COGS 109: Assignment #2

Due on Sunday, October 18, 2015

Tu, Zhuowen 2pm

Kyle Lee
A01614951

Problem 1

Please indicate below whether each problem is an example of supervised or unsupervised learning.

- Predict tomorrow's weather based on today's weather (a training set of the weather conditions in the past 10 years is provided to you).

  Solution: This is an example of supervised learning.

- Divide patients with lung cancers into 5 subgroups, based on their blood sugar levels (all patients blood sugar data are given to you).

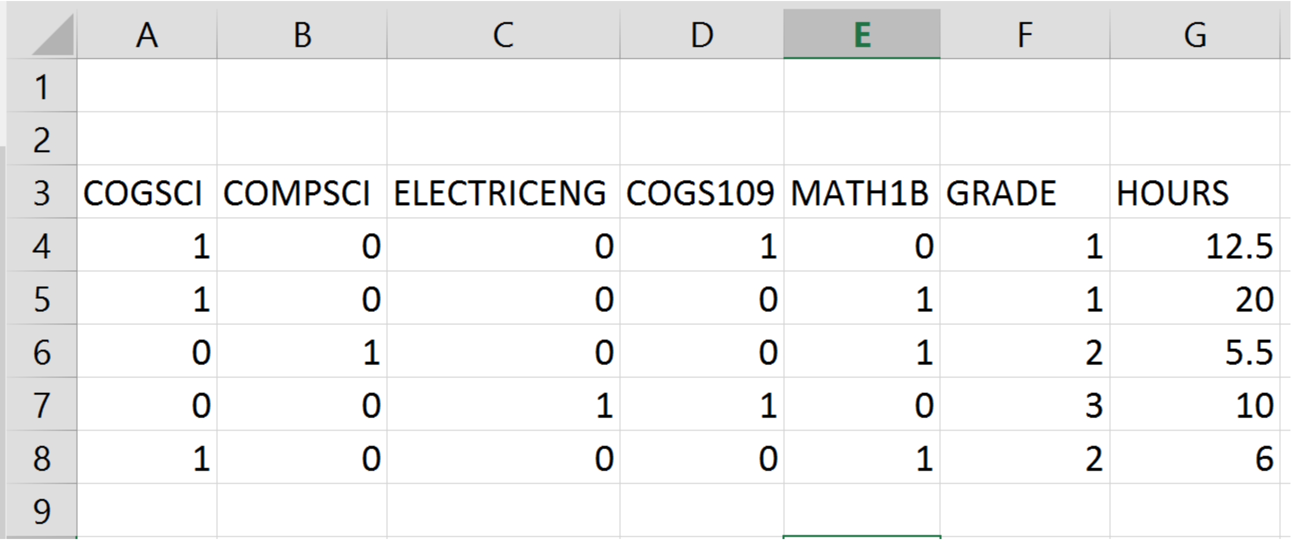
  Solution: This is an example of unsupervised learning.

- Check whether a persons height can predict his/her body weight.

  Solution: This is an example of unsupervised learning.

Problem 2

Please turn data above into a numeric matrix with proper coding for each type of feature by following the instructions given in the class about features. In your matrix, representing each sample as a row vector is preferred. If you decide to use column vector instead, please clearly state it in your answer.
Solution:



	A	B	C	D	E	F	G
1							
2							
3	COGSCI	COMPSCI	ELECTRICENG	COGS109	MATH1B	GRADE	HOURS
4	1	0	0	1	0	1	12.5
5	1	0	0	0	1	1	20
6	0	1	0	0	1	2	5.5
7	0	0	1	1	0	3	10
8	1	0	0	0	1	2	6
9							

I separated the majors into binary categorical data. I also encoded COGS109 and MATH1B to be their own separate binary categorical data. I encoded grades as 1-A,2-B,3-C. Hours remain the same. I demonstrated the data as row vectors.

