CS306: Data Analysis and Visualization
Lab 8: Report

Rajdeep Pinge 201401103
Aditya Joglekar 201401086

Objective:

To compare methods/models using correlation coefficient and CI, and draw inferences.

Experiment 1:

See Results_Raw_Data_24obs.xls. It provides subjective quality scores for a set of 144 videos (each video is rated by 24 observers) and the MOS. The objective quality for these videos from two objective methods A and B are also provided in the file.

1. Compute the linear correlation coefficient between MOS and the two methods separately. Which method gives higher correlation with MOS?

Correlation coefficient for MOS and method A	0.7905
Correlation coefficient for MOS and method B	0.8156

Both values are **very close to each other** and are in th**e higher quartile of correlation** coefficients.

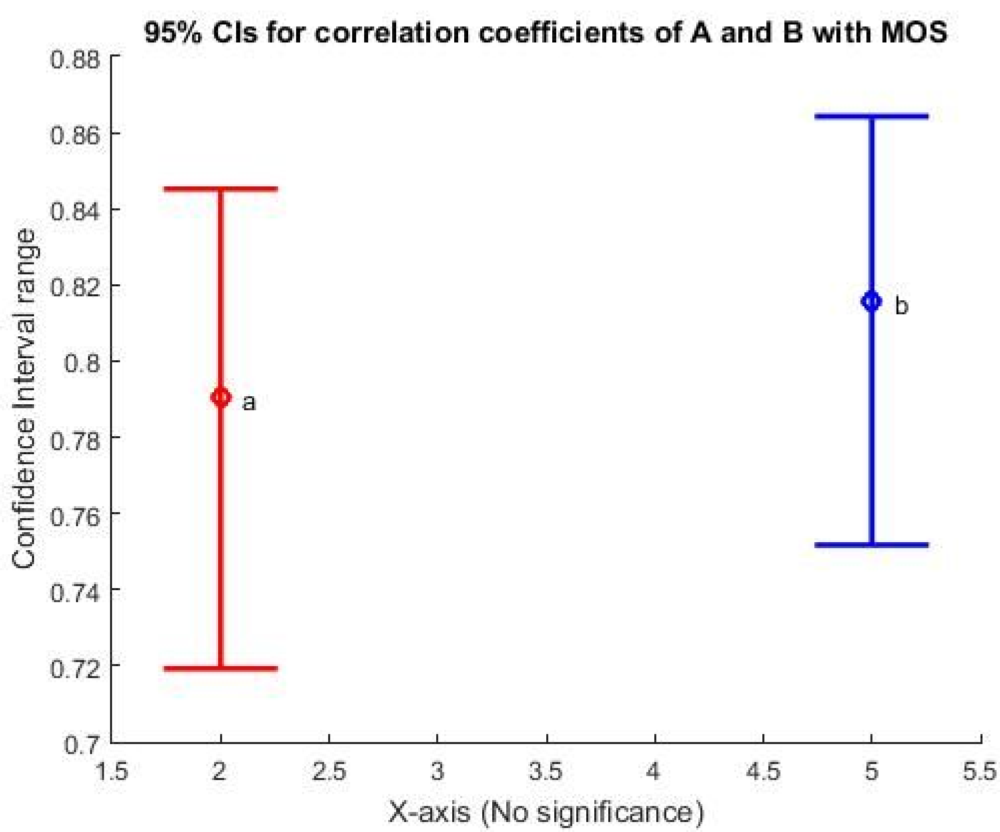
The reasoning behind c**omputing correlation coefficients is that we want to identify the method which can be used in place of the subjective MOS method**. Thus, the objective method must have scores similar to the MOS scores in the sense that even though the **scales might be different, a bad quality video should give less score in both the methods and the opposite must happen for the good quality case.**

If we assume that the **MOS scores are reliable then we can assume that if the video quality is high then MOS will be higher and MOS will be lower if the quality is lower.** If our objective method **really captures the video quality then it too must behave in the same manner**. Thus, if the two method**s are capturing the same thing, then one expects them to have a high degree of correlation**. This is certainly true **if they depend on a common causal variable.**

The above **correlation coefficients are quite close to each other** and show a **relatively high correlation with MOS**. This means that both tests are **likely to be good substitutes** for MOS bu**t Method B is slightly better** as far as its correlation with A is concerned.

However **further analysis is required** to determine the **statistical significance** of this observation.

