Dimensionality Reduction and Visualization
Exercise Set 7 Solutions

Md. Abdullah-Al Mamun

Part E: Nonlinear Dimensionality Reduction

Problem E1: Intrinsic Dimension Estimation by the Box-counting Dimension

It can be done by setting the length if the edges. Where the first and second coordinates are divided into intervals and the rest of the coordinates are divided into just one interval.

Problem E2: Intrinsic Dimension Estimation by Local PCA

For moving the box, we have created 3 nested loops. Data points in every box are measured using a logical test by comparing values to increment intervals in every loop. They are retrieved during five datapoints were there and PCA is used for implementation. To see the dimensions more than 0.90 of the variance, the explained array has been used. To get the intrinsic estimated dimensions, the corresponding dimensions were concatenated, and we have found a dimension mean of 1.934. **(code is in the index at the very last pages of this document)**

Problem E3: Various Suggestions for Nonlinear Dimensionality Reduction Cost Functions.

First Cost function: The problem in this cost function is that there is some irrelevance of the vectors that we have in projection space and in original space.

Second Cost function: There is no issue in this cost function as the distance measurement of two points is same in the projection space as in original vector space.

Third Cost function: This function may not work as it is mathematically not viable ads both variable x and y weigh same.

Fourth Cost function: The relative position of cases is not being considered in this model. And the variables can be just one dimension.

Problem E5: Multidimensional Scaling versus Sammon's Mapping, Shepard diagram.

Codes are in the index at the very last pages of this document.

Index

Problem E2:

In [ ]:

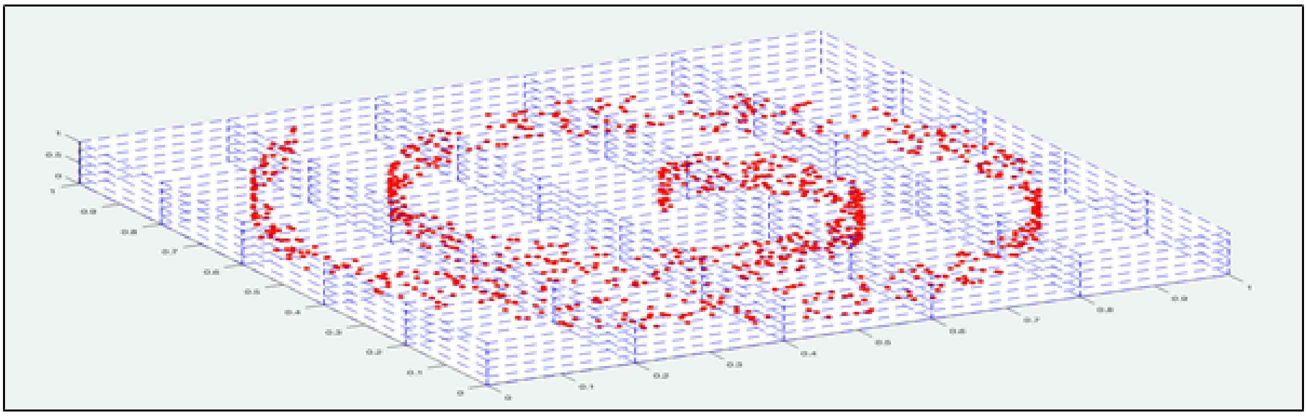
```
dims=[];
delta=0.2;
for i=0:4
    zk=[0+i*delta,0.2+i*delta];
    for j=0:4
        yk=[0+j*delta,0.2+j*delta];
        for k=0:4
            xk=[0+k*delta,0.2+k*delta];
            test=swissroll(:,1)> xk(1) & swissroll(:,1)< xk(2) & swissroll(:,2)>
yk(1) & swissroll(:,2)< yk(2) & swissroll(:,3)> zk(1) & swissroll(:,3)< zk(2);
            if sum(test) > 4
                I=find(test==1);
                X=swissroll(I,:);
                [COEFF, SCORE, LATENT, TSQUARED, EXPLAINED, MU] = pca(X);
                clear COEFF SCORE LATENT TSQUARED MU;
                for l=1:length(EXPLAINED)
                    if sum(EXPLAINED(1:l))/sum(EXPLAINED)>0.9
                        dims=[dims l];
                        break;
                    end
                end
                clear EXPLAINED;
            end
        end
    end
end

dims_mean=mean(dims)


% plotting

data=swissroll;
n = 0:0.2:1;
[X Y] = meshgrid(n,n);
x = [X(:) X(:)]';
y = [Y(:) Y(:)]';
z = [repmat(n(1),1,length(x)); repmat(n(end),1,length(x))];
col='--b';

hold all
plot3(x,y,z,col);
plot3(y,z,x,col);
plot3(z,x,y,col);
scatter3(data(:,1),data(:,2),data(:,3),'or','filled');
```



