```
clc

clear all
```

Here we show the classification results using SMOTE and Random undersampling as well as SVM vs BiLSTM

The metrics considered are Class specific F-measures, macro average and micro average F-measure, precision, specificity, sensitivity etc. We have also included accuracy even though it may not be a good metric for testing imbalanced dataset. In addition normalized confusion matrix is also provided for analysis.

SVM based Classification

REM vs Wake

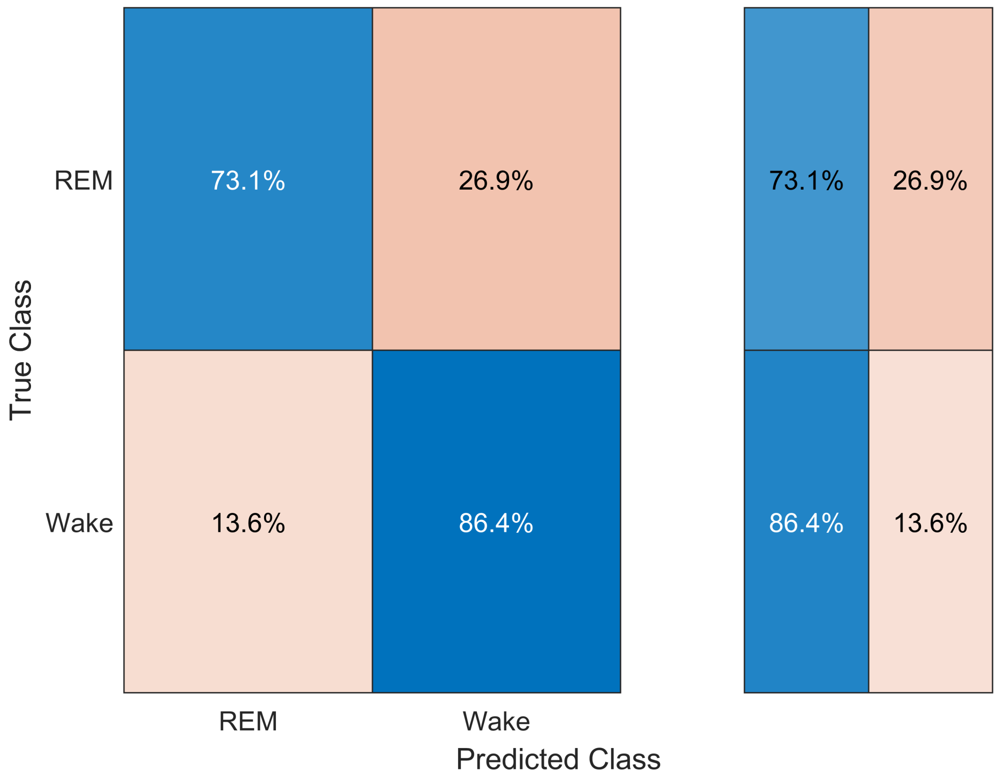
Model trained using Random Undersampling of Majority class (The models considered here are the best models obtained using hyperparameter optimization)

```

Model = load('C:\courses\ML\project\SVM_2_class\trained_model_SVM_US.mat'); % Model trained wit
Model = Model.model;
Testing = load('C:\courses\ML\project\SVM_2_class\Testingred_features_SVM_US.mat');
Testing = Testing.testFeatures1;
TestingLabels = load('C:\courses\ML\project\codes\TrainNetworkProject2\dataset_processed\Twocla
TestingLabels = TestingLabels.TwoClassTest_US_Labels';
[label,score] = predict(Model,Testing);
C = confusionmat(TestingLabels,categorical(label));
[stats] = statsOfMeasure(C, 1);
```

stats = 9×4 table

	name	classes		macroAVG	microAVG
1	"true_positive"	2385	716	1.5505e+03	1.5505e+03
2	"false_positive"	113	878	495.5000	495.5000
3	"false_negative"	878	113	495.5000	495.5000
4	"true_negative"	716	2385	1.5505e+03	1.5505e+03
5	"precision"	0.9548	0.4492	0.7020	0.7578
6	"sensitivity"	0.7309	0.8637	0.7973	0.7578
7	"specificity"	0.8637	0.7309	0.7973	0.7578
8	"accuracy"	0.7578	0.7578	0.7578	0.7578
9	"F-measure"	0.8280	0.5910	0.7095	0.7578

```
figure
confusionchart(TestingLabels,categorical(label),'Normalization','row-normalized','RowSummary','
```



Model trained using SMOTE (The models considered here are the best models obtained using hyperparameter optimization)