2. Show side-by-side the two correlation values computed in the previous part along with the corresponding 95% confidence intervals (use error bars to visualize the CI). Based on this, is one method statistically better than the other?



1. Clearly, the **CI for the correlation coefficient of the two methods** with the given MOS **scores are overlapping.**
2. We have **chosen a significance level of 0.05**. Thus, given that the experiment is repeated a large number of times, **we can infer that 95% of the time**, the **correlation coefficient of the population MOS and the objective methods** will lie in the respective CI intervals.
3. The intervals are **pretty narrow** and are located in the **region of high correlation**.
4. Thus, since the intervals overlap we cannot make a definitive statement with regards to which of the methods are better. This, indicates that the conclusion in part 1 might have been by chance.
5. However, from point 3 we can say that both are more or less capable for acting as substitutes for the MOS as both have a statistically significant high degree of correlation with the MOS.
6. But we can argue that according to a **95% confidence measure, B's population correlation coefficient cannot be worse than A's** and in some **cases will in fact perform better than it.** So if one has to choose between the two, we are **better off with B**.

3. An objective score (either from method A or B) is deemed an outlier if it lies outside the confidence interval of the corresponding MOS. Using this, compute the percentage of outliers for the two methods A and B. Based on this, which method, A or B, is better?

Percentage outlier A- 65.97

Percentage outlier B- 56.25

We analyze the implications of the above in Q 4. below

4. Based on the correlation coefficients in part 1 and the percentage of outliers in part 3, comment which method A, B, or none of the two, is likely to be practically useful.

Observations:

1. Based on the **correlation coefficients in part 1, we could daresay that method B is slightly better**. However, we mus**t realize that this was the result for the given sample of observers**. We needed **more analysis before** we could reach a conclusion.
2. In part 2, we **could not reach a decisive conclusion** about which is the better method because of the overlap.
3. In part 3, **method B has a slightly less number of scores which lie outside the CI** of the MOS for a particular video algorithm from the 144 types provided. Note, we have made **a big assumption** in how the outlier is defined.
4. We **say that if the objective method's score falls outside the MOS CI, it is an outlier**. However, remember that the **mean MOS of the population can fall anywhere in the CI**. There is a **possibility that a score falls outside the CI but is quite close to the population mean**[this will happen if score is just outside and the population mean is close to the edge of the interval]. We **assume that such cases are rare and that lying inside the CI is a good indicator of the closeness to the population mean MOS**.
5. Under **assumption in point 4, we can say that method B is again slightly better** as it has a lesser number of outliers.
6. However, a r**elatively high number of outliers (> 50 %) indicates that both these methods might not be suitabl**e because of the observation in point 4. In Spite of g**iving relatively high correlations, both methods give a large number of outliers**. For a better estimate: we shou**ld thus also look at the distance which rejected values[of A or B] are** at from the CI.

Experiment 2:

Load data_lab8.mat. 'MOS' represents the averaged subjective quality score for 866 images. The quality of these images was also computed using two objective methods: 'PSNR' and 'SSIM'.

1. Plot side-by-side the linear correlation coefficients for PSNR and SSIM with MOS, and also draw the corresponding 95% confidence intervals. Which method, PSNR or SSIM, is statistically better?

Correlation coefficient for MOS and PSNR	0.8046
Correlation coefficient for MOS and SSIM	0.8169

