Funded by NSF IIS-1161997, NSF IIS 1510741, NSF IIS 0803410, NSF IIS 0808770 and NSF IIS 0237918.

We always acknowledge gifts from a Smarter Planet Award from IBM and gifts from Google, Microsoft, MERL Labs, Samsung, NetAPP and Siemens. However, any errors or controversial claims, are due to us alone.

UCR Time Series Classification Archive

Please reference as:

Dau, Hoang Anh, Eamonn Keogh, Kaveh Kamgar, Chin-Chia Michael Yeh, Yan Zhu, Shaghayegh Gharghabi, Chotirat Ann Ratanamahatana, Yanping Chen, Bing Hu, Nurjahan Begum, Anthony Bagnall, Abdullah Mueen and Gustavo Batista (2018). "The UCR Time Series Classification Archive." https://www.cs.ucr.edu/~eamonn/time_series_data_2018/

Welcome!

Dear Colleague,

If you are reading this, you are interested in using the UCR Time Series Classification Archive. This archive is a *superset* of, and completely replaces [8]. The current version, thereafter, referred to as Fall 2018 expansion, will eventually replace Summer 2015 release [9]. The archive originally was born out of our frustration with papers reporting error rates on a single data set and claiming (or implicitly suggesting) that the results would generalize [6]. However, while we think the availability of previous versions of the UCR Archive has mitigated this problem to a great extent, it may have opened other problems.

1. Several researchers have published papers on showing "*we win some, we lose some*" on the UCR Archive. However, there are many trivial ways to get "*win some, lose some*" type results on these data sets (for example, just smoothing the data, or generalizing from 1-NN to *k*-NN etc.). Using the archive can therefore *apparently* add credence to poor ideas (very sophisticated tests are required to show *small* but *true* improvement effects [3][7]). In addition Gustavo Batista has pointed out that "*win some, lose some*" is worthless unless you know *in advance* which ones you will win on! [4]. Dau et al. discuss this in detail [10].

2. It could be argued that the goal of researchers should be to solve real-world problems, and that improving accuracy on the UCR Archive is at best a poor proxy for such real-world problems. Bing Hu has written a beautiful explanation as to why this is the case [2].

Despite the above, the community generally finds the archive to be a very useful tool, and to date, more than 1,200 people have downloaded the UCR archive, and it has been referenced several hundred times.

We are therefore delighted to share this resource with you. We encourage you to read the paper accompanies this new archive expansion [10]. The password you need to unlock the data download is available in this document, *read on* to find it.

Best of luck with your research.

Eamonn, Anh and the Team

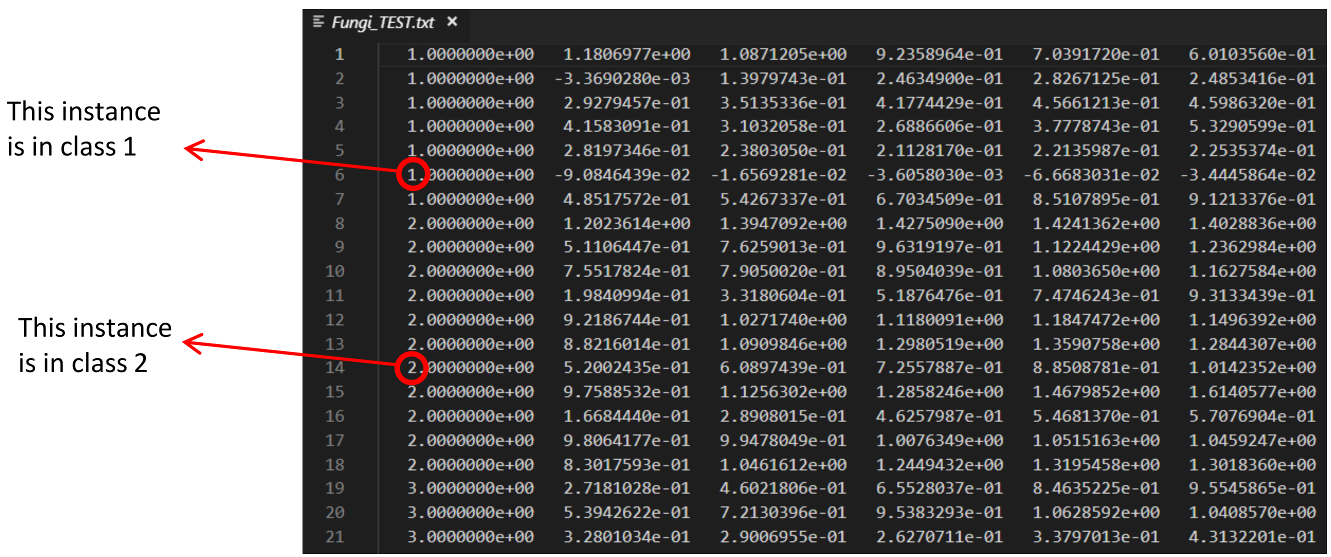
Data Format

Each of the data sets comes in two parts, a TRAIN partition and a TEST partition.
For example, for the ***Fungi*** data set we have two files, `Fungi_TEST.tsv` and `Fungi_TRAIN.tsv`
The two files will be in the same format but are generally of different sizes.
The files are in the standard ASCII format that can be read directly by most tools/languages.
For example, to read the data of ***Fungi*** data set into MATLAB, we can type…

```
>> TRAIN = load('Fungi_TRAIN.tsv');
>> TEST  = load('Fungi_TEST.tsv' );
```

…at the command line.

There is one time series exemplar per row. The first value in the row is the class label (an integer between 1 and the number of classes). The rest of the row are the data sample values. The order of time series exemplar carry no special meaning and is in most cases random. A small number of data sets have class label starting from 0 or -1 by legacy.



1.0000000e+00	1.1806977e+00	1.0871205e+00	9.2358964e-01	7.0391720e-01	6.0103560e-01
1.0000000e+00	-3.3690280e-03	1.3979743e-01	2.4634900e-01	2.8267125e-01	2.4853416e-01
1.0000000e+00	2.9279457e-01	3.5135336e-01	4.1774429e-01	4.5661213e-01	4.5986320e-01
1.0000000e+00	4.1583091e-01	3.1032058e-01	2.6886606e-01	3.7778743e-01	5.3290599e-01
1.0000000e+00	2.8197346e-01	2.3803050e-01	2.1128170e-01	2.2135987e-01	2.2535374e-01
1.0000000e+00	-9.0846439e-02	-1.6569281e-02	-3.6058030e-03	-6.6683031e-02	-3.4445864e-02
1.0000000e+00	4.8517572e-01	5.4267337e-01	6.7034509e-01	8.5107895e-01	9.1213376e-01
2.0000000e+00	1.2023614e+00	1.3947092e+00	1.4275090e+00	1.4241362e+00	1.4028836e+00
2.0000000e+00	5.1106447e-01	7.6259013e-01	9.6319197e-01	1.1224429e+00	1.2362984e+00
2.0000000e+00	7.5517824e-01	7.9050020e-01	8.9504039e-01	1.0803650e+00	1.1627584e+00
2.0000000e+00	1.9840994e-01	3.3180604e-01	5.1876476e-01	7.4746243e-01	9.3133439e-01
2.0000000e+00	9.2186744e-01	1.0271740e+00	1.1180091e+00	1.1847472e+00	1.1496392e+00
2.0000000e+00	8.8216014e-01	1.0909846e+00	1.2980519e+00	1.3590758e+00	1.2844307e+00
2.0000000e+00	5.2002435e-01	6.0897439e-01	7.2557887e-01	8.8508781e-01	1.0142352e+00
2.0000000e+00	9.7588532e-01	1.1256302e+00	1.2858246e+00	1.4679852e+00	1.6140577e+00
2.0000000e+00	1.6684440e-01	2.8908015e-01	4.6257987e-01	5.4681370e-01	5.7076904e-01
2.0000000e+00	9.8064177e-01	9.9478049e-01	1.0076349e+00	1.0515163e+00	1.0459247e+00
2.0000000e+00	8.3017593e-01	1.0461612e+00	1.2449432e+00	1.3195458e+00	1.3018360e+00
3.0000000e+00	2.7181028e-01	4.6021806e-01	6.5528037e-01	8.4635225e-01	9.5545865e-01
3.0000000e+00	5.3942622e-01	7.2130396e-01	9.5383293e-01	1.0628592e+00	1.0408570e+00
3.0000000e+00	3.2801034e-01	2.9006955e-01	2.6270711e-01	3.3797013e-01	4.3132201e-01

Sanity Check

In order to make sure that you understand the data format, you should run this simple piece of code to test *SyntheticControl* data set (you can cut and paste it, it is standard MATLAB).

Note that this is slow "teaching" code. To consider all the data sets in the archive, you will probably want to do something more sophisticated (indexing, lower bounding etc).

Nevertheless, we highly recommend you start here.

```
function UCR_time_series_test %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% (C) Eamonn Keogh %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
TRAIN = load('SyntheticControl_TRAIN.tsv'); % Only these two lines need to be changed to test a different data set. %
TEST  = load('SyntheticControl_TEST.tsv' ); % Only these two lines need to be changed to test a different data set. %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


TRAIN_class_labels = TRAIN(:,1);     % Pull out the class labels.
TRAIN(:,1) = [];                     % Remove class labels from training set.
TEST_class_labels = TEST(:,1);       % Pull out the class labels.
TEST(:,1) = [];                      % Remove class labels from testing set.
correct = 0; % Initialize the number we got correct
for i = 1 : length(TEST_class_labels) % Loop over every instance in the test set
      classify_this_object = TEST(i,:);
   this_objects_actual_class = TEST_class_labels(i);
   predicted_class = Classification_Algorithm(TRAIN,TRAIN_class_labels, classify_this_object);
   if predicted_class == this_objects_actual_class
       correct = correct + 1;
   end;
   disp([int2str(i), ' out of ', int2str(length(TEST_class_labels)), ' done']) % Report progress
end;
%%%%%%%%%%%%%%%%% Create Report %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
disp(['The dataset you tested has ', int2str(length(unique(TRAIN_class_labels))), ' classes'])
disp(['The training set is of size ', int2str(size(TRAIN,1)),', and the test set is of size ',int2str(size(TEST,1)),'.'])
disp(['The time series are of length ', int2str(size(TRAIN,2))])
disp(['The error rate was ',num2str((length(TEST_class_labels)-correct )/length(TEST_class_labels))])
%%%%%%%%%%%%%%%%% End Report %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Here is a sample classification algorithm, it is the simple (yet very competitive) one-nearest
% neighbor using the Euclidean distance.
% If you are advocating a new distance measure you just need to change the line marked "Euclidean distance"
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function predicted_class = Classification_Algorithm(TRAIN,TRAIN_class_labels,unknown_object)
best_so_far = inf;
 for i = 1 : length(TRAIN_class_labels)
     compare_to_this_object = TRAIN(i,:);
     distance = sqrt(sum((compare_to_this_object - unknown_object).^2)); % Euclidean distance
        if distance < best_so_far
          predicted_class = TRAIN_class_labels(i);
     best_so_far = distance;
    end
end;
```

```
>> UCR_time_series_test
1 out of 300 done
2 out of 300 done
…
299 out of 300 done
300 out of 300 done
The dataset you tested has 6 classes
The training set is of size 300, and the test set is of size 300.
The time series are of length 60
The error rate was 0.12
```

In this package we have produced a spreadsheet that gives basic information about the data sets (number of classes, size of train/test splits, length of time series etc)

In addition, we have computed the error rates for:

- Euclidean distance
- DTW, unconstrained
- DTW, after learning the best constraint in from the train set*
- Default rate (that is, *the most probable* class). To be consistent, we display default error rate, which is (1 – default_rate).

*Note that our simple method for learning the constraint is not necessary the best (as explained in the next slide).

You can download the entire spreadsheet displayed below in CSV format or Excel format.

ID	Type	Name	Train	Test	Class	Length	ED (w=0)	DTW (learned_w)	DTW (w=100)	Default rate	Data donor/editor
1	Image	Adiac	390	391	37	176	0.3890	0.3913 (3)	0.3960	0.9591	A. Jalba
2	Image	ArrowHead	36	175	3	251	0.2000	0.2000 (0)	0.2970	0.6971	L. Ye & E. Keogh
3	Spectro	Beef	30	30	5	470	0.3330	0.3333 (0)	0.3670	0.8000	K. Kemsley & A. Bagnall
4	Image	BeetleFly	20	20	2	512	0.2500	0.3000 (7)	0.3000	0.5000	J. Hills & A. Bagnall
5	Image	BirdChicken	20	20	2	512	0.4500	0.3000 (6)	0.2500	0.5000	J. Hills & A. Bagnall
6	Sensor	Car	60	60	4	577	0.2670	0.2333 (1)	0.2670	0.6833	J. Gao
7	Simulated	CBF	30	900	3	128	0.1478	0.0044 (11)	0.0030	0.6644	N. Saito
8	Sensor	ChlorineConcentration	467	3840	3	166	0.3500	0.3500 (0)	0.3520	0.4675	L. Li & C. Faloutsos
9	Sensor	CinCECGTorso	40	1380	4	1639	0.1030	0.0696 (1)	0.3490	0.7464	physionet.org
10	Spectro	Coffee	28	28	2	286	0.0000	0.0000 (0)	0.0000	0.5357	K. Kemsley & A. Bagnall
11	Device	Computers	250	250	2	720	0.4240	0.3800 (12)	0.3000	0.5000	J. Lines & A. Bagnall
12	Motion	CricketX	390	390	12	300	0.4230	0.2282 (10)	0.2460	0.8974	A. Mueen & E. Keogh
13	Motion	CricketY	390	390	12	300	0.4330	0.2410 (17)	0.2560	0.9051	A. Mueen & E. Keogh
14	Motion	CricketZ	390	390	12	300	0.4130	0.2538 (5)	0.2460	0.8974	A. Mueen & E. Keogh
15	Image	DiatomSizeReduction	16	306	4	345	0.0650	0.0654 (0)	0.0330	0.6928	ADIAC project
16	Image	DistalPhalanxOutlineAgeGroup	400	139	3	80	0.3741	0.3741 (0)	0.2302	0.5324	L. Davis & A. Bagnall
17	Image	DistalPhalanxOutlineCorrect	600	276	2	80	0.2826	0.2754 (1)	0.2826	0.4167	L. Davis & A. Bagnall
18	Image	DistalPhalanxTW	400	139	6	80	0.3669	0.3669 (0)	0.4101	0.7194	L. Davis & A. Bagnall
19	Sensor	Earthquakes	322	139	2	512	0.2878	0.2734 (6)	0.2806	0.7482	A. Bagnall
20	ECG	ECG200	100	100	2	96	0.1200	0.1200 (0)	0.2300	0.3600	R. Olszewski
21	ECG	ECG5000	500	4500	5	140	0.0750	0.0749 (1)	0.0760	0.4162	Y. Chen & E. Keogh
22	ECG	ECGFiveDays	23	861	2	136	0.2030	0.2033 (0)	0.2320	0.4971	physionet.org, Y. Chen & E. Keogh
23	Device	ElectricDevices	8926	7711	7	96	0.4483	0.3806 (14)	0.3990	0.7463	A. Bagnall & J. Lines

Worked Example

We can use the archive to answer the following question: *Is DTW better than Euclidean distance for all/most/some/any problems?*
As explained in [4], if DTW is only better on *some* data sets, this is not very useful unless we know ahead of time that it will be better. To test this we can build a Texas Sharpshooter plot (see [4] for details).
In brief, after computing the baseline (here, the Euclidean distance) we then compute the **expected improvement** we would get using DTW (at this stage, learning any parameters and settings), then compute **the actual improvement** obtained (using these now hardcoded parameters and settings).

When we create the Texas Sharpshooter plot , each data set fall into one of four possibilities.

In our worked example, we will try to optimize the performance of DTW, *looking only at the training data* and predict its improvement (which could be negative), in a very simple way.

Expected Improvement: We will search over different warping window constraints, from 0% to 100%, in 1% increments, looking for the warping window size that gives the highest 1-NN training accuracy (if there are ties, we choose the smaller warping window size).

Actual Improvement: Using the warping window size we learned in the last phase, we test the holdout test data on the training set with 1-NN.

Note that there are better ways to do this (learn with increments smaller than 1%, use *k*-NN instead of 1-NN, do cross validation within the test set etc). However, as the next slides show, the results are unambiguous even for this simple effort.

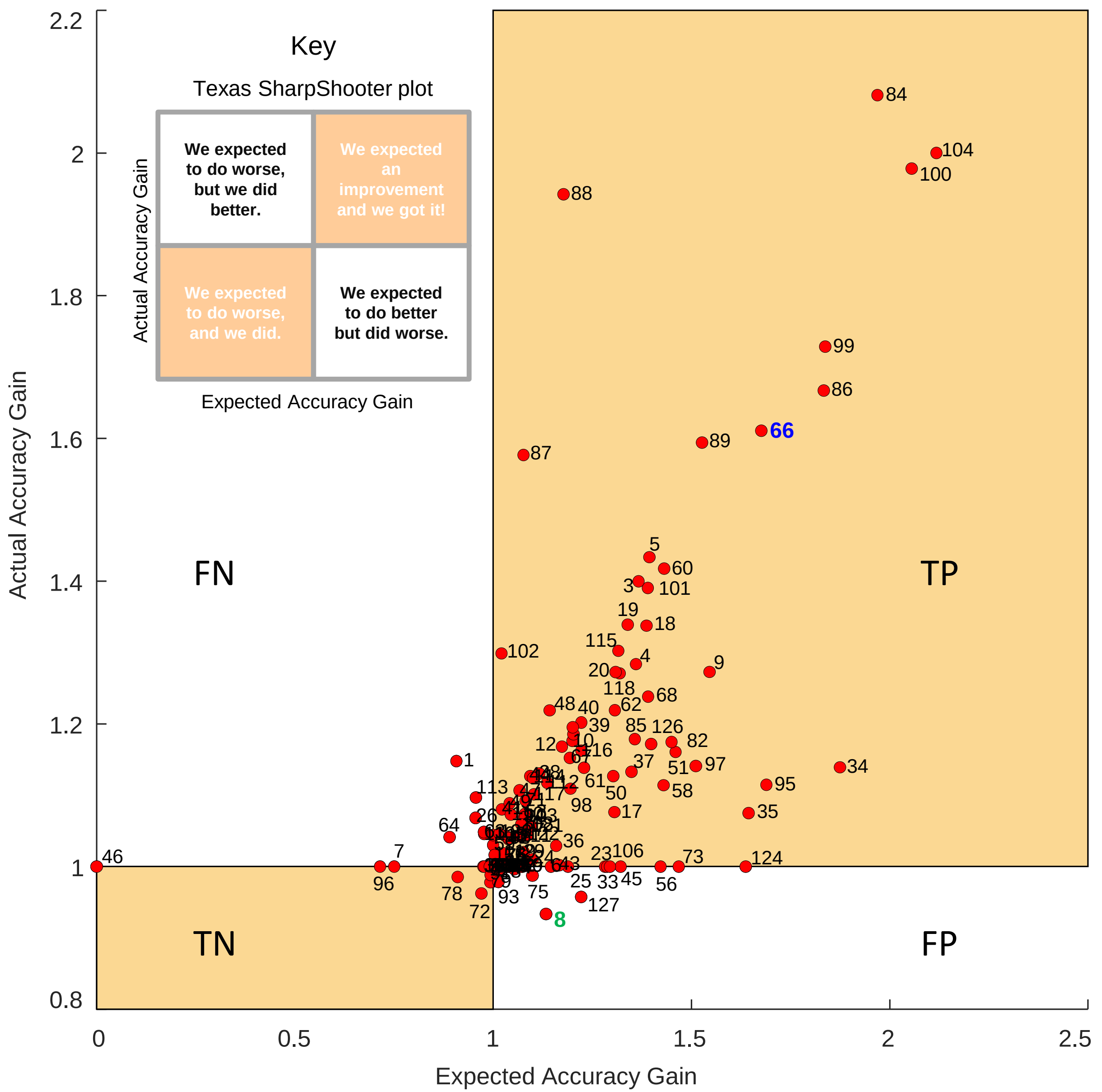
Texas Sharpshooter Plot [4]

The results are strongly supportive of the claim that "*DTW better than Euclidean distance for most problems.*"

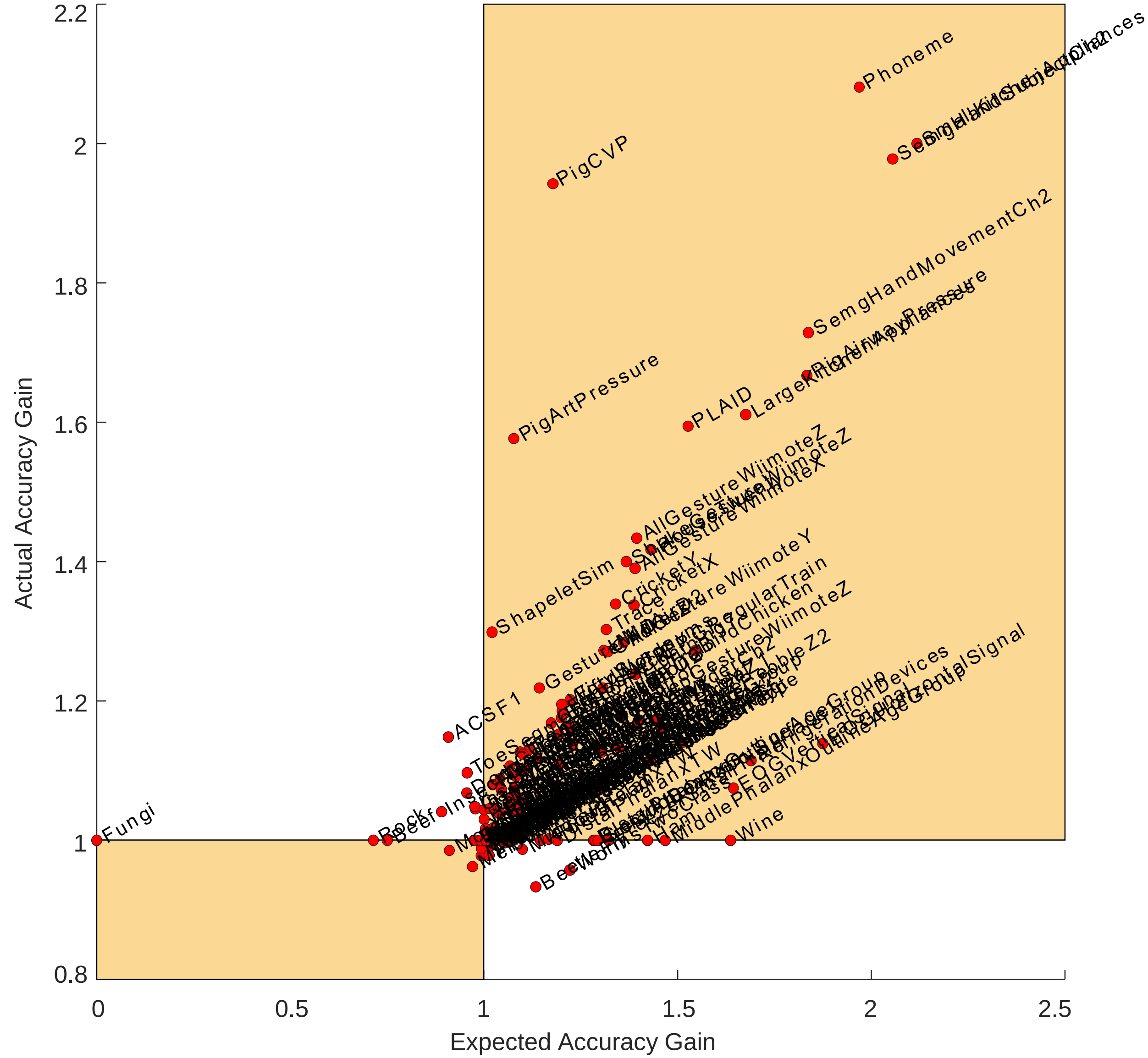
We sometimes have difficultly in predicting *when* DTW would be better/worse, but many of the training sets are tiny, making such tests very difficult.

For example, **8** is *BeetleFy*, with just 20 train and 20 test instances. Here we expected to do a little better, but we did a little worse.

In contrast, for **66** (*LargeKitchenAppliances*) we had 375 train and 375 test instances and were able to more accurately predict a large improvement.



(after plotting in MATLAB, the code is in Appendix A, you can zoom in to avoid the visual clutter seen to the right).



Suggested Best Practices/Hints

1. If you modify the data in anyway (add noise, add warping etc), please give the modified data back to the archive before you submit your paper (we will host it, and that way a diligent reviewer can test your claims while the paper is under review).
2. Where possible, we ***strongly*** advocate testing and publishing results ***on all data sets*** (to avoid cherry picking), unless of course you are making an explicit claim for only a certain type of data (i.e. classifying ***short*** time series). In the event you don't have space in your paper, we suggest you create an extended tech report online and point to it. Please see [4] (esp. Fig 14) for some ideas on how to visualize the accuracy results on many data sets.
3. If you have additional data sets, we ask that you donate them to the archive in our simple format.
4. When you write your paper, please make ***reproducibility*** your goal. In particular, explicitly state all parameters. A good guiding principle is to ask yourself: "*Could a smart grad student get the exact same results as claimed in this paper with a day effort*"?. If the answer is no, we believe that something is wrong. Help the imaginary grad student by rewriting your paper.
5. Where possible, make your code available (as we have done), it will make the reviewers task easier.
6. If you are advocating a new distance/similarity measure, we strongly recommend you test and report the 1-NN accuracy (as we have done). Note that this does ***not*** preclude the addition of other of tests (we strongly encourage additional test), however the 1-NN test has the advantage of having no parameters and allowing comparisons between methods.
7. Note that for 85 data sets of Summer 2015 release, the data are z-normalized by legacy. Paper [7] explains why this is very important. For 43 data sets of Fall 2018 expansion (this release), data are kept as is unless they were already z-normalized by donating source.

Suggested Reading

1. Wang, Xiaoyue, et al. "Experimental comparison of representation methods and distance measures for time series data." *Data Mining and Knowledge Discovery* 26.2 (2013): 275-309.
2. Hu, Bing, Yanping Chen, and Eamonn Keogh. "Time series classification under more realistic assumptions." *Proceedings of the 2013 SIAM International Conference on Data Mining*. Society for Industrial and Applied Mathematics, 2013.
3. Hills, Jon, et al. "Classification of time series by shapelet transformation." *Data Mining and Knowledge Discovery* 28.4 (2014): 851-881.
4. Batista, Gustavo EAPA, Xiaoyue Wang, and Eamonn J. Keogh. "A complexity-invariant distance measure for time series." *Proceedings of the 2011 SIAM international conference on data mining*. Society for Industrial and Applied Mathematics, 2011.
5. Keogh, Eamonn, and Shruti Kasetty. "On the need for time series data mining benchmarks: a survey and empirical demonstration." *Data Mining and knowledge discovery* 7.4 (2003): 349-371.
6. Rakthanmanon, Thanawin, et al. "Addressing big data time series: Mining trillions of time series subsequences under dynamic time warping." *ACM Transactions on Knowledge Discovery from Data (TKDD)* 7.3 (2013): 10. If you are claiming that DTW is too slow… Maybe, but read this first.
7. Lines, Jason, Sarah Taylor, and Anthony Bagnall. "Time Series Classification with HIVE-COTE: The Hierarchical Vote Collective of Transformation-Based Ensembles." *ACM Transactions on Knowledge Discovery from Data (TKDD)* 12.5 (2018): 52.
8. Keogh, E., Zhu, Q., Hu, B., Hao. Y., Xi, X., Wei, L. & Ratanamahatana, C. A. (2011). "The UCR Time Series Classification/Clustering Homepage".
9. Chen, Yanping, Eamonn Keogh, Bing Hu, Nurjahan Begum, Anthony Bagnall, Abdullah Mueen, and Gustavo Batista. 2015. "The UCR Time Series Classification Archive." https://www.cs.ucr.edu/~eamonn/time_series_data/
10. Dau, Hoang Anh, Anthony Bagnall, Kaveh Kamgar, Chin-Chia Michael Yeh, Yan Zhu, Shaghayegh Gharghabi, Chotirat Ann Ratanamahatana and Eamonn Keogh, "The UCR Time Series Archive." 2018 https://arxiv.org/abs/1810.07758 Early adopters (late 2018) please cite this, after early 2020, please check for a peer-reviewed version of this paper.
11. Bagnall, Anthony, et al. "The great time series classification bake off: a review and experimental evaluation of recent algorithmic advances." *Data Mining and Knowledge Discovery*31.3 (2017): 606-660.

Appendix A:
Texas Sharpshooter Plots

Here is the code we used to produce the Texas Sharpshooter plots.

```matlab
function [] = plot_texas_sharpshooter(result_file)
% Compute a Texas Sharpshooter plot of DTW over Euclidean Distance. See
% Batista, Wang and Keogh (2011) A Complexity-Invariant Distance Measure
% for Time Series. SDM 2011
% % Last updated April 2019 by Hoang Anh Dau

% For example, if we want to construct the figure for comparison between
% Euclidean distance (ED) and Dynamic Time Warping distance (DTW), we
% compute the following statistics:
%
% expected_accuracy_gain = DTW_train_accuracy / ED_train_accuracy
% actual_accuracy_gain = DTW_test_accuracy / ED_test_accuracy

% By definition, the train accuracy of DTW is always greater than the train
% accuracy of ED, which makes expected accuracy gain meaningless in this
% context. Therefore, we only use ED test accuracy, considering it as the
% normalizing factor.

% In the result spreadsheet texas_plot_2018.csv:
% row 1 is header
% column 1 is datataset name
% column 2 is ED train error rate
% column 3 is ED test error rate
% column 4 is DTW train error rate
% column 5 is DTW test error rate

% Example of usage:
% result_file = 'texas_plot_2018.csv';
% plot_texas_sharpshooter(result_file);

%%
% read in result spreadsheet
result = importdata(result_file, ',', 1);
error_rates = result.data;

% convert error to accuracy, by subtracting from 1
texas_values = 1 - error_rates;

expected_accuracy_gain = texas_values(:,3)./texas_values(:,2);
actual_accuracy_gain   = texas_values(:,4)./texas_values(:,2);

% produce plot just so we can get Xlim and Ylim
figure;
scatter(expected_accuracy_gain,actual_accuracy_gain, 20, 'r', 'filled');

Xaxis = get(gca,'XLim');
Yaxis = get(gca,'YLim');

hold on; axis square;

% bottom left quadrant
patch([Xaxis(1) 1 1 Xaxis(1)],[Yaxis(1) Yaxis(1) 1 1],[0.9843 0.8471 0.5765]);
% top right quadrant
patch([1 Xaxis(2) Xaxis(2) 1],[1 1 Yaxis(2) Yaxis(2)],[0.9843 0.8471 0.5765]);

scatter(expected_accuracy_gain,actual_accuracy_gain, 20, 'r', 'filled');

xlabel('Expected Accuracy Gain');
ylabel('Actual Accuracy Gain');

% plot with symbol as number
for i = 1: length(texas_values(:,1))
    text(expected_accuracy_gain(i),actual_accuracy_gain(i),int2str(i))
end

% % uncomment this to plot with symbol as data set name
% % note that the order of texas_names and texas_values must be the same.
% texas_names  = result.textdata(2:end, 1);
% for i = 1: length(texas_values(:,1))
%     text(expected_accuracy_gain(i),actual_accuracy_gain(i),texas_names(i,:),'rotation',+30)
% end

end
```

Here the result summary file for making the Texas Sharpshooter plot.

texas_plot_2018.csv

- row 1 is header
- column 1 is data set name
- column 2 is ED train error rate
- column 3 is ED test error rate
- column 4 is DTW train error rate
- column 5 is DTW test error rate

```
Name, ED train, ED test, DTW train, DTW test
ACSF1,0.57,0.46,0.51,0.38
Adiac,0.3949,0.3887,0.3897,0.3913
AllGestureWiimoteX,0.5033,0.4843,0.2833,0.2829
AllGestureWiimoteY,0.4433,0.4314,0.2267,0.27
AllGestureWiimoteZ,0.5967,0.5457,0.3667,0.3486
ArrowHead,0.0833,0.2,0.0833,0.2
Beef,0.5,0.3333,0.5,0.3333
BeetleFly,0.45,0.25,0.15,0.3
BirdChicken,0.3,0.45,0.15,0.3
BME,0.1,0.1667,0,0.02
Car,0.3,0.2667,0.2833,0.2333
CBF,0.1667,0.1478,0,0.0044
ChinaTown,0.05,0.0466,0.05,0.0466
ChlorineConcentration,0.3662,0.35,0.3662,0.35
CinCECGTorso,0.15,0.1029,0.075,0.0645
Coffee,0,0,0,0
Computers,0.444,0.424,0.248,0.38
CricketX,0.4026,0.4231,0.2,0.2282
CricketY,0.4564,0.4333,0.241,0.241
CricketZ,0.4231,0.4128,0.2256,0.2538
Crop,0.2928,0.2883,0.2928,0.2883
DiatomSizeReduction,0.0625,0.0654,0.0625,0.0654
DistalPhalanxOutlineAgeGroup,0.1975,0.3741,0.1975,0.3741
DistalPhalanxOutlineCorrect,0.2167,0.2826,0.2117,0.2754
DistalPhalanxTW,0.2475,0.3669,0.2475,0.3669
DodgerLoopDay,0.5385,0.45,0.4744,0.4125
DodgerLoopGame,0.25,0.1159,0.05,0.0725
DodgerLoopWeekend,0.05,0.0145,0,0.0217
Earthquakes,0.2578,0.2878,0.2329,0.2734
ECG200,0.14,0.12,0.14,0.12
ECG5000,0.066,0.0751,0.064,0.076
ECGFiveDays,0.1739,0.2033,0.1739,0.2033
ElectricDevices,0.2911,0.4492,0.2911,0.4492
EOGHorizontalSignal,0.2652,0.5829,0.2182,0.5249
EOGVerticalSignal,0.3398,0.558,0.2735,0.5249
EthanolLevel,0.7044,0.726,0.6825,0.718
FaceAll,0.1125,0.2864,0.0375,0.1917
FaceFour,0.3333,0.2159,0.125,0.1136
FacesUCR,0.245,0.2307,0.075,0.0878
FiftyWords,0.3644,0.3692,0.2289,0.2418
Fish,0.24,0.2171,0.2,0.1543
FordA,0.3244,0.3348,0.3091,0.3091
FordB,0.3251,0.3938,0.2929,0.3926
FreezerRegularTrain,0.1933,0.1951,0.12,0.093
FreezerSmallTrain,0.1071,0.3242,0.1071,0.3242
Fungi,1,0.1774,1,0.1774
GestureMidAirD1,0.4615,0.4231,0.3846,0.3615
GestureMidAirD2,0.524,0.5077,0.4375,0.4
GestureMidAirD3,0.6635,0.6538,0.6394,0.6231
GesturePebbleZ1,0.1591,0.2674,0.0455,0.1744
GesturePebbleZ2,0.1712,0.3291,0.0205,0.2215
GunPoint,0.04,0.0867,0.04,0.0867
GunPointAgeSpan,0.0519,0.1013,0.0296,0.0348
GunPointMaleVersusFemale,0,0.0253,0,0.0253
GunPointOldVersusYoung,0.1029,0.0476,0.0221,0.0349
Ham,0.1468,0.4,0.1468,0.4
HandOutlines,0.143,0.1378,0.143,0.1378
Haptics,0.4839,0.6299,0.471,0.5877
Herring,0.5938,0.4844,0.4844,0.4688
HouseTwenty,0.25,0.3361,0.05,0.0588
InlineSkate,0.7,0.6582,0.58,0.6109
InsectEPGRegularTrain,0.2419,0.3213,0.1129,0.1727
InsectEPGSmallTrain,0.4118,0.3373,0.3529,0.3052
InsectWingbeatSound,0.5182,0.4384,0.5,0.4152
ItalyPowerDemand,0.0448,0.0447,0.0448,0.0447
LargeKitchenAppliances,0.384,0.5067,0.1733,0.2053
Lightning2,0.2833,0.2459,0.1,0.1311
Lightning7,0.3571,0.4247,0.2,0.2877
Mallat,0.0182,0.0857,0.0182,0.0857
Meat,0,0.0667,0,0.0667
```

The Password

- As noted above. My one regret about creating the UCR Archive is that some researchers see improving accuracy on it as a *sufficient* task to warrant a publication. I am not convinced that this should be the case (unless the improvements are very significant, or the technique is so novel/interesting that it might be of independent interest).
- However, the archive is in a very contrived format. In many cases, taking a real-world data set, and putting it into this format, is a *much* harder problem than classification itself!
- Bing Hu explains this nicely in the introduction to her paper [2], I think it should be required reading for anyone working in this area.
- The password is the missing words from this sentence "*Why would ******* use the archive and not acknowledge it*?
- The sentence is in the Introduction of [10]. The paper is available for download on the UCR Archive webpage or at https://arxiv.org/abs/1810.07758

Personal note from Eamonn

I am somewhat bemused by the hundreds of papers that use the UCR Archive, but do not acknowledge or thank the archivists.