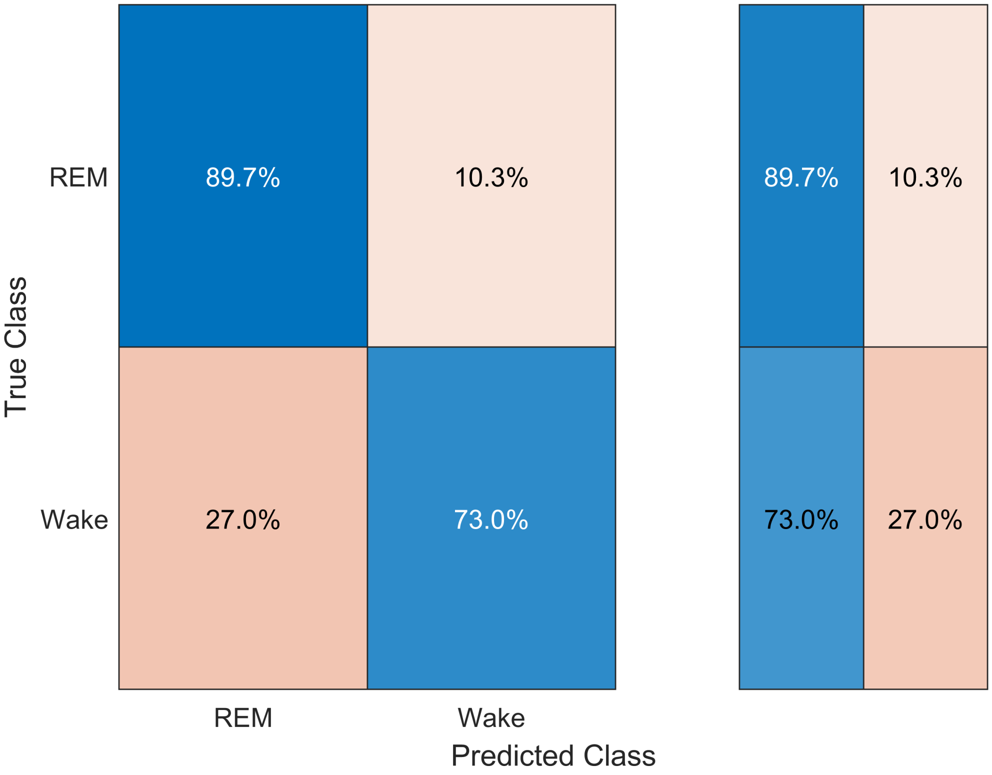
```
Model = load('C:\courses\ML\project\SVM_2_class\trained_model_SVM_SMOTE.mat'); % Model trained
Model = Model.model;
Testing = load('C:\courses\ML\project\SVM_2_class\Testingred_features_SVM_SMOTE.mat');
Testing = Testing.testFeatures1;
TestingLabels = load('C:\courses\ML\project\codes\TrainNetworkProject2\dataset_processed\Twocla
TestingLabels = TestingLabels.TwoClassTest_SMOTE_Labels';
[label,score] = predict(Model,Testing);
C = confusionmat(TestingLabels,categorical(label));
[stats] = statsOfMeasure(C, 1);
```

stats = 9×4 table

	name	classes		macroAVG	microAVG
1	"true_positive"	2927	605	1766	1766
2	"false_positive"	224	336	280	280
3	"false_negative"	336	224	280	280
4	"true_negative"	605	2927	1766	1766
5	"precision"	0.9289	0.6429	0.7859	0.8631
6	"sensitivity"	0.8970	0.7298	0.8134	0.8631

	name	classes		macroAVG	microAVG
7	"specificity"	0.7298	0.8970	0.8134	0.8631
8	"accuracy"	0.8631	0.8631	0.8631	0.8631
9	"F-measure"	0.9127	0.6836	0.7982	0.8631

```
figure
confusionchart(TestingLabels,categorical(label),'Normalization','row-normalized','RowSummary','
```



NREM vs Wake

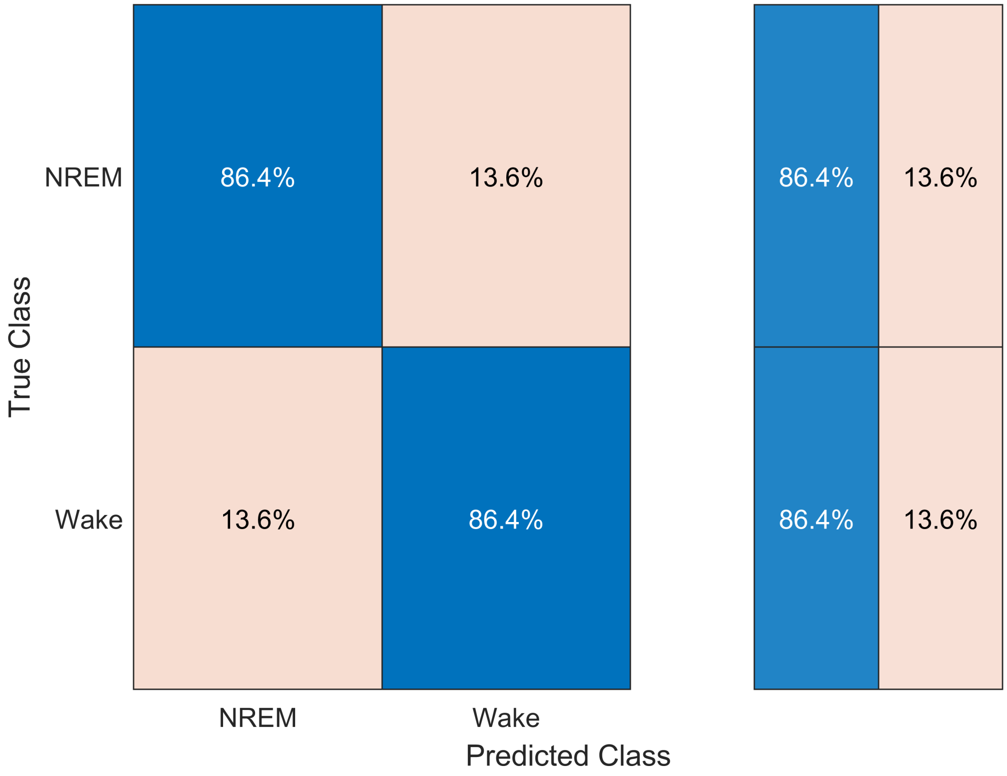
Model trained using Random Undersampling of Majority class (The models considered here are the best models obtained using hyperparameter optimization)

```
Model = load('C:\courses\ML\project\SVM_3_class\trained_model_SVM_US.mat'); % Model trained wit
Model = Model.model;
Testing = load('C:\courses\ML\project\SVM_3_class\Testingred_features_SVM_US.mat');
Testing = Testing.testFeatures1;
TestingLabels = load('C:\courses\ML\project\codes\TrainNetworkProject2\dataset_processed\Threec
TestingLabels = TestingLabels.ThreeClassTest_US_Labels';
[label,score] = predict(Model,Testing);
C = confusionmat(TestingLabels,categorical(label));
[stats] = statsOfMeasure(C, 1);
```

```
stats = 9×4 table
```

	name	classes		macroAVG	microAVG
1	"true_positive"	13184	716	6950	6950
2	"false_positive"	113	2076	1.0945e+03	1.0945e+03
3	"false_negative"	2076	113	1.0945e+03	1.0945e+03
4	"true_negative"	716	13184	6950	6950
5	"precision"	0.9915	0.2564	0.6240	0.8639
6	"sensitivity"	0.8640	0.8637	0.8638	0.8639
7	"specificity"	0.8637	0.8640	0.8638	0.8639
8	"accuracy"	0.8639	0.8639	0.8639	0.8639
9	"F-measure"	0.9233	0.3955	0.6594	0.8639

```
figure
confusionchart(TestingLabels,categorical(label),'Normalization','row-normalized','RowSummary','
```