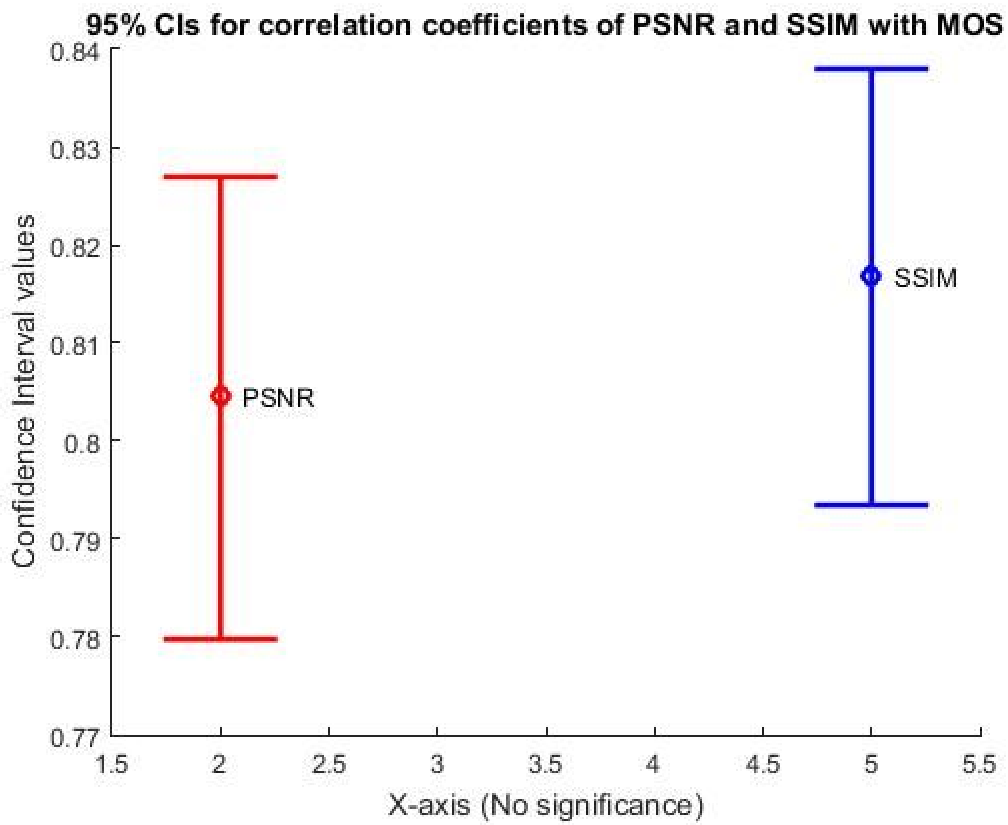
- The above correlation coefficients are very high indicating that there is very less difference between the subjective and objective methods. Also, **the SSIM method seems to be the better method among the two**. But there are **some reservations** which must not be overlooked while concluding this. We need to **test statistical significance** to resolve them.
- It is not clear **how representative the observers chosen are to the total population i.e. if we change the people who determined the quality of the images the MOS** which is subjective might change and the correlation coefficient also.
- We also don't know if images of various different pixel ranges (i.e. pixel quality) are being considered or not. It may be possible that the PSNR is better for one type of pixel range images while SSIM is for different. If images are mixture which is balancing out such differences, then it would not be a good representation. Hence we need to do additional work before concluding anything.

Here, **since the underlying images are same for all the quality measurement methods**, the degree of freedom for the t score calculation is taken as df = n-1 = 866-1 = 865

For the 95% confidence interval $\alpha = 0.05$

Therefore $\alpha/2 = 0.025$

T score = 1.963



From the above graph, we can clearly see that the **two CIs overlap**.
The result can be interpreted based on the following image:

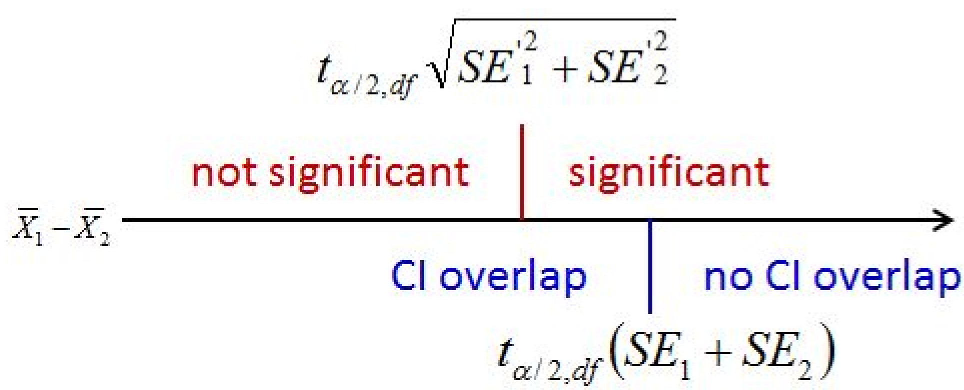


Fig.: Inference from CIs

Now, as per the above rule, **we cannot conclude anything because the difference may or may not be significant. Hence we cannot comment on which method among the given two (PSNR and SSIM) is statistically better.**

But note that if **we have to choose one, we should go with SSIM, hoping that because from the above figure we see that it can't perform worse than the other and in some cases it will perform better**.

