DistParzen_v3.1[#]

```
├── 5F_SubjectA.mat                | BCI-5F dataset
├── ChangeFigProperties.m          | Code to enhance and save result of toy datasets
├── ClassificationResults_v3.mat   | Pre-computed results as reported in submitted paper
├── code                           | Sub-folder
│   ├── adLaplacian.m              | Optimal neighborhood code file
│   ├── customClassifiers.m        | SVM and kNN classifiers
│   ├── dataDivision.m             | Code to create training and test indices
│   ├── eigenlaplacian.m           | Calculating Laplacian
│   ├── find_nn.m                  | Code required for LLE*
│   ├── gLaplacian.m               | Baseline graph Laplacian code as available from authors website*
│   ├── Isomap.m                   | ISOMAP code as available from authors website*
│   ├── L2_distance.m              | Code required for LLE/ISOMAP*
│   ├── lle.m                      | LLE code as available from authors website*
│   ├── ltsa.m                     | LTSA code as available from authors website*
│   └── ml_options.m               | Structured param for Laplacian*
├── experimentclassification.m     | Code for classification experiments
├── experimentnldr.m               | Code for dimensionality reduction
├── HaLT_SubjectA.mat              | BCI-HaLT dataset
├── Hasy_v2.mat                    | HaSy dataset
├── NaturalImage.mat               | Natural image dataset
├── NLDRCoil.mat                   | Synthetic dataset-spiral on cylinder
├── NLDRElevatedSwissRoll.mat      | Synthetic dataset-elevated Swiss roll
├── nldr_interactive.fig           | Matlab interactive UI file
├── nldr_interactive.m             | Interactive UI to examine synthetic dataset
├── NLDRSineCylinder.mat           | Synthetic dataset-sine wave on cylinder
├── NLDRSineHyperboloid.mat        | Synthetic dataset-sine wave on hyperboloid/Hourglass
├── NLDRSineRot.mat                | Synthetic dataset-sine wave with random twists
├── NLDRSineSphere.mat             | Synthetic dataset-sine wave on sphere
├── NLDRToroidalHelix.mat          | Synthetic dataset-toroidal helix
└── runExperiment.m                | Sample code to run all methods on all classification datasets
```

*These files are downloaded and bundled "as it is" available on their official websites and has only been used for evaluation purpose. Authors of the paper titled "Optimal manifold neighborhood and kernel width for robust non-linear dimensionality reduction" make no claim over them to be their own.

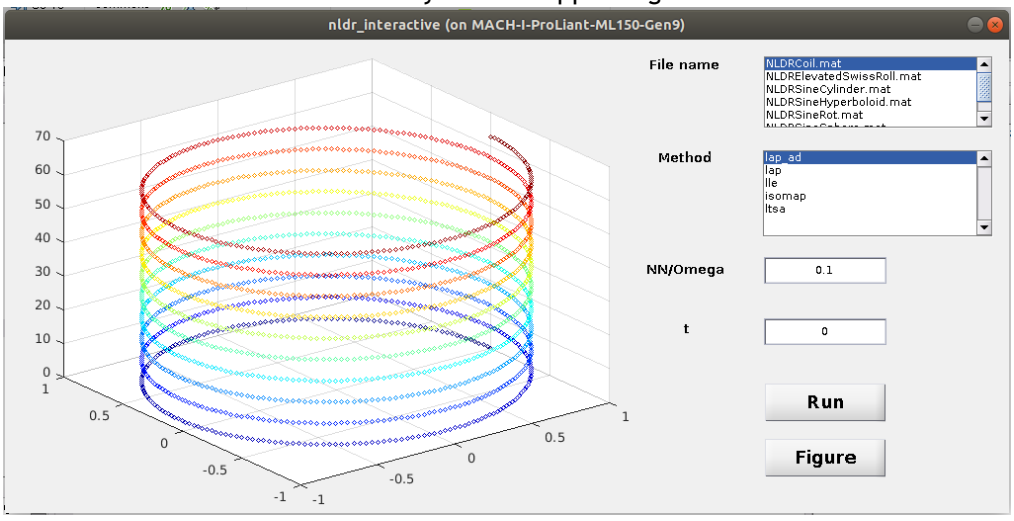
[#]The code base for robust Laplacian has been developed using Matlab R2018b.

How to proceed

1. User can open and run *nldr_interactive.m* file in base folder

   File name- User can select any file name appearing on the list containing synthetic dataset.

   Method- User can select any method appearing on the list



   lap_ad- Robust Laplacian
   lap- Baseline graph Laplacian
   lle- Local linear embedding
   isomap- Isometric mapping
   ltsa- Local tangent space alignment

   NN/Omega- Custom omega threshold for robust Laplacian and nearest neighbor for other methods.

   t- Kernel width parameter for 'lap' if =0, Silverman's rule of thumb will be used.

   Run- Executes the selected dataset with chosen algorithm and give parameters, result will be shown in the embedded figure axes.

   Figure- Enhances and save the figure plot in *./results* folder.

2. User can execute dimensionality reduction code with any dataset stored in a file with *data* matlab variable using *experimentnldr.m*
   Syntax:

fileName=tmpdata.mat method=lap_ad targetDimension=2 omega=5e-2 t=0	mapX= experimentnldr(fileName, method, targetDimension, omega, t);

3. The classification code can be executed in the similar fashion using *experimentclassification.m*, execution syntax is same as of experimentnldr.m except for few more variables as listed below are required in the file to proceed
   a. Datasets- structure variable created using *./code/dataDivision.m* file. It contains training and testing index along with their respective labels index.
   b. Label- nx1 data where n is the number of observations and the only column contains their respective label.
   c. NN- nearest neighbor factor for kNN classifier.
4. *runExperiment.m* contains sample format to execute all methods on all classification dataset at once. User can refer to this file and extend or copy for further use.
5. To create new classification training/test index use *./code/dataDivision.m*
   a. Load label variable in matlab workspace and run *./code/dataDivision.m*
6. The attached code base switches from base directory to code directory and switches back to base directory, if an error occurs in between, kindly switch back to base directory and execute the code once again.
7. ClassificationResults_v3.mat contains pre-computed results.

rmserror

1x6 struct with 3 fields

Fields	dname	errSVM	errKNN
1	'5F_Subj...	5x5 double	5x5 double
2	'CVPR.mat'	5x67 dou...	5x67 dou...
3	'FashioM...	5x10 dou...	5x10 dou...
4	'HaLT_Su...	5x9 double	5x9 double
5	'Hasy_v2...	5x9 double	5x9 double
6	'Naturall...	5x8 double	5x8 double

dname: classification data file name.
errSVM: 5xC, C is the number of distinct label in the original data.
   1st row- lap_ad
   2nd row- lap
   3rd row- lle
   4th row- isomap
   5th row- ltsa
errKNN: similar to *errSVM*.

These results contain mean of number of trials done for classification.

Reported results are mean of each row of *errSVM* and *errKNN*.