Many such papers thank funding agencies, people that donated CPU time, friends that gave feedback etc. But many of these papers could not have been written without access to dozens of labeled time series data sets.

These dozens of labeled data sets were provided, completely for free! And these data sets represent (now) at least a thousand hours of work by my students and collaborators, to create or collect, to clean and annotate, to compute benchmarks etc.

It does seem like an acknowledgment would be classy ;-)

Acknowledgments

The authors would like to thank Prof. Eamonn Keogh and all the people who have contributed to the UCR time series classification archive for their selfless work. We also thank the anonymous reviewers for their valuable advice.

This work has been supported by Major Project of High Resolution Earth Observation System of China (Grant No.03-Y20A04-9001-15/16), the CNES TOSCA-VEGIDAR Program, and CAS-CNRS Joint Doctoral Promotion Program.

References

[1] Bailly, A., Malinowski, S., Tavenard, R., Guyet, T., Chapel, L., 2015. Bag-of-Temporal-SIFT-Words for time series classification. In: ECML/PKDD Workshop on Advanced Analytics and Learning on Temporal Data.

[2] Bartolini, I., Ciaccia, P., Patella, M., 2005. Warp: Accurate retrieval of shapes using phase of fourier descriptors and time warping distance. IEEE Transactions on Pattern Analysis and Machine Intelligence 27 (1), 142–147.

[3] Batista, G. E., Wang, X., Keogh, E. J., 2011. A complexity-invariant distance measure for time series. In: Proceedings of SIAM International Conference on Data Mining. Vol. 11. SIAM, pp. 699–710.

[4] Belongie, S., Malik, J., Puzicha, J., 2002. Shape matching and object recognition using shape contexts. IEEE Transactions on Pattern Analysis and Machine Intelligence 24 (4), 509–522.

[5] Bergmann, B., Hommel, G., 1988. Improvements of general multiple test procedures for redundant systems of hypotheses. In: Multiple Hypothesenprüfung/Multiple Hypotheses Testing. Springer, pp. 100–115.

About the baseline results reported – Before you ask

- Did you z-normalize the data before passing to the algorithm?
- There can be different implementations of DTW. Some implementations divide the distance by the warping path length; some use a different step patterns etc. We use MATLAB implementation of DTW [r1].

```
dist = dtw(time_series_1, time_series_2, window_size, 'squared');
```

- We use MATLAB implementation of *k*-NN [r2]

```
mdl = fitcknn(train_data, train_label, 'Standardize', 0, 'NSMethod', 'exhaustive');
```

- We use leave-one-out cross-validation to learn the warping constraint

```
cross_validation = crossval(mdl, 'LeaveOut', 'on');
```

- For constrained warping, if the percentage of time series length results in a real number, you can round up or round down. We round up.
- We round the error rate to four decimal places. For a more comprehensive result comparison and other resources, we recommend the UEA & UCR Time Series Classification Repository [r3].

[r1] https://www.mathworks.com/help/signal/ref/dtw.html
[r2] https://www.mathworks.com/help/stats/classificationknn.html
[r3] http://www.timeseriesclassification.com/index.php

About the baseline results reported – How we handle special cases

- For time series of different lengths:
  - In storing data: We pad NaN (to the end) to the length of the longest time series. This makes it convenient when loading data into MATLAB.
  - In computing baselines: We add low amplitude random numbers (to the end) to the length of the longest time series to make all time series of equal length.

```
% pad_len is the length of the padding portion
time_series = [time_series, rand(1, pad_len)/1000];
```

- For time series with missing values
  - In storing data: Missing values are represented with NaN (if NaN is at the end of the time series, it is not real missing values).
  - In computing baselines: We use linear interpolation.

```
time_series = fillmissing(time_series, 'linear', 2, 'EndValues', 'nearest');
```

- There are 15 data sets fall into either of these special cases. No data sets is both of variable-length and with missing values. In the interest of reproducible research, we also provide the processed version (equal length, no missing values) of data that we used to produce the baseline results.

You are in good company, just some of the people that requested the UCR Archive prior to Summer 2015. Page 1

Shiyuan Liu, Li Lv.
Thiago Santos Quirino, Mei-Ling Shyu
Pierre-Franqois Marteau,
Cosmin Bocaniala, Lancaster University.
Yueguo Chen,Anthony K.H. Tung, Beng Chin Ooi, National University of Singapore.
Vernon Rego, Vernon Rego. Purdue University.
Mislav Malenica, Tomislav Smuc
Man Hon WONG, ZHOU Mi, The Chinese University of Hong Kong.
Guillaume Bouchard Xerox Research Centre Europe.
Hoa Vo and David Joslin Seattle University.
Carlotta Orsenigo , University degli Studi di Milano.
Dr. Paolo Ciaccia
Xiaoqing Weng, Jiaotong University.
Longin Jan Latecki and Qiang Wang, Temple University.
Tony Bagnall
Michail Vlachos, IBM.
Bernard Hugueney
Sourav Mukherjee
Marcos M. Campos (Oracle)
Ludmila I. Kuncheva
Edward Omiecinski and Jun Li
Victor Eruhimov, Intel
Rob Jasper
Andre Coelho
Gernot Herbst
Vit Niennattrakul
Nozomi Matsuda
Flavio Miguel Varejao and Idilio Drago
HAORIANTO COKROWIJOYO TJIOE
Molnar Miklos
Steinn Gudmundsson and Thomas Runarsson
Niall Adams and Sai Wing Man
Isabel Maria Marques da Silva, Maria Eduarda da Rocha Pinto Augusto da Silva and Joaquim Fernando Pinto da Costa
Panagiotis Papapetrou and George Kollios
Huang Tan
Sergio Guadarrama
Alicia Troncoso Lora
Pyry Avist
Peng-Yi Lai
Yong Fu
Soheil Bahrampour
Long Yao and Meng Bo
Robert Moskovitch and Yuval Shahar
Abdellali Kelil
Puspadevi Kuppusamy
Qun Dai and Songcan Chen
Lisa Gralewski
Maria Teresinha Arns Steiner and Rosangela Villwock
Amir Ahmad and Galvin Brown
Abhijit Jayant Kulkarni
Xingquan (Hill) Zhu
Amol Deshpande and Qiang Qiu
Vercellis Carlo and Gianni Alberti
Pamela Nerina Llop
Tobias Scheffer
Jochen Fischer
Mao Ye and Yingying Zhu
Cintia Lera
George Runger and Rohit Das
Omar U. Florez and Seungjin Lim
Ruy Luiz Milidiu and Pedro Teixeira
Mykola Galushka and Dave Patterson
Rahul Sinha
Minh Hoai Nguyen and Fernando de la Torre
Eric Eaton
NGUYEN Van Hanh
Lucas F. Rosada
Nicky Van Thuyne
Skopal Tomas and Michal Vajbar
Carlo Piccardi and Martina Maggio
Muhammad Aamir Khan
Larry Deschaine
Janosa Andras
Andrew Starkey
Karthik Marudhachalam and Ansaf Salleb-Aouissi
Rayner Alfred and Samry Mohd Shamrie Sainin
Evins lio and Oumsis Oumsis
Hanjing, Su
Marco Cuturi
Wang Zhimin
Miao Zeng and Yubao Liu
Rene Vidal and Rizwan Chaudhry
Susan Cheng and Min Ding
Amit Ganatra and Dhaval Bhoi

awei Han & Manish Gupta
linstry Liang and Qin Lv
Eirik Benum Reksten
Lucas Gallindo Martins Soares
Sungyoung Lee and La The Vinh
Huidong (Warren) Jin
Santiago Velasco
Jairo Cugliari
Quazi Abidur Rahman
Hungyu Henry Lin and James Davis
Azuraliza Abu Baka and Almahdi Mohammed Almahdi
Benjamin Bustos and Victor Sepulveda
Krisztian Buza
Young Xin
Shen Wang and Haimonti Dutta
Victor Garcia-Portilla
Samir Al-Stouhi
Guo Fei
Teodor Costachioiu
Stephen Pollard
Eser Kandogan
lida rashidi
Takashi Washio and Satoshi Hara
Meizhu Liu and Baba C. Vemuri
Jose Principe and Sohan Seth
Ramanuja Simha and Rahul Tripath
Rodolphe JENATTON and Francis Bach
Siladitya Dey and Ambuj Singh
Mahsa Orang and Nematollaah SHIRI V.
Saket Saurabh
Wu Wush
Deepti Dohare
Subhajit Dutta.
Evgeny Pyatkov
Ion George TODORAN
Qin Zou
K.S.SUBHASHINI
Feng Gu, Julie Greensmith
Pedro Felzenszwalb
Onur Seref
Alessia Albanese
Seniha Esen Yuksel and Paul Gader
Alexandros Iosifidis
Rania Hussein and Neamat Farouk El-Gayar
Matteo Danieletto
Ying Zhang
Iman Rashidi
Xiaobo Wu and S Chen
Sun Lei and Yu-Jiu Yang
Belen Martin Barragan
Mustafa Ahmad and Hua-Liang Wei
KHALIL BRAHMA
Maria Luisa Sapino and Rosaria Rossini
Peiman Barnaghi and Azuraliza Abu Bakar
Hammoud Aljoumaa and Dirk Soeffker
Haris Gavranovic
Matthias Mouton
Mohammad Reza Daliri
FABIO STELLA
Rohit J. Kate
Ruoqian Liu, Yi Lu Murphey
jennifer bavani Vishwanath Jhoe and Rahul Singh
George Runger and Mustafa Baydogan
Mohamed Ghalwash and ZORAN OBRADOVIC
Jieyue Li
Josif Grabocka
Marco Cristani
Tomas Olsson
Tianwei Liu
Cexus Jean-Christophe
Antonello Rizzi and Antonello Rizzi
Zhai Ting Ting
Ming Zhang
Shanghai Jiao Tong and Zhang Zhongfei
Zhang Zhang and Dacheng Tao
Trevor Tian
Ta Minh Thuy
Yin Zhou and Kenneth Barner
Suzanne Tamang and Simon Parsons
carlotta orsenigo and carlo vercellis
Bahaeddin Eravci and Hakan Ferhatosmanoglu
Haobo Jiang
Bingyi Kang
Jing Zhang
Jon Froehlich and Yi-Chun Ko
Matthias Klusch and Josenildo Silva
Jing Yang
Komang Sidhi Artha
Philip Knaute and Nick Jones
Ivan Mitzev and Nick Younan
Juhua Hu and Jian Pei
Hasari Tosun and John Sheppard
Md. Abdul Awal
Andrey Sukhanov
Cecil Schmidt
Sebastian Schmitz,Marcin Grzegorzek, Lukas Köping

Benjamin Bustos and Heider Sanchez Enriquez
Thinh Vuong and Thinh Vuong
Yiorgos Adamopoulos
Zhao Xiaohui
Xue Bai
Baker Abdalhaq
Paria Shirani and Mohammad Abdollahi Azgomi
Manuel Oviedo de la Fuente.
JiRinki Pakshwar
Nikita Mishra and Somesh Kumar
Mike Jones
Deák Szilárd Nicola Rebagliati
Ira Assent and Søren Chrestensen
Marcela Svarc
Bankó Zoltán
yasuko matsubara and Yasushi Sakurai
Liu, Yueming and Su, Jianzhong
hongxiong ye and QIAN-LI MA
Huseyin Kaya
Hezi Halpert and Mark Last
Paolo Missier and Tudor Miu
Jiandong Wang and Peng-Cheng Zou
Shahriar Shariat Talkhoonche and Vladimir Pavlovic
Hideo Bannai
Gerard Medioni and Qian Gong
Shengfa Miao and Arno Knobbe
Pengfei Ravikumar
Nabil Alshurafa and Majid Sarrafzadeh
Vladimir Miškovic
David Providakes and Jason Wang
Wei Ding and Yang Mu
Meng-Jung Shih and Shou-de Lin
OSman Gunay
Lu Min and Xiaoru Yuan
Hiba Hamrouni
Laurence Chu
Puneet Singh
Joaquim Vinicius C Assuncao
B Kalyan Kumar and debarun kar
Jim Austin and Alexander Fargus
Cheng Li and Osmar R. Zaiane
Bilel Ben ali
Oya Celiktutan, bulent sankur and Cem Bakan
Michael Pettigrew
Katarzyna Kaczmarek and Olga Hryniewicz
Jun ha Park
Daniele Cecasa and Fabio Antonio Stella
Anqi Wu and Ragha Raghavendra
Nur Zuria Haryani and Abdul Razak
Hamdan Ana Arribas Gil
Zhen Bo and Junpeng Bao
Alexio Kotsifakos and Vassilis Athitsos
Vincent S. Tseng
Ryan Kleck and Gu Quanquan
Gavin Smith
Usue Mori, Jose A. Lozano, Alexander Mendiburu
Mohammad Hasan Alveshah
Sarah Brockhaus and Sonja Greven
Fanghuan Li
Wang Yu
Mayank Mohta and Adit Madan
Sreenivasulu Reddy and Naga sundaram
Chiatung Mao and Jia-Dong M
Maciej Luczak
Zhiguang Wang and Tim Oates
Lukas Pfahler and Marco Stolpe
Rickey Agarwal
Yada Zhu and Jingrui He
Thanhvinh Vu and Duong Tuan Anh
Chris Carbone
Xiaohui Huang
Nicolas Ragot
Tanmoy Mondal
Guillem Rigaill
Fatemeh Kaveh-Yazdy and mohammad reza zare
Azer Kerimov
Hanan Shteingart
Xin Qi
S. Mohanavalli
Emanuele Ruffaldi and Leonard Johard
Tomasz Pander
Pierre Allain and Thomas Corpetti
Changcheng Xiang
Wojciech Mioduszewski and Jurek Blaszczynski
Zhang Daokun
Xiangzeng Kong
Amparo Alonso Betanzos
Olivia Maoy and Yixin Chen
Carlo GAETAN and Paolo Girardi
Farzad Noorian

Greg Fanslow
Xiaojin Li
Korkinof, Dimitrios
Yixin Chen and Yujie He
Christina Yassouridis
itti laurent and jiaping ZHAO
Ahmad Al-Hasanat
Artur Dubrawski, Mathieu Guillame-Bert
Rodica Potolea and Victor Ionescu
JANETH CAROLINA RENDON AGUIRRE
Javier Prieto
Gianniotis, Nikos
Wesley Chen & Pavlos Protopapas
Sahar Torkamani and Volker Lohweg
Najarian, Kayvan
Xiaojiang Qian
Hudson Fujikawa and Renato Ishii
Antonio Gattone
Alexandra M. Carvalho and Susana Vieira and Lucia Cruz
Yanci Lyu and James Kwok
Peiqi Zhao
Sergey Milanov
Zhibo Zhu
Stefan Kramer and Atif Raza
Brijnesh-Johannes Jain
Qian Xiaochen
Felix Reinhart and Vitali Aswolinskiy
Kwangho Heo
Cun Ji
Sara Carolina Gouveia Rodrigues and Cláudia Antunes
Kumar Vasimalla
Jose Alejandro Cordero
Liu Xiao
Carlos Francisco Soares de Souza
Yurong Luo
Wang, Kaile(Keller)
Babak Hosseini
Dong Hao
Colin O'Reilly
Alon Gold
Rahim Khan
Adam Oliner
Arvind Balasubramanian and B. Prabhakaran
Soon-hwan Kwon, Jong Hoon Lee
Chen Yun
ndre Rodrigo Sanches and Nina Sumiko Tomita Hirata
Claudio Piciarelli and Gian Luca Foresti
Wei T. Yue
Michael Botsch and Josef A. Nossek
Bingyu Sun
Sun Fu-Shing
Babak Amiri
Xing ChunXiao and Du Xutao, Tsinghua University
Elloumi Samir , Sondess Bentekaya
Li Shijin
Erik Learned-Miller, Marwan A. Mattar
Chiranjib Bhattacharya,Karthik K
Nicandro Cruz Ramirez
Jiankui Guo Fudan University
Bin Z Zhang IBM
Yi-Dong Shen and Zhiyong Shen
Georgios Evangelidis, Leonidas Karamitopoulos
Hendrik Purwins
Jignesh M. Patel and Michael Morse
Gert Van Dijck and Marc Van Hulle
Chao Hui Lee and Vincent Tseng
Linh Tran (Boeing)
Hugo Alonso Vilares Monteiro and Joaquim Fernando Pinto da Costa
David Minnen
Tsuyoshi Mikami
Qiang Yang and Sinno Pan
Paolo Tormene
Hui Ding and Peter Scheuermann
Ronaldo Cristiano Prati
Christian Gruber and Bernhard Sick
Silvia Chiappa
Ankur Jain
Maria Cristina Ferreira de Oliveira and Aretha Barbosa Alencar
Febri Andriani
Myeong-Seon Gi
Pengtao jia
Farid Seifi.
Clodoaldo Aparecido de Moraes Lima
Konstantinos Blekas
Juan Prada
Ben Fulcher and Nick Jones
Victor Sheng
Cedric Frambourg and Ahlame Douzal Chouakria
Rakia JAZIRI and Mustapha Lebbah
Dave Marshall and Andrew Aubrey
Omar Torfason
Le Huu Thanh and Duong Tuan Anh.
Laila Fatehy and Mahmoud Gabr
Alle veenstra
Nima Riahi
Mohiuddin Ahmad and md. Abdul Awal
Hyokyeong Lee and Rahul Singh

Inderjit Dhillon and HYUK CHO
Aida Valls
Narayanan Chatapuram Krishnan and Sethuraman Panchanathan
Stephan Gunnemann and Thomas Seidl
Thirumaran Ekambaram and M. Narasimha Murty
Christine Preisach and Lars Schmidt-Thieme
Dino Isa and Rajprasad Kumar
Feibao Zhuo
Frans van den Bergh
Kfir Glik
Xiao Yu
Yingying Zhu
Rosanna Verde and Antonio Balzanella
Paul Baggenstoss
Koichi ASAKURA and Wei Fan
Lucia Sacchi and Iyad Batal
Morne Neser
Luca Chiaravalloti
vikram deshmukh
Harri M.T. Saarikoski
Caio Nogara Andreatta and Neusa Grando
Ying Xie
Ulf Grobekathofer
Christophe Genolini
Celine Robardet
Musa Chemisto
Wangmeng Zuo
Paolo Remagnino
Soumi Ray and Tim Oates
Pierre Ganarski and Francois Petitjean
Joydeep Ghosh and John Tourish
Alireza-Xaker
Wilfgang nejdl and ERNESTO DIAZ-AVILES
Romain Tavenard and Laurent Amsaleg
Hahn-Ming Lee, Christos Faloutsos,Hsing-Kuoh Pao, Ching-Hao (Eric) Mao
Woong-Kee Loh
bikesh singh
Marco Grimaldi, Cesare Furlanello and Giuseppe Jurman
Chonghui Guo
Saeid Rashidi
Yanchang Zhang
S R Kannan
FRANCISCO JAVIER CUBEROS GARCIA-BAQUERO
Dr. SOTIRIOS P. CHATZIS
Ming Luo and Igor G. Zurbenko
Yuya Sakamoto
Yosi Keller
David Corney
Alexander Gray and Nishant Mehta
Tomoyuki Hiroyasu and Takuma Nishii
Blaz Strle and Martin Mozina
Yasin Bakis
Mrinal Mandal and Cheng Lu
Hendrik Blockeel and Kurt Driessens
Jonas Haustad and Yongluan Zhou
Julien Rabin
Oscar Gerardo Sanchez Siordia and Isaac Martin de Diego
Scott Deeann Chen
Lixia Wu
Tewari, Ashutosh
RANGEENA T V radhakrishnan
Zhou Zhou
Troy Raeder
Amit Thombre
Nhut Ta Hoang
Oisin Mac Aodha
Ali Farahpoor
Xingwang Zhao
Wu Gang
Tetsuji Hidaka
Wang He Nan
David Weston
Dr. Huayang (Jason) Xie
Jia-Lien Hsu
Saeed R. Aghabozorgi
Paolo Giussani
Zhenhui (Jessie) Li and Jiawei Han
Jeff Patti
Muthyala Kartheek,Navneet Goyal
Mojtaba Najafi, Dr. Kangavari
Philippe ESLING
Zoe Falomir Llansola
Xiaoxu He
Cheng Po Man and Wai-Kuen Cham
Doina Precup and Jordan Frank
Laurens van der Maaten
GUSTAVO CACERES CASTELLANOS
Deepa Krishnaswamy
Wang Jialin

You are in good company, just some of the people that requested the UCR Archive prior to summer 2015. Page 2

Tao Quang Bang
Jeremie Mary
Kai-Wei Chang and Dan Roth
Joan Serrà and Josep Lluis Arcos
Risa Myers and Risa Myers
Ankit Gupta
Weng-Keen Wong and Yonglei Zheng
Cássio M. M. Pereira and Rodrigo Fernandes de Mello
Subutai Ahmad
Visuru Lafi Mohamed Rizny
Eunice Carrasquinha, Ana Pires, Conceição Amado
Huaming Huang and Mehrotra
Abdulallah Balamesh
Feng Zhou and Fernando De la Torre
Johannes Schneider
José Manuel Benítez Sánchez (and Students)
Xin Zhao and Xue Li
Daniel Gordon, Danny Hendler, Lior Rokach
Anirvan Chakraborty and Probal Chaudhuri
Qing He and DongZhi
Ioannis Paparrizos and Luis Gravano
Marcos Quiles.
Chris Grieves and Rida E. Moustafa (Shalash)
Ben Marlin
maryam tayefi
Huiping Cao and Zhe Xie
Christian Hundt and Elmar Schömer
M. en C. Jorge Ochoa Somuano
Goce Ristanoski, James Bailey and Wei Liu
Paul Viola (Yes, as in Viola- Jones)
JYH-CHARN (STEVE) LIU and Hao Wang
TSUJIMOTO Takaaki and Prof. Uehara
Tomáš Kepic and Gabriela Kosková
Eva Ceulemans and Joke Heylen
Alexis BONDU
Che-Jui Hsu
Wang Chao
Wong, Weng-Keen and Nick Sullivan
Pavel Senin
Qian Xu
Fengzhen Tang and Peter Tino
Qiang Wang
Hesam Izakian
Witold Pedrycz
Yun Chen and Yang, Hui
Tanatorn Tanantong and Ekawit Nantajee[illegible]arawat
Cauchy Lee
Chedy Raissi
BARRE Anthony and RIU Delphine
Yang ZHANG
Flavio Vinicius Diniz de Figueiredo
Jussara Marques de Almeida
Dr. J. Figueroa-Nazuno
Victor Romero-Cano and Juan Nieto
Jianhua Zhao
Leslie S. Smith, Abdulrahman A.
Samuel J. Blasiak and Huzefa Rangwala
Harm de Vries and Azzopardi, G. (George)
Folly Adjogou and Alejandro Murua
Dustin Harvey and Todd, Michael D.
Kilian Weinberger
Daniel Kohlsdorf and Thad Starner
Zengwen Mo
Faezeh Eshragh and Xin Wang
Tuan Dang and Leland Wilkinson
Kang Li and Yun Raymond Fu
Andrés Eduardo Castro Ospina
Zheng Zhang
Peter Fox and Yanning(Yu) Chen
Jeremiah Deng and Feng Zhou
Franz Pernkopf and Nikolaus Mutsam
Alexey Chernyshev
Benjamin Auder
Jiawei Yuan and Shucheng Yu
Heidelinde Roeder and Jan Freund
Toon Calders and Thanh Lam Hoang
Feixiang Gao
Nancy Pérez-Castro
Tracy Hu H
André Gensler
zairul hadi
Beau Piccart
Tomas Pfister
John Costanzo and Nathan Cahill
Conceição Amado and Diogo Silva
Zhang, Zhifei and Qi, Hairong
Diego Rivera García.

Tanzy Love and Kyra Singh
Manuele Bicego and Pietro Lovato
Vladimer Kobayashi
Talayeh Razzaghi and Petros Xanthopoulos
Mohit Sharma
Juergen Schmidhuber and Klaus Greff
Heriberto Avelino
Manoj Apte
Nehal Magdy Saber
Webber Chen and Dan Stashuk
Huaihou Chen and Philip Reiss
Seung-Kyu Lee
Marcelo de Azevedo Ávila
Subhajit Dutta and Anil K. Ghosh
MAILLARD, Bruno
Yazhou Ren and Carlotta Domeniconi
Eithon Cadag and Johan Grahnen
Dinkar Sitaram and Varun Shenoy
Chan Syin and Vuong Nhu Khue
Rodrigo Araujo
Mohamed Kamel
Borgwardt Karsten M. and Llinares Lopez Felipe
Jiang Liyang
Timothy Ravasi and Gregorio Alanislobato
James Zhang
Heloisa de Arruda Camargo and Fábio José Justo dos Sant[illegible]
D.Pradeep Kumar
Steffen Dienst and Stefan Kühne
Christian Hahn and gudrun stockmanns
haoran xu
Chuanlei Zhang
Youngha Hwang and Saul Gelfand
Tetsuya Nakamura
Daniel Alejandro Garcia Lopez
Richard J. Povinelli
Yi-Dong Shen [illegible] Shen
Emmanuel [illegible]nnet
Remi [illegible]din [illegible]olas nicoloyannis
Phi[illegible] Gro[illegible]
Franc[illegible]s Po[illegible]et
Qi He , Dr. Kuiyu Chang, Dr. Ee-Peng Li[illegible]
Fabio Antonio Pereira Reis
Damien Tessier
Cuvelier Etienne
Moataz El Ayadi & M[illegible]ed S. [illegible]amel
Hillol Kargupta
Andrey Ustyuzhanin
Hussa[illegible]shraideh
Jo[illegible] M. [illegible]enkle
[illegible]ris Martin Jimenez & Francisco Herrera Fernandez
Juan Jos[illegible] Rodriguez
Zolt[illegible] [illegible]anko
[illegible]jan Goswami & Debashis Mondal
Simon Kagwi Mwangi
Simone Fontolan and A[illegible]andro Garg[illegible]etti
Lin Zhang and Joe [illegible]
Zhenzhi Huang and Zh[illegible]feng [illegible]e
Wale[illegible] K[illegible]us
Hao Hu [illegible]nd Qi[illegible]g [illegible]ang
[illegible]y Rob[illegible]s and Dragana Veljkovic
Da[illegible]el Pen[illegible] Regina Kaiser, Ana Lau[illegible] Bad[illegible]iani
Dani[illegible] [illegible]aves and Witold Pedrycz
Anne Denton
Julia Hunter and Martin Colley
Lucia Sacchi
Bernhard Seege[illegible] Michael [illegible]rau.
John Maindonal[illegible]
Balazs Torma
Nikolaos Chatzis
Daniel Smith
Abdul Razak, Khairuddin Omar
Elwin (Yong) lee
Alex Smola and Xinhua Zhang
Rudolf Kruse and Christian [illegible]oew[illegible]s
Pekka Siirtola
Michael Berthold
David Bong and James Tan
Zhengzheng (Crystal) Xing and Jian Pei
Leticia Arco Garcia and Rafael Bello
Nuno Constantino Castro and Paulo Azeved[illegible]
Ng Kam Swee
Antonio Irpino
Jong Myoung Ko
Jonas Richiardi
Hassani, Marwan and Sebastian Schaub
Chen, Po-Yu,McCann, Julie, Yu, Weiren
Laetitia Chapel
Michelle Zhang and Sirajul Salekin

Dhaval Patel, Wynne Hsu and Lee Mong Lee.
Ville Hautamaki
Peter Sunehag
Richard Clements
Hichem Frigui and Walid MISSAOUI
KASHIMA Toru
Tang Ke and Guojie Song
Jinfu Fan
Ruchira Guha
Fan Zhou and Wu Yue
Chen Duansheng
Cheng wencong
Xiaoli Li
Fedor Zhdanov and Vladimir Vovk
WU Quan-Yuan
Anuradha Kodali
Keith Noah Snavely
Hilario Navarro Veguillas [illegible]d Jesu[illegible] Bou[illegible]o
Myong K. Jeong
Luca Ga[illegible]aldi
Dileep George
Roni [illegible]ard[illegible] and Yuyang WANG
Alexa[illegible]dros N[illegible]opoulos
Be[illegible]ei [illegible]ang and Rong Chen
Sile [illegible]'Modhrain
Am[illegible] McGovern
Ya-Ju Fan and W. Art Chaovalitwongse
Xiaobin Li
Smruti Sarangi
Peter Grabus[illegible]
Fernando Cela[illegible]az
Thomas [illegible]hre N[illegible]lsen and Shengtong Zhong
Zhenzhou [illegible]hen
[illegible]esley Kerr [illegible] Paul Cohen
[illegible]rent baumes
Lu[illegible] Antunes
Chih-Chun and Zeeshan Syed
Rene Vidal and Merve Kaya
Zhaozhong Wang
Duangmalai Klongd[illegible]e and [illegible]hu[illegible] Jar
Rakesh Babu
Bao Yubin
Rayner Alfred
Ji[illegible] Howard
Xiao[illegible]n Zeng and Geng Li
[illegible]tonio Neme
N[illegible]APALLI SAI SUDHEER and V.B[illegible]avani
Julien JACQUES
Abdulla Al Maruf and Ky[illegible]awa[illegible]e
Neuza Filipa Ma[illegible]tins N[illegible]es a[illegible] Hu[illegible]o Gamboa
Liang Ge
Geovanny Gi[illegible]gana [illegible]nd [illegible]loeger
Ugo Vespier [illegible]d Arno [illegible]oopman
Boa[illegible] Nad[illegible]r
jan[illegible]m luo
Mike [illegible]ess[illegible]er and Sumeet Dua
Chang[illegible]n Gao
YIN Hong sheng
Baiming Ma
Ramon Huerta
Petr Volny
Francois Rheaume
Alexander Kolesnikov
Vishwajeet Singh Thakur
Tomasz Gorecki
Kerem Muezzinoglu
Ritwik Kumar
YANG Yuhang
Ali Shokoufandeh , Terence Tuhinanshu, Ernst Pisch
Maria Titah Jatipaningrum
Anita Santanna, Nicholas Wickstrom
seyma ket[illegible]
Kristian [illegible]ndbe[illegible]
Godtliebsen Fre[illegible]
Mary She and [illegible]ng [illegible]ang
[illegible]ust[illegible] Bayer a[illegible] Patrick van der Smagt
A[illegible]mal Al-Anazi
[illegible]evin [illegible]hen
Stephan Chalup and Arash Jalalian
[illegible]eandro MinkuMichele Dallachiesa and Themis
[illegible]alpanas
Andreas Brandmaier and von Oertzen, Timo
Garcia-Trevino, Edgar and Javier A. Barria
Hao Zhu and He Zhongsh
Nipotephat Muangkote
Aleksandar Pechkov
Myat Su Yin