2. Instead of obtaining the correlation over all the 866 images, we decide to analyze 'local' correlation coefficients. To that end, we randomly select 20% images and compute the correlation between their corresponding MOS and the two methods (PSNR and SSIM). This process is repeated over a large number of iterations (use 20000) and in each iteration we obtain one correlation coefficient for PSNR and another for SSIM. Finally, we obtain the mean correlation coefficient for PSNR and SSIM over 20000 iterations, and the corresponding 95% CI. Based on this approach, which method, PSNR or SSIM, is statistically better? Does the conclusion agree with that in part 1?

Correlation coefficient for MOS and PSNR	0.8047
Correlation coefficient for MOS and SSIM	0.8172

Observe that **due to a large number of iterations, these values are very close the the correlation coefficients got in the previous part[recall law of large numbers]**. The similar arguments as given in part 1 apply here but the reservations mentioned are also same. Hence further work is required before concluding anything.

Here also since the underlying images are same for all the quality measurement methods, the degree of freedom for the t score calculation is taken as n-1.

But this **n is for the randomly sampled 20% images[** sample's size is smaller**]** from the given images

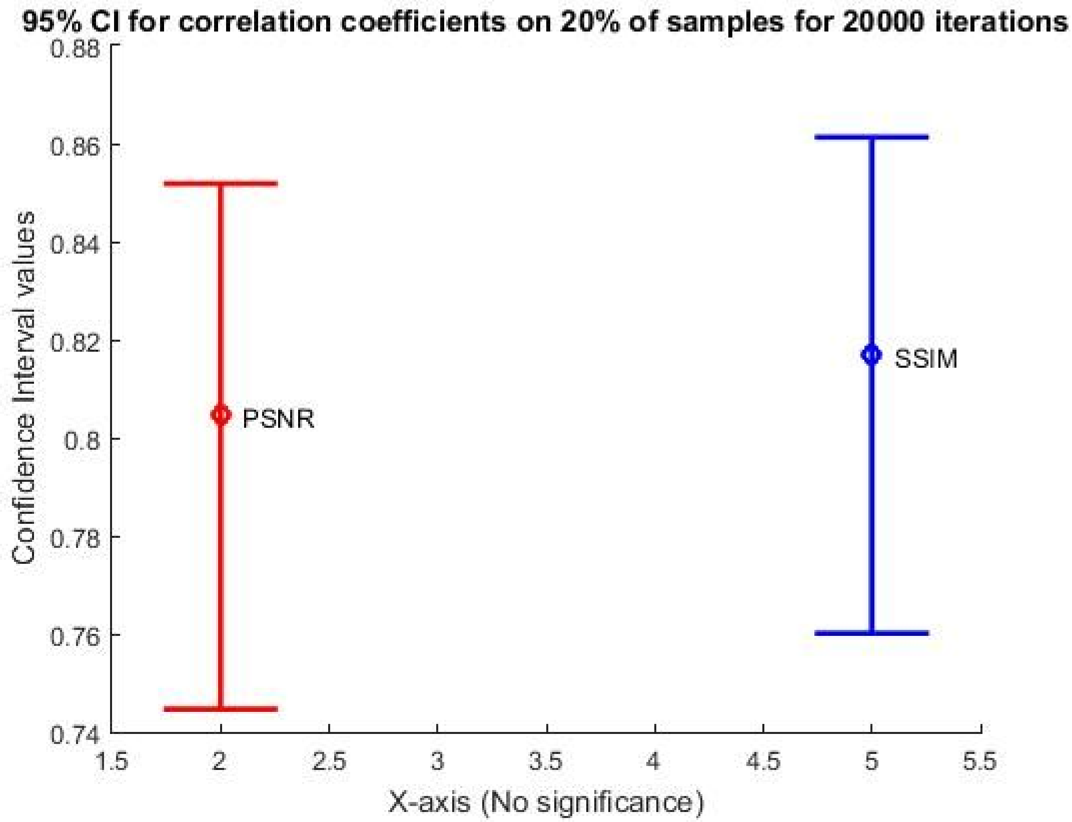
Hence n = 20% of 866 = 174

Therefore df = 174-1 = 173

For the 95% confidence interval $\alpha = 0.05$

Therefore $\alpha/2 = 0.025$

T score = 1.974



Observe that here the **CIs cover a larger range [attributable to the smaller sample size] but still both the CIs overlap**. Hence as per the above rule, **we cannot conclude anything because this may either mean that the difference is not significant or that the difference is significant but is not visible in CI.**

Hence we cannot comment on which method among the given two (PSNR and SSIM) is statistically better.

