{x} = y \cdot v = xw_1 v$$

$$\therefore \delta_v = u(x - xw_1 v) \; (= u(x - yv))$$

$$dv = u(x - xw_1 v)\hat{x} = u(x - xw_1 v)(xw_1 v) = u(x-yv)(yv)$$

$$\delta_w = v \cdot \delta_v$$

$$dw = u \cdot \delta_w x$$

$$= u^3 (x - yv)^2 yv \cdot x$$

$$= u^3 (x^3 vy + xvy^3 - 2x^2y^2v)$$

$$= u^3 vxy(x^2 + y^2 - 2xy)$$

since $y = xw$

$$\therefore dw = u^3 v x^2 w(-2xy + y^2 + x^2)$$

Compare it to Oja's algo,

Well, $u(yX - y^2 W)$