Abhishek Sharma
shouyi wang and W. Art Chaovalitwongse
Hyrum Anderson
Qian Chen
Tomas Bartos and To[illegible]mas Skopal
Vitali Loseu
Alessandro Anto[illegible]cci
Andrii Cherni[illegible] and [illegible]dimi[illegible] Zadorozhny
Julia Ga[illegible]er an[illegible] Stepha[illegible] Spiegel
Wa[illegible]l Ma[illegible]i
Pi[illegible]r [illegible]bolewski and Michal Wozniak
I[illegible]s A[illegible]e, Michael Berthold and Sebastian Peter
Omid Geramifard and Xu Jianxin
Ng Wee Keong and NGUYEN HAI LONG
Li Tao
Willian Zal[illegible]wsk an[illegible] Fab[illegible]no Silva
Neelanjana [illegible]tta, [illegible]ri[illegible]n Chellappa, Donald Wunsch II
Dominique G[illegible]
Mai Th[illegible]i S[illegible] an[illegible] Christian Böhm
Ioan[illegible]s Kalo[illegible]amp[illegible]s and Dave Marshall
[illegible]YAT [illegible]LYI[illegible]
Aip[illegible]g Zh[illegible]u
Ho[illegible]tao De[illegible]g and George Ru[illegible]ger
[illegible]anyu Li and Dr. Dong
May D Wang and Ja[illegible]es Cheng
Thomas Rueck[illegible]tiess and Patrick v[illegible] der Smagt
Rebecca Willett
krystal [illegible]ylor
Pengfei Z[illegible]
Bohdan Ko[illegible]rzews[illegible]
[illegible]ngang Li
S[illegible] Xiangzhi
Yuta[illegible] Fan
D[illegible]mitri Mavris, Megan A Halsey, [illegible]rtis [illegible]ata
Wang Wei
MªIsabel Borrajo García
Dennis Shasha and [illegible]te[illegible] Soul[illegible]
Andrew Cohe[illegible]
Weng-Keen Wo[illegible]g an[illegible] Xinze Guan
[illegible]agesh, K[illegible]rthik
[illegible]homas Ste[illegible]ke [illegible]nd Patrick Schäfer
Antonio [illegible]enta
Katheri[illegible]e Anderson and Castle, Joseph P
Bressan, Stephane
Michael Böhlen and Mourad Khayati
Yi Zheng
Lijuan Zhong
Eduardo Gerlein and Martin McGinnity
Dr.Ing. MAURICIO OROZCO-ALZATE
Kevin Shi and Layne Lori
Sharon Goldwater and HAIDER Adnan
Youqiang Sun
Koki Nakatani and Kumiko Tanaka-Ishii
Uttam Kumar Sarkar
Aalaa Mojahed and Beatriz De La Iglesia
Prof. L. Zhang
Hrishav Agarwal
Jimin Wang
Piyush Kumar
Qing Xie and Xiangliang Zhang
Abhay Harpale, Tianbao Yang and Daniel Marthaler
Andrew Finch
Mona Rahimi
Fred Nicolls
Paulo Borges and George Darmiton da Cunha Cavalcant
John Lach and Davis Blalock
Heggere S. Ranganath and Vineetha Bettaiah
Keith Henderson
Steve Lalley and Herbert Xiang Zhu
Behzad Mansouri
Megha Agrawal
Zhihua Chen and Wangmeng Zuo
Diep Nguyen Ngoc
BATCHELOR Andrew
Jinwook Kim and Min-Soo Kim.
Warat Safin
pradeep polisetty
Houshang Darabi and anooshiravan sharabiani
Chen Qiang
Liying ZHENG
Sarat Kr. Chettri
Graham Taylor and Ethan Buchman
Cai Qinglin
Anuj Srivastava and J. Derek Tucker
Ajithkumar Warrier
Kyle Johnston, and Adrian Peter
Serim Park and Gustavo Kunde Rohde
Lee Seversky and Jack M Fischer

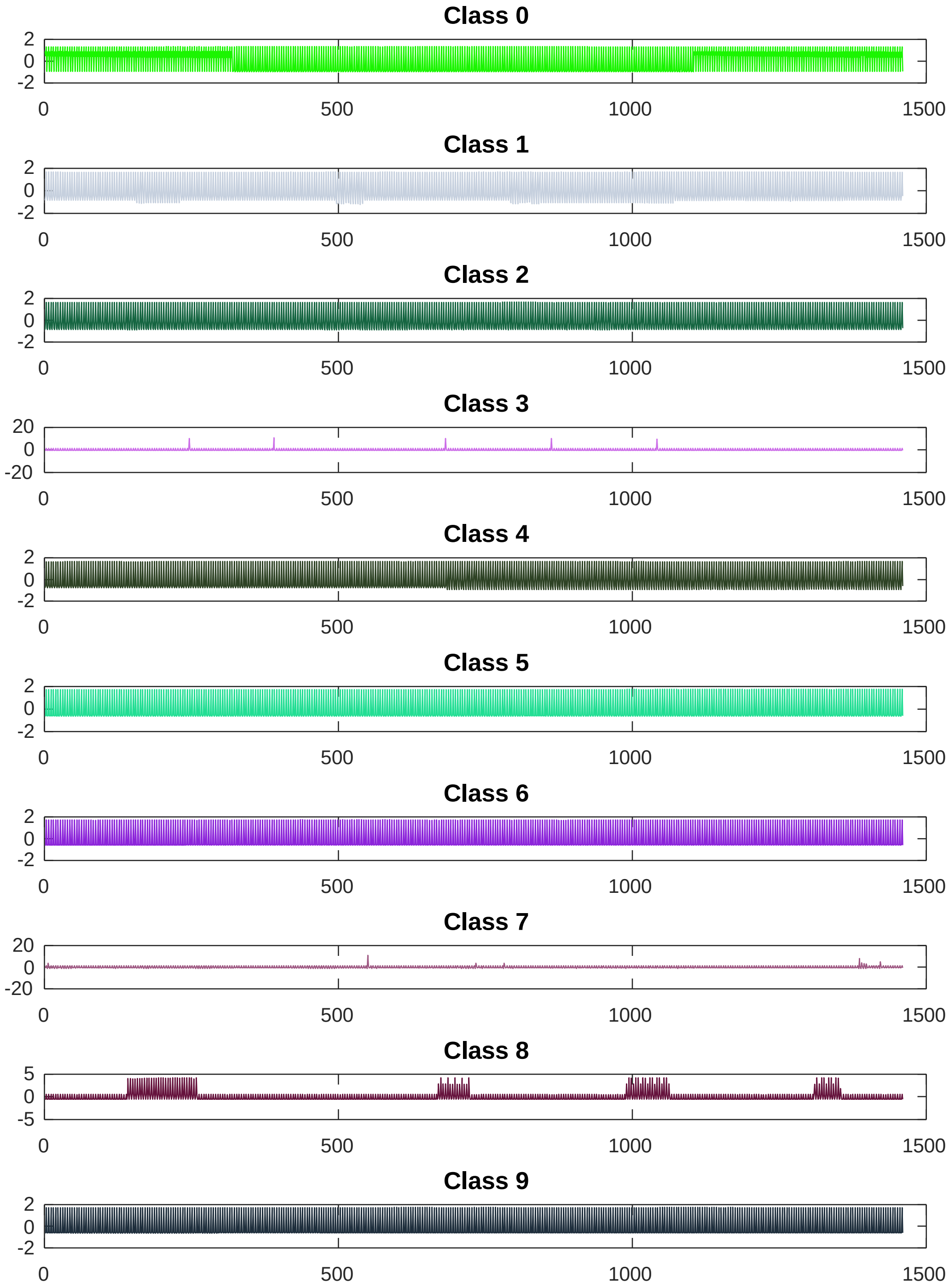
Ahmad, Faraz and Smith, David
Velislav Batchvarov
Gautier Marti
Dimitrios Kosmopoulos
Rana Haber and Adrian Peter
Shachar Afek Kaufman and Ruth Heller
Chengzhang Qu and Jianhui Zhao
Basabi Chakraborty
Grzegorz Dudek
Sindhu Ghanta and Jennifer Dy
Yu Fang and Huy Xuan Do
Ali Raza Syed and Andrew Rosenberg
Agada Joseph
ricardo garcia rodenas
Om Prasad Patri
Leonardo Nascimento Ferreira
Michal Prílepok
Ke Yi and LUO,Ge
Bruno Ferraz do Amaral and Elaine Sousa
Edward GONG and Si Yain-Whar
Chaoyi Pang
Vida Groznik and Aleksander Sadikov
Abdulhakim Qahtan and Xiangliang Zhang
Cao Duy Truong
Oliver Obst
Yung-Kyun Noh
Wei Song
Yu Su
Uwe Aickelin, Eugene Ch'ng, Xinyu Fu
Quang Nguyen
Or Zuk, Avishai Wagner, Tom Hope
Subhadeep Mukhopadhyay
WEN GUO
poyraz umut hatipoglu and Cem Iyigun
Yongjie Cai
Mehdi Faramarzi
Tarhio Jorma and Chhabra Tamanna
Xiaoqing Weng
Simon Shim
Yan Gao Gao and Daqing Hou
Maria Rifqi, Marcin Detyniecki and Xavier Renard
Frank Englert and Sebastian Kößler
Paul Grant
juan colonna
Elias Egho
QUANG LE and Chung Pham
Jitesh Khandelwal
Karima Dyussekeneva
Wu, Ruhao and Wang, Bo
Yuki Kashiwaba
Ade Bailly
Abhinav Venkat
Michael Hauser and Asok Ray
Tuan-Minh Nguyen and Michael Bromberger
Ruako TAKATORI
Rahul Singh and Ryan Eshleman
Laura Beggel
Tom Arodz
Victoria Sanchez
Mike Sips and Carl Witt
Segolene Dessertine-Panhard
Zhao Xu
Lovekesh Vig
Angelo Maggioni e Silva
Abdelwaheb Ferchichi and Mohamed Saleh Gouider
Wenlin Chen and Yixin Chen
Gui Zi-Wen and Yuh Jye
Romain Tavenard
Chandan Gautam
Jinglin Peng
Ahlame Douzal and Jidong Yuan
Chen Jing
Zeda Li and Alan J. Izenman
Peng Zhang
Hanyuan Zhang and Xuemin Tian
Georgios Giantamidis and Stavros Tripaki
Goutam Chakraborty and Takuya Kamiyama
Pipiras, Vlada and Stone, Eric
Ping Li
Herbert Kruitbosch
Adam Oliner, Jacob Leverich, Tian Chen, Tom LaGatta
Yulin Sergey
Huan Liu
Vadim Vagin
Nikolaos Passalis
Graham Mueller

43 data sets added in Fall 2018

The figures follow are intended to offer a quick inspection of the data. For readability, depending on the scenario, the data may be normalized or may be not, the number of exemplars per class may be one, three or many.