But note that if **we have to choose, we should go with SSIM, because from the above figure we see that it can't perform worse than the other and in some cases it will perform better**.

Codes:

Code Q1- Part 1 and 2

```
clear;
close all;

data=xlsread('Results_Raw_Data_24obs.xls');
%data= load('Results_Raw_Data_24obs.xls')

mos = data(:,26);
a = data(:,27);
b = data(:,28);

corr_m_a=corr(mos,a); % r_a
corr_m_b=corr(mos,b); % r_b

% calc f, then calc f CI, then r CI

f_m_a= 0.5*log( (1+corr_m_a)/(1-corr_m_a) ); % f of corr of m and A

f_m_b= 0.5*log( (1+corr_m_b)/(1-corr_m_b) );
% old code, % doubt what is deg of freedom

n= length(a);
SEM1_m_a = 1/sqrt(n-3);              % Standard Error
ts_m_a = tinv([0.025  0.975],n-1);   % T-Score, deg of freedom, alpha= 0.05,
returns 2 critical values upper and lower
CI_m_a = f_m_a + ts_m_a*SEM1_m_a                 % takes care of both upper and
lower
%CI1_L= mean(island1_txt) - ts1*SEM1


SEM2_m_b = 1/sqrt(n-3);              % Standard Error
ts_m_b = tinv([0.025  0.975],n-1);   % T-Score, deg of freedom, alpha= 0.05,
returns 2 critical values upper and lower
CI_m_b = f_m_b + ts_m_b*SEM2_m_b

% takes care of both upper and lower

CI_r_m_a_lower= ( exp(2*CI_m_a(1))  -1)/ ( exp(2*CI_m_a(1) ) +1)
CI_r_m_a_higher= ( exp(2*CI_m_a(2))  -1)/ ( exp(2*CI_m_a(2) ) +1)
%CI_r_m_b= exp(1).^(2*CI_m_b-1)/ (exp(1).^(2*CI_m_b +1))
```

```
CI_r_m_b_lower= (exp(1)^(2*CI_m_b(1))-1)/ (exp(1)^(2*CI_m_b(1))+1)
CI_r_m_b_higher= (exp(1)^(2*CI_m_b(2))-1)/ (exp(1)^(2*CI_m_b(2))+1)



%axis([1 10 CI_r_m_a_lower CI_r_m_b_higher+1])
line([4-0.25,4+0.25],[CI_r_m_a_higher,CI_r_m_a_higher],[0,0],'LineStyle','-','C
olor','r','LineWidth',2);
line([4-0.25,4+0.25],[CI_r_m_a_lower,CI_r_m_a_lower],[0,0],'LineStyle','-','Col
or','r','LineWidth',2);
line([4,4],[CI_r_m_a_lower,CI_r_m_a_higher],[0,0],'LineStyle','-','Color','r','
LineWidth',2);

hold on

line([7-0.25,7+0.25],[CI_r_m_b_higher,CI_r_m_b_higher],[0,0],'LineStyle','-','C
olor','r','LineWidth',2);
line([7-0.25,7+0.25],[CI_r_m_b_lower,CI_r_m_b_lower],[0,0],'LineStyle','-','Col
or','r','LineWidth',2);
line([7,7],[CI_r_m_b_lower,CI_r_m_b_higher],[0,0],'LineStyle','-','Color','r','
LineWidth',2);

title('CI for correlation MOS_A and correlation MOS_B','FontSize',16)
ylabel('Beak sizes','FontSize',16)
```

Code Q1- Part 3 and 4

```
clear;
close all;

data=xlsread('Results_Raw_Data_24obs.xls');
%data= load('Results_Raw_Data_24obs.xls')

mos = data(:,26);
a = data(:,27);
b = data(:,28);
```

```
%corr_m_a=corr(mos,a); % r_a
%corr_m_b=corr(mos,b); % r_b

% calc f, then calc f CI, then r CI
n= 24; % imp 24 observers

A_count=0;

B_count=0;

ts_mos = tinv([0.025  0.975],n-1);

for i=1:144


SEM_mos = std(data(i,1:24))/sqrt(n);              % Standard Error
      % T-Score, deg of freedom, alpha= 0.05, returns 2 critical values upper
and lower
CI_mos = data(i,26) + ts_mos*SEM_mos; % mos ci                    % takes care
of both upper and lower
%CI1_L= mean(island1_txt) - ts1*SEM1


if (data(i,27)>=CI_mos(1) & data(i,27)<=CI_mos(2)) % does A lie
A_count= A_count+1;
end

if (data(i,28)>=CI_mos(1) & data(i,28)<=CI_mos(2)) % does A lie
B_count= B_count+1;
end

end

A_per_outlier= 1- A_count/144
B_per_outlier= 1- B_count/144
%%% last

%%
```

