Problem Set - Resampling

1. Bootstrap

```
%Problem Set
%
%The variables x and y represent errors made by a particular subject when
%reaching with his hand to visual targets. The x = before prism goggles,
%y = after goggles.

%The dataset consists of N=10 reaches with each hand, The data should be
%obtained using the supplied function GetSample.m:   [X,Y] = GetSample;
% Although GetSample can take arguments, for now you should use the form
%above (no input arguments).

%Variables to define!!
N = 10; % Number of samples in the experiment
B = 100000; %number of times to repeat the "experiment"


%Code
[X,Y] = GetSample(N);  %Acquire sample from Flip's code


%Do the goggles affect reaching performance? Are the means of the groups
%different?


%1A) Consider the statistic d = x ? ? y ?, which is a measure of
adaptation
%Use the Bootstrap to estimate the standard deviation of d and its 95%
%Confidence Interval. Use the ?percentile interval?, as defined in the
notes.
%Is there evidence of adaptation, i.e. are the means  significantly
different?

%Bootstrap
d_boot = zeros(B,1);
for b = 1:B
    X_boot = X( ceil(N*rand(N,1)) );
    Y_boot = Y( ceil(N*rand(N,1)) );
    d_boot(b) = mean(X_boot)- mean(Y_boot);
end

mean_d_boot = mean(d_boot) %-0.6997
std_d_boot = std(d_boot) %= 0.6317
CI_95__d_boot = prctile(d_boot,[2.5 97.5])  %-1.9452    0.5313


d_XY = mean(X) - mean(Y)  %   -0.7003
%The sample difference in means is within the 95% CI of the
```

```
boot mean difference, so the groups are likely not different.


%1B) Paired bootstap of above.

%Bootstrap
d_boot = zeros(B,1);
for b = 1:B
    idx = ceil(N*rand(N,1));
    X_boot = X(idx);
    Y_boot = Y(idx);
    d_boot(b) = mean(X_boot-Y_boot);
end

mean_d_boot = mean(d_boot) %-0.7003
std_d_boot = std(d_boot) %= 0.1537
CI_95__d_boot = prctile(d_boot,[2.5 97.5])  %-0.9992   -0.3992


d_XY = mean(X-Y)  %   -0.7003
%The sample difference in means is within the 95% CI of the
boot mean difference, so the groups are likely not different.
```

2. Permutation Test

```
%2A) Permutation Test of X & Y, unpaired

d_XY = mean(X) - mean(Y)  %   -0.7003
Nx = length(X);
Ny = length(Y);


D_perm = zeros(B,1);
Z = [X;Y];

for b = 1:B
    [tmp, i] = sort(rand(Nx+Ny,1));
    Zperm = Z(i,:);
    D_perm(b) = mean(Zperm(1:Nx,:))-mean(Zperm(Nx+[1:Ny],:));
end

p = mean( abs(D_perm) > abs(d_XY) )  %   0.3065


%2B) Permutation Test of X & Y, PAIRED

d_XY = mean(X) - mean(Y)  %   -0.7003
Nx = length(X);
Ny = length(Y);

D_perm = zeros(B,1);

for b = 1:B
```

```
    for k = 1:length(X)
        coin = rand(1,1) > 0.5;
        if coin
            Aperm(k) = X(k);
            Bperm(k) = Y(k);
        else
            Aperm(k) = Y(k);
            Bperm(k) = X(k);
        end
    end
    D_perm(b) = mean(Aperm-Bperm);
end


p = mean( abs(D_perm) > abs(d_XY) )  %        0.0039
```

3. Power Analysis

```
%3) Power Analysis
clear
clc

D = [0.12, 0.25, 0.5, 1, 2];
N = [10 20];

B = 500;
R = 200;
Betas_u = zeros(2,5);
Betas_p = zeros(2,5);

for w = 1:length(D);

    reject_u = zeros(length(R));
    reject_p = zeros(length(R));

    for j = 1:R
        [X,Y] = GetSample(N(1),D(w)); %Select Sample Size and Difference
Here!!
        d_XY = mean(X) - mean(Y);
        Nx = length(X);
        Ny = length(Y);

        %unpaired
        D_perm = zeros(B,1);
        Z = [X;Y];

        for b = 1:B
            %unpaired
            [tmp, i] = sort(rand(Nx+Ny,1));
            Zperm = Z(i,:);
```

```
            D_perm_u(b) = mean(Zperm(1:Nx,:))-mean(Zperm(Nx+[1:Ny],:));
            %paired
            for k = 1:length(X)
                coin = rand(1,1) > 0.5;
               if coin
                    Aperm(k) = X(k);
                    Bperm(k) = Y(k);
               else
                    Aperm(k) = Y(k);
                    Bperm(k) = X(k);
               end
             D_perm_p(j) = mean(Aperm-Bperm);
            end
        end
        reject_u(j) = mean(abs(D_perm_u) > abs(d_XY)) < 0.05;
        reject_p(j) = mean(abs(D_perm_p) > abs(d_XY)) < 0.05;

    end
    Betas_u(1,w) = mean(reject_u);  %Choose correct spot for answer
    Betas_p(1,w) = mean(reject_p);  %Choose correct spot for answer


    reject_u = zeros(length(R));
    reject_p = zeros(length(R));

    for j = 1:R
        [X,Y] = GetSample(N(2),D(w)); %Select Sample Size and Difference
Here!!
        d_XY = mean(X) - mean(Y);
        Nx = length(X);
        Ny = length(Y);

        %unpaired
        D_perm = zeros(B,1);
        Z = [X;Y];

        for b = 1:B
            %unpaired
            [tmp, i] = sort(rand(Nx+Ny,1));
            Zperm = Z(i,:);
            D_perm_u(b) = mean(Zperm(1:Nx,:))-mean(Zperm(Nx+[1:Ny],:));
            %paired
            for k = 1:length(X)
                coin = rand(1,1) > 0.5;
               if coin
                    Aperm(k) = X(k);
                    Bperm(k) = Y(k);
               else
                    Aperm(k) = Y(k);
                    Bperm(k) = X(k);
               end
             D_perm_p(j) = mean(Aperm-Bperm);
            end
        end
```