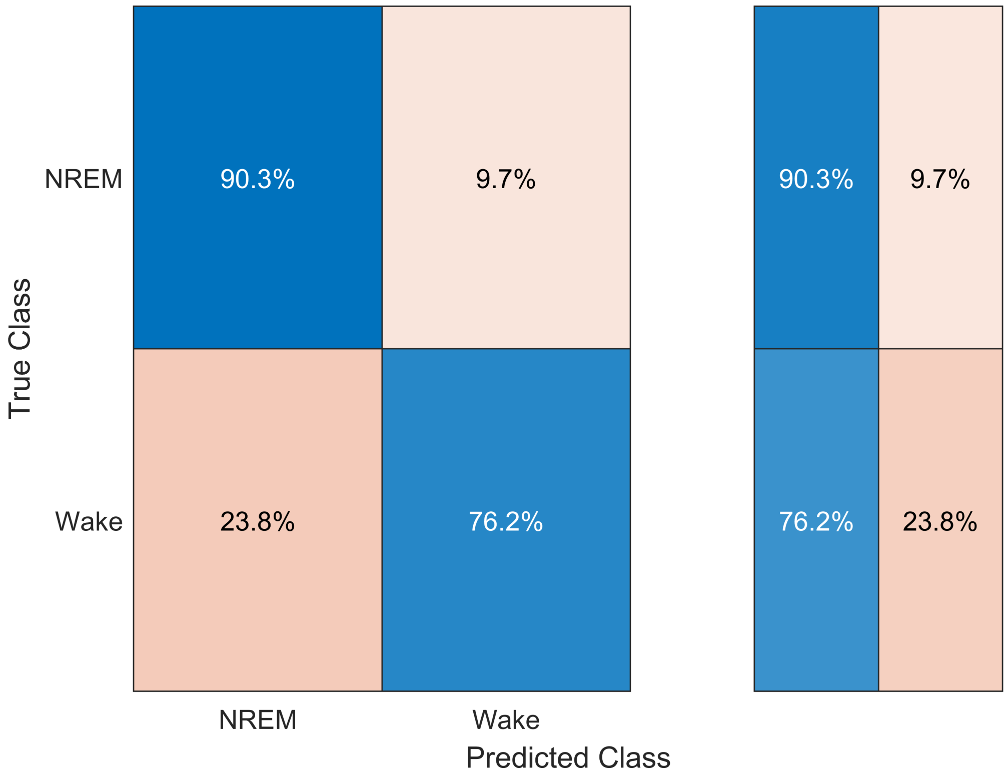
Model trained using SMOTE (The models considered here are the best models obtained using hyperparameter optimization)

```
Model = load('C:\courses\ML\project\SVM_3_class\trained_model_SVM_SMOTE.mat'); % Model trained
Model = Model.model;
Testing = load('C:\courses\ML\project\SVM_3_class\Testingred_features_SVM_SMOTE.mat');
Testing = Testing.testFeatures1;
TestingLabels = load('C:\courses\ML\project\codes\TrainNetworkProject2\dataset_processed\Threee
TestingLabels = TestingLabels.ThreeClassTest_SMOTE_Labels';
[label,score] = predict(Model,Testing);
C = confusionmat(TestingLabels,categorical(label));
[stats] = statsOfMeasure(C, 1);
```

stats = 9×4 table

	name	classes		macroAVG	microAVG
1	"true_positive"	13780	632	7206	7206
2	"false_positive"	197	1480	838.5000	838.5000
3	"false_negative"	1480	197	838.5000	838.5000
4	"true_negative"	632	13780	7206	7206
5	"precision"	0.9859	0.2992	0.6426	0.8958
6	"sensitivity"	0.9030	0.7624	0.8327	0.8958
7	"specificity"	0.7624	0.9030	0.8327	0.8958
8	"accuracy"	0.8958	0.8958	0.8958	0.8958
9	"F-measure"	0.9426	0.4298	0.6862	0.8958

```
figure
confusionchart(TestingLabels,categorical(label),'Normalization','row-normalized','RowSummary','
```