Q2. Part 1

```
% Author - Rajdeep Pinge
% Date 7th March, 2017
```

```
% Code to compare the two objective methods: PSNR and SSIM of measuring image
quality
% using the help of their correlation factor with MOS subjective
% measurement taken from observers.

clear;
close all;


load('data_lab8');

mos_vals = MOS';
psnr_vals = PSNR';
ssim_vals = SSIM';

% find correlation coefficient for psnr
r_psnr = corr(mos_vals, psnr_vals);

% find correlation coefficient for ssim
r_ssim = corr(mos_vals, ssim_vals);

% use Fischer transformation
f_psnr = 0.5 * log( (1+r_psnr) / (1-r_psnr) );
f_ssim = 0.5 * log( (1+r_ssim) / (1-r_ssim) );


% generate CI of r for psnr for alpha = 0.05, df = 865
% NOTE: since underlying images on which the tests have been performed is
% same, the degree of freedom = n-1
n = numel(mos_vals);
t_0_975 = 1.963;         % alpha = 0.05, df = 865

% find bounds of f value
f_lower_psnr = f_psnr - t_0_975 / sqrt(n-3);
f_upper_psnr = f_psnr + t_0_975 / sqrt(n-3);

% Find lower and upper bound for CI
r_lower_psnr = (exp(2*f_lower_psnr) - 1) / (exp(2*f_lower_psnr) + 1);
r_upper_psnr = (exp(2*f_upper_psnr) - 1) / (exp(2*f_upper_psnr) + 1);



% generate CI of r for ssim
n = numel(mos_vals);
t_0_975 = 1.963;         % alpha = 0.05, df = 865

% find bounds of f value
```

```
f_lower_ssim = f_ssim - t_0_975 / sqrt(n-3);
f_upper_ssim = f_ssim + t_0_975 / sqrt(n-3);

% Find lower and upper bound for CI
r_lower_ssim = (exp(2*f_lower_ssim) - 1) / (exp(2*f_lower_ssim) + 1);
r_upper_ssim = (exp(2*f_upper_ssim) - 1) / (exp(2*f_upper_ssim) + 1);


% plot CI of both for significance level = 0.05
figure
line([2-0.25,2+0.25],[r_upper_psnr,
r_upper_psnr],[0,0],'LineStyle','-','Color','r','LineWidth',2);
line([2-0.25,2+0.25],[r_lower_psnr,r_lower_psnr],[0,0],'LineStyle','-','Color',
'r','LineWidth',2);
line([2,2],[r_lower_psnr,r_upper_psnr],[0,0],'LineStyle','-','Color','r','LineW
idth',2);
hold on
plot([2,2], r_psnr, 'ro','LineWidth',2)
text(2+0.1,r_psnr,'PSNR')

line([5-0.25,5+0.25],[r_upper_ssim,
r_upper_ssim],[0,0],'LineStyle','-','Color','b','LineWidth',2);
line([5-0.25,5+0.25],[r_lower_ssim,r_lower_ssim],[0,0],'LineStyle','-','Color',
'b','LineWidth',2);
line([5,5],[r_lower_ssim,r_upper_ssim],[0,0],'LineStyle','-','Color','b','LineW
idth',2);
hold on
plot([5,5], r_ssim, 'bo','LineWidth',2)
text(5+0.1,r_ssim,'SSIM')
title('95% CIs for correlation coefficients of PSNR and SSIM with MOS')
xlabel('X-axis (No significance)')
ylabel('Confidence Interval values')
```

