The **Sample MATLAB Code and dataset** for "Systems Biology and Machine Learning Approaches Identify Drug Targets in Diabetic Nephropathy" paper, submitted to Scientific Reports

Part 1. The dataset

Two excel files, 'BioFeatures.xlsx' and 'TopBioFeature.xlsx,' were provided. The first file contains biochemical network features, while the second file has network topological and biochemical features. Note that each file has two sheets, drug target, and non-drug target class. The first row has the name of the features, while the first column contains UniProt ID (Figs. 1 and 2).

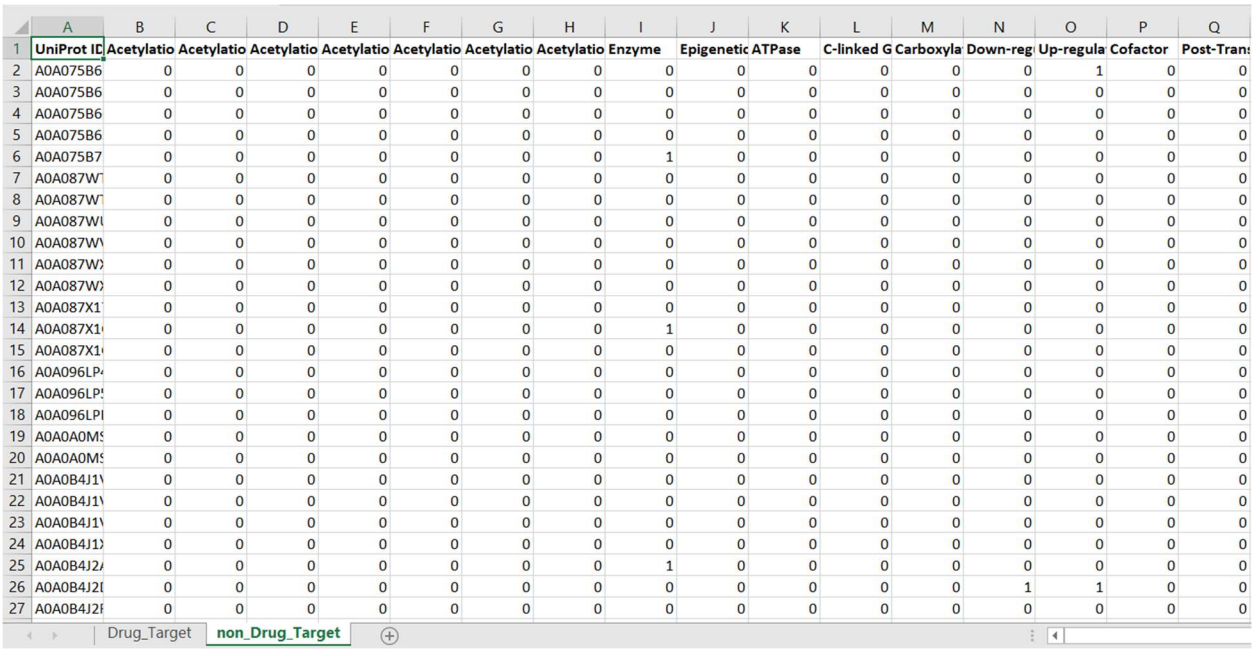
	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1	UniProt ID	Average S	Betweenn	Betweenn	Closeness	Closeness	Clustering	Degree	Eccentricit	Eigenvecto	LAC	Neighborh	Network	SelfLoops	Stress
2	O00180	4.622142	373.6189	0	0.001344	0.21635	0	3	9	0.001407	2	52.5	3.5	1	0
3	O00204	5.728635	18512	0	0.001344	0.174562	0	2	10	0.000175	0	6.5	0	0	0
4	O00206	4.484543	79769.73	0.000828	0.001345	0.222988	0.045455	22	9	0.00624	2.5	19.31818	4.837519	0	508536
5	O00264	6.083123	814.1193	8.31E-05	0.001344	0.164389	0	2	11	0.000145	1	15	2	0	39053
6	O00305	4.732534	6895.639	3.08E-06	0.001344	0.211303	0.02381	7	9	0.000979	0.714286	28	0.866667	0	1013
7	O00329	4.64448	27362.38	0.000345	0.001345	0.215309	0.037879	13	9	0.006903	2.923077	27.33333	6.933333	1	204874
8	O00459	4.037697	49911.97	0.000709	0.001345	0.247666	0.099798	32	8	0.040474	6.90625	56.5	8.230203	0	514442
9	O00555	5.402366	1968.611	1.24E-05	0.001344	0.185104	0.033333	6	10	0.000715	0.5	12.5	0.6	0	7233
10	O00763	0	0	0	0.001343	0	0	1	0	5.39E-05	0	10	0	0	0
11	O14594	4.652633	7182.3	0	0.001345	0.214932	0.066667	6	9	0.001393	1	29.83333	1.2	0	0
12	O14646	4.67776	3617.8	5.72E-05	0.001345	0.213778	0	7	9	0.003291	0.428571	39.28571	0.5	0	29632
13	O14649	3.987699	6268.224	0	0.001345	0.250771	0.111111	9	8	0.008899	2.333333	84.44444	2.625	0	0
14	O14717	4.747634	14038.54	7.21E-05	0.001345	0.210631	0	9	9	0.00338	0.666667	64.22222	0.75	0	55169
15	O14727	4.19511	21077.23	0.000532	0.001345	0.238373	0.075	16	9	0.006762	3.0625	33.4375	5.119048	0	265931
16	O14732	6.238921	0	0	0.001343	0.160284	0	1	11	4.86E-05	0	9	0	0	0
17	O14746	3.986278	141342.5	0.000965	0.001346	0.250861	0.037549	24	8	0.020256	4	52.08696	8.785128	1	516793
18	O14764	5.394795	0	0	0.001344	0.185364	0.5	2	10	0.000589	2	96	4	0	0
19	O14786	4.861199	62404.96	0.000546	0.001344	0.205711	0.069853	18	9	0.003675	4.666667	14.05882	13.20686	1	374792
20	O14788	3.941956	34234.11	0.00034	0.001345	0.253681	0.032967	15	9	0.013091	3.266667	55.07143	3.745671	1	176976
21	O14832	5.186593	34231.16	0.000468	0.001344	0.192805	0	8	10	0.000702	2	19.14286	2.5	1	217322
22	O14842	5.4217	69.01018	0	0.001344	0.184444	0	2	10	0.000523	1	45	2	0	0
23	O14880	0	0	0	0.001342	0	0	1	0	1.4E-05	1	19	0	0	0
24	O14920	3.833281	136358.3	0.001546	0.001346	0.260873	0.06753	48	8	0.038846	8.75	48.55319	15.40028	1	1020409
25	O14939	3.852681	132589.2	0.000953	0.001346	0.259559	0.035	25	8	0.02179	2.24	51.36	2.678477	0	506693
26	O15111	3.777287	179304.8	0.002602	0.001346	0.26474	0.063498	54	8	0.051952	9.12963	56.35849	16.9568	1	1670095
27	O15269	5.318877	18512	7.4E-05	0.001344	0.18801	0	2	10	0.000153	0	17.5	0	0	33444