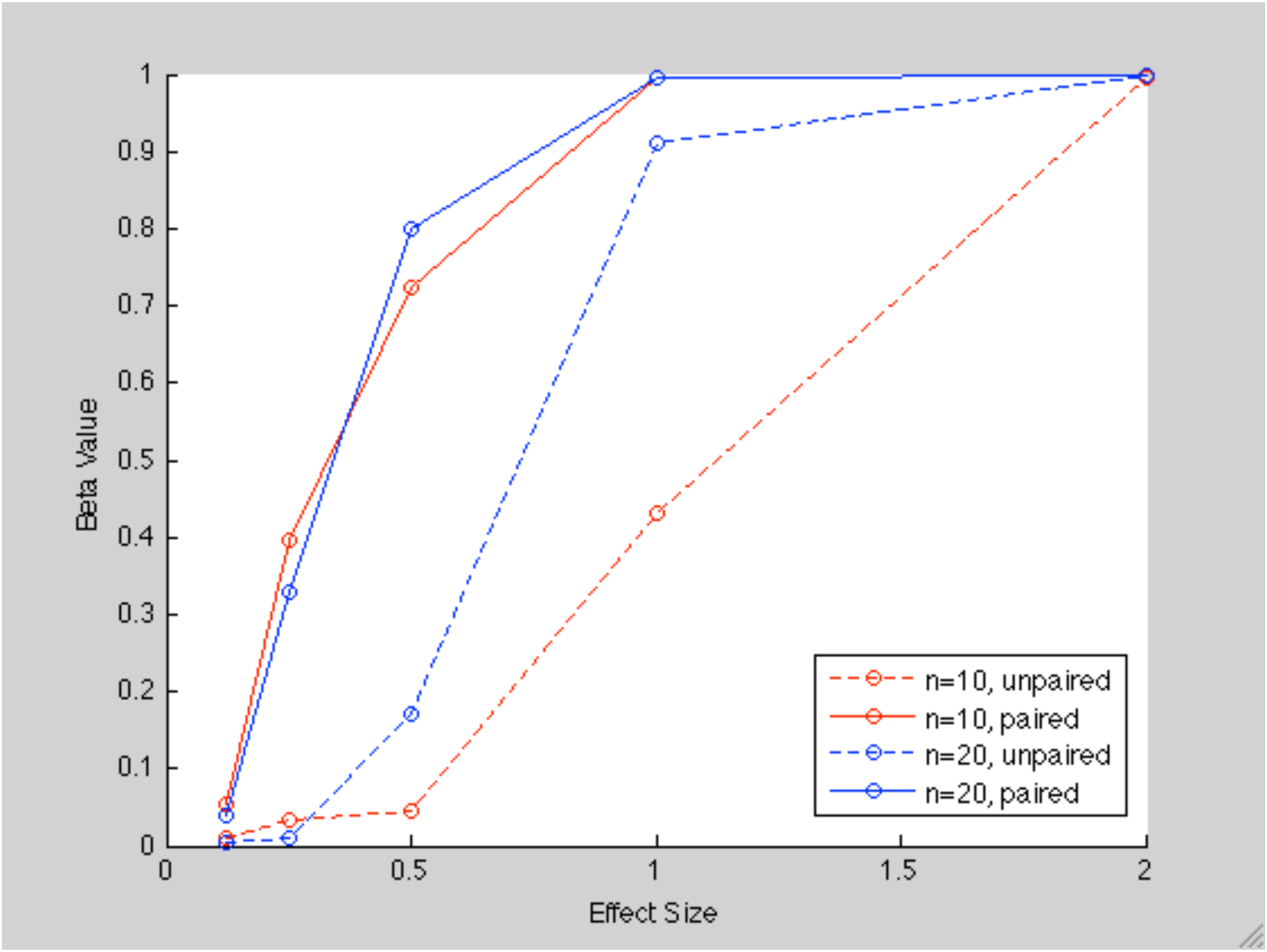
```matlab
        reject_u(j) = mean(abs(D_perm_u) > abs(d_XY)) < 0.05;
        reject_p(j) = mean(abs(D_perm_p) > abs(d_XY)) < 0.05;

    end
    Betas_u(2,w) = mean(reject_u);  %Choose correct spot for answer
    Betas_p(2,w) = mean(reject_p);  %Choose correct spot for answer
end

Betas_u
Betas_p

betas = figure;
hold on
plot(D, Betas_u(1,:), '--or');
plot(D, Betas_p(1,:), '-or');
plot(D, Betas_u(2,:), '--ob');
plot(D, Betas_p(2,:), '-ob');
xlabel('Effect Size')
ylabel('Beta Value')
legend('n=10, unpaired','n=10, paired','n=20, unpaired','n=20, paired')
hold off
```

Betas_u =

0.0100	**0.0350**	**0.0450**	**0.4300**	**0.9950**
0.0050	**0.0100**	**0.1700**	**0.9100**	**1.0000**

Betas_p =

0.0550	**0.3950**	**0.7250**	**0.9950**	**1.0000**
0.0400	**0.3300**	**0.8000**	**0.9950**	**1.0000**


Beta Value
Effect Size
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
0
0.5
1
1.5
2
n=10, unpaired
n=10, paired
n=20, unpaired
n=20, paired