Q2. Part 2 - taking 20% samples and iterating 20000 times

```
% Author - Rajdeep Pinge
% Date 7th March, 2017

% Code to compare the two objective methods: PSNR and SSIM of measuring image
quality
% using the help of their correlation factor with MOS subjective
```

```
% measurement taken from observers.

% Here the local correlation factors are computed for a large number of
iterations
% by taking 20% of the total samples each time, finding their correlation
% factors and then taking the mean as the correlation coefficient of the
% whole process. The CIs are found out and the results are compared with
% part 1

clear;
close all;

load('data_lab8');

mos_mat = MOS';
psnr_mat = PSNR';
ssim_mat = SSIM';

iterations = 20000;

% calculate total size
n = numel(mos_mat);

% calculate 20% size
n_20perc = ceil(0.2 * n);

% create matrices to store correlation coefficients in each iteration
r_psnr_arr = zeros(iterations, 1);
r_ssim_arr = zeros(iterations, 1);

% Iterate 20000 times to remove the effect of randomness
for i = 1:iterations

      % find samples randomly
      index_20perc = randperm(n, n_20perc);

      mos_vals = mos_mat(index_20perc);
      psnr_vals = psnr_mat(index_20perc);
      ssim_vals = ssim_mat(index_20perc);

      % find correlation coefficient for psnr
      r_psnr_arr(i) = corr(mos_vals, psnr_vals);

      % find correlation coefficient for ssim
      r_ssim_arr(i) = corr(mos_vals, ssim_vals);

end
```

```
% take mean of the stored values as the actual correlation coefficient
r_psnr = mean(r_psnr_arr);

r_ssim = mean(r_ssim_arr);

% use Fischer transformation
f_psnr = 0.5 * log( (1+r_psnr) / (1-r_psnr) );
f_ssim = 0.5 * log( (1+r_ssim) / (1-r_ssim) );

% generate CI of r for psnr for alpha = 0.05, df = 173
% NOTE: since underlying images on which the tests have been performed is
% same, the degree of freedom = n-1
t_0_975 = 1.974;         % alpha = 0.05, df = 173

% find bounds of f value
f_lower_psnr = f_psnr - t_0_975 / sqrt(n_20perc-3);
f_upper_psnr = f_psnr + t_0_975 / sqrt(n_20perc-3);

% Find lower and upper bound for CI
r_lower_psnr = (exp(2*f_lower_psnr) - 1) / (exp(2*f_lower_psnr) + 1);
r_upper_psnr = (exp(2*f_upper_psnr) - 1) / (exp(2*f_upper_psnr) + 1);



% generate CI of r for ssim for alpha = 0.05, df = 173
t_0_975 = 1.974;         % alpha = 0.05, df = 173

% find bounds of f value
f_lower_ssim = f_ssim - t_0_975 / sqrt(n_20perc-3);
f_upper_ssim = f_ssim + t_0_975 / sqrt(n_20perc-3);

% Find lower and upper bound for CI
r_lower_ssim = (exp(2*f_lower_ssim) - 1) / (exp(2*f_lower_ssim) + 1);
r_upper_ssim = (exp(2*f_upper_ssim) - 1) / (exp(2*f_upper_ssim) + 1);


% plot CI of both for significance level = 0.05
figure
line([2-0.25,2+0.25],[r_upper_psnr,
r_upper_psnr],[0,0],'LineStyle','-','Color','r','LineWidth',2);
line([2-0.25,2+0.25],[r_lower_psnr,r_lower_psnr],[0,0],'LineStyle','-','Color',
'r','LineWidth',2);
line([2,2],[r_lower_psnr,r_upper_psnr],[0,0],'LineStyle','-','Color','r','LineW
idth',2);
hold on
plot([2,2], r_psnr, 'ro','LineWidth',2)
text(2+0.1,r_psnr,'PSNR')
```

```
line([5-0.25,5+0.25],[r_upper_ssim,
r_upper_ssim],[0,0],'LineStyle','-','Color','b','LineWidth',2);
line([5-0.25,5+0.25],[r_lower_ssim,r_lower_ssim],[0,0],'LineStyle','-','Color',
'b','LineWidth',2);
line([5,5],[r_lower_ssim,r_upper_ssim],[0,0],'LineStyle','-','Color','b','LineW
idth',2);
hold on
plot([5,5], r_ssim, 'bo','LineWidth',2)
text(5+0.1,r_ssim,'SSIM')
title('95% CI for correlation coefficients on 20% of samples for 20000
iterations')
xlabel('X-axis (No significance)')
ylabel('Confidence Interval values')
```