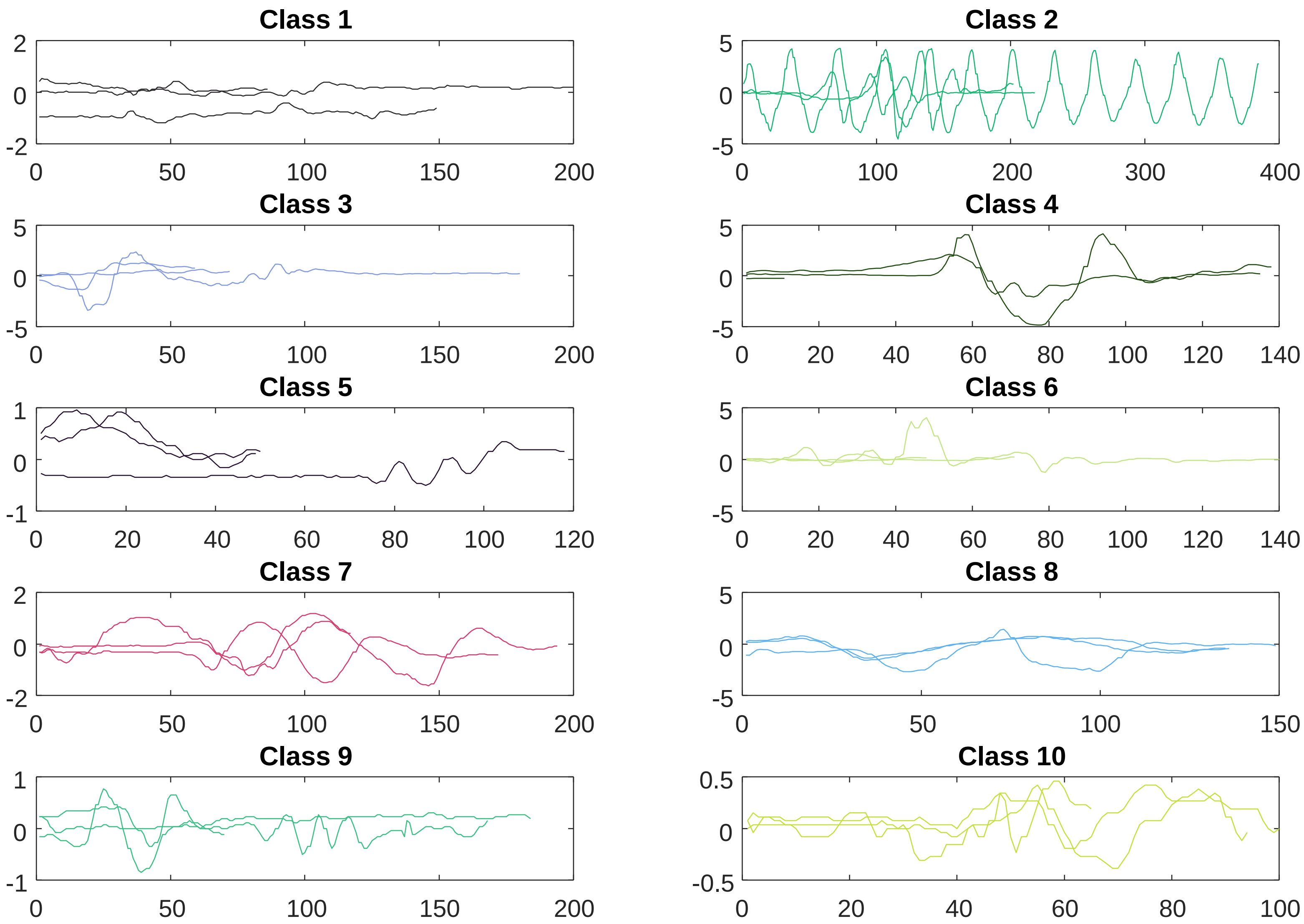
ACSF1

One exemplar per class, with z-normalization



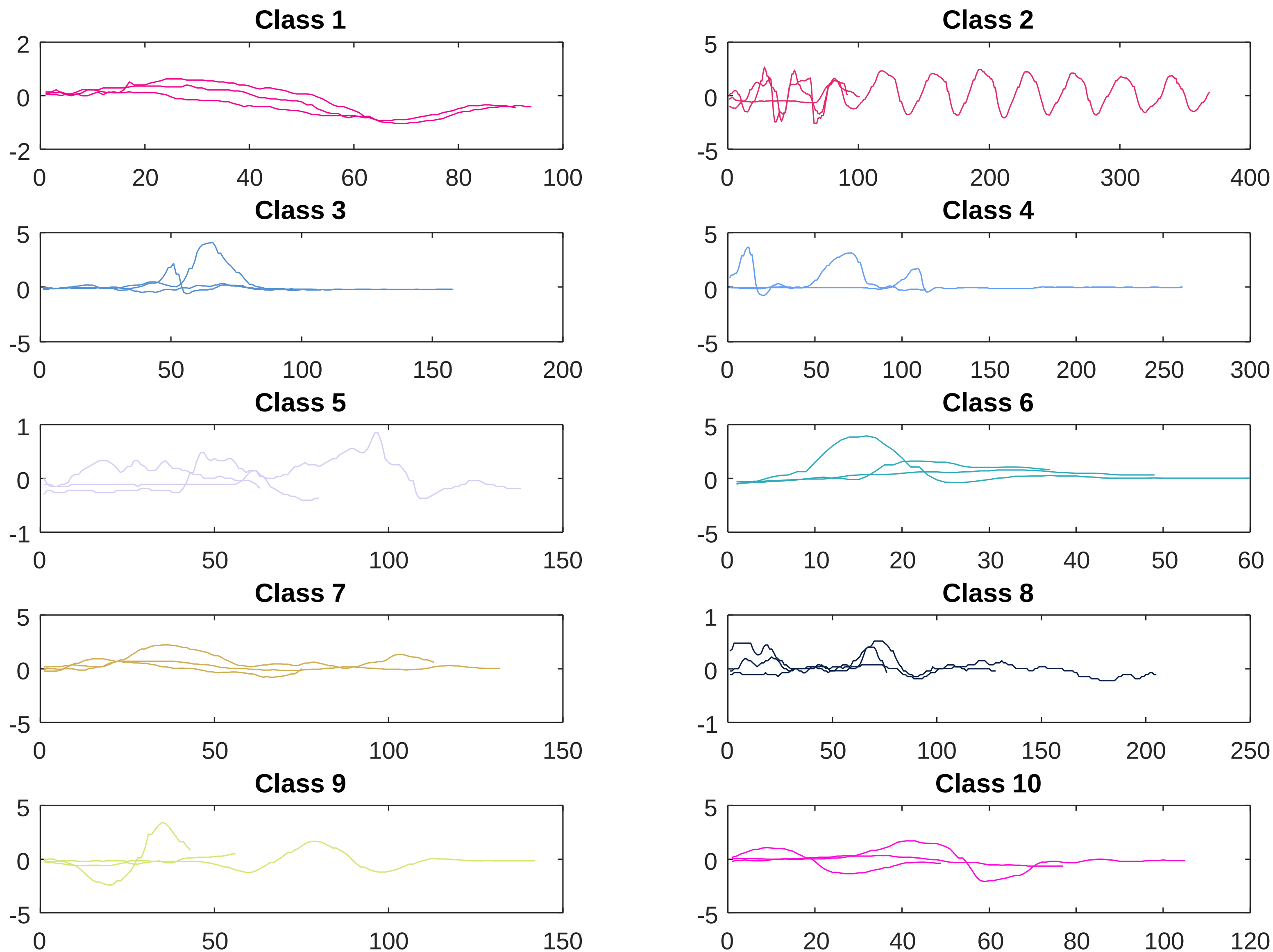
AllGestureWiimoteX

Three exemplars per class,
without z-normalization



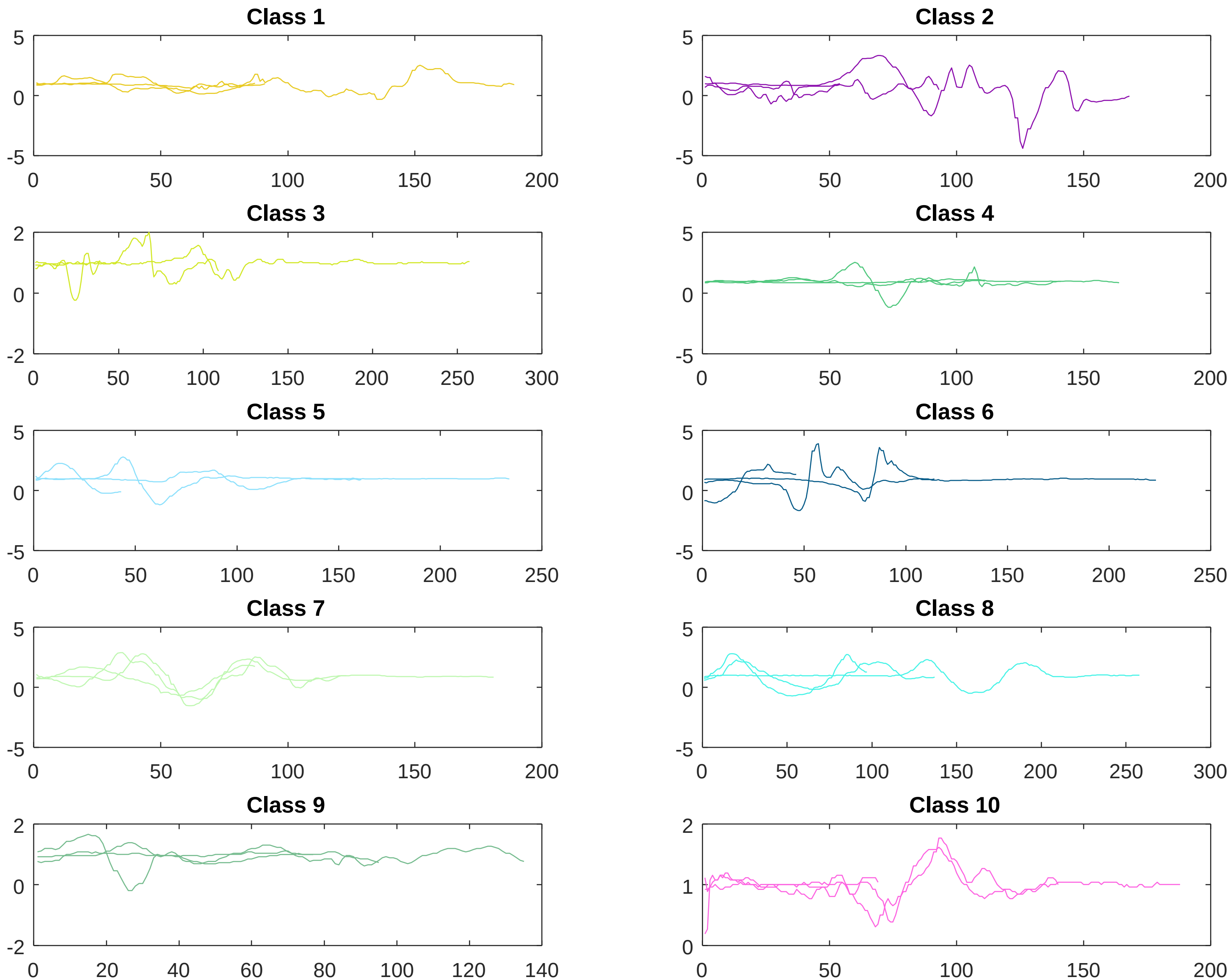
AllGestureWiimoteY

Three exemplars per class,
without z-normalization



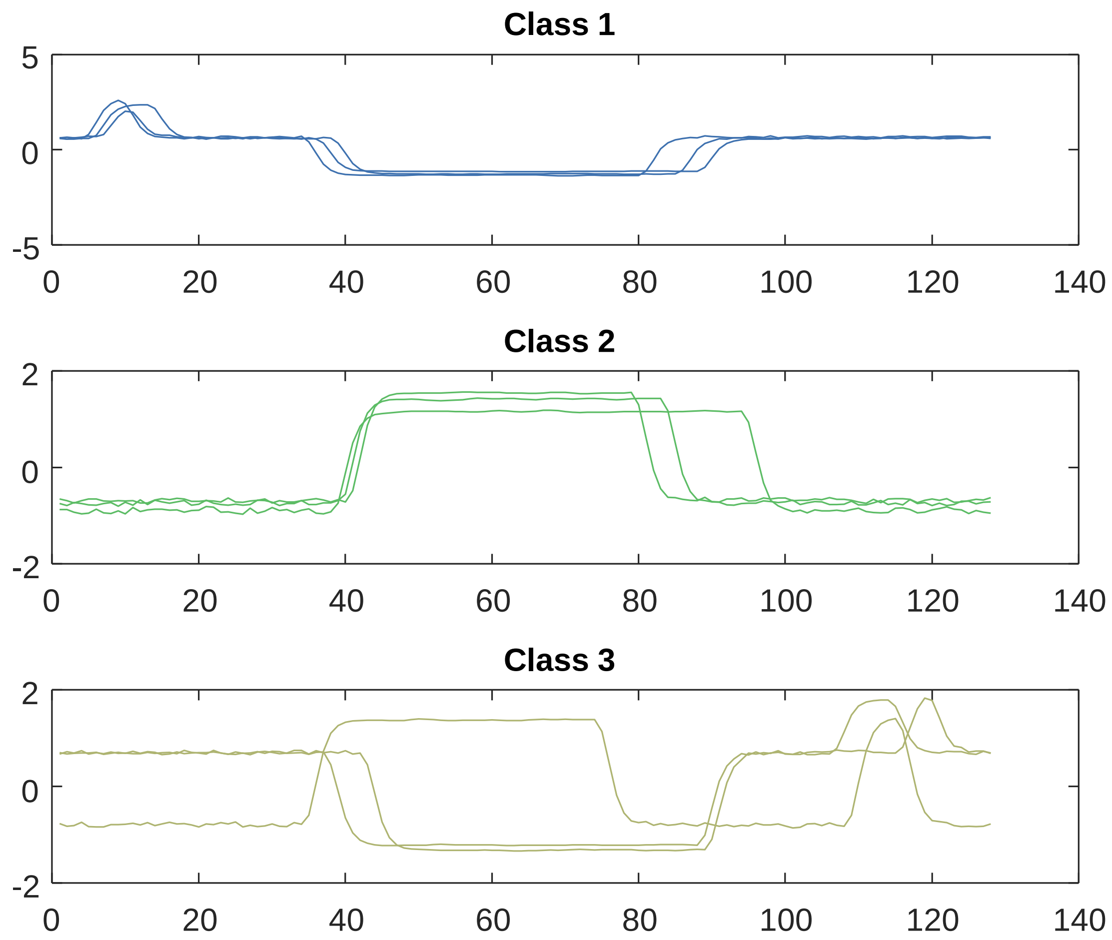
AllGestureWiimoteZ

Three exemplars per class,
without z-normalization



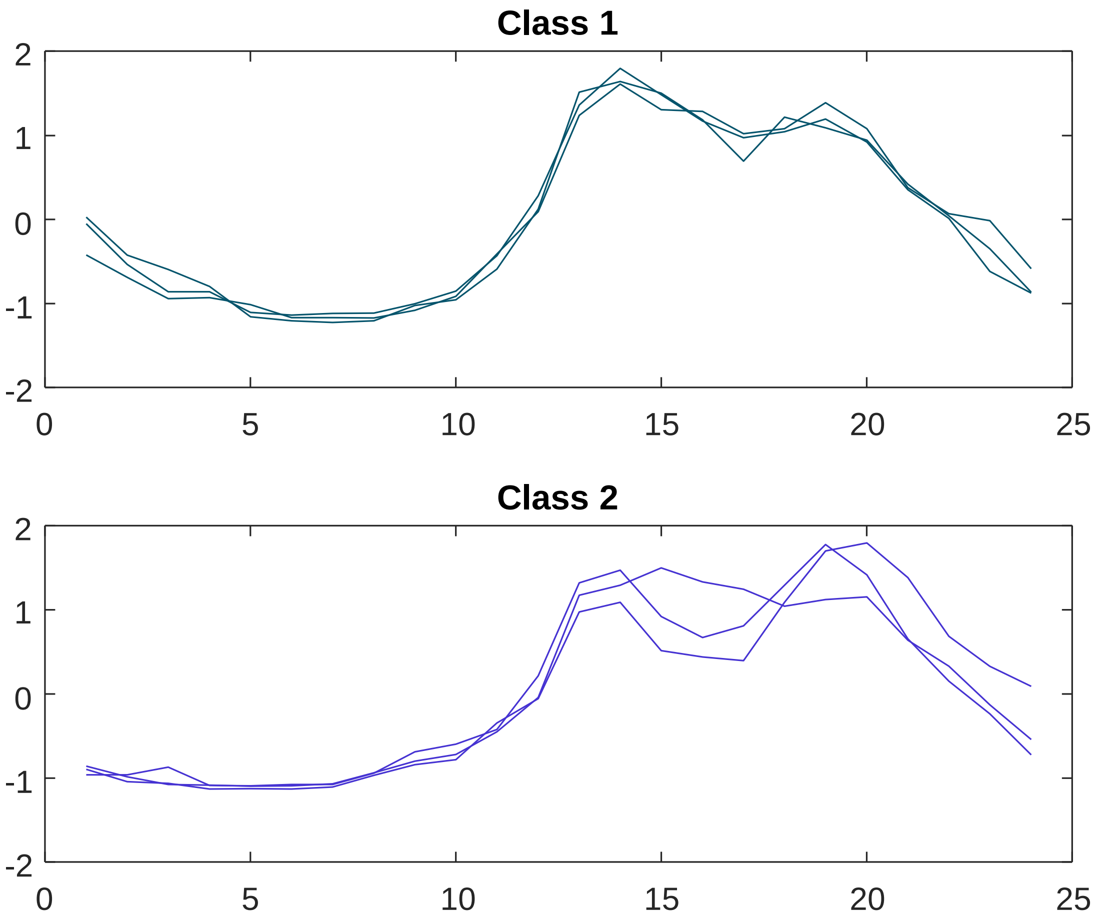
BME

Three exemplars per class,
with z-normalization



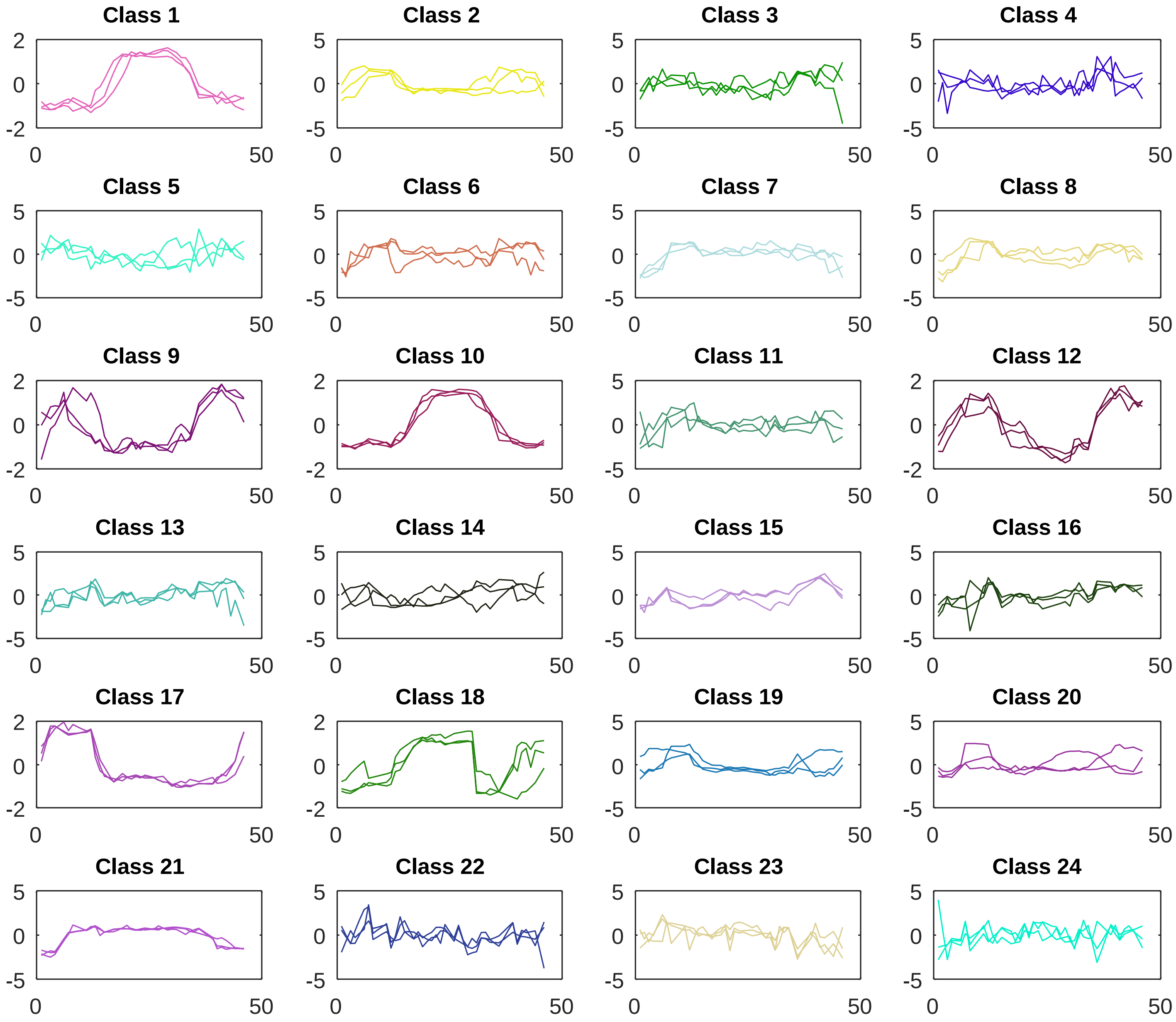
Chinatown

Three exemplars per class,
with z-normalization



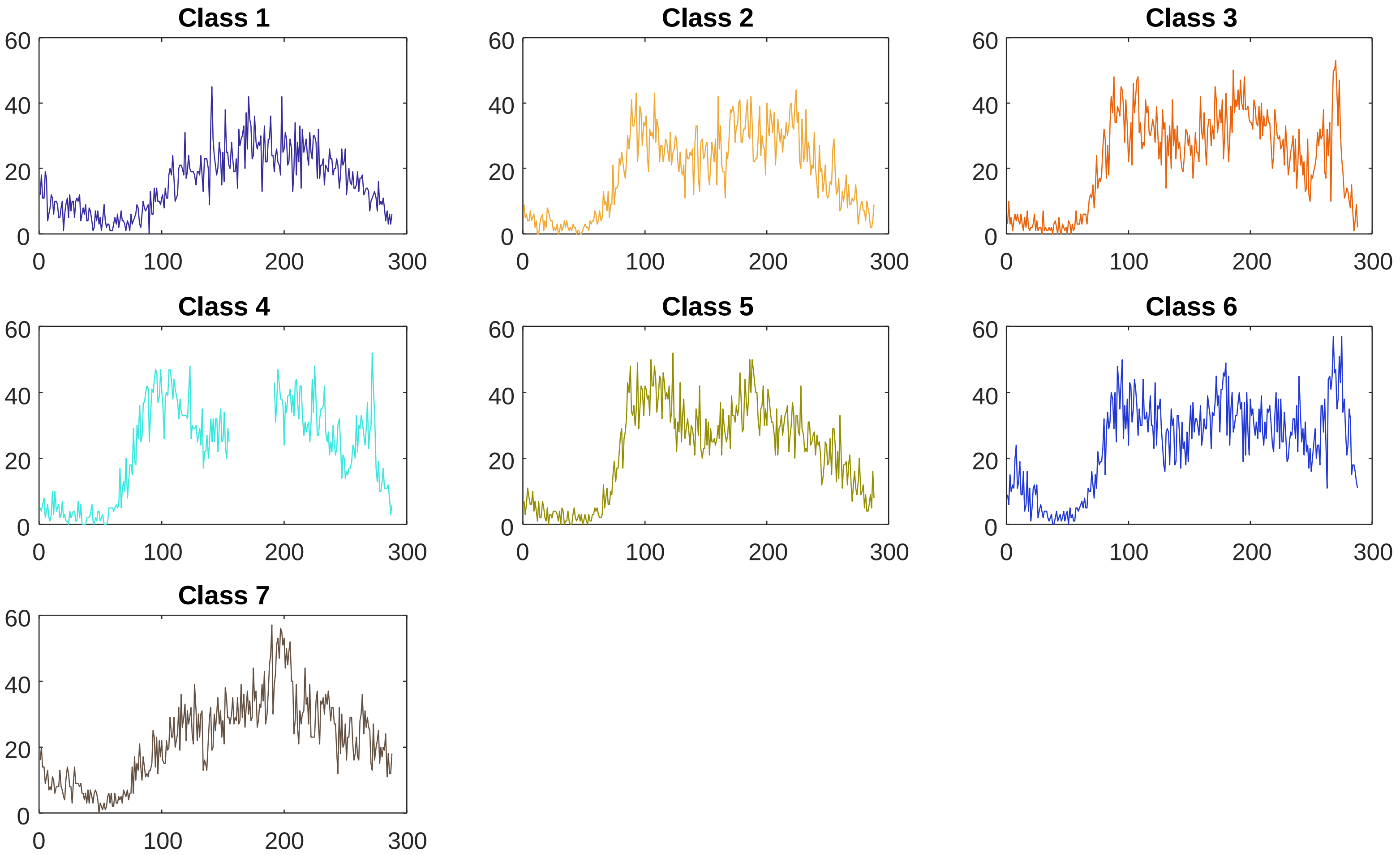
Crop

Three exemplars per class, with z-normalization



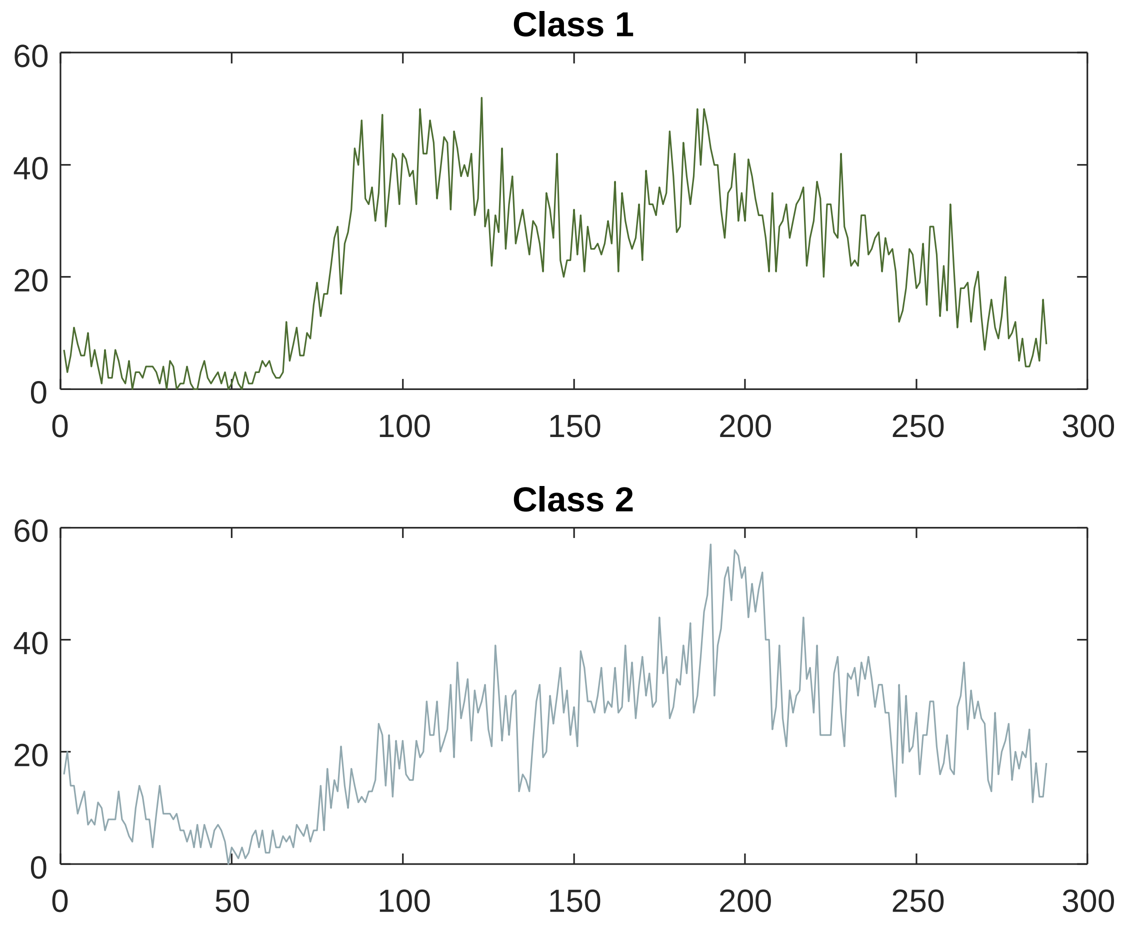
DodgerLoopDay

One exemplar per class,
with z-normalization



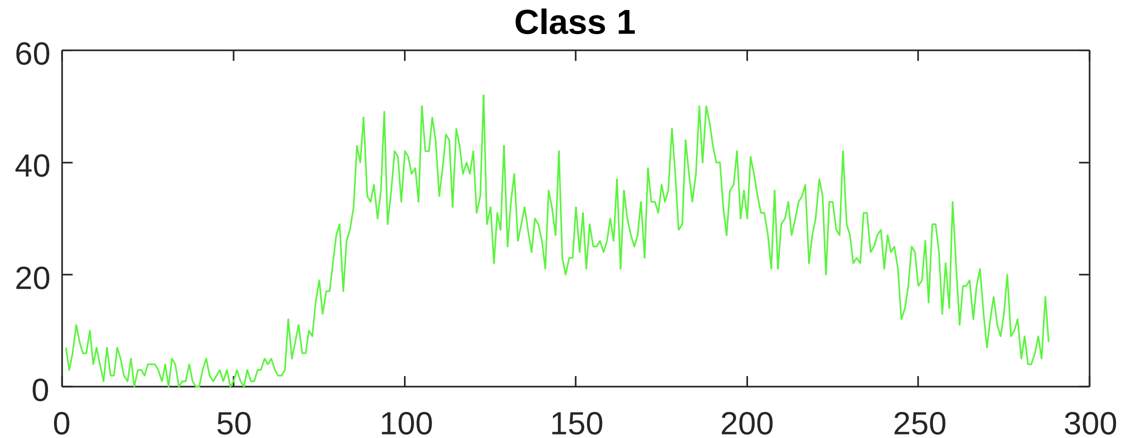
DodgerLoopGame

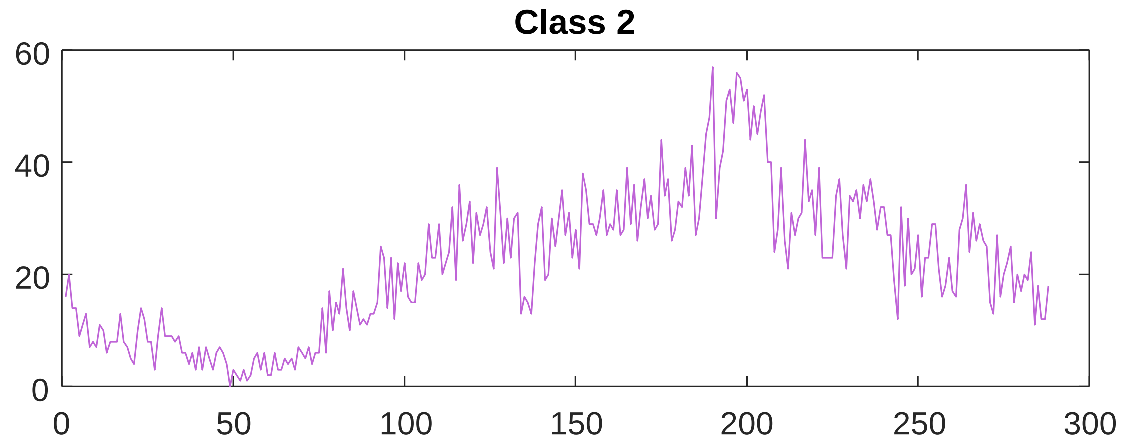
One exemplar per class,
without z-normalization



DodgerLoopWeekend

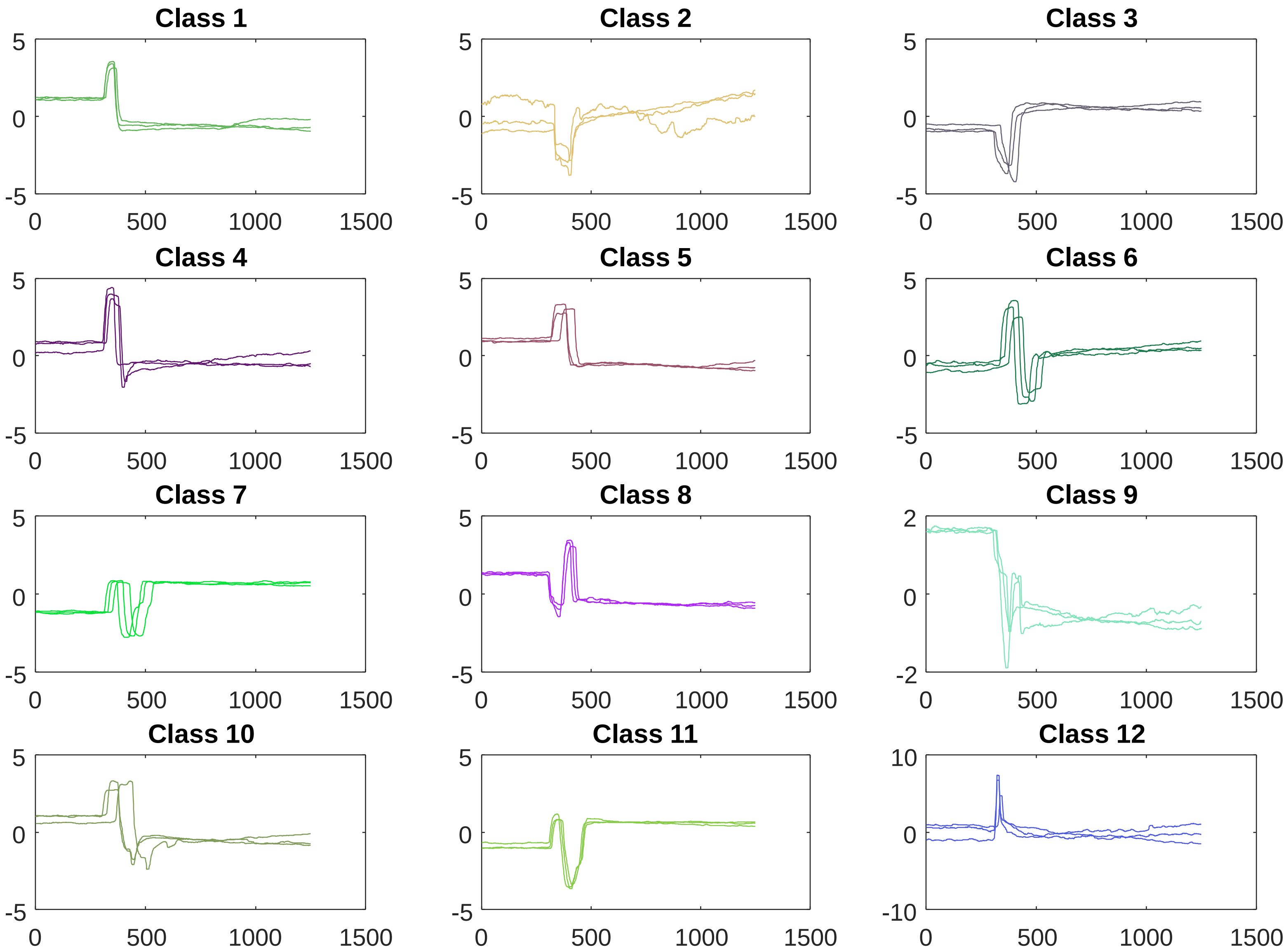
One exemplar per class,
without z-normalization





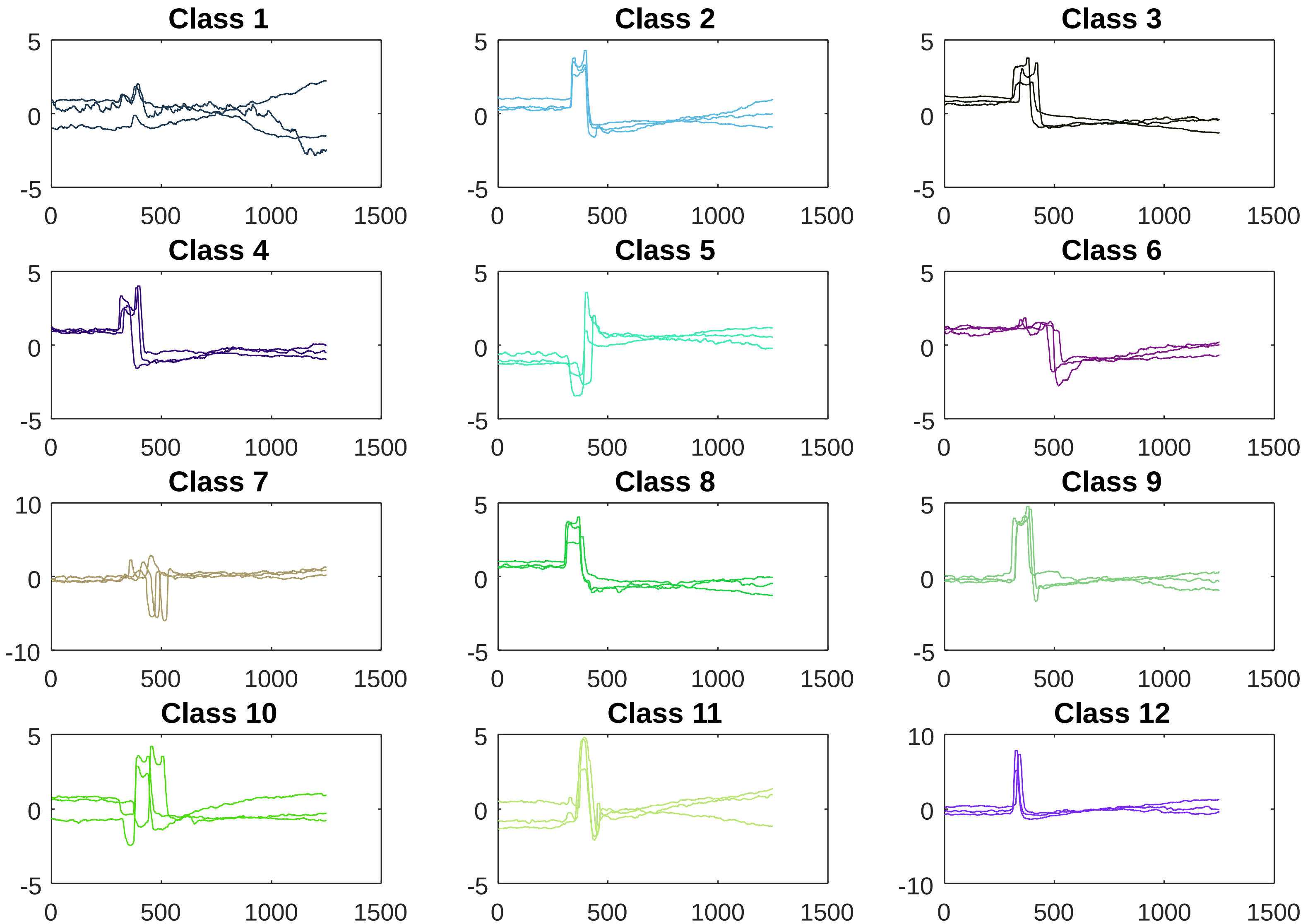
EOGHorizontalSignal

Three exemplars per class,
with z-normalization



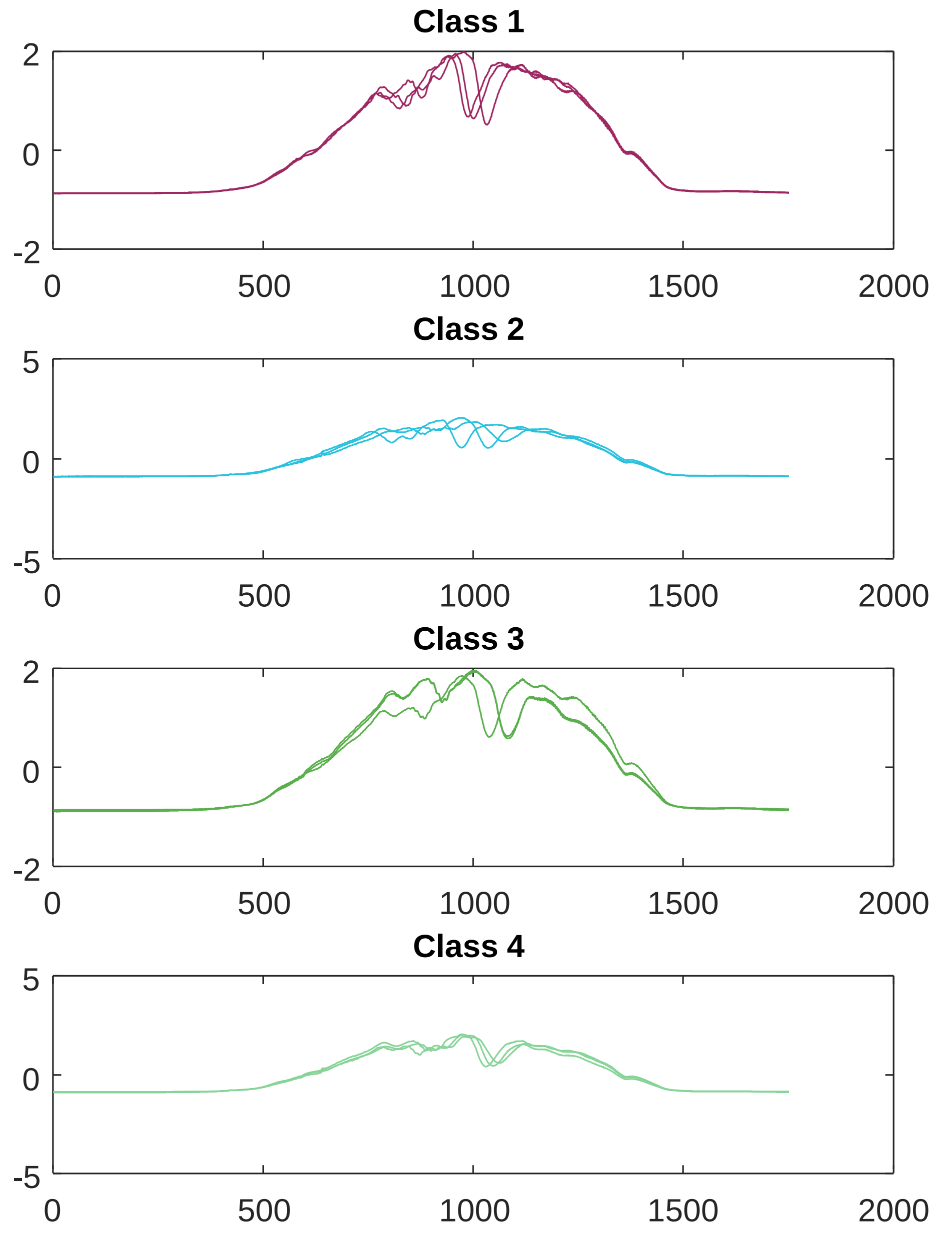
EOGVerticalSignal

Three exemplars per class,
with z-normalization



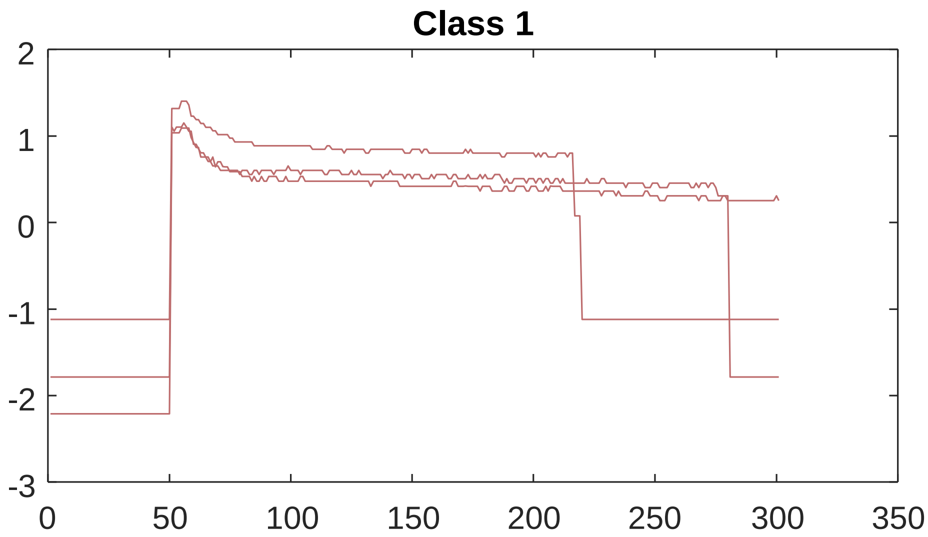
EthanolLevel

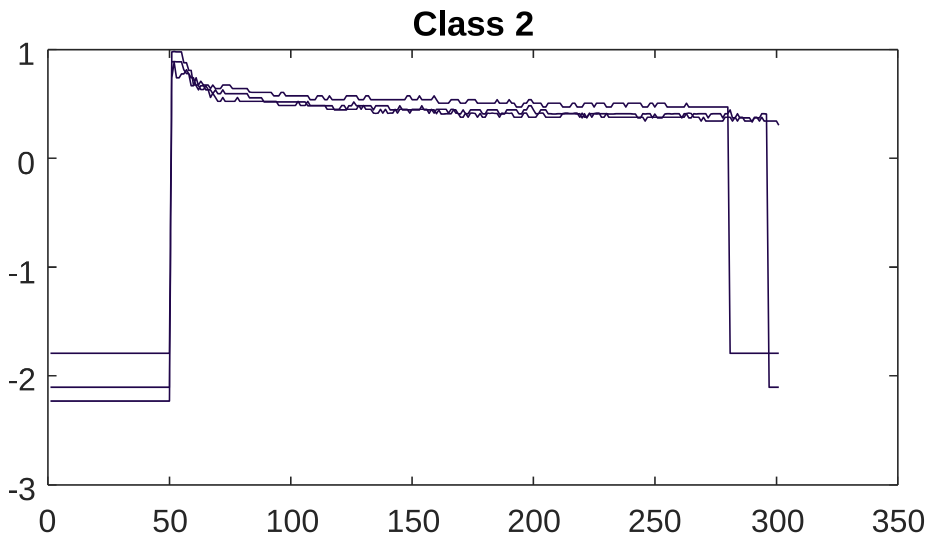
Three exemplars per class,
with z-normalization



FreezerRegularTrain

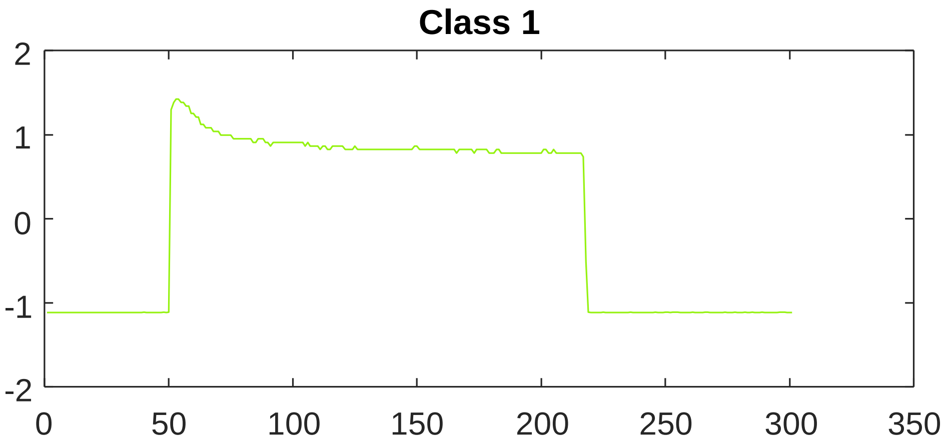
Three exemplars per class,
with z-normalization

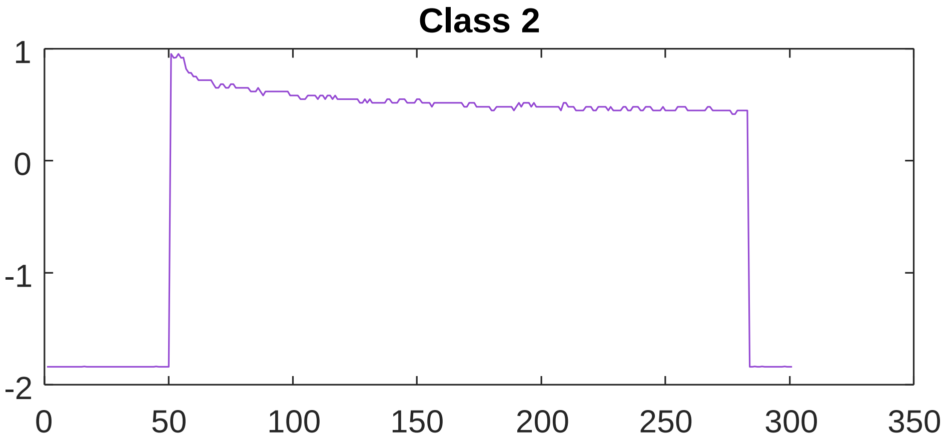




FreezerSmallTrain

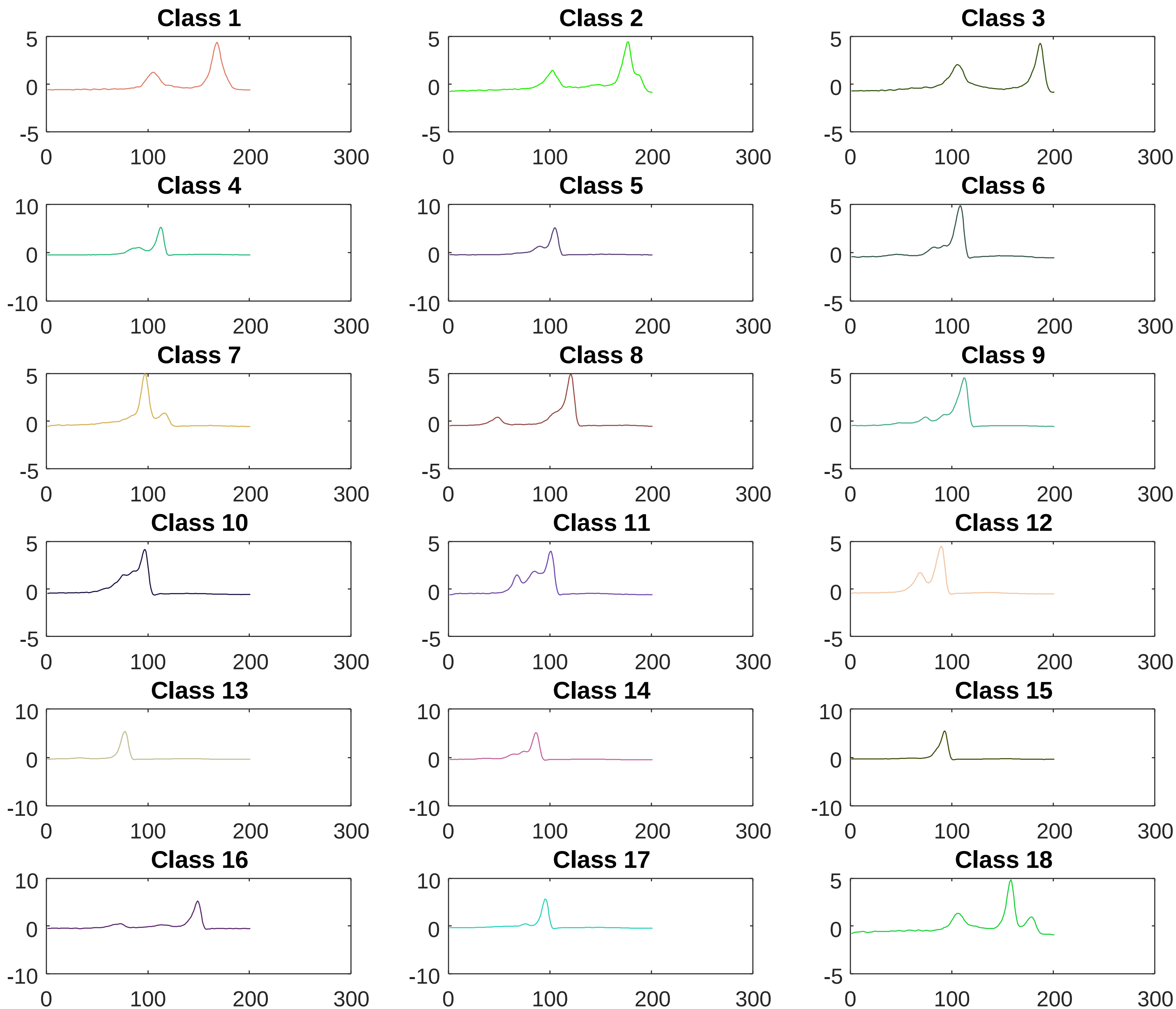
One exemplar per class,
with z-normalization