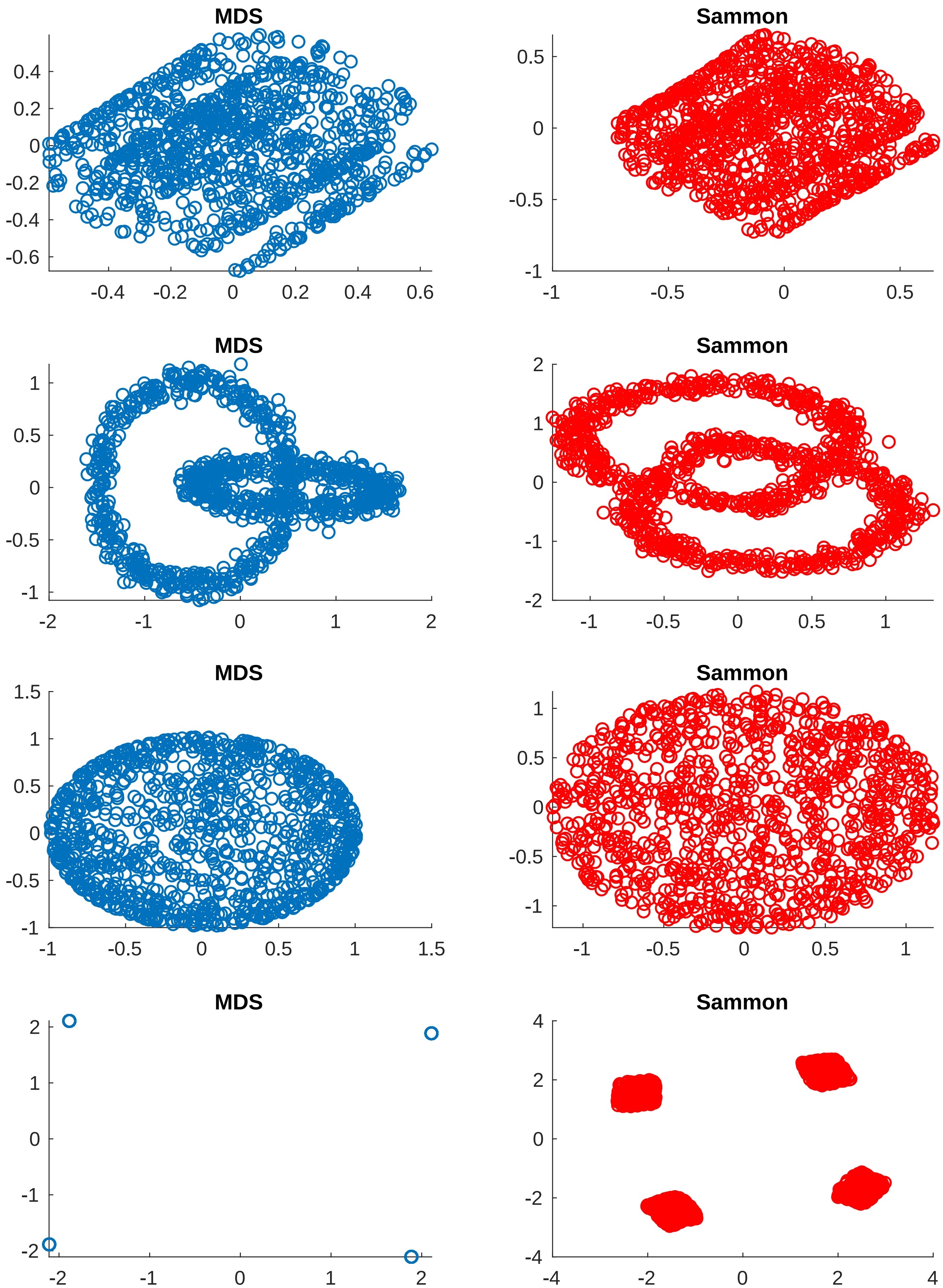
Problem E5:

In [ ]:

```
data={swissroll, tworings,halfsphere,foursquares};
hold all
locs={[1 2],[3,4],[5,6],[7,8]}

for i=1:4
    msd_res=mds(data{i},2);
    sammon_res=sammon(data{i});
    subplot(4,2,locs{i}(1));
    scatter(msd_res(:,1),msd_res(:,2));
    title('MDS');

    subplot(4,2,locs{i}(2));
    scatter(sammon_res(:,1),sammon_res(:,2),'r');
    title('Sammon')
end
```


MDS
0.4
0.2
0
-0.2
-0.4
-0.6
-0.4 -0.2 0 0.2 0.4 0.6
Sammon
0.5
0
-0.5
-1
-1 -0.5 0 0.5
MDS
1
0.5
0
-0.5
-1
-2 -1 0 1 2
Sammon
2
1
0
-1
-2
-1 -0.5 0 0.5 1
MDS
1.5
1
0.5
0
-0.5
-1
-1 -0.5 0 0.5 1 1.5
Sammon
1
0.5
0
-0.5
-1
-1 -0.5 0 0.5 1
MDS
2
1
0
-1
-2
-2 -1 0 1 2
Sammon
4
2
0
-2
-4
-4 -2 0 2 4