Drug_Target | non_drug_target

Fig. 1. Part of the Excel file "TopBioFeature.xlsx" shows the network topological and biochemical features. It has two sheets, drug target, and non-drug target features. Except for the first row and the first column, the data of the other cells can be used to classify drug and no-drug targets.

Sample Code:

First, The Matlab version must be R2021b or later to run the code. The main function is "example.m". It contains a demo in which a Matlab benchmark breast cancer dataset is loaded to classify Benign and Malignant cases (Fig. 3).



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1	UniProt ID	Acetylatio	Acetylatio	Acetylatio	Acetylatio	Acetylatio	Acetylatio	Acetylatio	Enzyme	Epigenetic	ATPase	C-linked G	Carboxyla	Down-reg	Up-regula	Cofactor	Post-Trans
2	A0A075B6	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
3	A0A075B6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	A0A075B6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	A0A075B6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	A0A075B7	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
7	A0A087W1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	A0A087W1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	A0A087W(	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	A0A087W\	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11	A0A087W2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	A0A087W2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13	A0A087X1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
14	A0A087X1(	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
15	A0A087X1(	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
16	A0A096LP4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
17	A0A096LP!	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
18	A0A096LPI	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
19	A0A0A0M!	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
20	A0A0A0M!	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
21	A0A0B4J1\	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
22	A0A0B4J1\	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
23	A0A0B4J1\	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
24	A0A0B4J1)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
25	A0A0B4J2/	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
26	A0A0B4J2[	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0
27	A0A0B4J2I	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Drug_Target | non_Drug_Target

Fig. 2. Part of the Excel file "BioFeature.xlsx" shows the biochemical features. It has two sheets, drug target, and non-drug target features. Except for the first row and the first column, the data of the other cells can be used to classify drug and no-drug targets.

```
1   %% Loading Data
2   %Inputs : M*N  matrix, N:Number of sample , M:number of feature
3   [Inputs, Targets] = cancer_dataset(); % loading the sample dataset
4   Targets=Targets(1,:);
5
```

Fig. 3. The starting section of the code "example.m". It opens the Matlab cancer benchmark dataset, by default as the demo.

It is possible to run the code on other datasets. To do so, lines 3-4 must be removed, and the feature matrix "Inputs" (an M by N matrix, N: Number of samples, M: number of features) and the class label vector "Targets" (a one by N vector, N: Number of samples) must be provided. The class labels must contain zeros and ones as the class labels, while the features must be numerical.

Validation Framework Parameters

The code uses stratified sampling for the training and test splits, and the training set has the estimation and validation splits to avoid overfitting. The related parts of the code and their line numbers are shown (Fig. 4).

```
 7 -   Train_Ratio=.7; % The percentage of the training set
41 -   pTrain_pValidation=.7;  % the percentage of the estimation in the training set
```

Fig. 4. The training and test split. Line 7 indicates the percentage of the training set in the entire dataset, while line 41 shows the percentage of the estimation set in the training set.