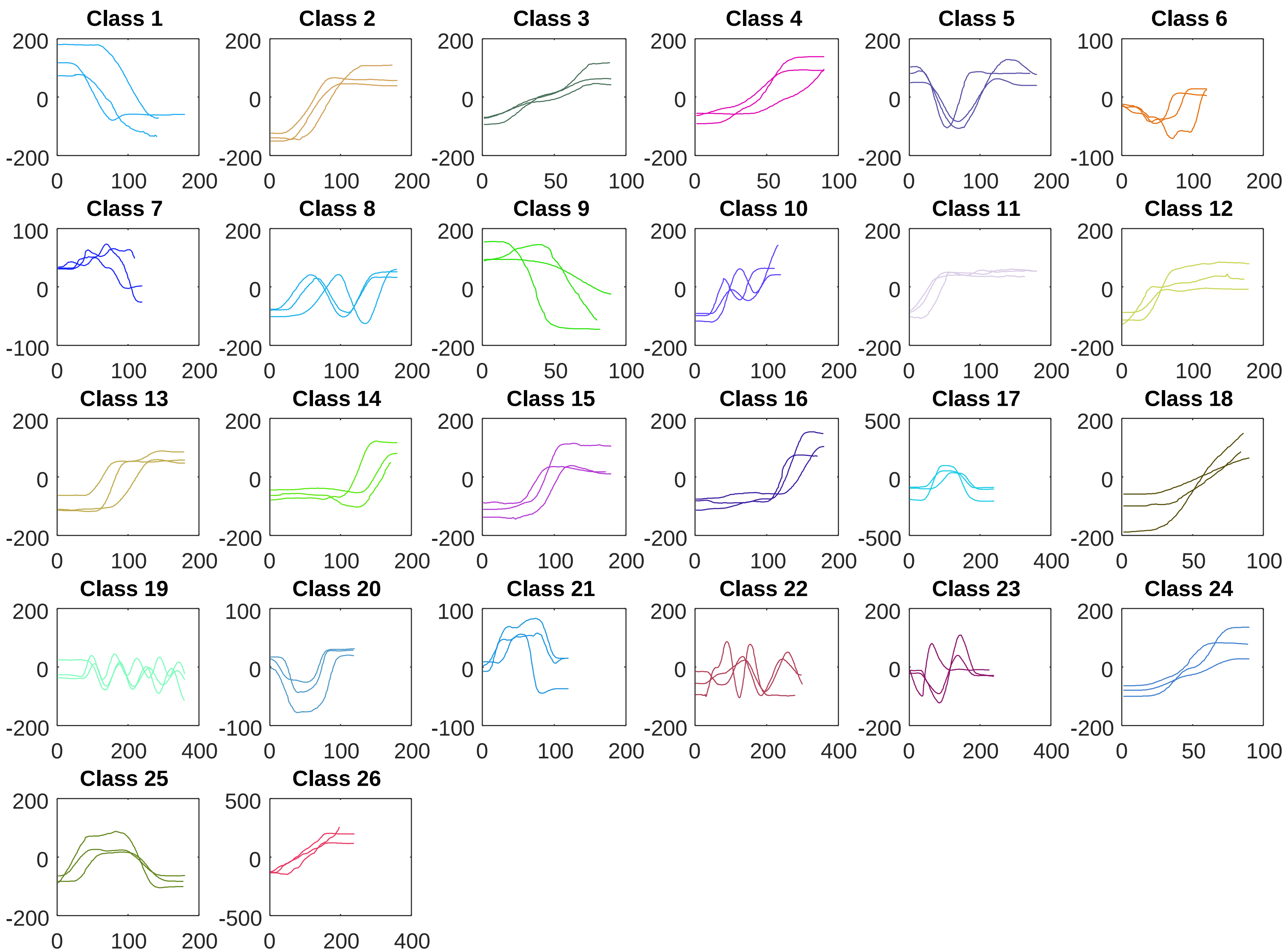
Fungi

One exemplar
per class, with
z-normalization



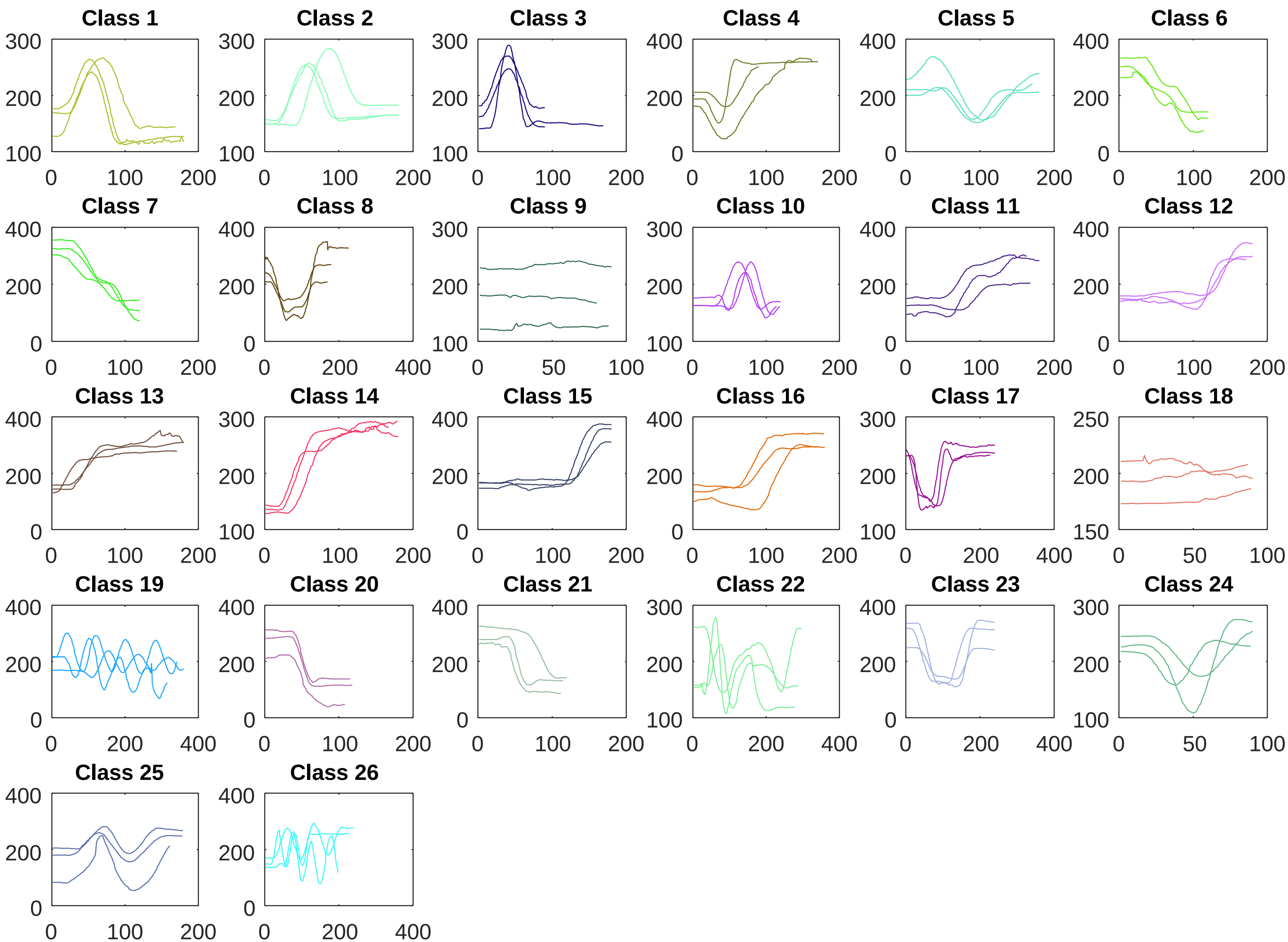
GestureMidAirD1

Three exemplars per class, without z-normalization



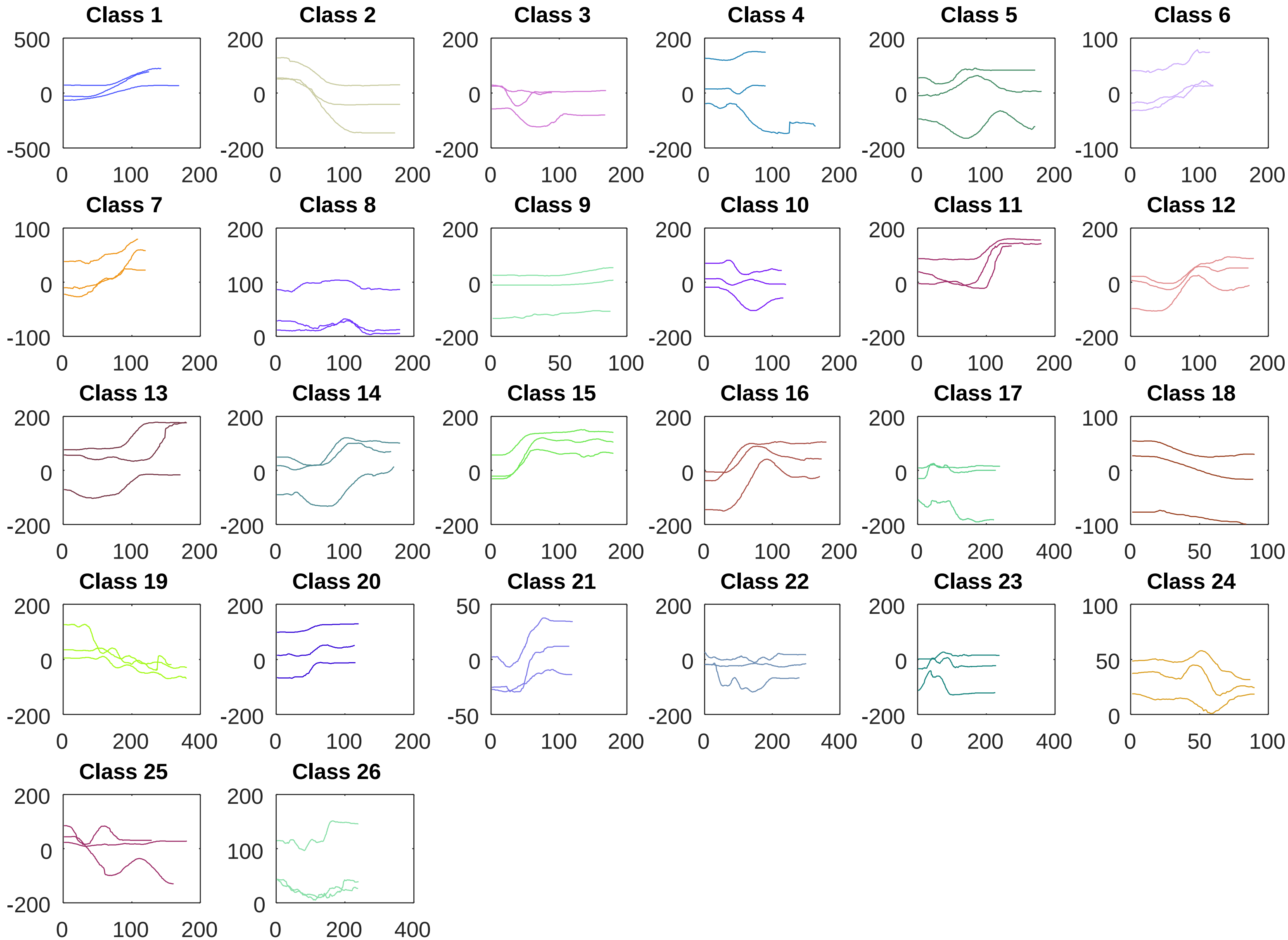
GestureMidAirD2

Three exemplars per class,
without z-normalization



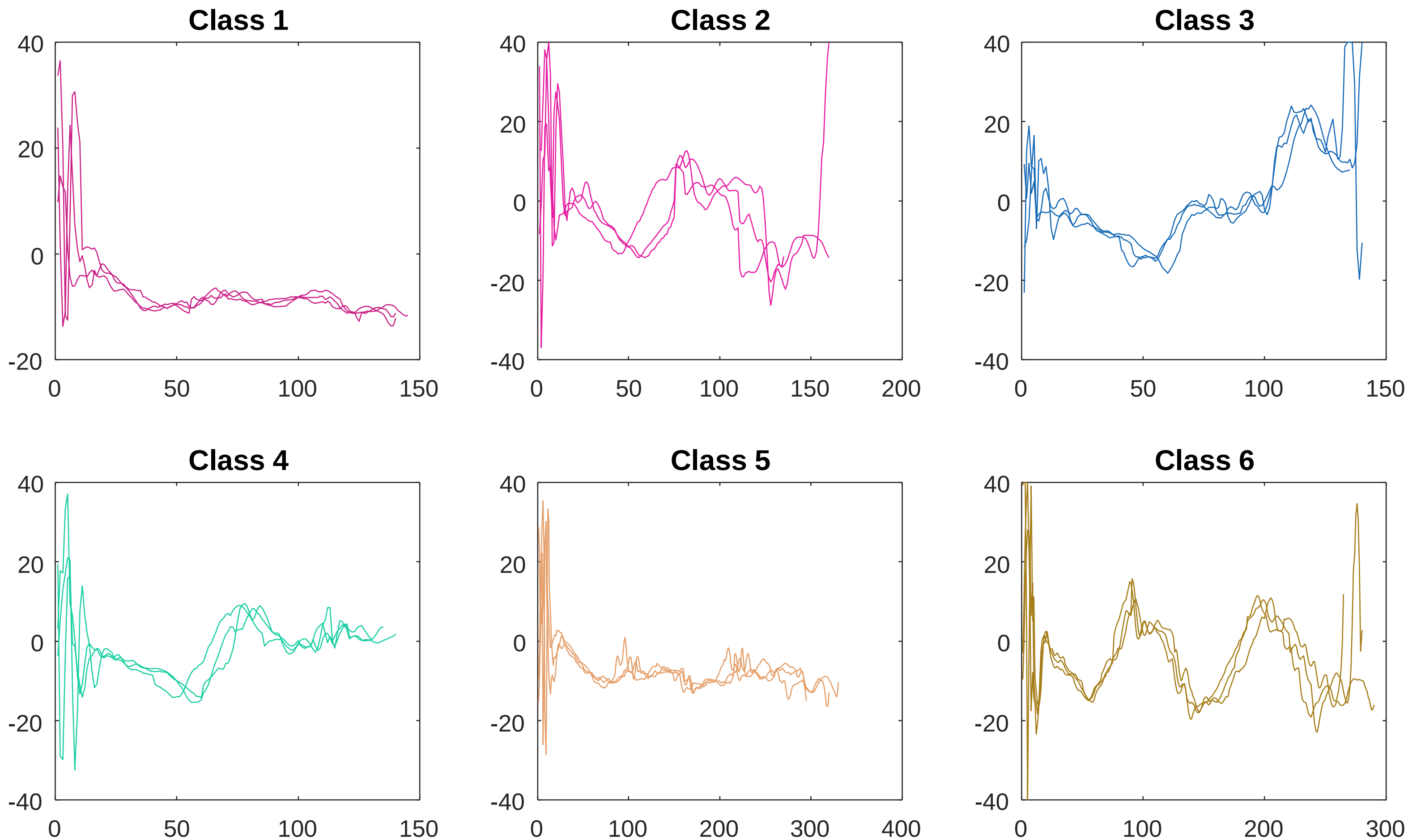
GestureMidAirD3

Three exemplars per class,
without z-normalization



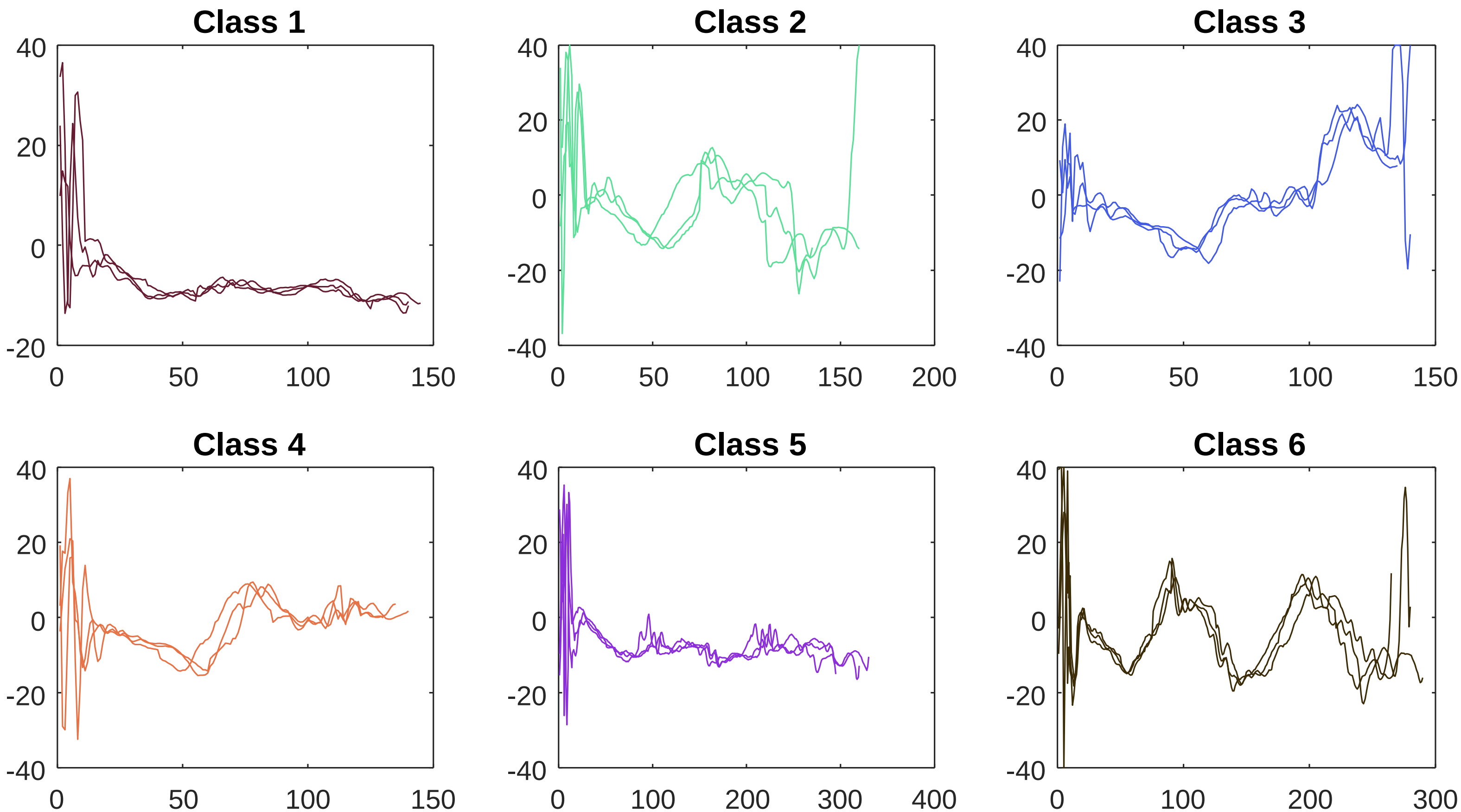
GesturePebbleZ1

Three exemplars per class,
without z-normalization



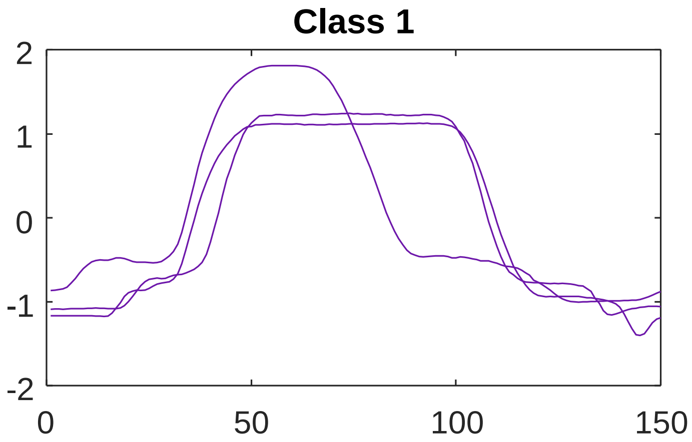
GesturePebbleZ2

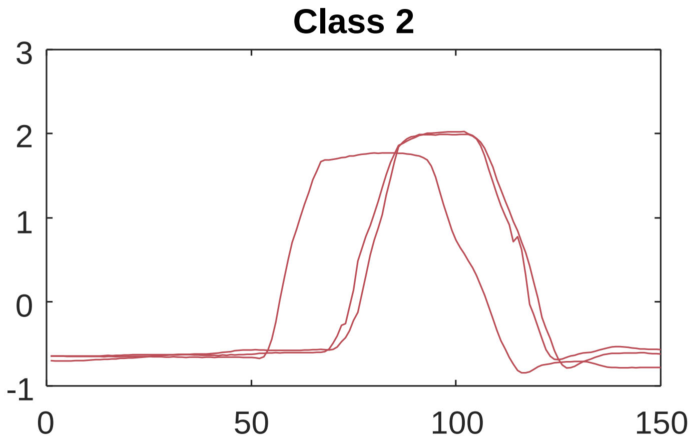
Three exemplars per class,
without z-normalization

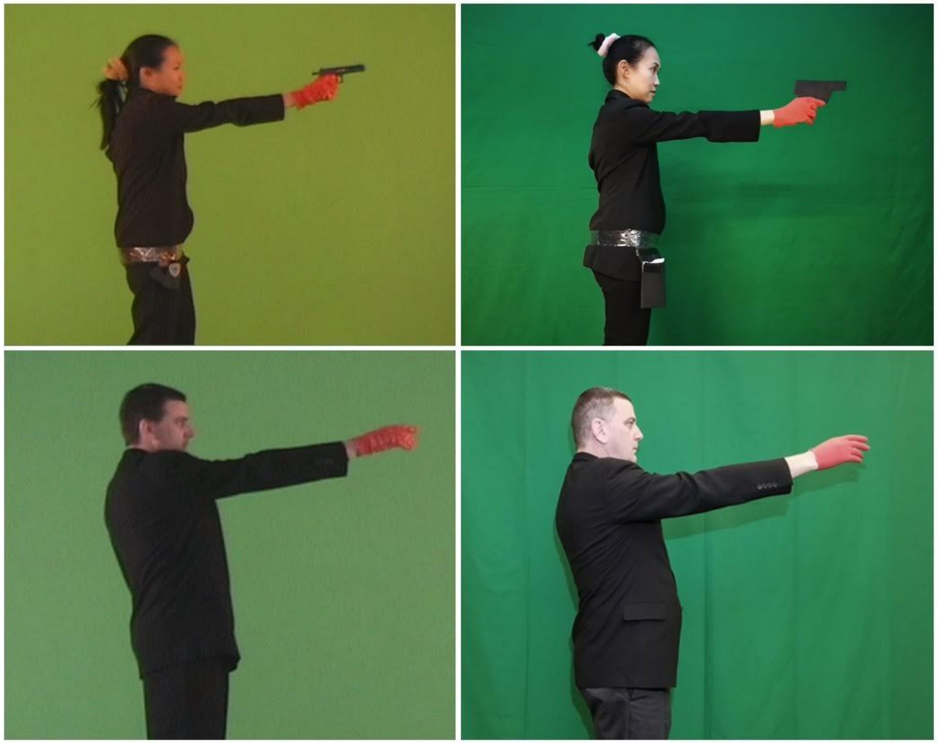


GunPointAgeSpan

Three exemplars per class,
with z-normalization



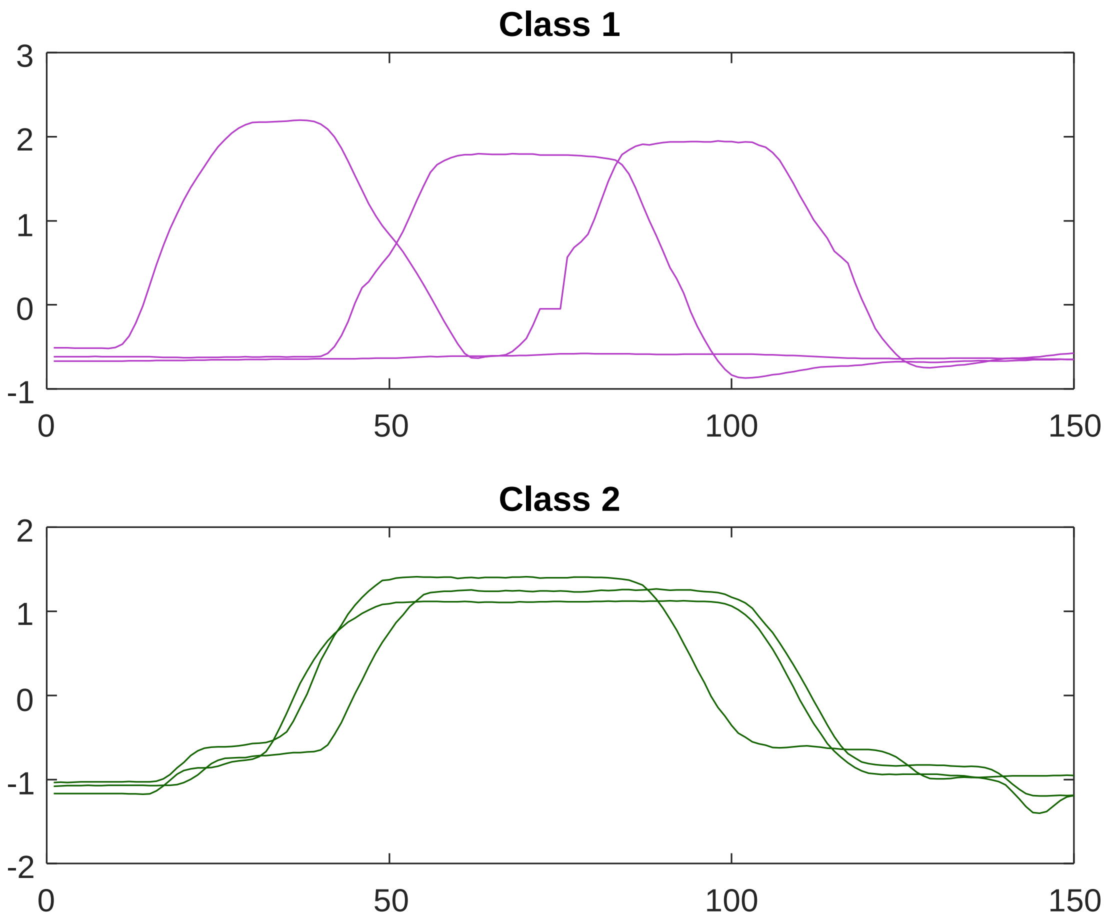




Left) GunPoint recording of 2003, *right*) GunPoint recording of 2018.
Top) Ann Ratanamahatana, *bottom*) Eamonn Keogh.
The female and male actors are the same individuals recorded fifteen years apart.