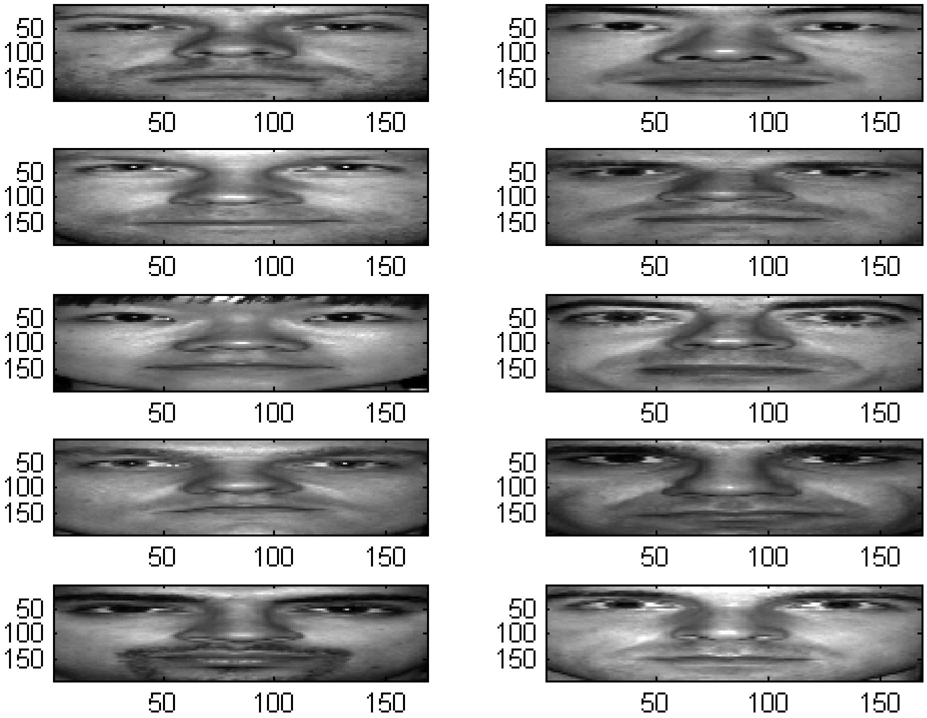
Problem 3

Please download the Yale face dataset (matlab matrices in double) from this link: https://sites.google.com/site/ucsdcogs109fall2015/assignments/assigment2/facesD.mat If you load this file into matlab, the first face/matrix can be accessed as e.g.$Yale_face1$. This dataset contains 10 faces of 10 individuals. Each face (image) is a 192 by 168 matrix.

Problem 3 continued on next page. . .

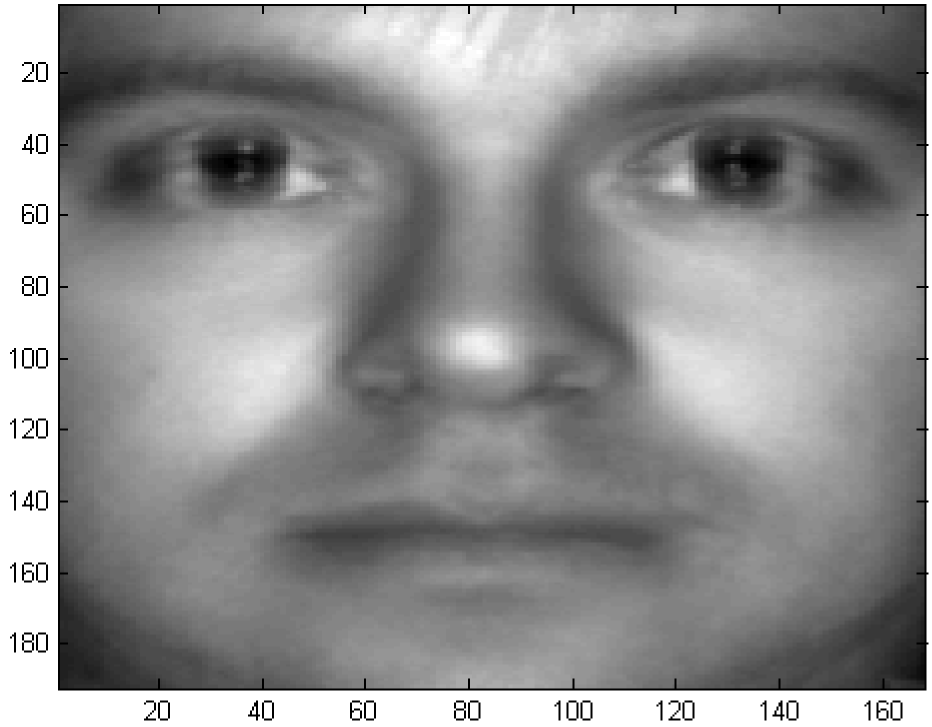
1. Create a five by two grid of subplots and plot these faces one by one using imagesc(). Images are in grayscale; therefore, remember to use colormap(gray) to make the corresponding colors (another way to display a matrix A is by using, e.g., imshow(A, []) where [] refers to doing the automatic scaling of values in A to 0-255).