The Pre-processing and GMDH Network Parameters

The pre-processing and GMDH input parameters are provided in Fig.5. It controls whether an outlier detection (line 43) or feature normalization (line 44) is used. The input parameters to control the GMDH network were also provided in lines 46-49. Noted that the core GMDH algorithm ("GMDH.m") was used, and the modifications addressed in the submitted paper were implemented.

```
43 -   AnormallyF=false; % Outlier detection
44 -   isNormalize=true; % doing normalization
45
46 -   MaxLayerNeurons=15; % maximum neurons in each layer
47 -   MaxLayers=5;  % maximum layer
48 -   Selection_Pressure=.9; % Selection pressure
49 -   NumberOfBestFeatureSelected=6; % Number of selected features
```

Fig. 5. The pre-processing (Line 43-44) and the GMDH input parameters (Line 46-49).

The Outputs of the Program

Variety of outputs are provided after running the program, including the GMDH layer structure "gmdh", the target and outcome of the training set "TrainTargets," and "TrainOutputs," as well as those for the test set "TestTargets" and "TestOutputs," the performance indices of the training and test sets "train," "test." Such indices were provided in Fig.6 for the tutorial.

To ensure optimal performance, the sensitivity analysis (or the grid search) must be performed on the GMDH input parameters (Fig. 5, lines 46-49). Moreover, if necessary, the customized cost function can be added to the function "Cost.m" [line 30]. The current cost function is the weighted average of the "Sensitivity," "Specificity," and "Precision" of the classifier (Fig.7).

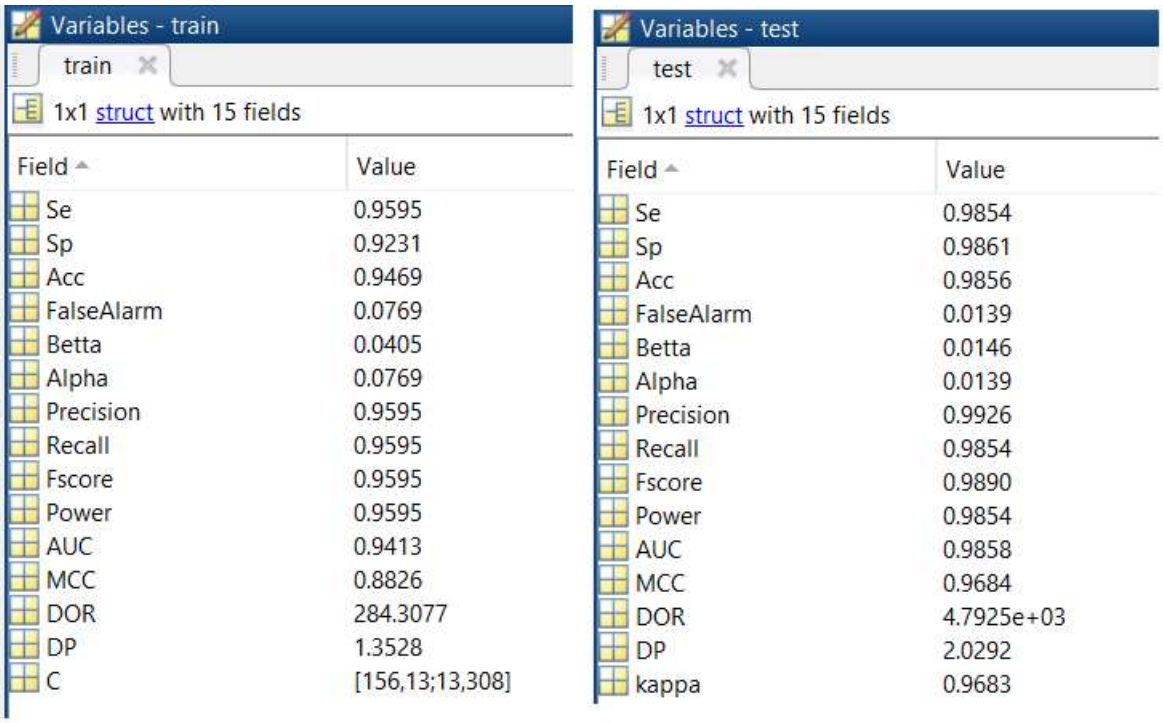


Fig. 6. The performance indices of the algorithm on the training and test sets when running the tutorial.

```
30 -        z=1-(1.5*test.Se+test.Sp+1.5*test.Precision)/4;
```

Fig. 7. The cost function definition is in the "Cost.m" function.

Citations of the functions used in the algorithm

Bjarke Skogstad Larsen (2021). Synthetic Minority Over-sampling Technique (SMOTE) (https://github.com/dkbsl/matlab_smote/releases/tag/1.0), GitHub. Retrieved October 30, 2021.

Cardillo G. (2007) Cohen's kappa: compute the Cohen's kappa ratio on a 2x2 matrix. http://www.mathworks.com/matlabcentral/fileexchange/15365.

Navid Rezaei (2021). GMDH (https://www.mathworks.com/matlabcentral/fileexchange/53249-gmdh), MATLAB Central File Exchange. Retrieved October 30, 2021.