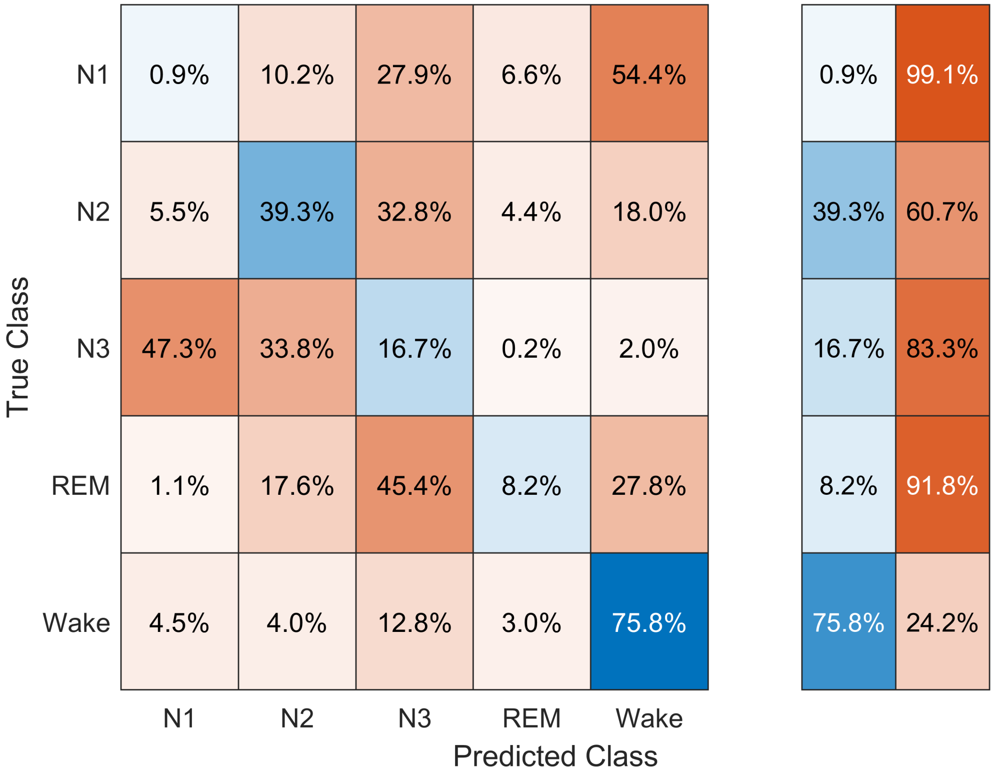
Five class Classification

```
Model = load('C:\courses\ML\project\SVM_5_class\AUTOMATEDCODE\trained_model_SVM_US.mat'); % Mod
Model = Model.model;
Testing = load('C:\courses\ML\project\SVM_5_class\AUTOMATEDCODE\Testingred_features_SVM_US.mat
Testing = Testing.testFeatures1;
TestingLabels = load('C:\courses\ML\project\codes\TrainNetworkProject2\dataset_processed\Fivecl
TestingLabels = TestingLabels.Class_Testing';
[label,score] = predict(Model,Testing);
C = confusionmat(TestingLabels,categorical(label));
[stats] = statsOfMeasure(C, 1);
```

```
stats = 9×4 table
                          ...
```

	name
1	"true_positive"
2	"false_positive"
3	"false_negative"
4	"true_negative"
5	"precision"
6	"sensitivity"
7	"specificity"
8	"accuracy"
9	"F-measure"

```
figure
confusionchart(TestingLabels,categorical(label),'Normalization','row-normalized','RowSummary',
```



Bidirectional Long Short Term Memory (BiLSTM) based classification

REM vs Wake

Model trained using Random Undersampling of Majority class (The models considered here are the best models obtained using hyperparameter optimization)

```
Model = load('C:\courses\ML\project\codes\TrainNetworkProject2\BiLSTM_fiveclass\hyperparameter_
Model = Model.netLSTM;
Testing = load('C:\courses\ML\project\codes\TrainNetworkProject2\dataset_processed\TwoclassTest
Testing = Testing.Balancedsleep2ClasstestUSCell;
TestingLabels = load('C:\courses\ML\project\codes\TrainNetworkProject2\dataset_processed\Twocla
TestingLabels = TestingLabels.TwoClassTest_US_Labels';
[label,score] = classify(Model,Testing)
```

```
label = 4092×1 categorical
REM
REM
REM
REM
REM
REM
Wake
Wake
REM
REM
      ⋮

score = 4092×2 single matrix
```

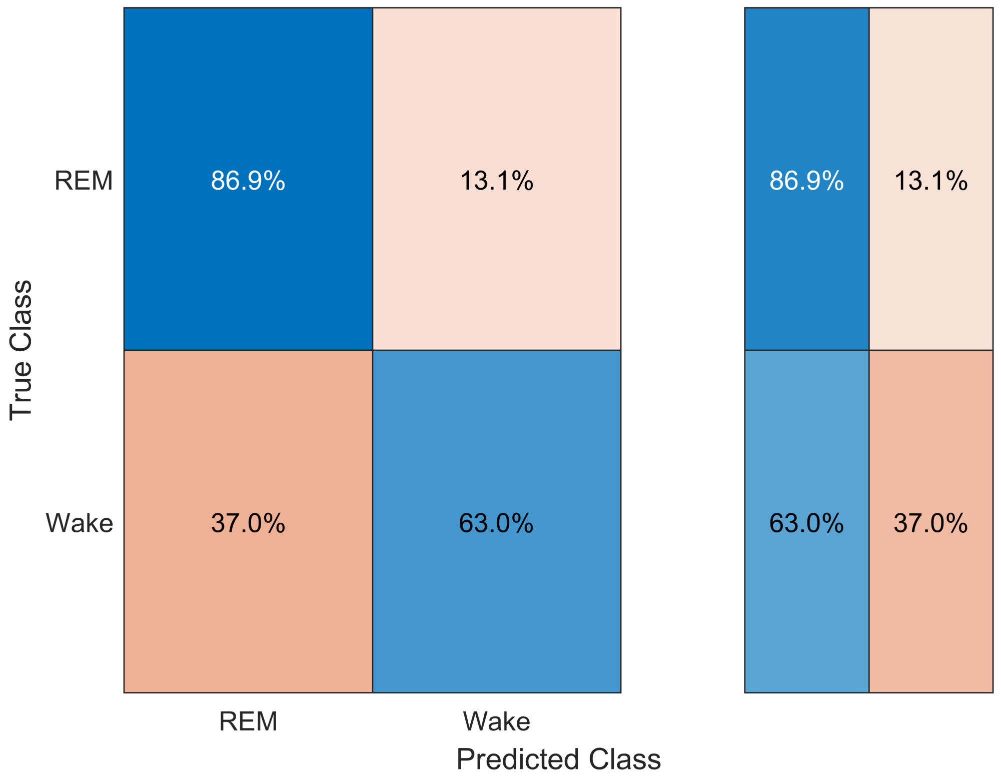
```
    0.9400    0.0600
    0.9822    0.0178
    0.8458    0.1542
    0.9981    0.0019
    0.9988    0.0012
    0.9978    0.0022
    0.0114    0.9886
    0.0762    0.9238
    0.9956    0.0044
    0.9952    0.0048
      ⋮
```

```
C = confusionmat(TestingLabels,categorical(label));
[stats] = statsOfMeasure(C, 1);
```

stats = 9×4 table

	name	classes		macroAVG	microAVG
1	"true_positive"	2835	522	1.6785e+03	1.6785e+03
2	"false_positive"	307	428	367.5000	367.5000
3	"false_negative"	428	307	367.5000	367.5000
4	"true_negative"	522	2835	1.6785e+03	1.6785e+03
5	"precision"	0.9023	0.5495	0.7259	0.8204
6	"sensitivity"	0.8688	0.6297	0.7493	0.8204
7	"specificity"	0.6297	0.8688	0.7493	0.8204
8	"accuracy"	0.8204	0.8204	0.8204	0.8204
9	"F-measure"	0.8852	0.5868	0.7360	0.8204

```
figure
confusionchart(TestingLabels,categorical(label),'Normalization','row-normalized','RowSummary',
```