   Solution:



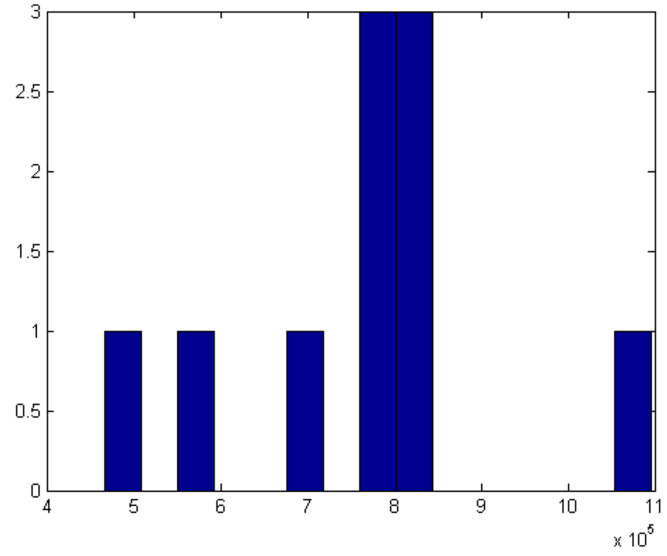
2. Compute mean face by averaging those 10 faces. Use imagesc() to show what this mean face looks like. (Hint: there are several ways to do this process. You can add all the faces together and then divide the matrix with a scalar, 10. Or, you can create a for-loop script to do the heavy work for you if you download the second file).

   Solution:



3. Calculate the distance between each face and the mean face based on the following instructions: subtract the face and the mean face; compute the absolute values of the subtracted image, and then sum the values in every matrix element into one single scalar value. This value represents the distance between faces (or more specifically, the L1 norm of the differences between faces). After obtaining the distance of each face from the mean face, use hist() with 15 bins to show the distribution of the distances.

   Solution:



Problem 4

This dataset contains a 50,000 by 1 column vector that represents the year of song production, and a 50,000 by 36 matrix in which each row represents 36 features of a song. Arrange the following plots in a figure with a two by two grid of subplots.

1. Plot out the histogram of the 1st sound feature of all 50,000 samples using the hist() function with 50 bins. **Solution: SEE BELOW FIGURE**

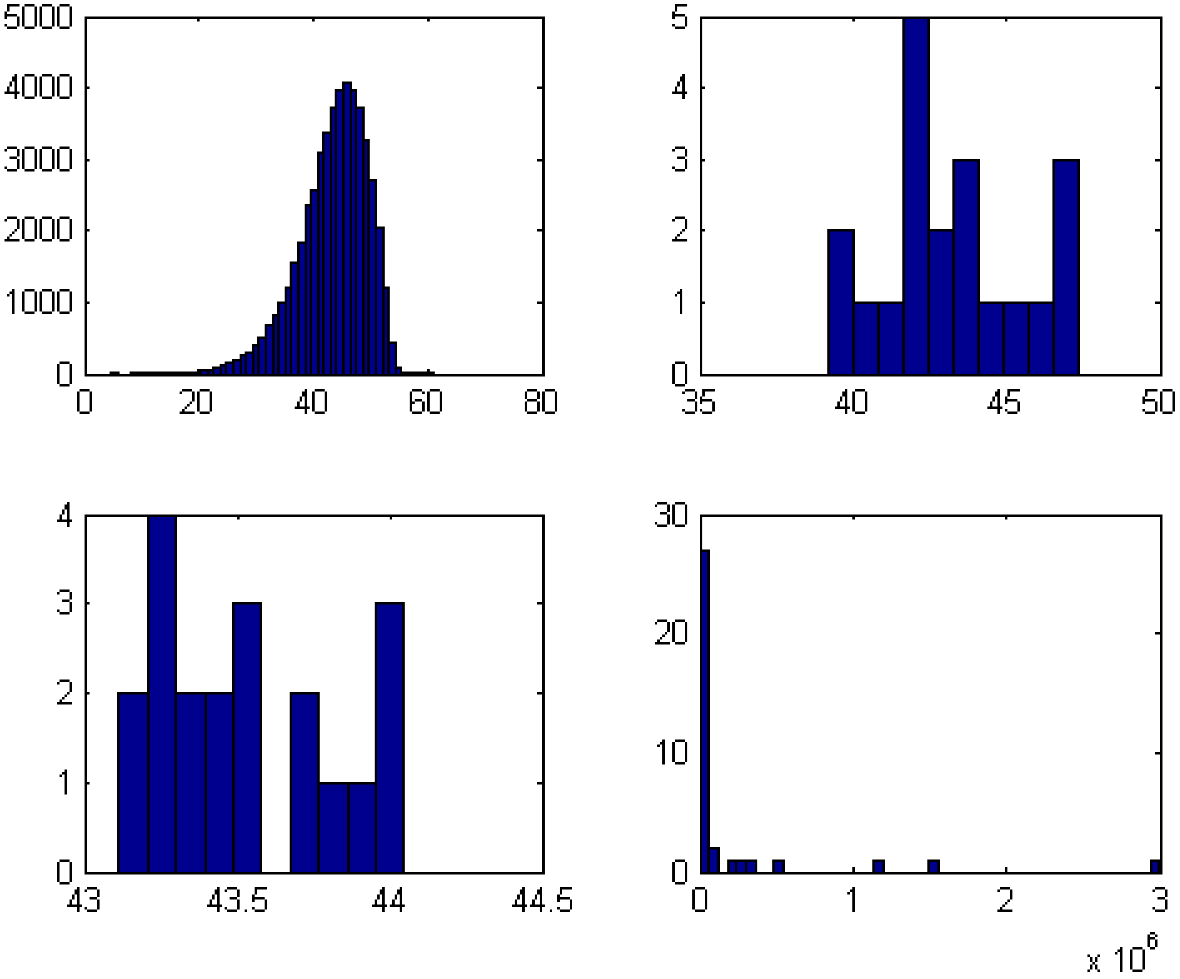
2. Calculate the sample mean of the 1st song feature for 10 songs randomly drawn from the dataset of 50,000 songs. You can use, for example, datasample(SoundCharacter, 10) to obtain 10 row vectors out of 50,000 samples. Repeat this procedure 20 times, and then use the hist() function with 10 bins to plot out the histogram of the calculated means. **Solution: SEE BELOW FIGURE**