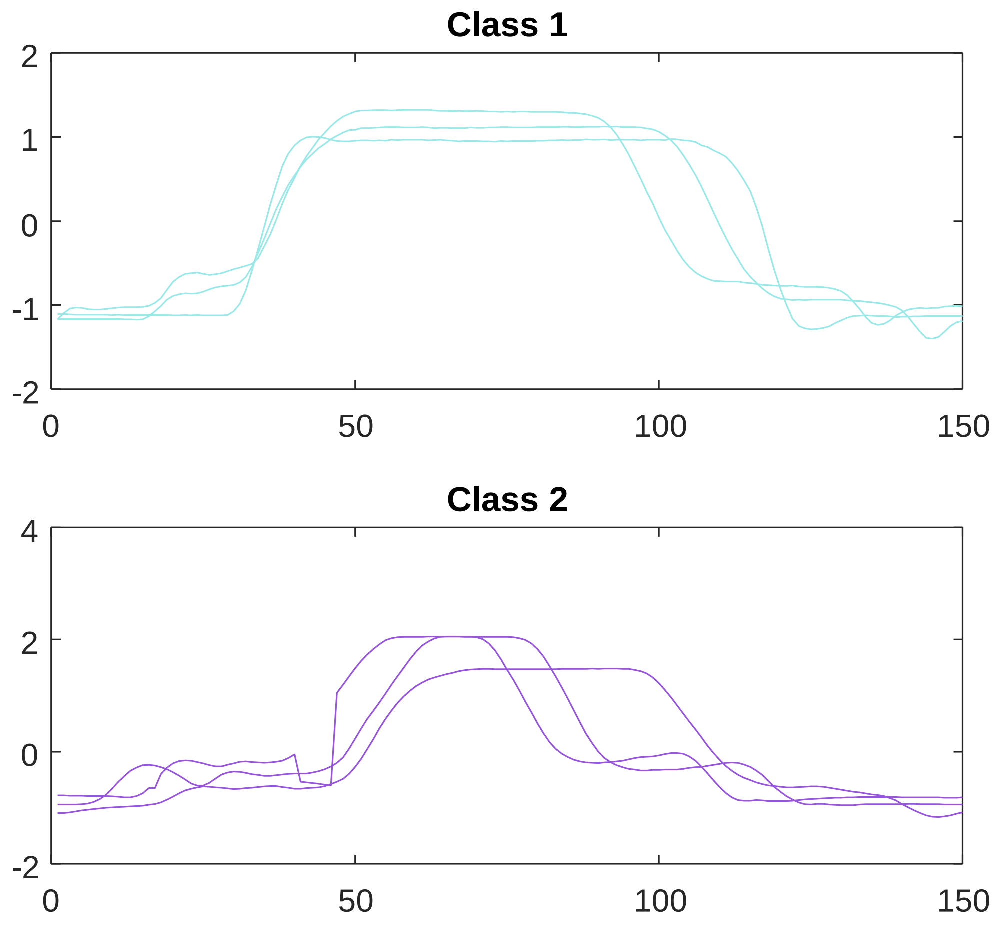
GunPointMaleVersusFemale

Three exemplars per class,
with z-normalization



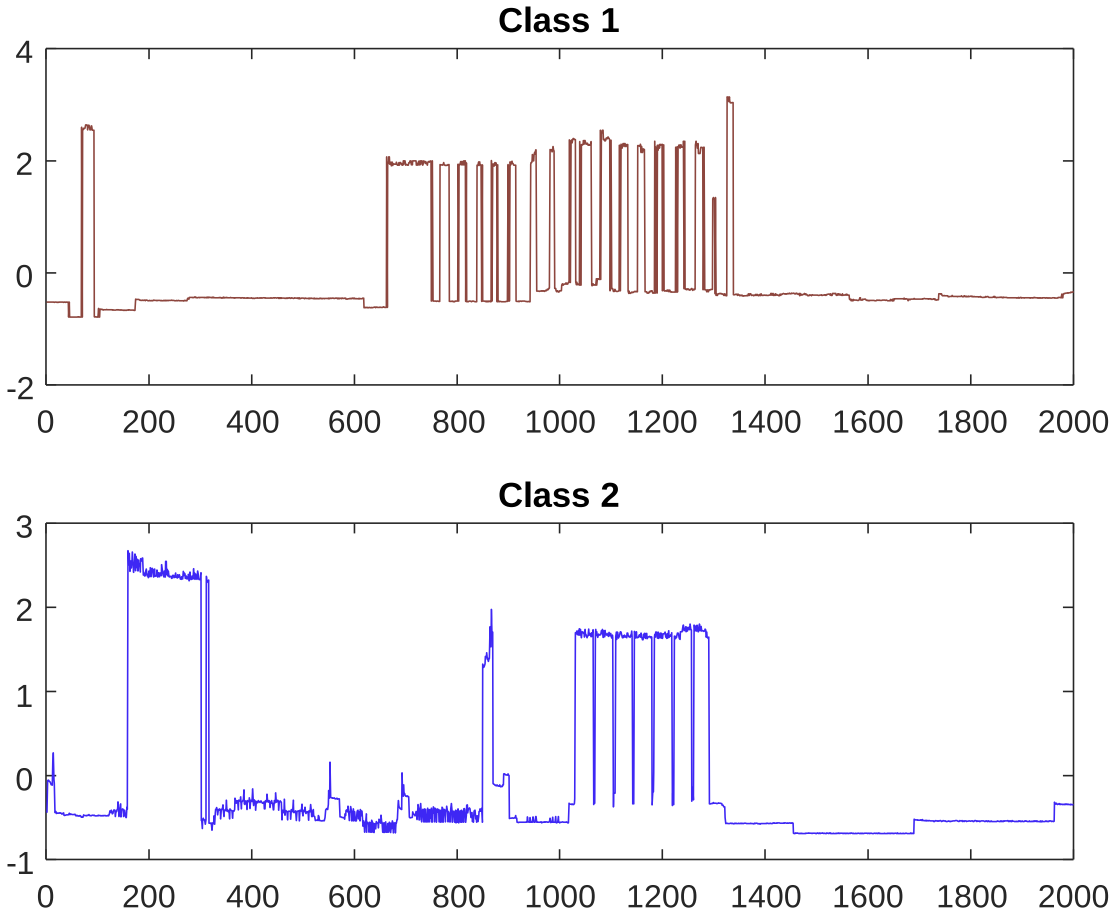
GunPointOldVersusYoung

Three exemplars per class,
with z-normalization



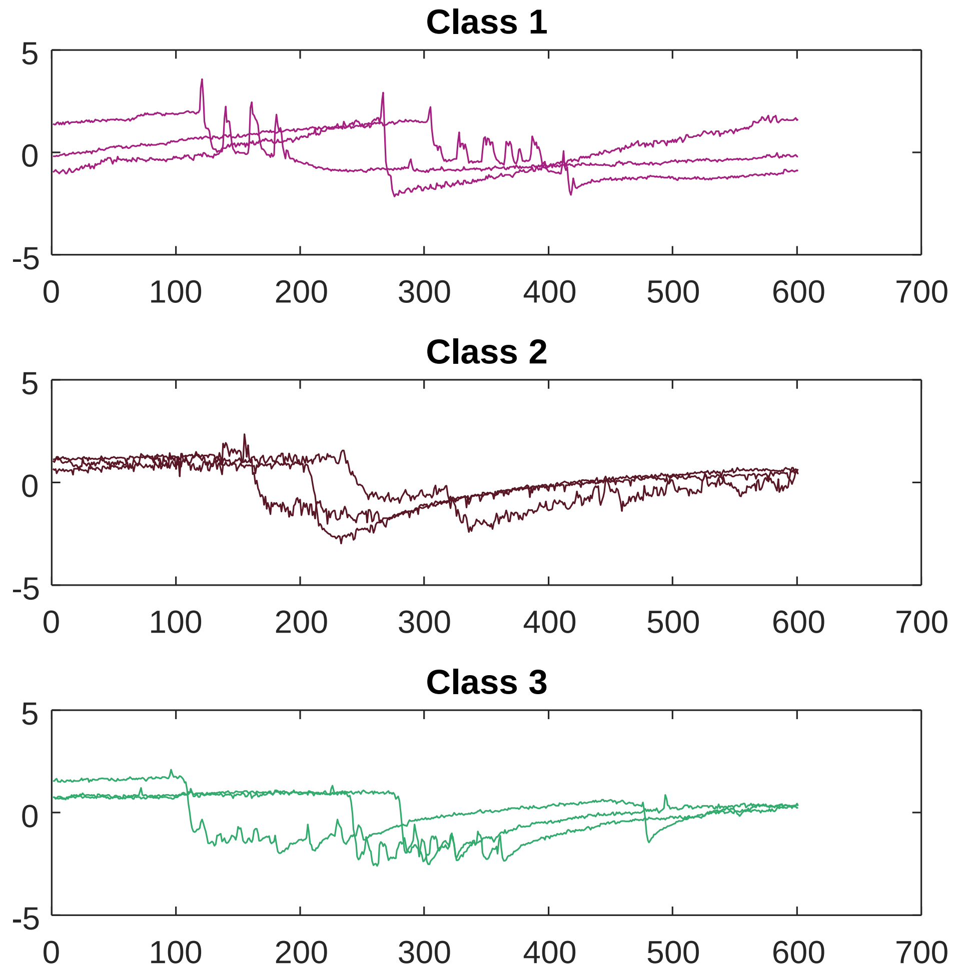
HouseTwenty

One exemplars per class,
with z-normalization



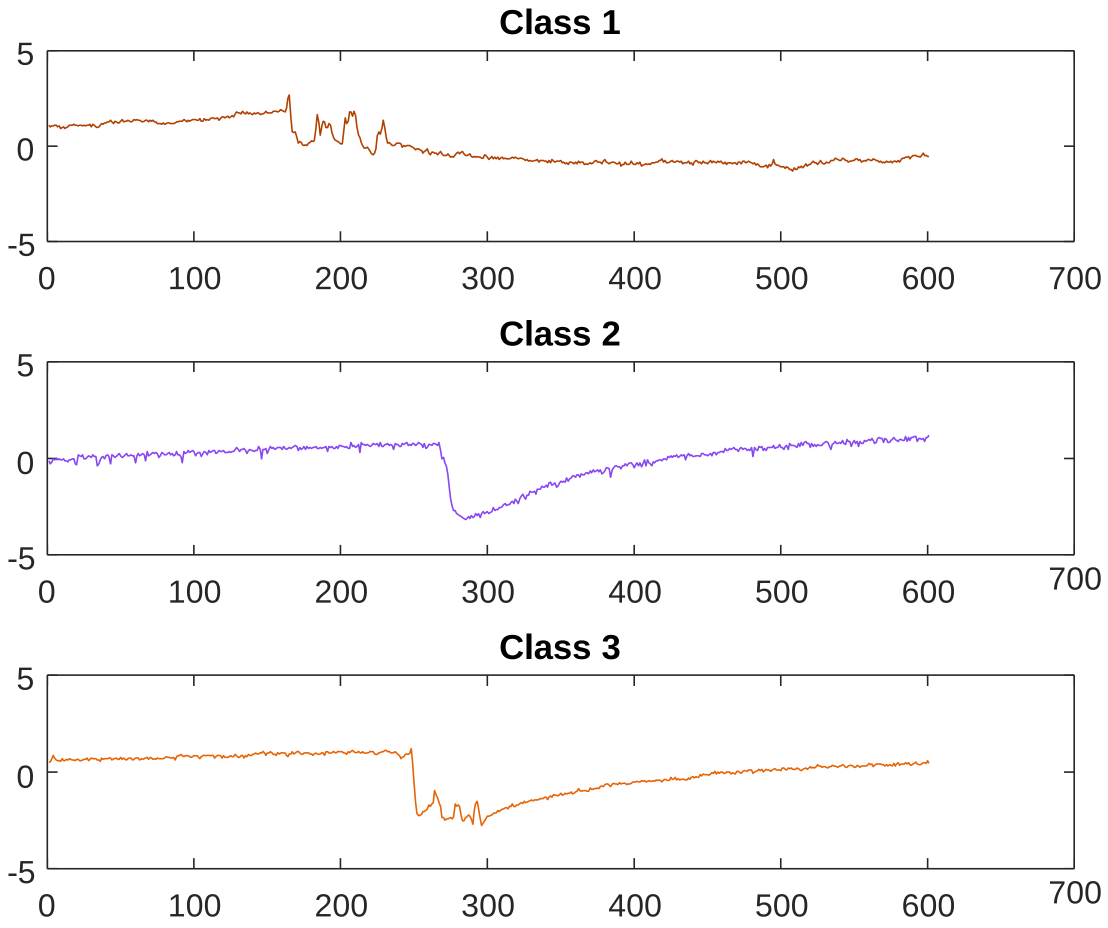
InsectEPGRegularTrain

Three exemplars per class,
with z-normalization



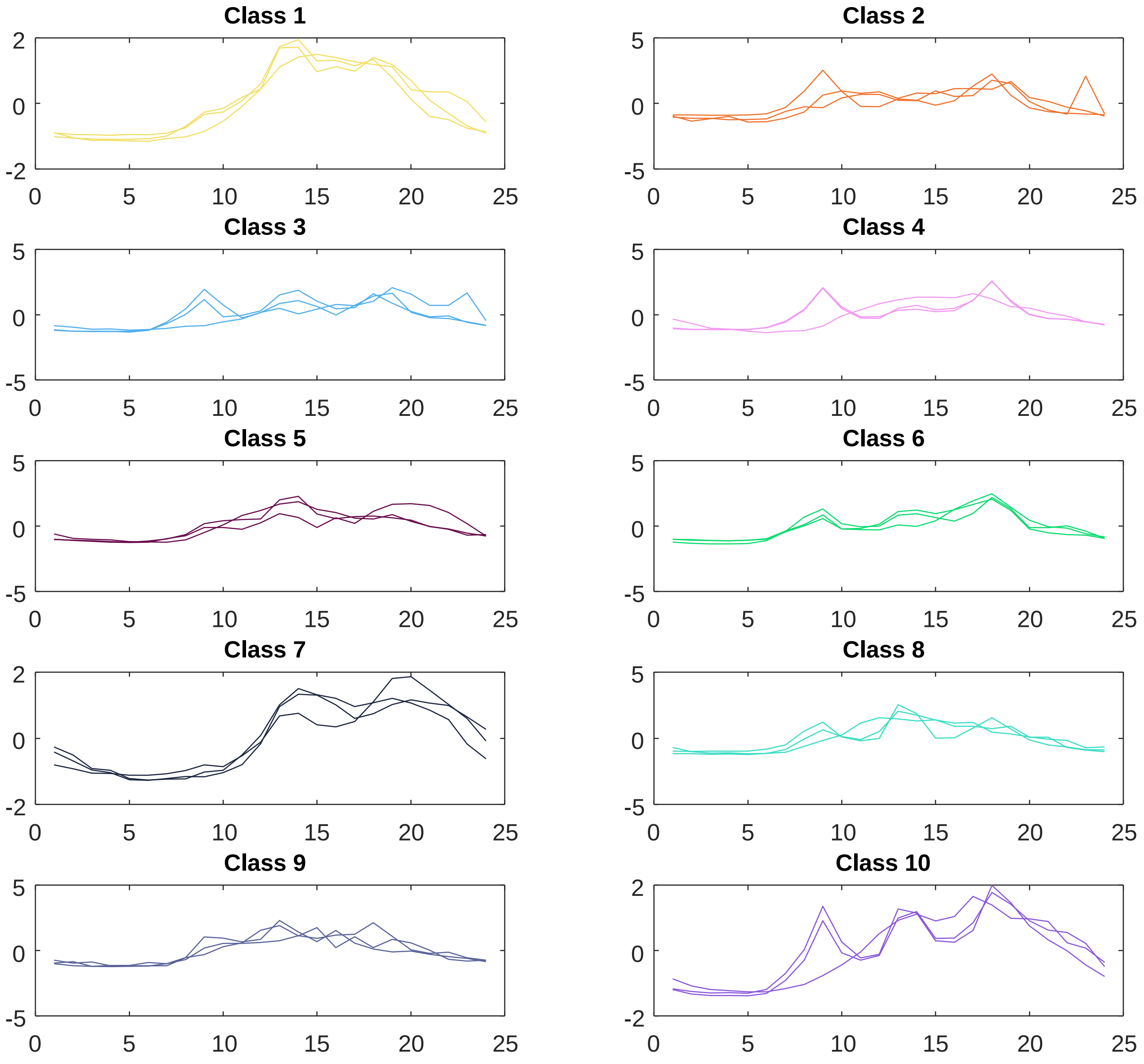
InsectEPGSmallTrain

One exemplars per class,
with z-normalization



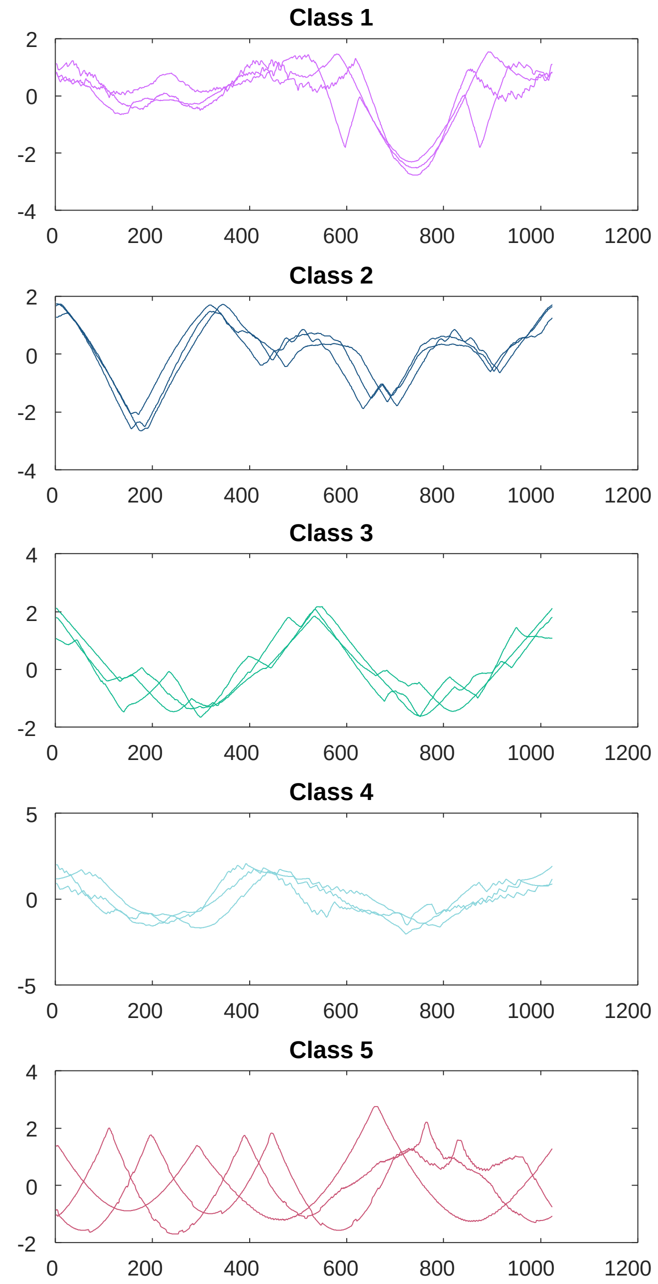
MelbournePedestrian

Three exemplars per class, with z-normalization



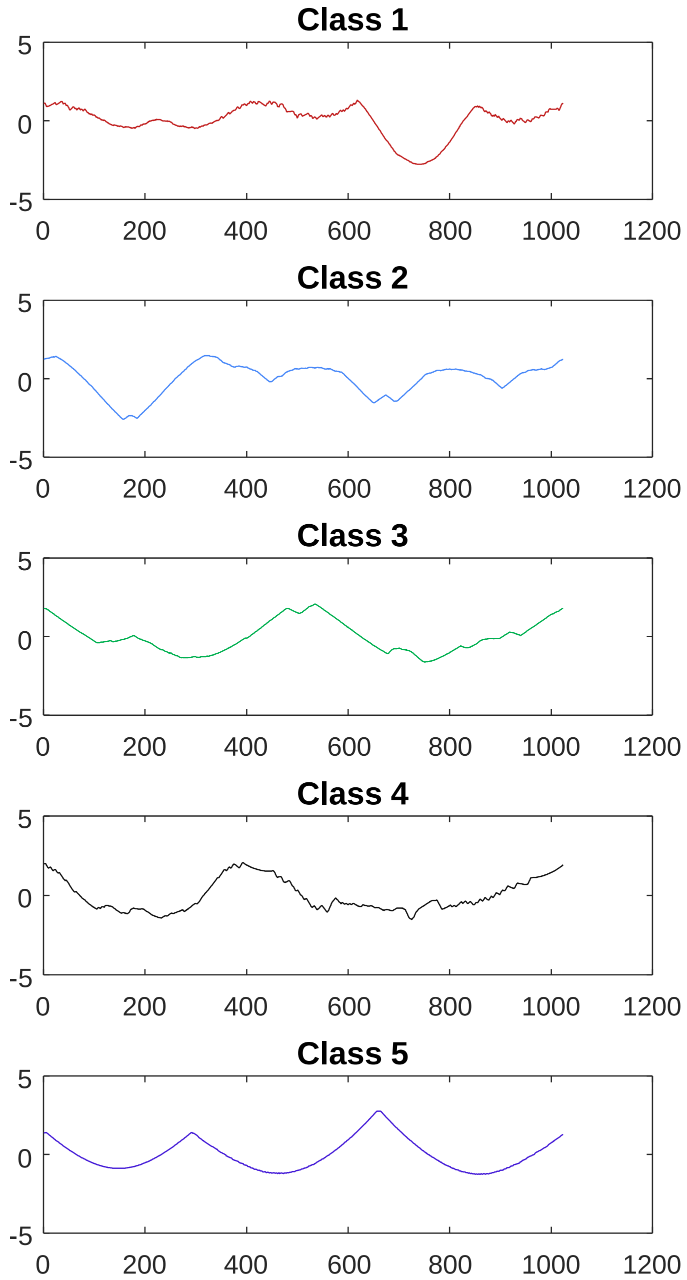
MixedShapesRegularTrain

Three exemplars per class,
with z-normalization



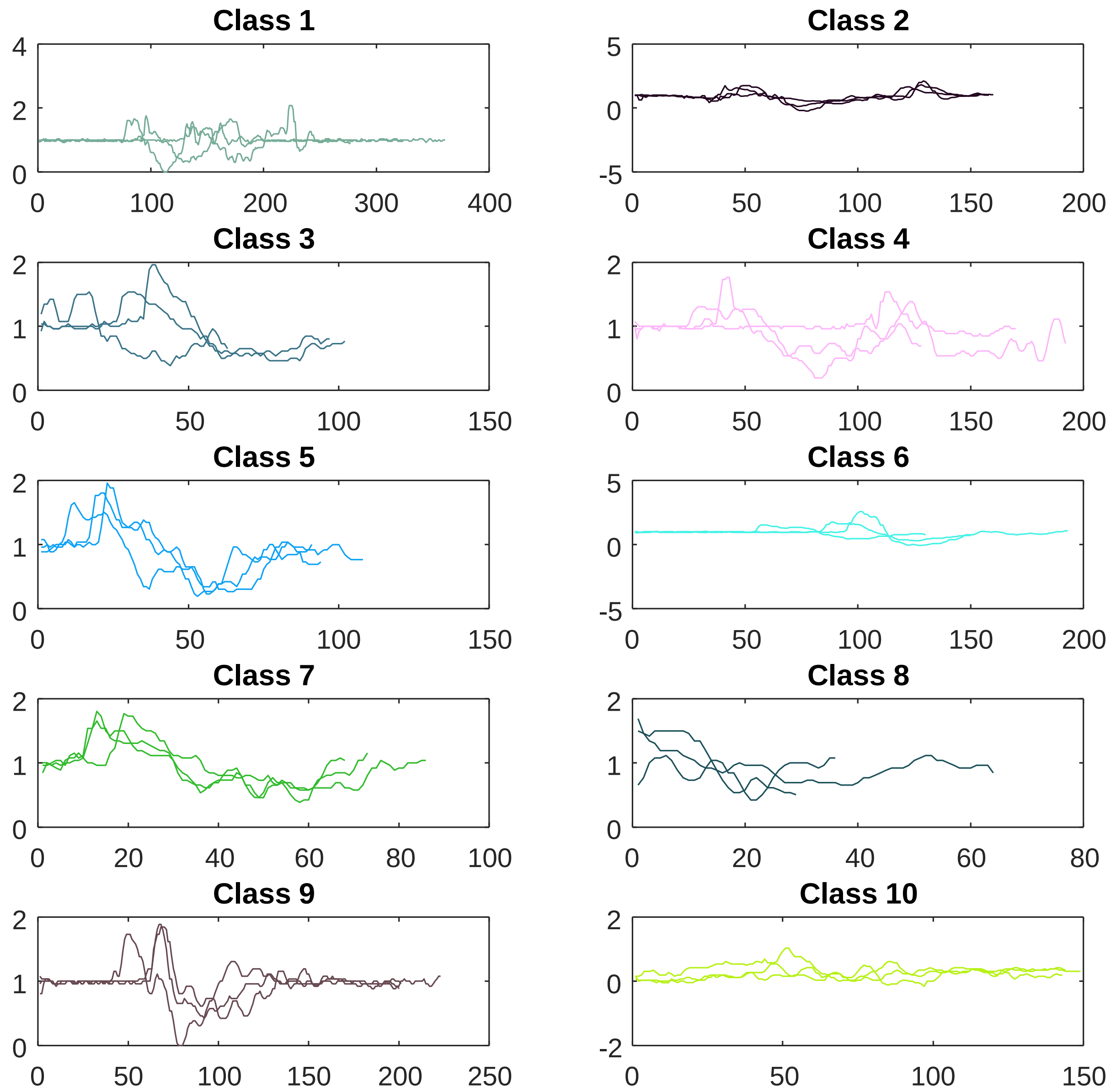
MixedShapesSmallTrain

One exemplar per class,
with z-normalization



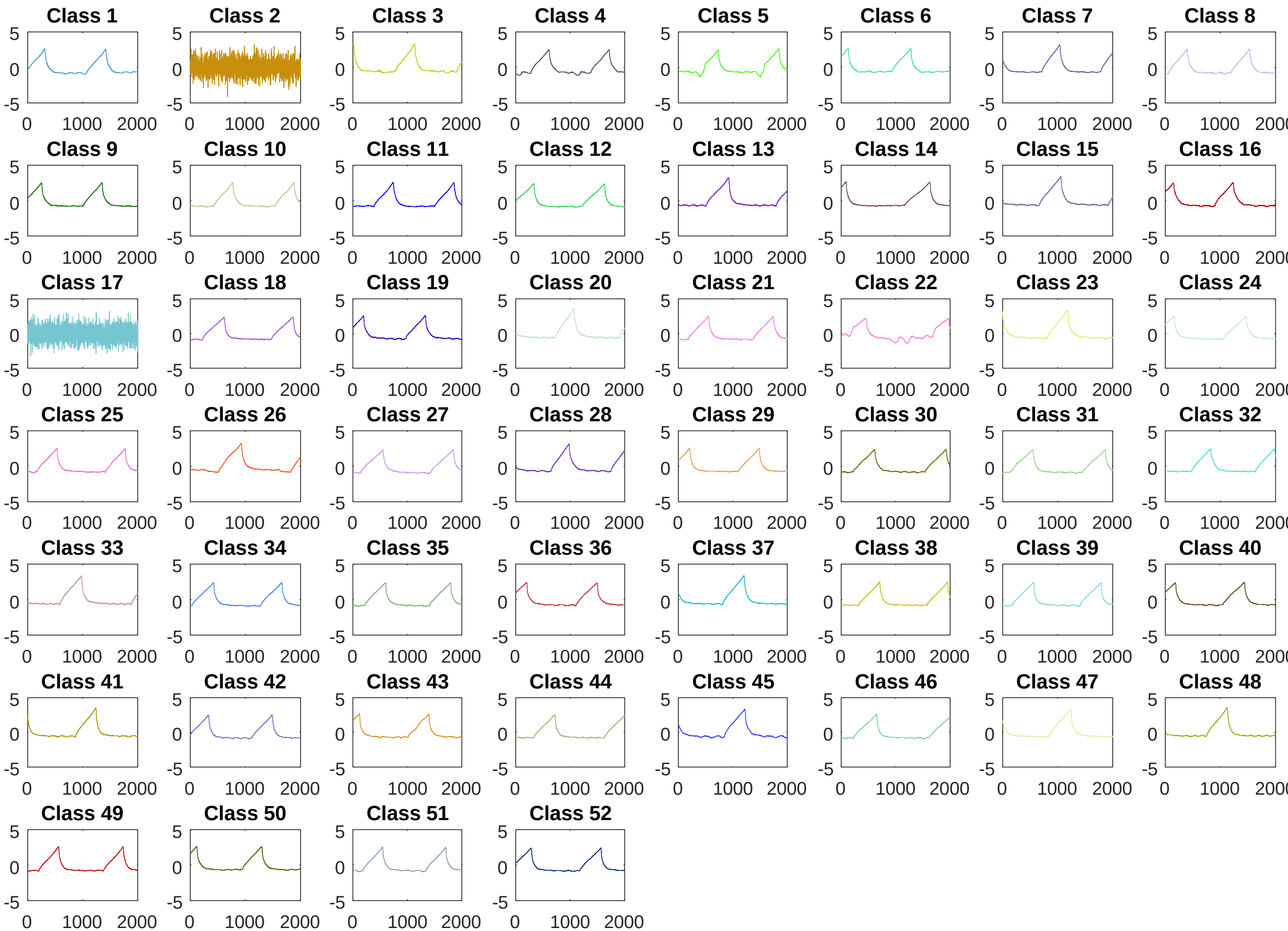
PickupGestureWiimoteZ

Three exemplars per class, without z-normalization



PigAirwayPressure

One exemplar per class,
with z-normalization

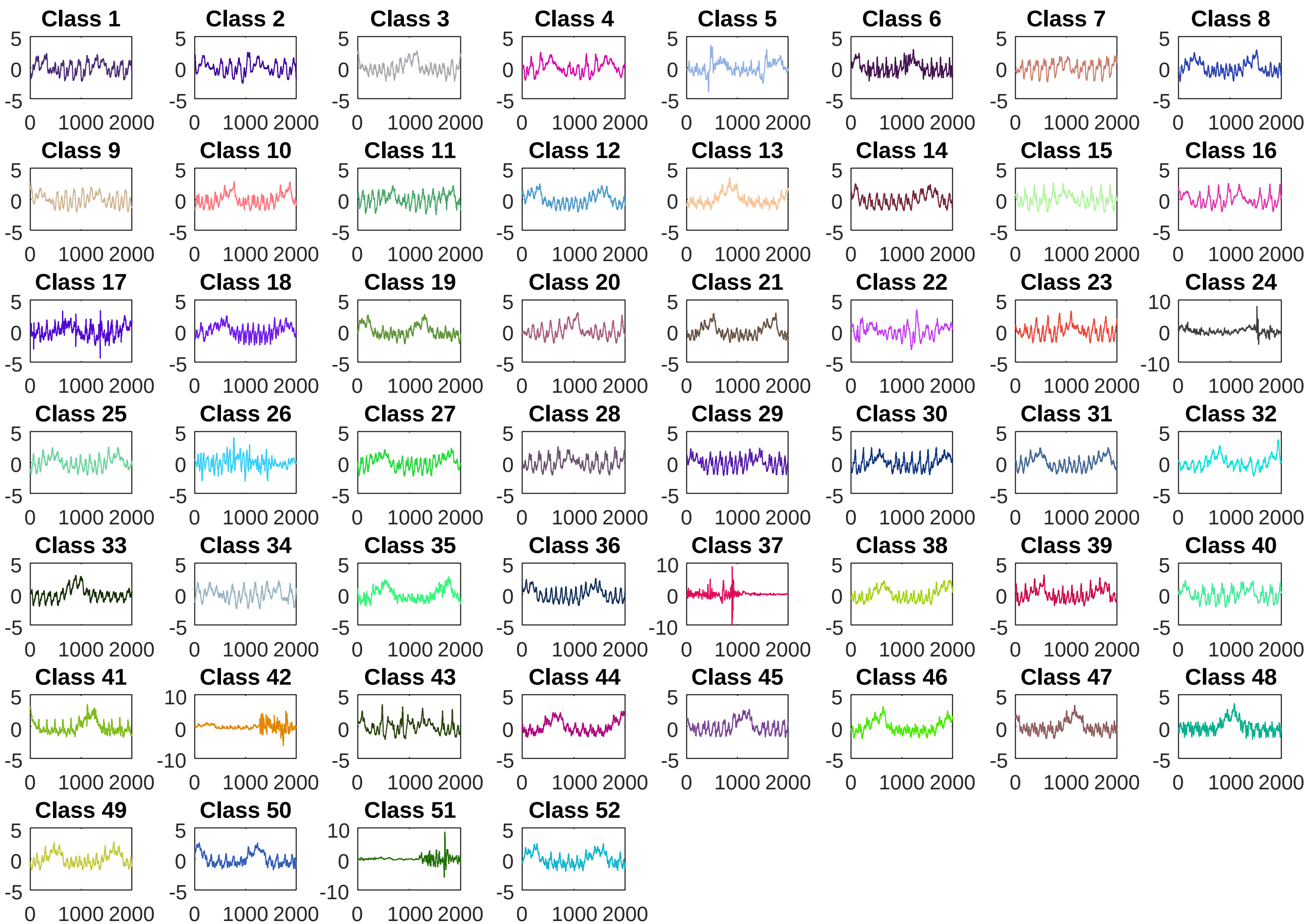


PigArtPressure

One exemplar per class,
with z-normalization

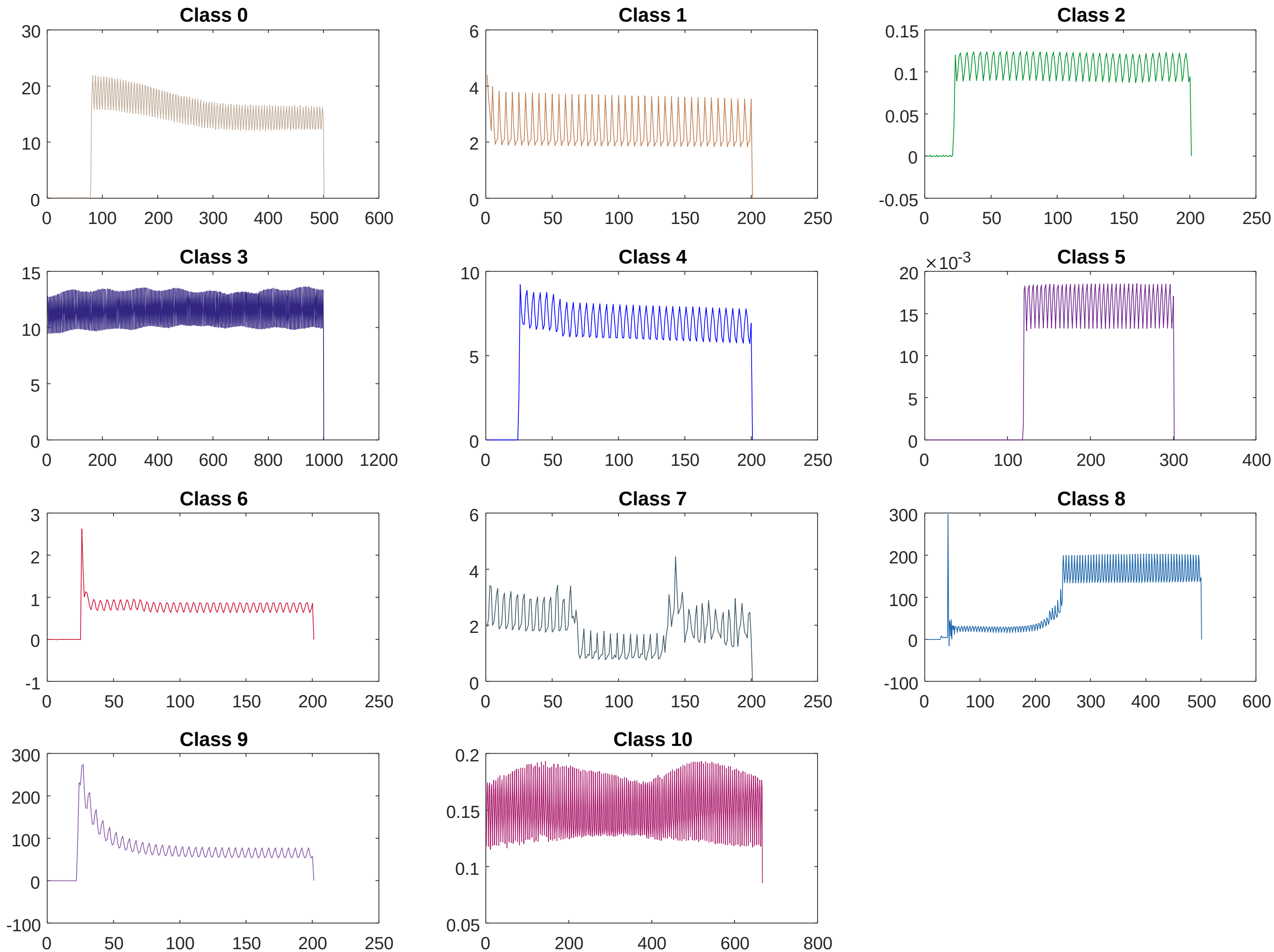
PigCVP

One exemplar per class,
with z-normalization



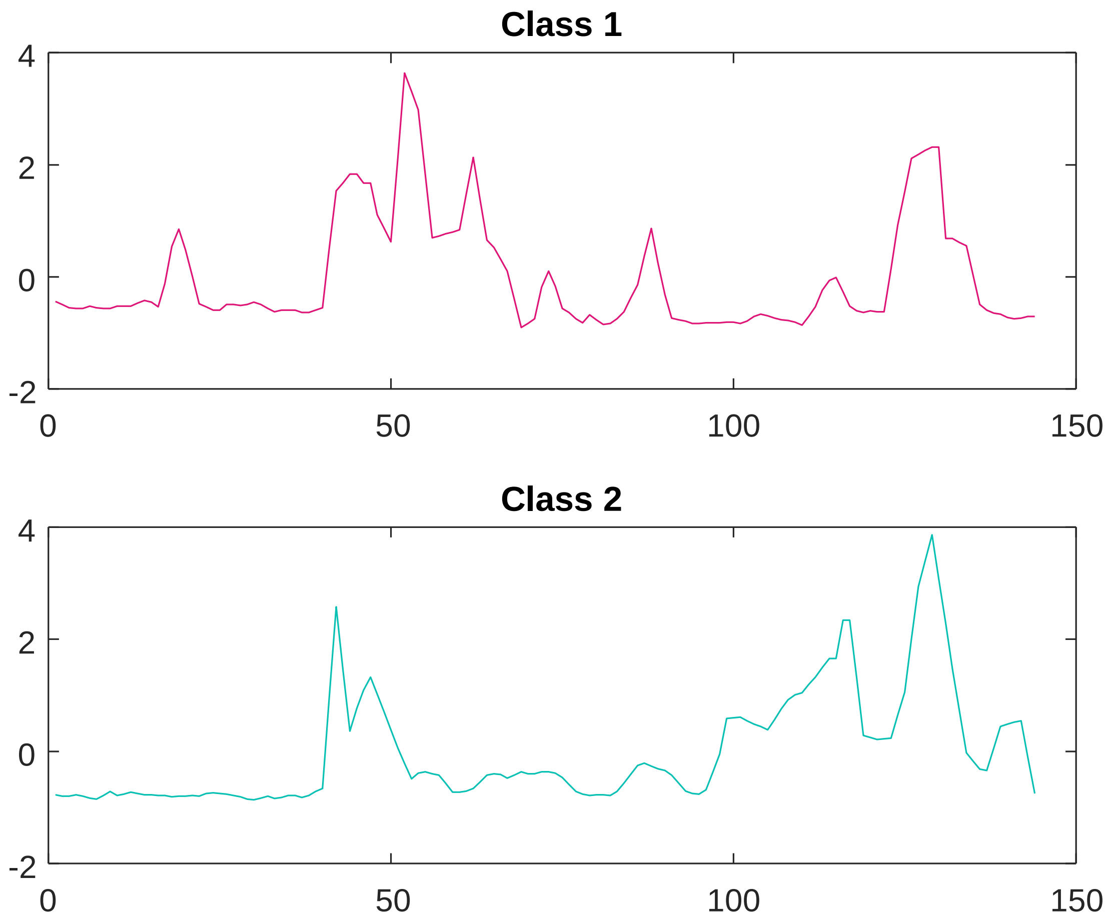
PLAID

One exemplar per class,
without z-normalization



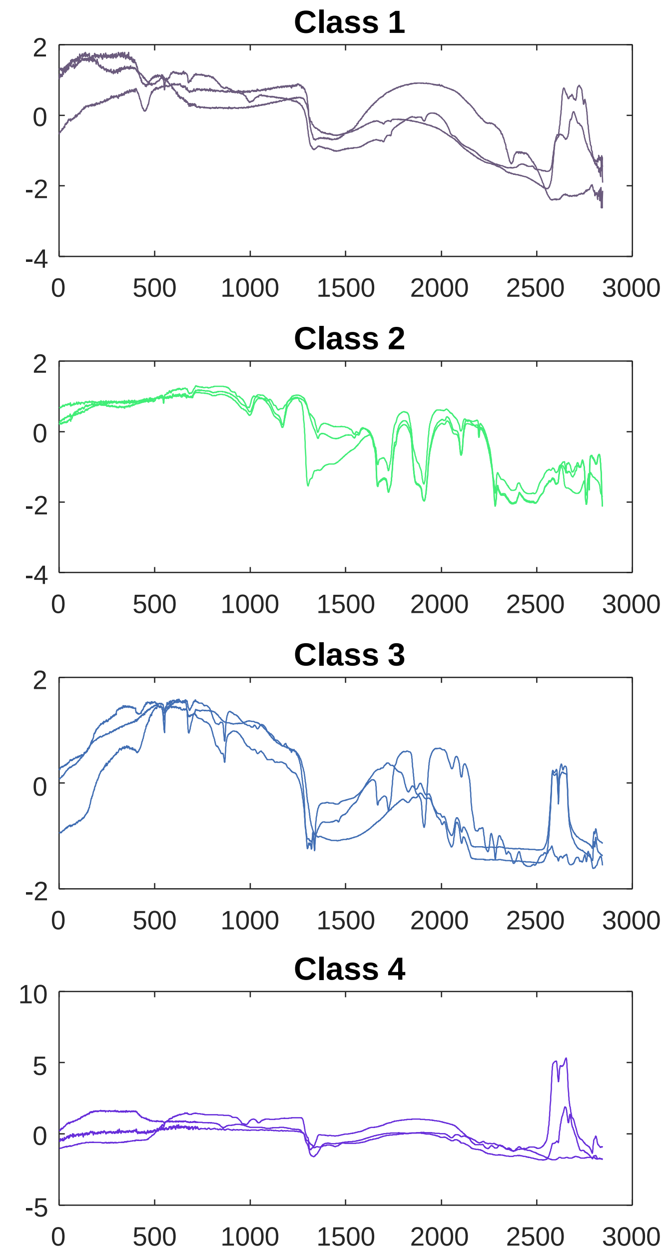
PowerCons

One exemplar per class,
with z-normalization



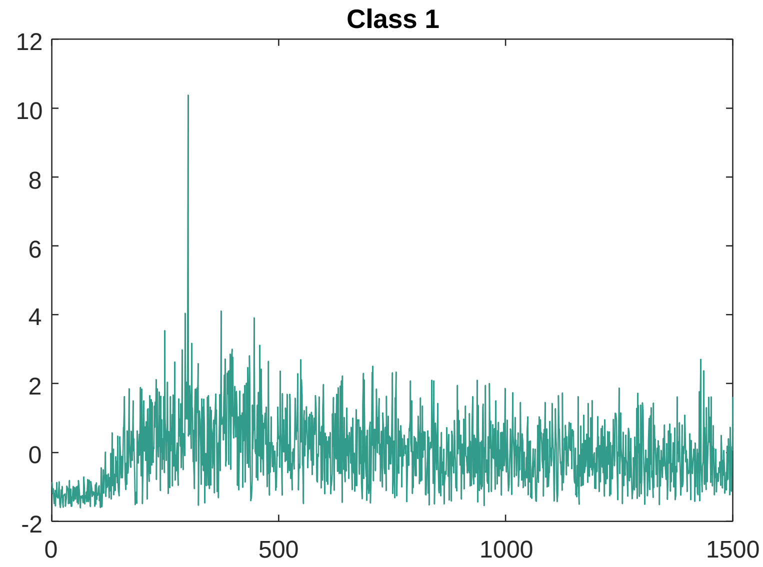
Rock

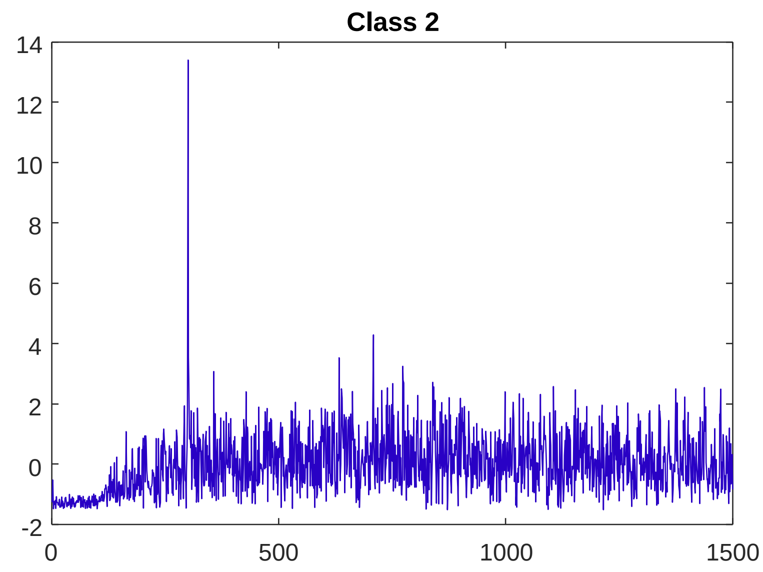
Three exemplars per class,
with z-normalization