Model trained using SMOTE (The models considered here are the best models obtained using hyperparameter optimization)

```
Model = load('D:\ML\hyperparametertuning_SMOTE_BILSTM_2CLASS\0.22749.mat'); % Best Model traine
```

```
Warning: While loading an object of class 'nnet.cnn.TrainingOptionsADAM':
'CheckpointFrequency' is not an option for solver 'adam'.
```

```
Model = Model.netLSTM;
Testing = load('C:\courses\ML\project\codes\TrainNetworkProject2\dataset_processed\TwoclassTest
Testing = Testing.Balancedsleep2ClasstestSMOTECell;
TestingLabels = load('C:\courses\ML\project\codes\TrainNetworkProject2\dataset_processed\Twocla
TestingLabels = TestingLabels.TwoClassTest_SMOTE_Labels';
[label,score] = classify(Model,Testing)
```

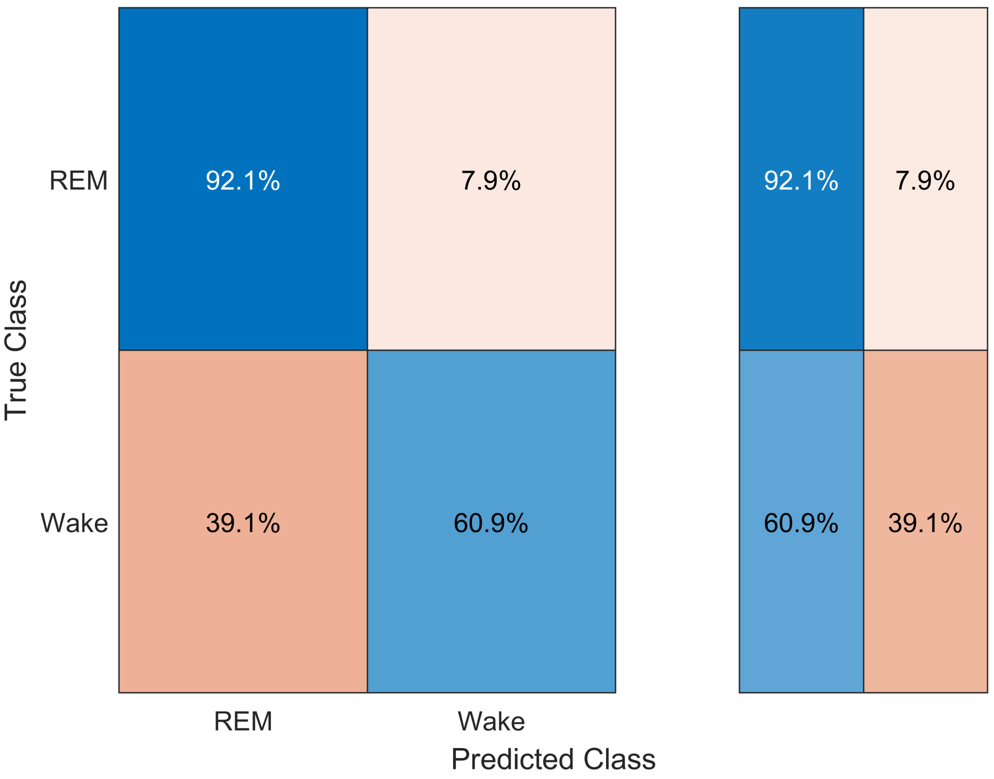
```
label = 4092×1 categorical
REM
REM
REM
REM
REM
REM
Wake
Wake
REM
REM
      ⋮

score = 4092×2 single matrix
    0.9557    0.0443
    0.9990    0.0010
    0.9346    0.0654
    0.9931    0.0069
    0.9985    0.0015
    0.9686    0.0314
    0.0968    0.9032
    0.0771    0.9229
    0.9849    0.0151
    0.9945    0.0055
      ⋮
```

```
C = confusionmat(TestingLabels,categorical(label));
[stats] = statsOfMeasure(C, 1);
```

```
stats = 9×4 table
```

	name	classes		macroAVG	microAVG
1	"true_positive"	3006	505	1.7555e+03	1.7555e+03
2	"false_positive"	324	257	290.5000	290.5000
3	"false_negative"	257	324	290.5000	290.5000
4	"true_negative"	505	3006	1.7555e+03	1.7555e+03
5	"precision"	0.9027	0.6627	0.7827	0.8580
6	"sensitivity"	0.9212	0.6092	0.7652	0.8580
7	"specificity"	0.6092	0.9212	0.7652	0.8580
8	"accuracy"	0.8580	0.8580	0.8580	0.8580
9	"F-measure"	0.9119	0.6348	0.7733	0.8580

```
figure
confusionchart(TestingLabels,categorical(label),'Normalization','row-normalized','RowSummary',
```



NREM vs Wake

Model trained using Random Undersampling of Majority class (The models considered here are the best models obtained using hyperparameter optimization)

```
Model = load('C:\courses\ML\project\codes\TrainNetworkProject2\BiLSTM_fiveclass\Hyperparameter_
Model = Model.netLSTM;
Testing = load('C:\courses\ML\project\codes\TrainNetworkProject2\dataset_processed\ThreeclassTe
Testing = Testing.Balancedsleep3ClasstestUSCell;
TestingLabels = load('C:\courses\ML\project\codes\TrainNetworkProject2\dataset_processed\Threec
TestingLabels = TestingLabels.ThreeClassTest_US_Labels';
[label,score] = classify(Model,Testing)
```