3. Calculate the sample mean of the 1st song feature for 500 songs randomly drawn from the dataset of 50,000 songs. You can use, for example, datasample(SoundCharacter, 500) to obtain 500 row vectors out of 50,000 samples. Repeat this procedure 20 times, and then use the hist() function with 10 bins to plot out the histogram of the calculated means. **Solution: SEE BELOW FIGURE**

4. Calculate the variance of each feature of the 50,000 samples. Use hist() with 50 bins to show the histogram of the feature variances. Indicate which feature has the smallest variance and which one has the largest variance.

   Solution:

```

%% Command Window %%
homework2

% Refers to index of minimum variance
ans =

    11

% Refers to index of maximum variance
ans =

    14
```



Problem 5

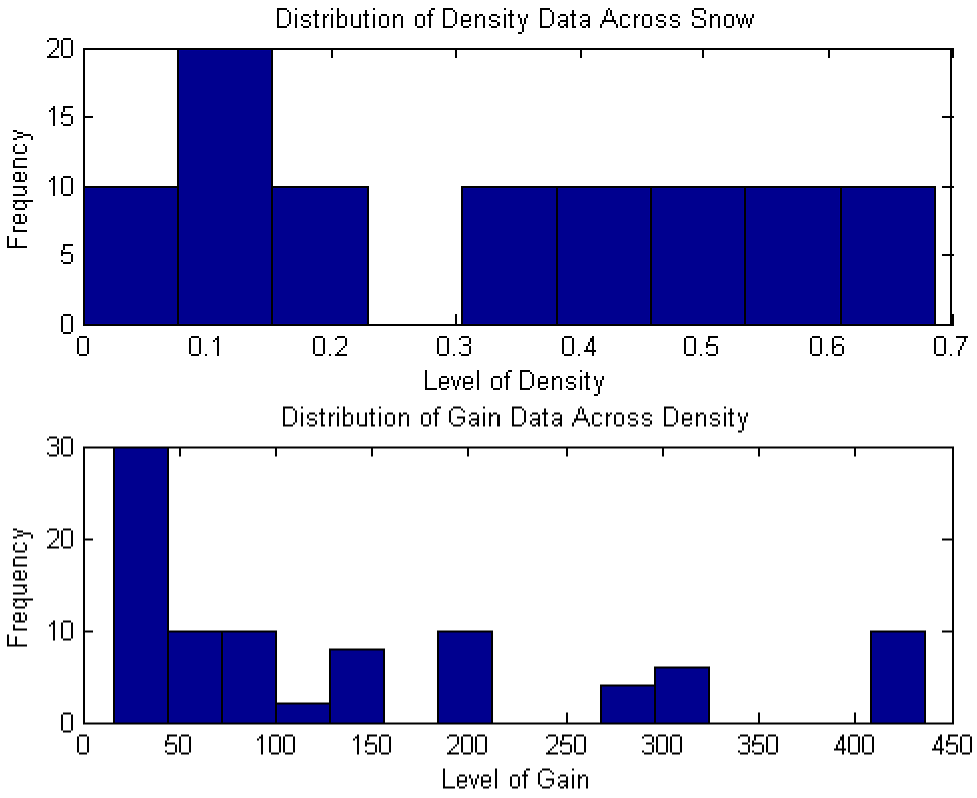
Using your dataset from HW1 or a new data set, choose 2-3 numerical (continuous or discrete) features from your data. Briefly describe each feature and plot each in a separate histogram. Choose the appropriate bin size for each feature to best represent the distribution of the data. Label the axes and provide a title for each plot.