SemgHandGenderCh2

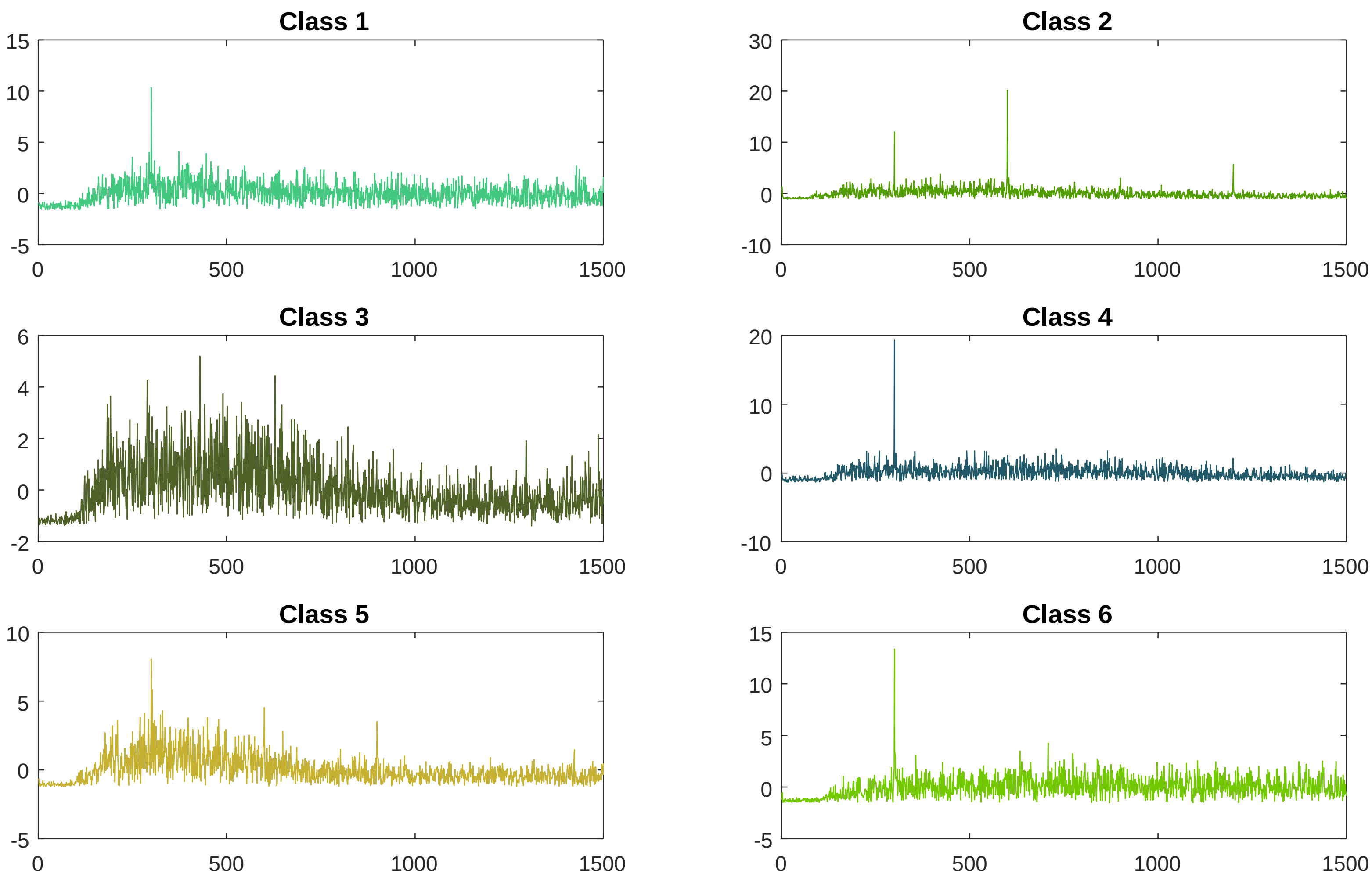
One exemplar per class,
with z-normalization





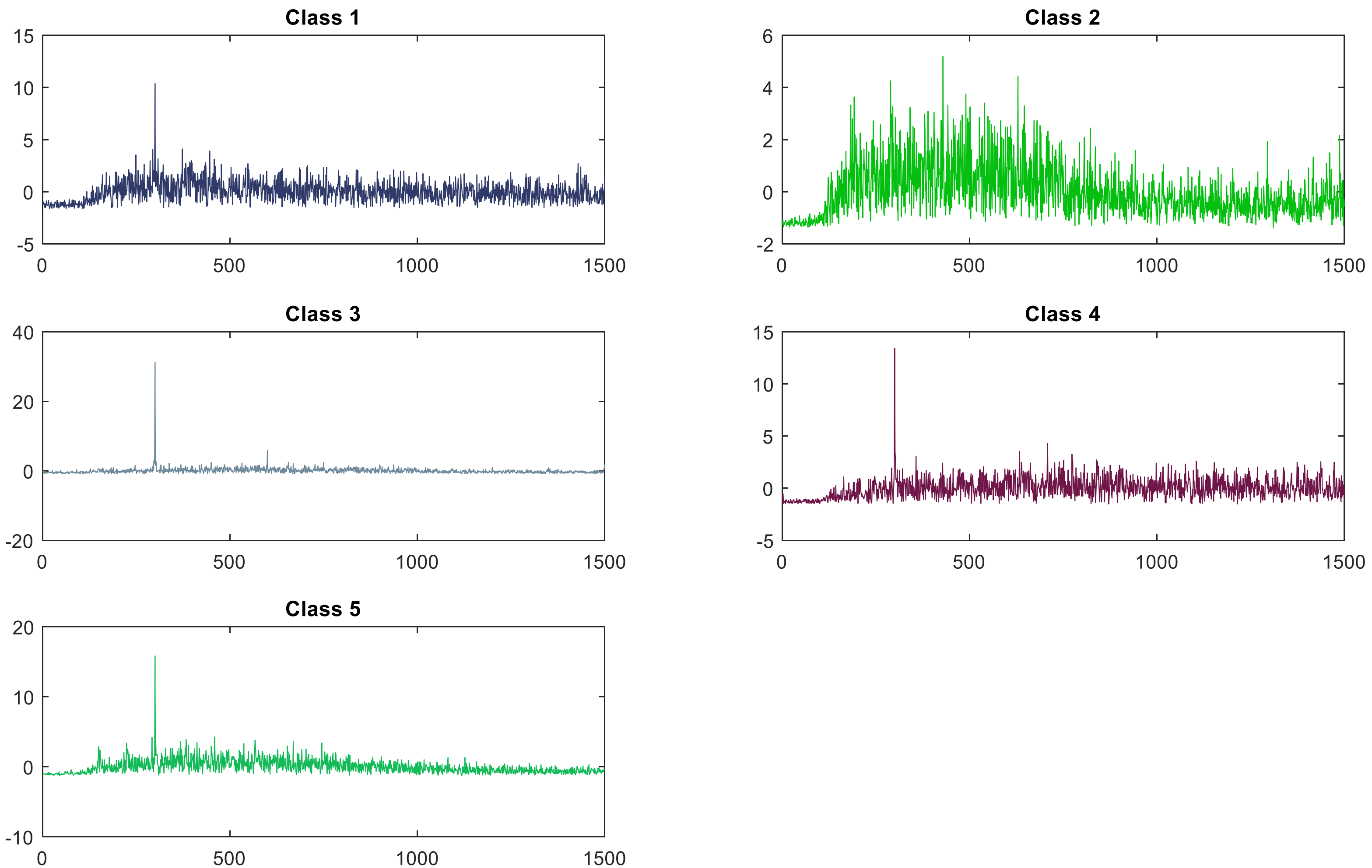
SemgHandMovementCh2

One exemplar per class,
with z-normalization



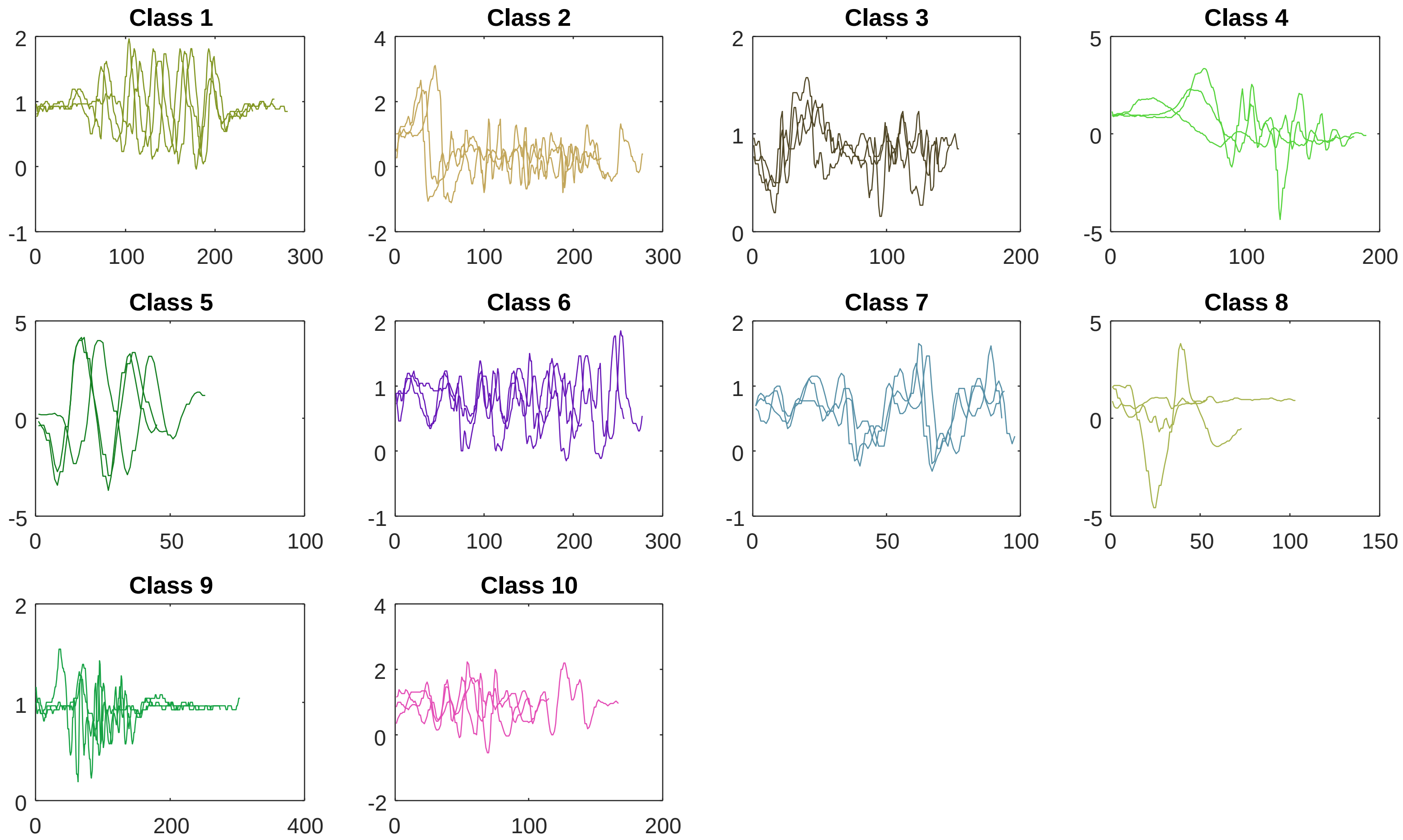
SemgHandSubjectCh2

One exemplar per class,
with z-normalization



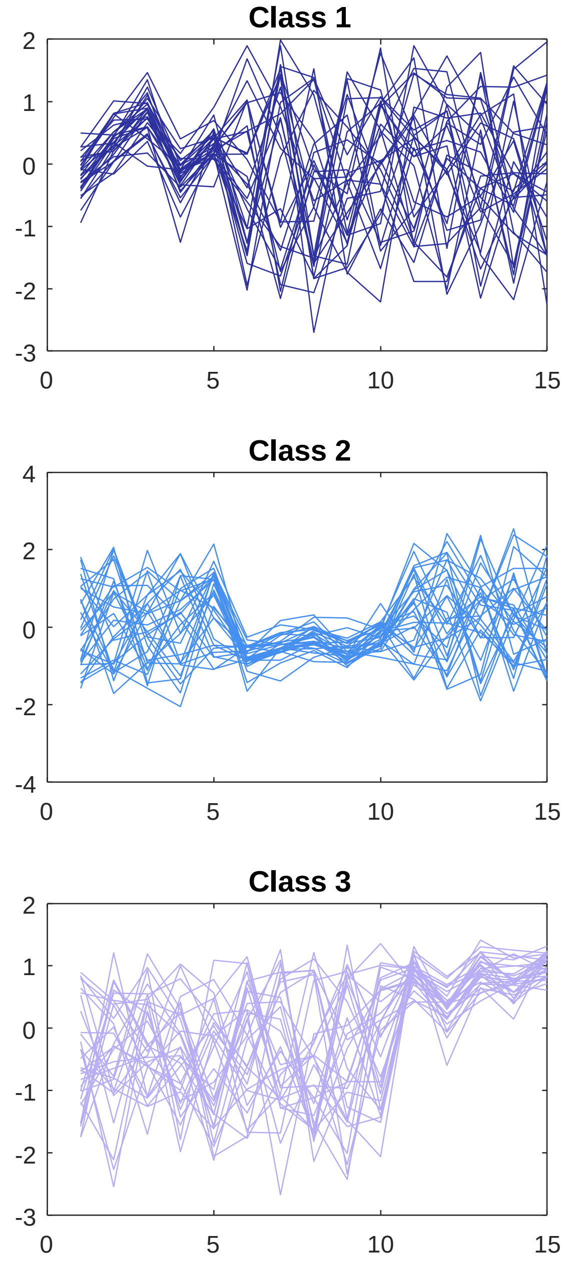
ShakeGestureWiimoteZ

Three exemplars per class,
without z-normalization



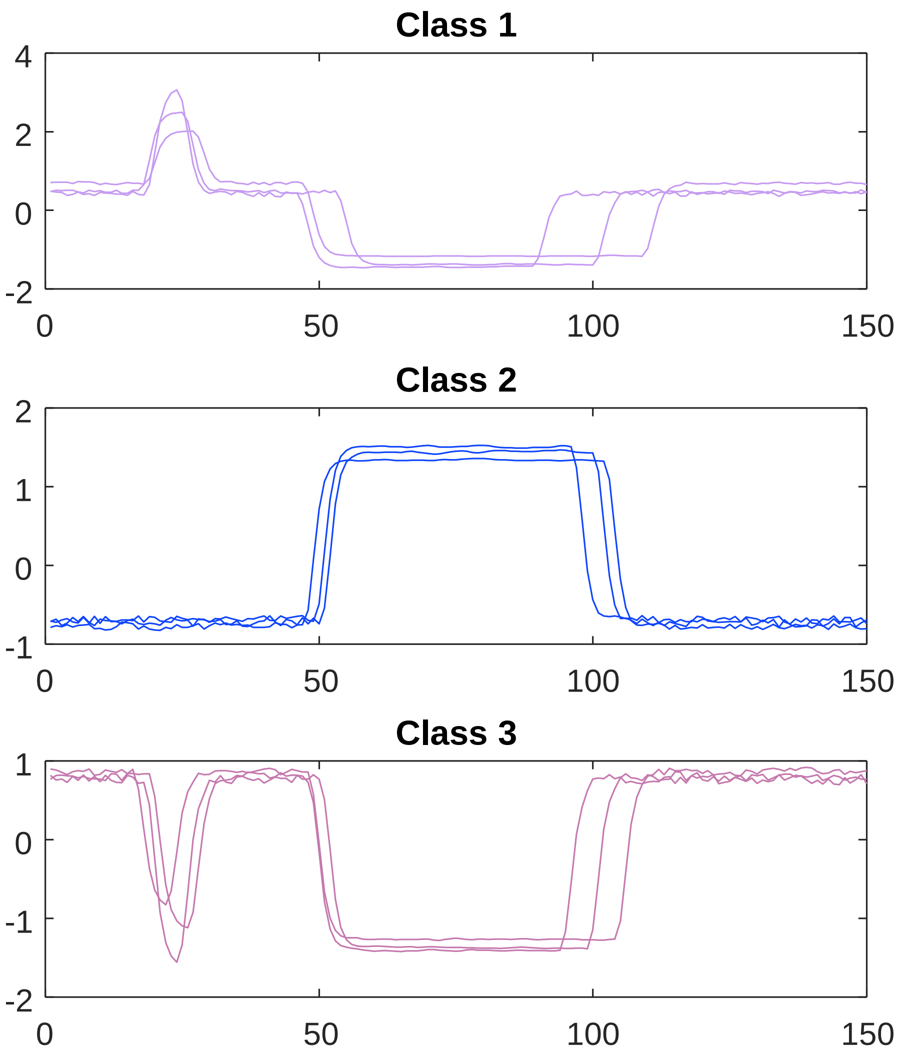
SmoothSubspace

Thirty exemplars per class, with z-normalization



UMD

Three exemplars per class,
with z-normalization

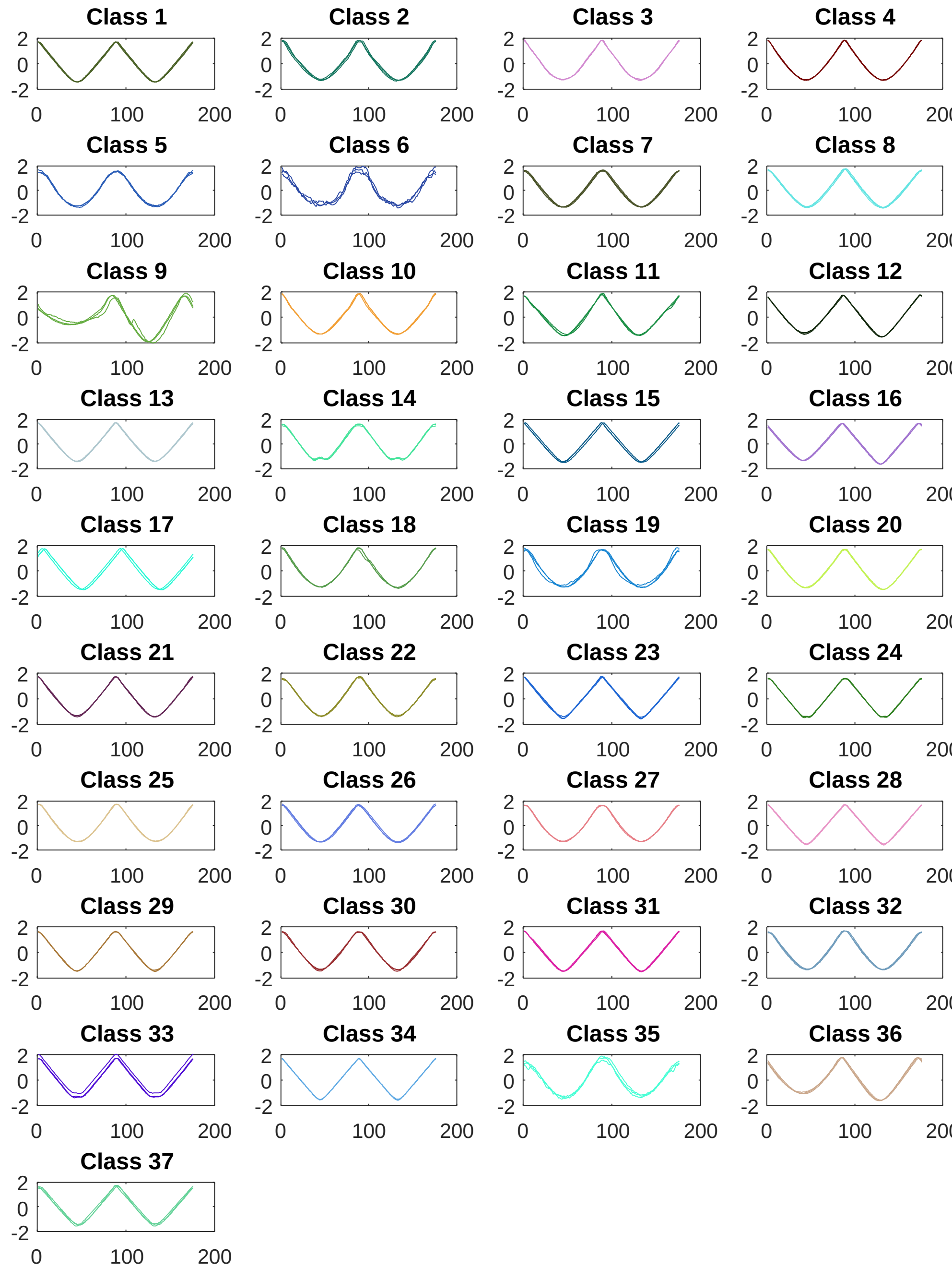


85 data sets from Summer 2015 release

The figures follow are intended to offer a quick inspection of the data. For readability, depending on the scenario, the data may be normalized or may be not, the number of exemplars per class may be one, three or many.

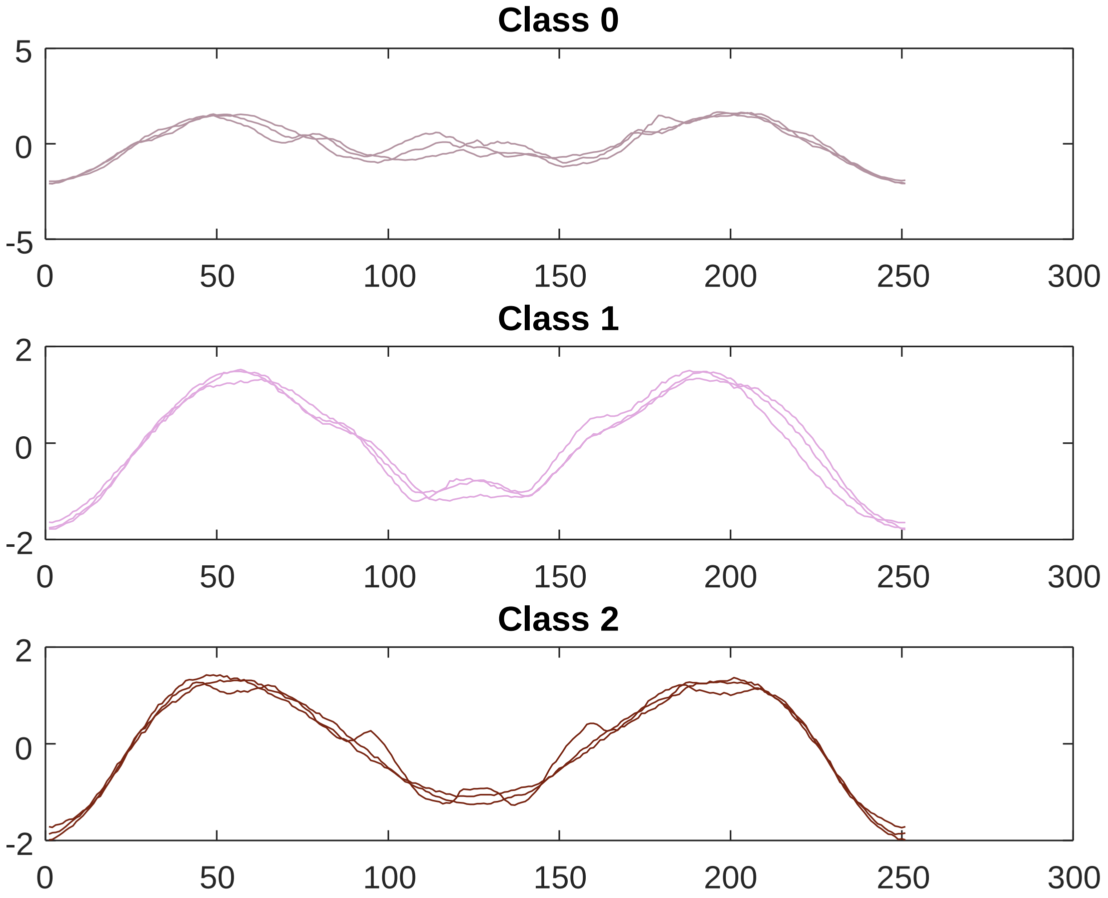
Adiac

Three exemplars per class, with z-normalization



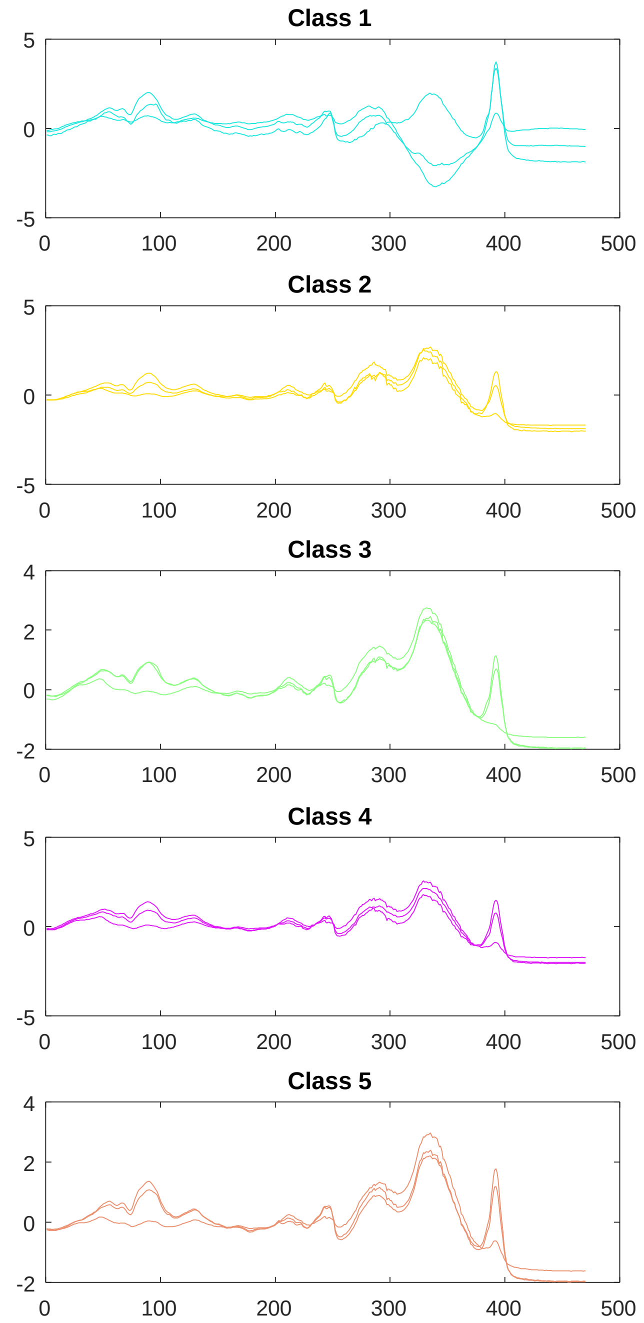
ArrowHead

Three exemplars per class,
with z-normalization



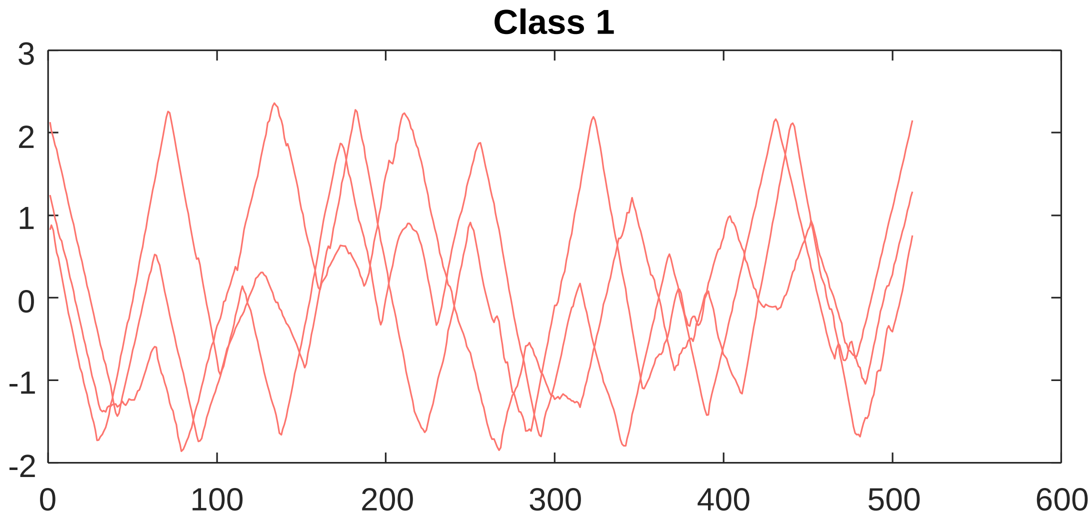
Beef

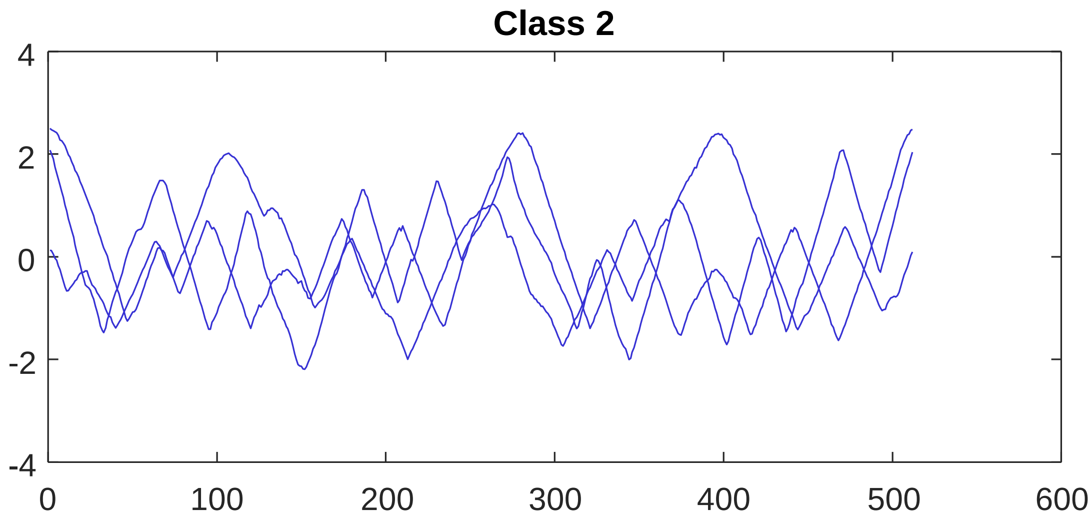
Three exemplars per class,
with z-normalization