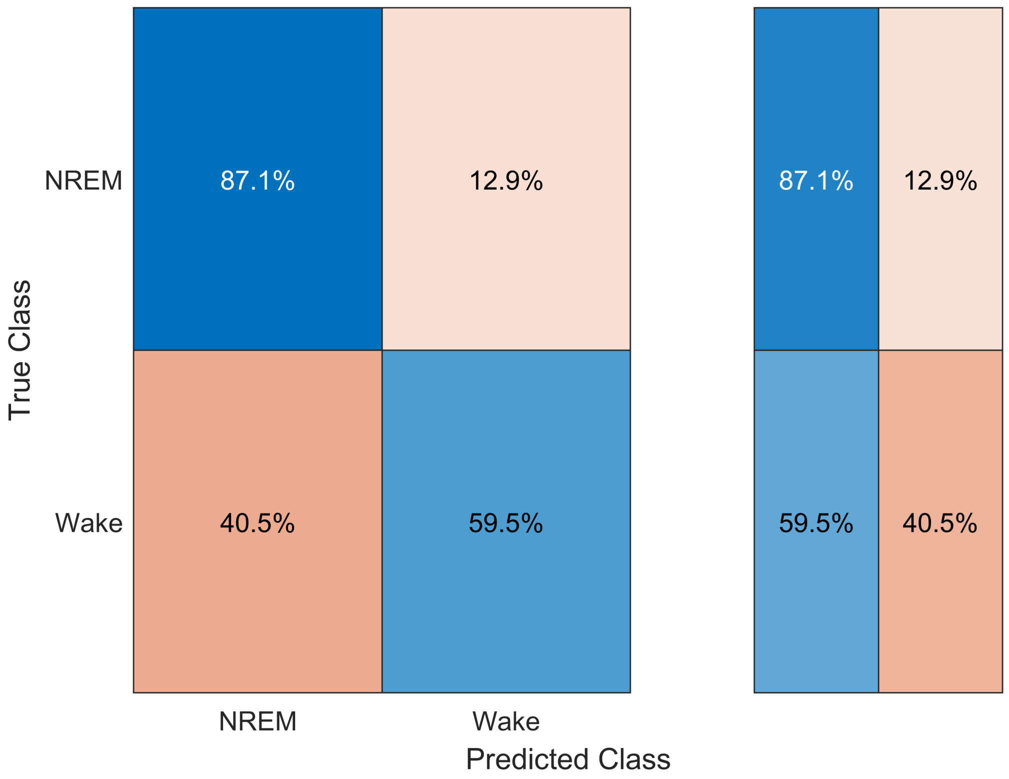
```
label = 16089×1 categorical
NREM
Wake
Wake
Wake
NREM
NREM
NREM
Wake
NREM
Wake
      ⋮

score = 16089×2 single matrix
    0.7407    0.2593
    0.0128    0.9872
    0.2245    0.7755
    0.2174    0.7826
    0.7075    0.2925
    0.6582    0.3418
    0.5454    0.4546
    0.4442    0.5558
    0.7784    0.2216
    0.2841    0.7159
      ⋮
```

```
C = confusionmat(TestingLabels,categorical(label));
[stats] = statsOfMeasure(C, 1);
```

stats = 9×4 table

	name	classes		macroAVG	microAVG
1	"true_positive"	13295	493	6894	6894
2	"false_positive"	336	1965	1.1505e+03	1.1505e+03
3	"false_negative"	1965	336	1.1505e+03	1.1505e+03
4	"true_negative"	493	13295	6894	6894
5	"precision"	0.9754	0.2006	0.5880	0.8570
6	"sensitivity"	0.8712	0.5947	0.7330	0.8570
7	"specificity"	0.5947	0.8712	0.7330	0.8570
8	"accuracy"	0.8570	0.8570	0.8570	0.8570
9	"F-measure"	0.9204	0.3000	0.6102	0.8570

```
figure
confusionchart(TestingLabels,categorical(label),'Normalization','row-normalized','RowSummary',
```



Model trained using SMOTE (The models considered here are the best models obtained using hyperparameter optimization)

```
Model = load('C:\courses\ML\project\codes\TrainNetworkProject2\BiLSTM_fiveclass\Hyperparameter_
Model = Model.netLSTM;
Testing = load('C:\courses\ML\project\codes\TrainNetworkProject2\dataset_processed\ThreeclassTe
Testing = Testing.Balancedsleep3ClasstestSMOTECell;
TestingLabels = load('C:\courses\ML\project\codes\TrainNetworkProject2\dataset_processed\Threec
TestingLabels = TestingLabels.ThreeClassTest_SMOTE_Labels';
[label,score] = classify(Model,Testing)
```