Solution:

Main source of Water for Northern California comes from the Sierra Nevada mountains. To help monitor the water supply, the Forest Service of the United States Department of Agriculture (USDA) operates a gamma transmission snow gauge in the Central Sierra Nevada near Soda Springs, CA. The gauge is used to determine a death profile of snow density. The snow gauge does not disturb the snow in the measurement process, which means the same snow-pack can be measured over and over again. With this replicate measurements on the same volume of snow, researchers can study snow-pack settlement over the course of the winter season and the dynamics of rain on snow. When rain falls on snow, the snow absorbs the water up to a certain point, after which flooding occurs. The denser the snow pack the less water it can absorb. Analysis the snow pack profile may help with monitoring the water supply and flood management. The gauge does not directly measure snow density. The density reading is converted from a measurement of gamma ray emissions. Due

Problem 5 continued on next page...

to instrument wear and radioactive source decay, there may be changes over the seasons in the functions used to cover the measured values into density readings. To adjust the conversion method, a calibration run is made each year at the beginning of the winter season.



The first histogram shows that data was evenly collected throughout the profiling. The second histogram shows the level of gains which we observe does not have a uniform distribution.

Homework Code

```
%% Homework #2 %%
%% Problem 3 %%
%% PART A %%
% Create 5x2 grid
figure
for i = 1:10
    % Plot each face to each index
        subplot(5,2,i);
        imagesc(Yale_faces{i});
end
colormap(gray)
%% PART B %%
% Initialize variables
sum2 = 0;
A = 0;
for i = 1:10
```

```
    % Convert to double
    A = double(Yale_faces{i});
    % Sum through every iteration
    sum2 = sum2 + A;
end
% Calculate mean
average = sum2/10;
% Create new figure, use gray color mapping, and plot
figure
colormap(gray)
subplot(1,1,1)
imagesc(average)
%% Part C %%
figure
z = zeros(1,10);
for i = 1:10
    % Calculate L1 Norm
    a = [abs(double(Yale_faces{i}) - average)];
    % Sum matrix
    z(i) = sum(sum(a));
end
hist(z,15)
%% Problem 4%%
figure
%% PART A %%
% Plot out histogram of 1st sound feature of all 50,000 samples
% using hist() function with 50 bins
subplot(2,2,1)
hist(SoundCharacter(:,1),50);

%% PART B %%
subplot(2,2,2)
%Initialize y1 to hold means
y1 = zeros(1,20);
for i=1:20
    % Sample 10 songs
    ithsample = datasample(SoundCharacter(:,1), 10);
    y1(i) = mean(ithsample);
end
% Print histogram with 10 bins
hist(y1,10)

%% Part C %%
subplot(2,2,3)
% Initialize y2 to hold means
y2 = zeros(1,20);
for i=1:20
    % Sample 500 songs
    ithsample = datasample(SoundCharacter(:,1), 500);
    y2(i) = mean(ithsample);
end
% Print histogram
hist(y2,10)
```

```

%% PART D %%
subplot(2,2,4)
% Initialize y3 to hold variances
y3 = zeros(1,36);
for i=1:36
    y3(i) = var(SoundCharacter(:,i));
end
% Print out histogram with 50 bins
hist(y3,50)
% Part d

% Find min and max of variance
min_var = min(y3);
max_var = max(y3);

% Find indices of minimum variance and maximum variance
find(y3 == min_var)
find(y3 == max_var)

%% EXTRA CREDIT %%
% I will be using a set on gauge data
figure
subplot(2,1,1)
hist(densit,9)
title(’Distribution of Density Data Across Snow’)
xlabel(’Level of Density’)
ylabel(’Frequency’)

%2nd Histogram
subplot(2,1,2)
hist(ygain,15)
title(’Distribution of Gain Data Across Density’)
xlabel(’Level of Gain’)
ylabel(’Frequency’)
```