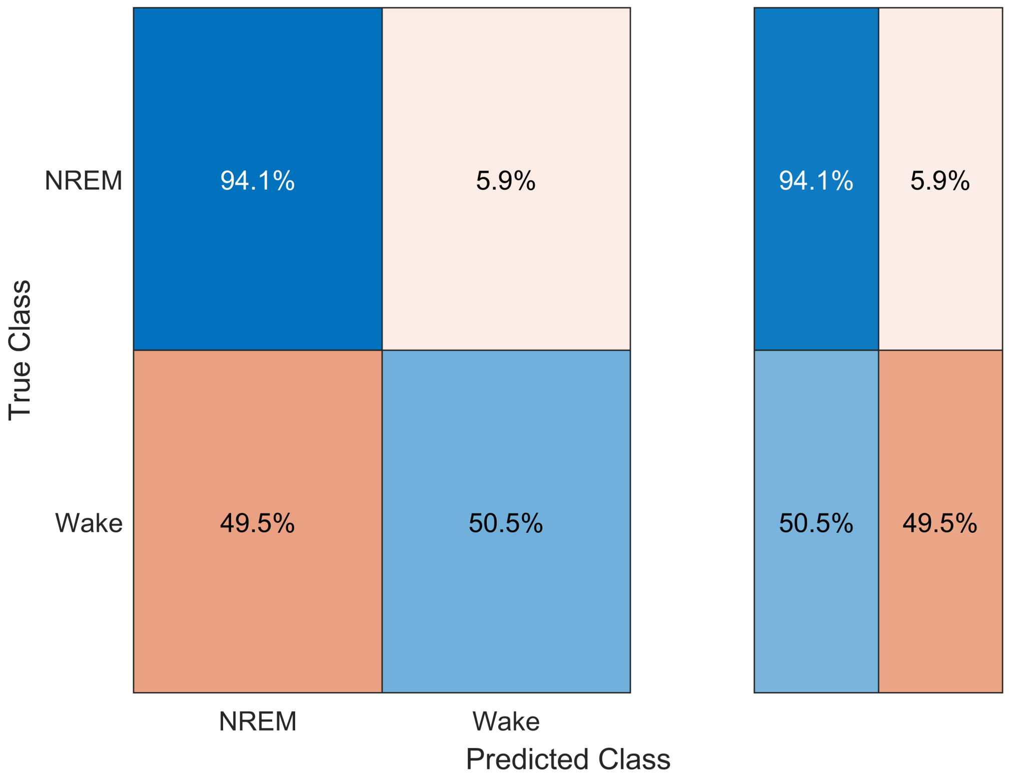
```
label = 16089×1 categorical
NREM
Wake
NREM
NREM
NREM
NREM
Wake
NREM
NREM
Wake
      ⋮

score = 16089×2 single matrix
    0.7798    0.2202
    0.3349    0.6651
    0.6163    0.3837
    0.6663    0.3337
    0.6256    0.3744
    0.8045    0.1955
    0.1394    0.8606
    0.6510    0.3490
    0.8357    0.1643
    0.3852    0.6148
      ⋮
```

```
C = confusionmat(TestingLabels,categorical(label));
[stats] = statsOfMeasure(C, 1);
```

```
stats = 9×4 table
```

	name	classes		macroAVG	microAVG
1	"true_positive"	14366	419	7.3925e+03	7.3925e+03
2	"false_positive"	410	894	652	652
3	"false_negative"	894	410	652	652
4	"true_negative"	419	14366	7.3925e+03	7.3925e+03
5	"precision"	0.9723	0.3191	0.6457	0.9190
6	"sensitivity"	0.9414	0.5054	0.7234	0.9190
7	"specificity"	0.5054	0.9414	0.7234	0.9190
8	"accuracy"	0.9190	0.9190	0.9190	0.9190
9	"F-measure"	0.9566	0.3912	0.6739	0.9190

```
figure
confusionchart(TestingLabels,categorical(label),'Normalization','row-normalized','RowSummary',
```



Five Class classification

Model trained using Random Undersampling of Majority class (The models considered here are the best models obtained using hyperparameter optimization)

```
Model = load('C:\courses\ML\project\codes\TrainNetworkProject2\BiLSTM_fiveclass\HYPERPARAMETER_
Model = Model.netLSTM;
Testing = load('C:\courses\ML\project\codes\TrainNetworkProject2\dataset_processed\FiveclassTes
Testing = Testing.BalancedsleepDataTestingCell;
TestingLabels = load('C:\courses\ML\project\codes\TrainNetworkProject2\dataset_processed\Fivecl
TestingLabels = TestingLabels.Class_Testing';
[label,score] = classify(Model,Testing)
```

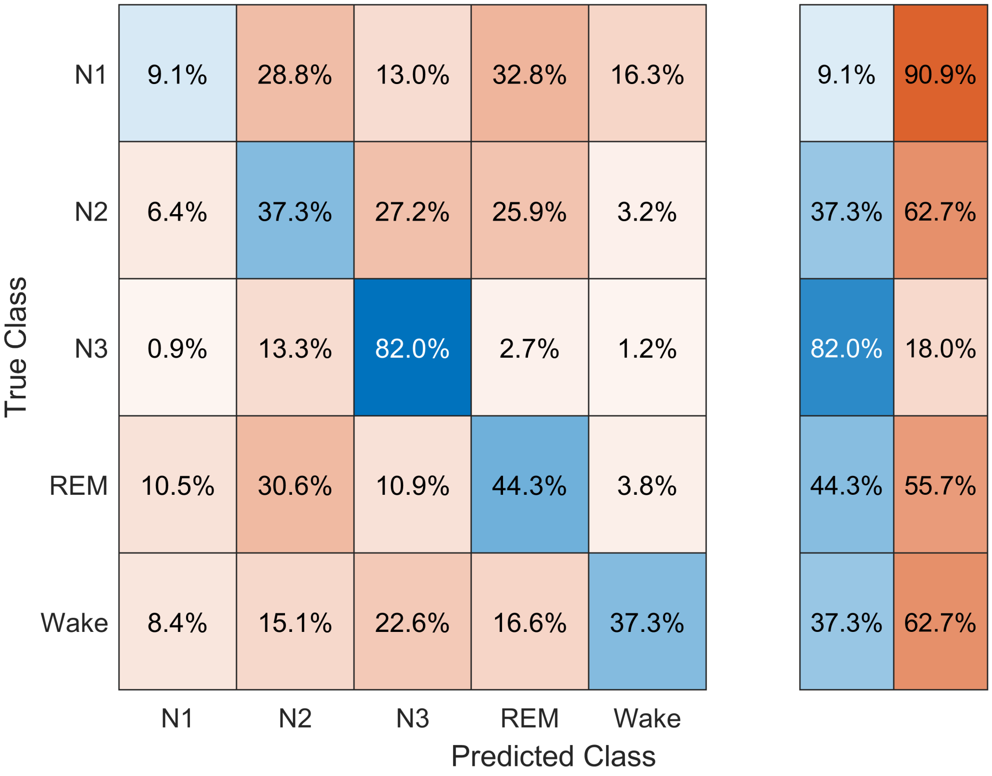
```
label = 19352×1 categorical
Wake
Wake
Wake
N2
N2
N3
Wake
N2
N3
Wake
      ⋮

score = 19352×5 single matrix
    0.0295    0.1163    0.1966    0.0041    0.6535
    0.0530    0.0839    0.2921    0.2769    0.2941
    0.1144    0.0664    0.2754    0.0247    0.5191
    0.1395    0.3937    0.2566    0.1097    0.1004
    0.0223    0.4391    0.3134    0.0291    0.1961
    0.0536    0.3070    0.4888    0.0843    0.0663
    0.1298    0.1992    0.0520    0.2576    0.3614
    0.1285    0.4375    0.0928    0.2532    0.0880
    0.1092    0.2144    0.3630    0.1461    0.1673
    0.0882    0.1299    0.0548    0.1778    0.5493
       ⋮
```

```
C = confusionmat(TestingLabels,categorical(label));
[stats] = statsOfMeasure(C, 1);
```

```
stats = 9×4 table
                         ...
```

	name
1	"true_positive"
2	"false_positive"
3	"false_negative"
4	"true_negative"
5	"precision"
6	"sensitivity"
7	"specificity"
8	"accuracy"
9	"F-measure"

```
figure
confusionchart(TestingLabels,categorical(label),'Normalization','row-normalized','RowSummary',
```



Model trained using SMOTE (The models considered here are the best models obtained using hyperparameter optimization)

```
Model = load('C:\courses\ML\project\codes\TrainNetworkProject2\BiLSTM_fiveclass\trainedModels_
Model = Model.netLSTM;
Testing = load('C:\courses\ML\project\codes\TrainNetworkProject2\dataset_processed\FiveclassTes
Testing = Testing.BalancedsleepDataTestingSMOTECell;
TestingLabels = load('C:\courses\ML\project\codes\TrainNetworkProject2\dataset_processed\Fivecl
TestingLabels = TestingLabels.Class_TestingSMOTE';
[label,score] = classify(Model,Testing)
```

```
label = 19352×1 categorical
N1
N2
N1
N2
N1
N2
REM
N2
REM
Wake
     ⋮

score = 19352×5 single matrix
    0.6301    0.2694    0.0017    0.0023    0.0965
    0.1846    0.6457    0.0057    0.0949    0.0692
    0.5772    0.2275    0.0625    0.0191    0.1137
    0.2231    0.6506    0.0178    0.0407    0.0679
    0.4869    0.2097    0.0240    0.0076    0.2719
    0.4082    0.4418    0.0015    0.0039    0.1447
    0.0196    0.1478    0.0370    0.4203    0.3754
    0.1025    0.4102    0.1385    0.1919    0.1570
    0.1044    0.2867    0.1724    0.3084    0.1281
    0.0692    0.1869    0.1244    0.1528    0.4667
      ⋮
```

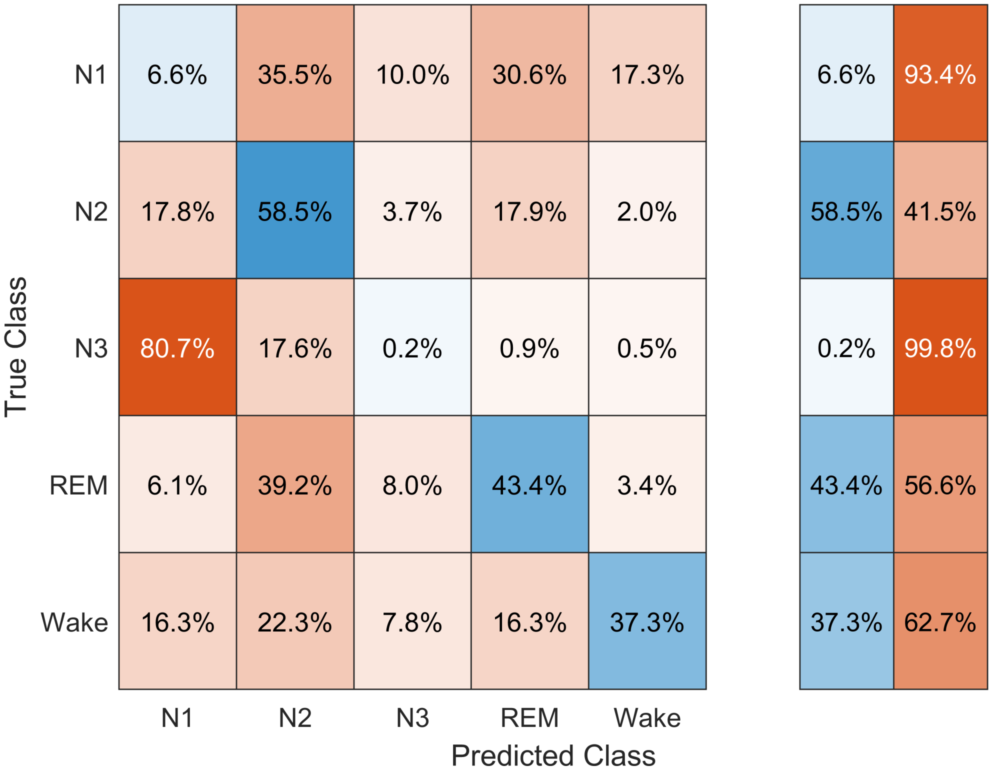
```
C = confusionmat(TestingLabels,categorical(label));
[stats] = statsOfMeasure(C, 1);
```

```
stats = 9×4 table
```

...

	name
1	"true_positive"
2	"false_positive"
3	"false_negative"
4	"true_negative"
5	"precision"
6	"sensitivity"
7	"specificity"
8	"accuracy"
9	"F-measure"

```
figure
confusionchart(TestingLabels,categorical(label),'Normalization','row-normalized','RowSummary',
```



	N1	N2	N3	REM	Wake		
N1	6.6%	35.5%	10.0%	30.6%	17.3%	6.6%	93.4%
N2	17.8%	58.5%	3.7%	17.9%	2.0%	58.5%	41.5%
N3	80.7%	17.6%	0.2%	0.9%	0.5%	0.2%	99.8%
REM	6.1%	39.2%	8.0%	43.4%	3.4%	43.4%	56.6%
Wake	16.3%	22.3%	7.8%	16.3%	37.3%	37.3%	62.7%