BeetleFly

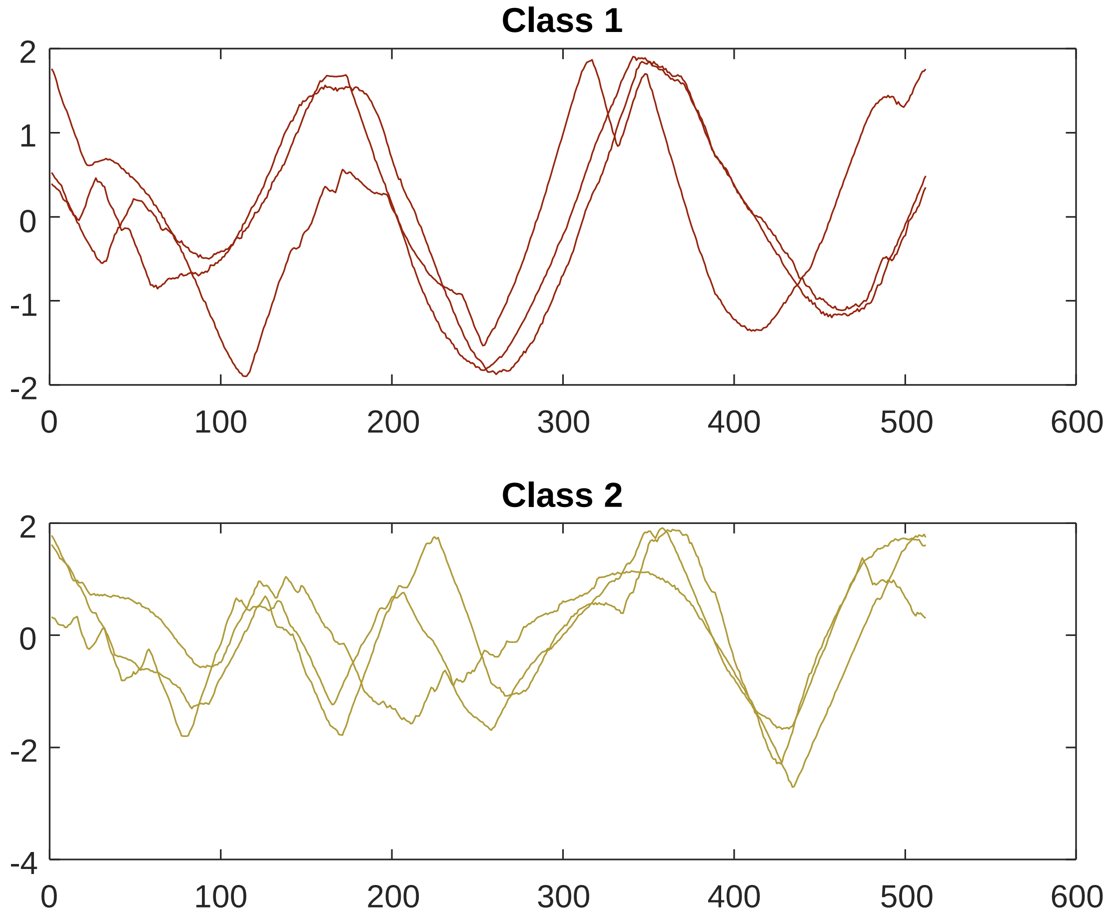
Three exemplars per class, with z-normalization





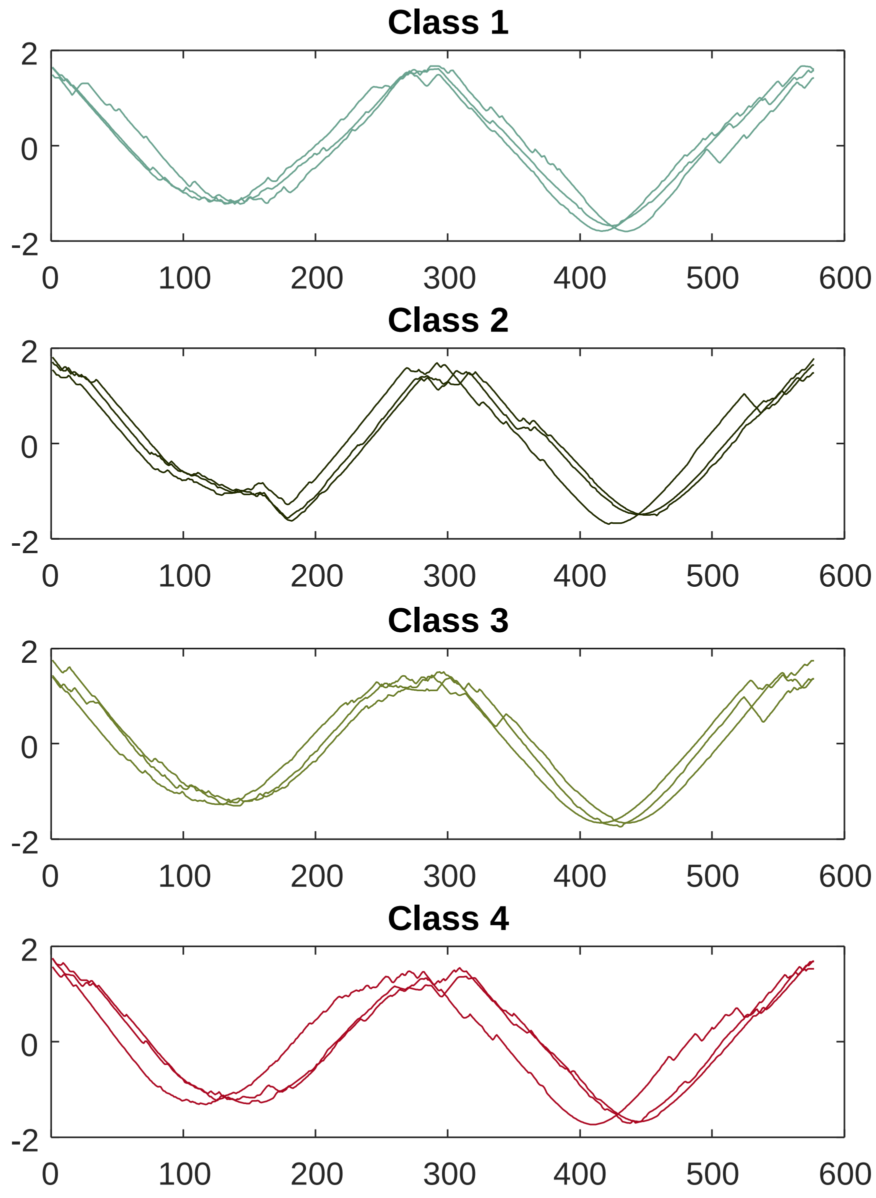
BirdChicken

Three exemplars per class,
with z-normalization



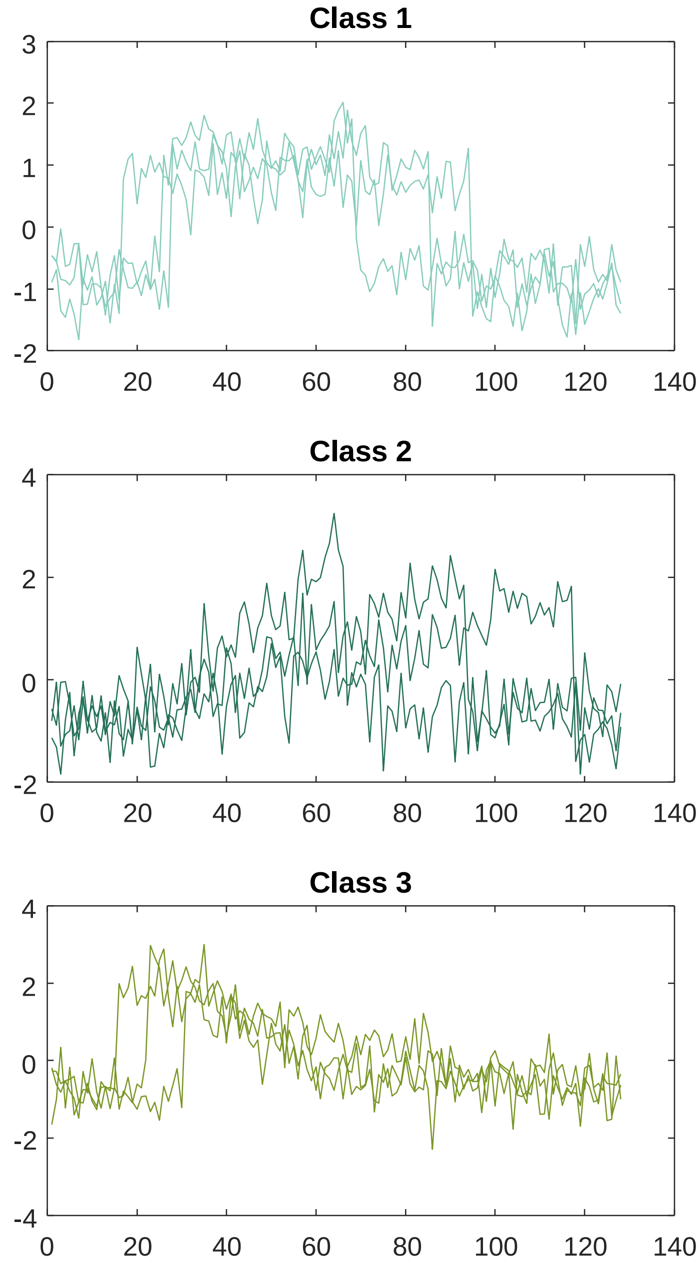
Car

Three exemplars per class,
with z-normalization



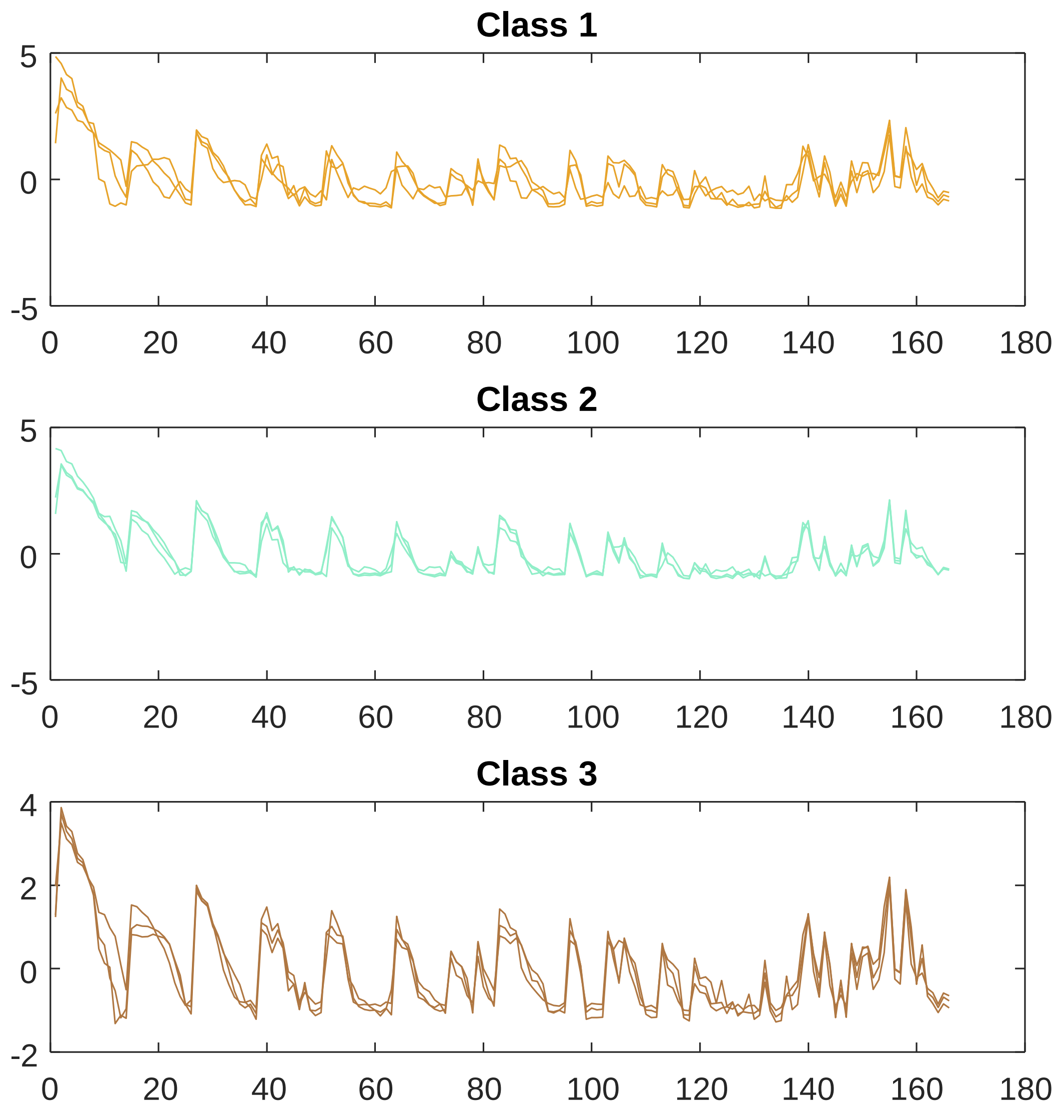
CBF

Three exemplars per class,
with z-normalization



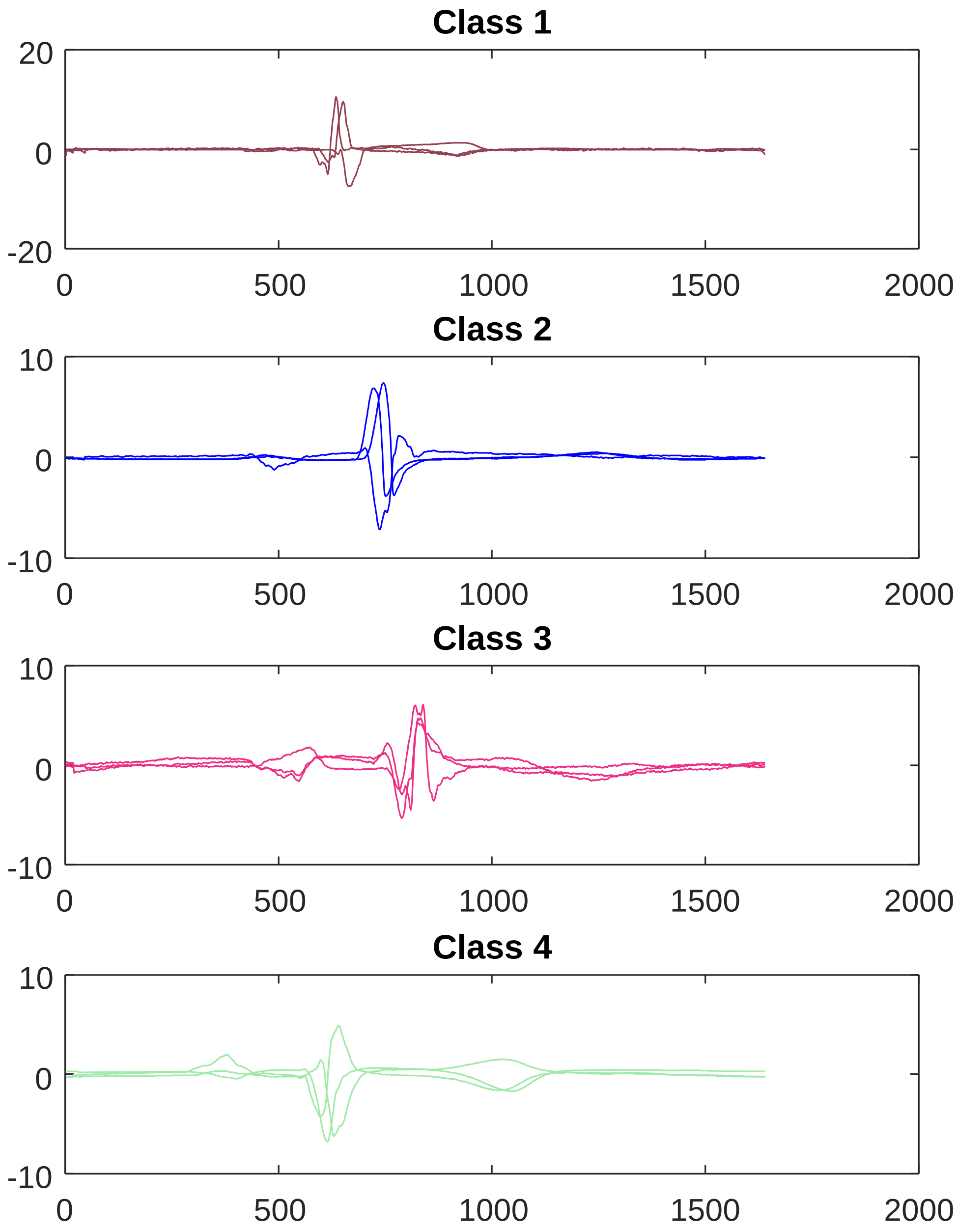
ChlorineConcentration

Three exemplars per class, with z-normalization



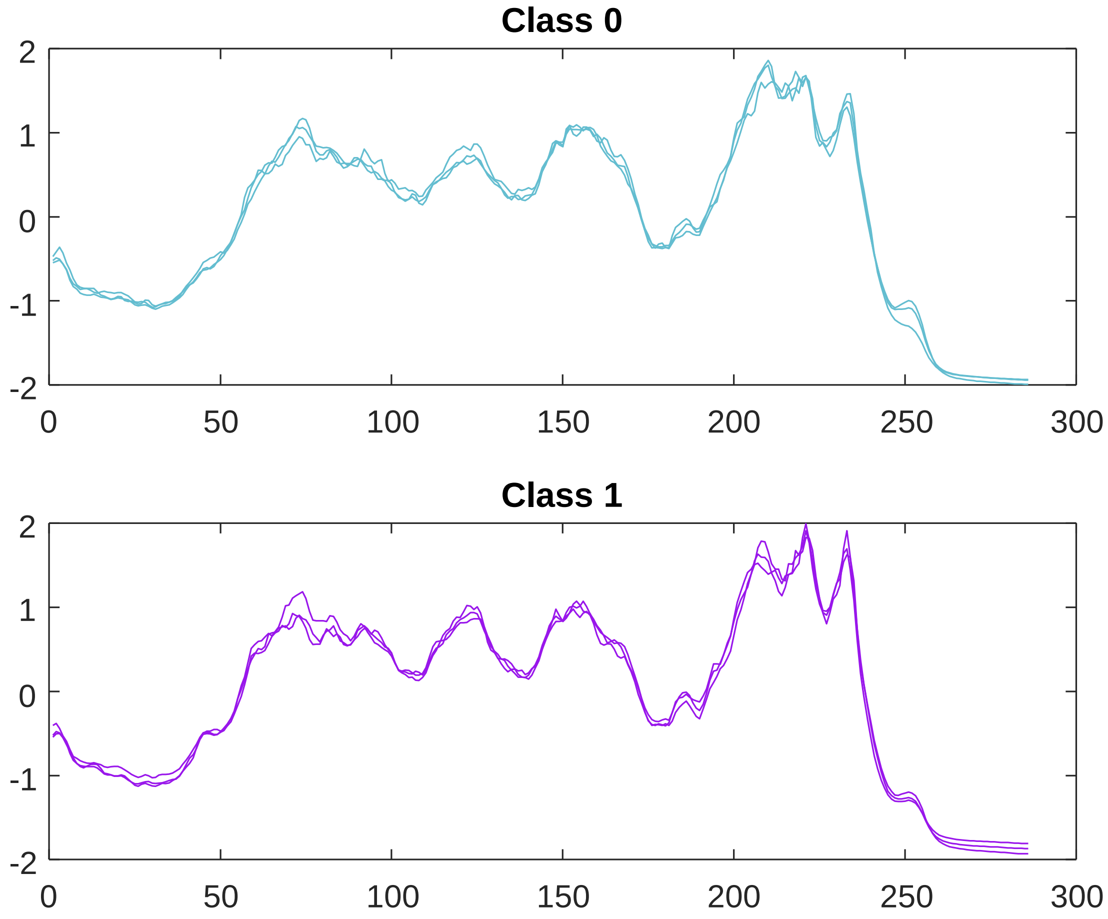
CinCECGTorso

Three exemplars per class, with z-normalization



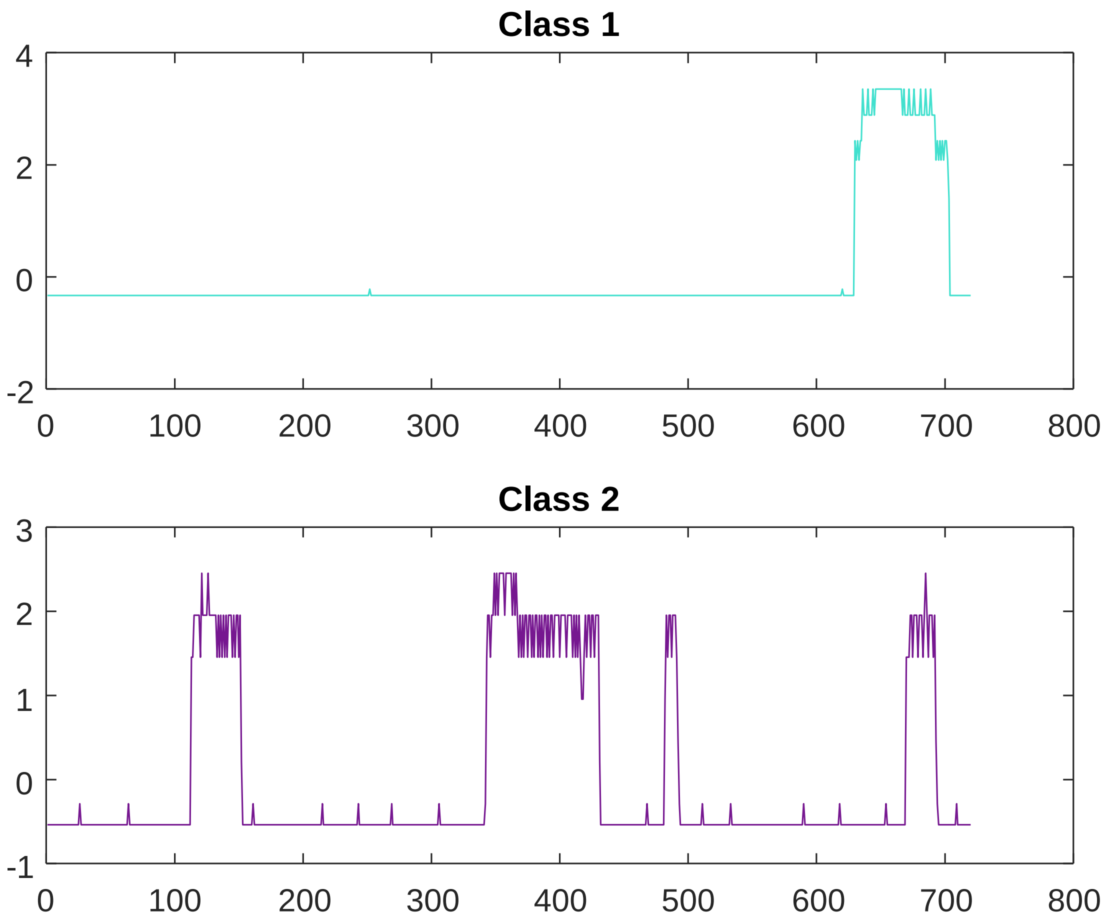
Coffee

Three exemplars per class,
with z-normalization



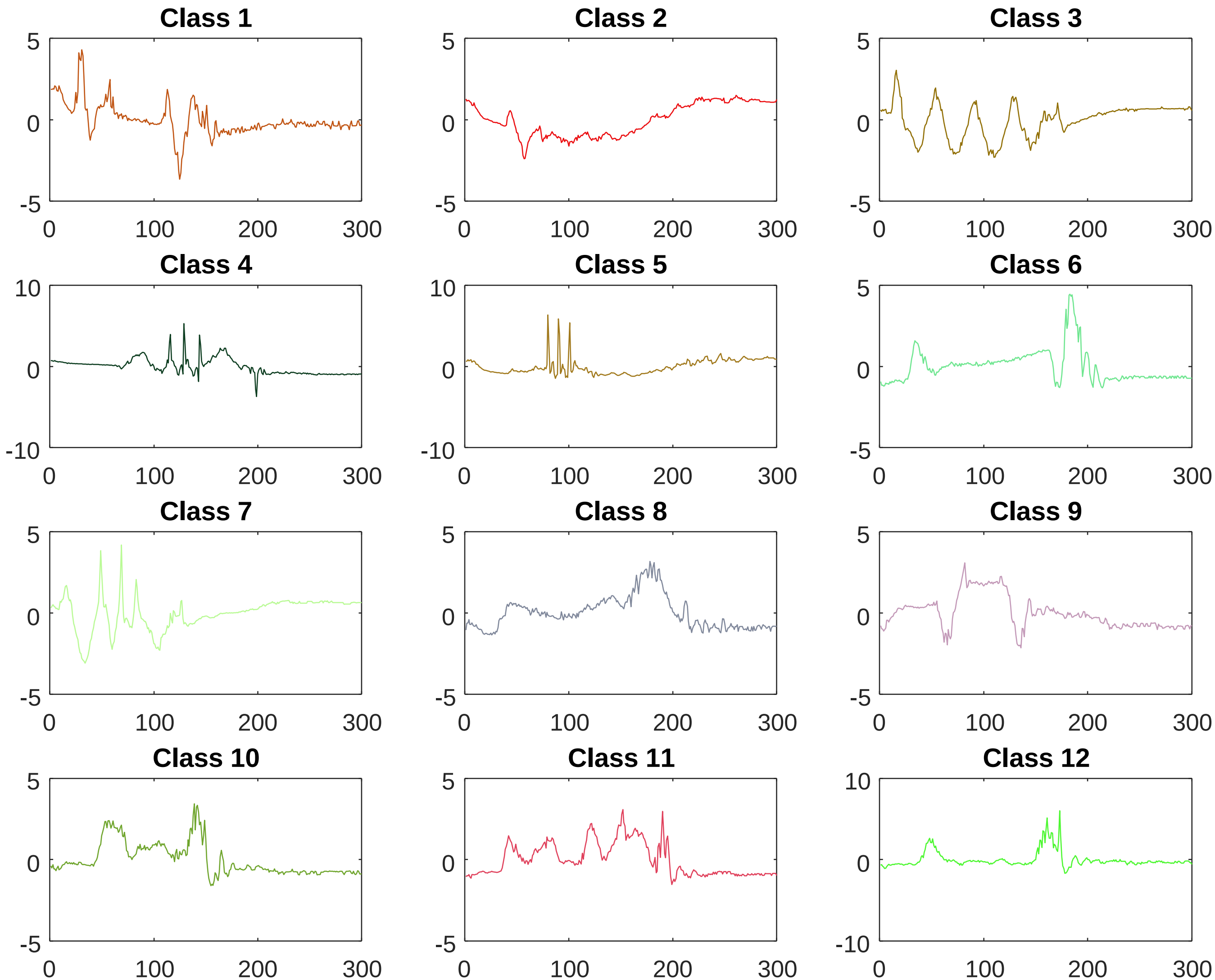
Computers

One exemplar per class,
with z-normalization



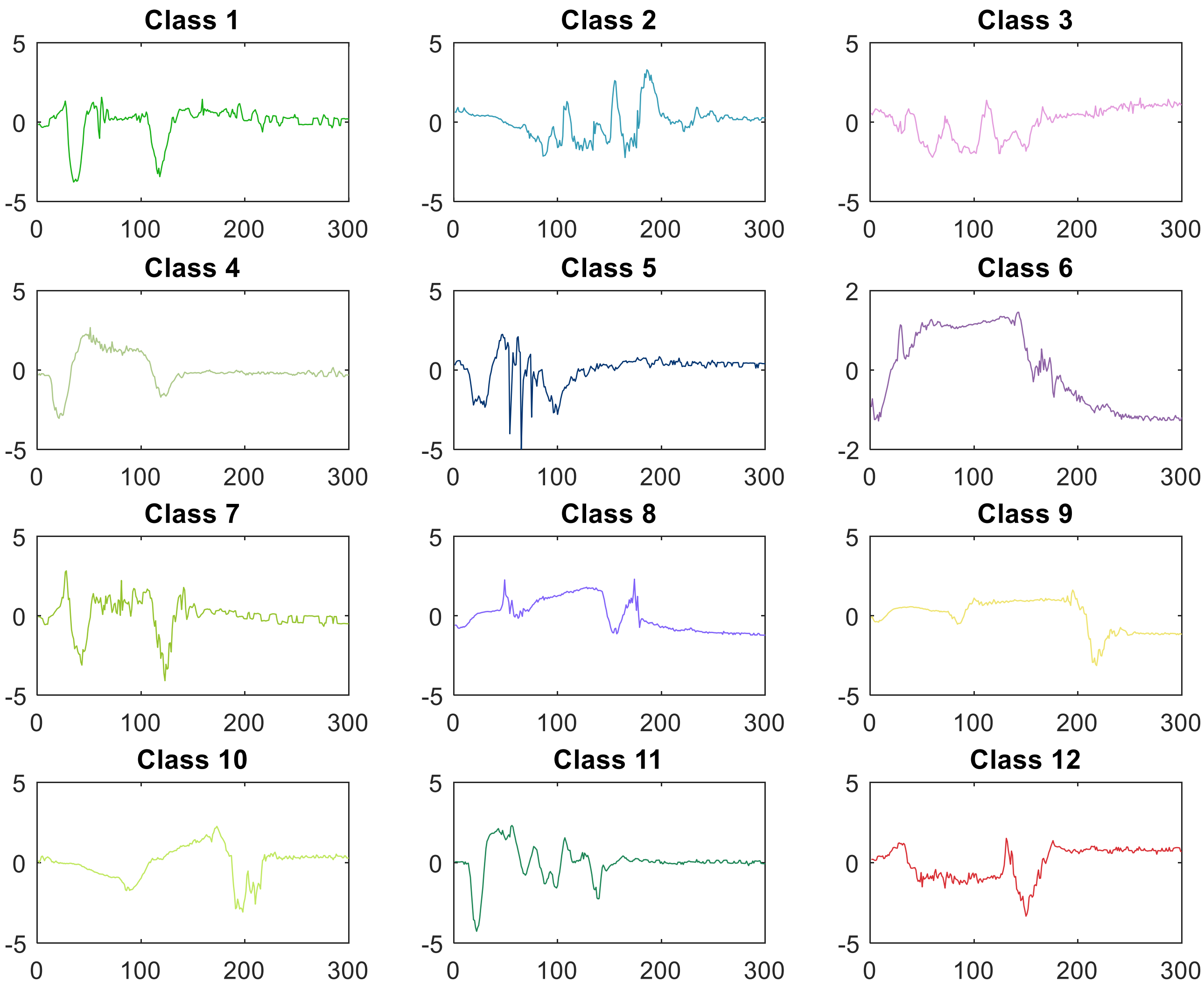
CricketX

One exemplar per class, with z-normalization



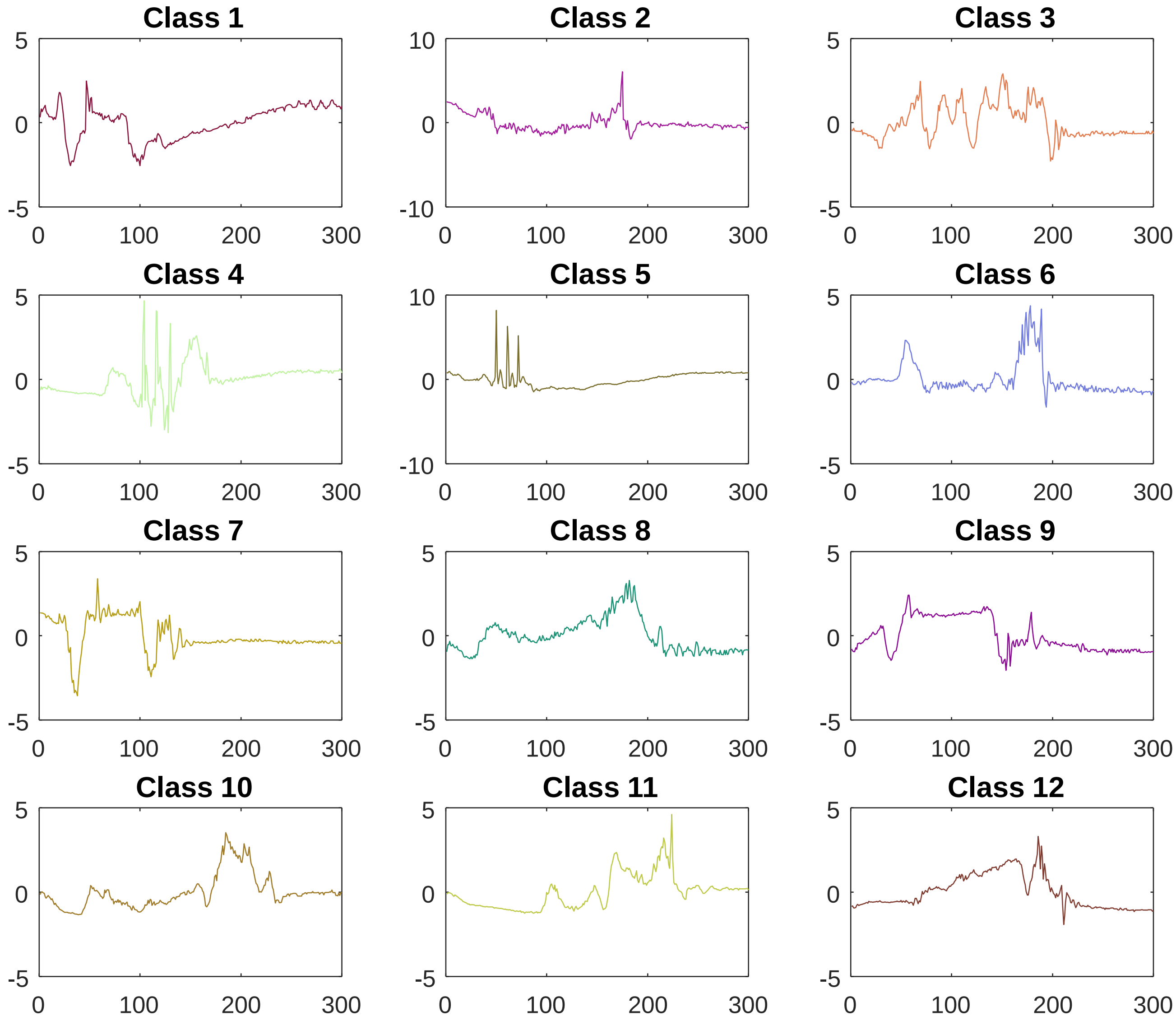
CricketY

One exemplar per class,
with z-normalization



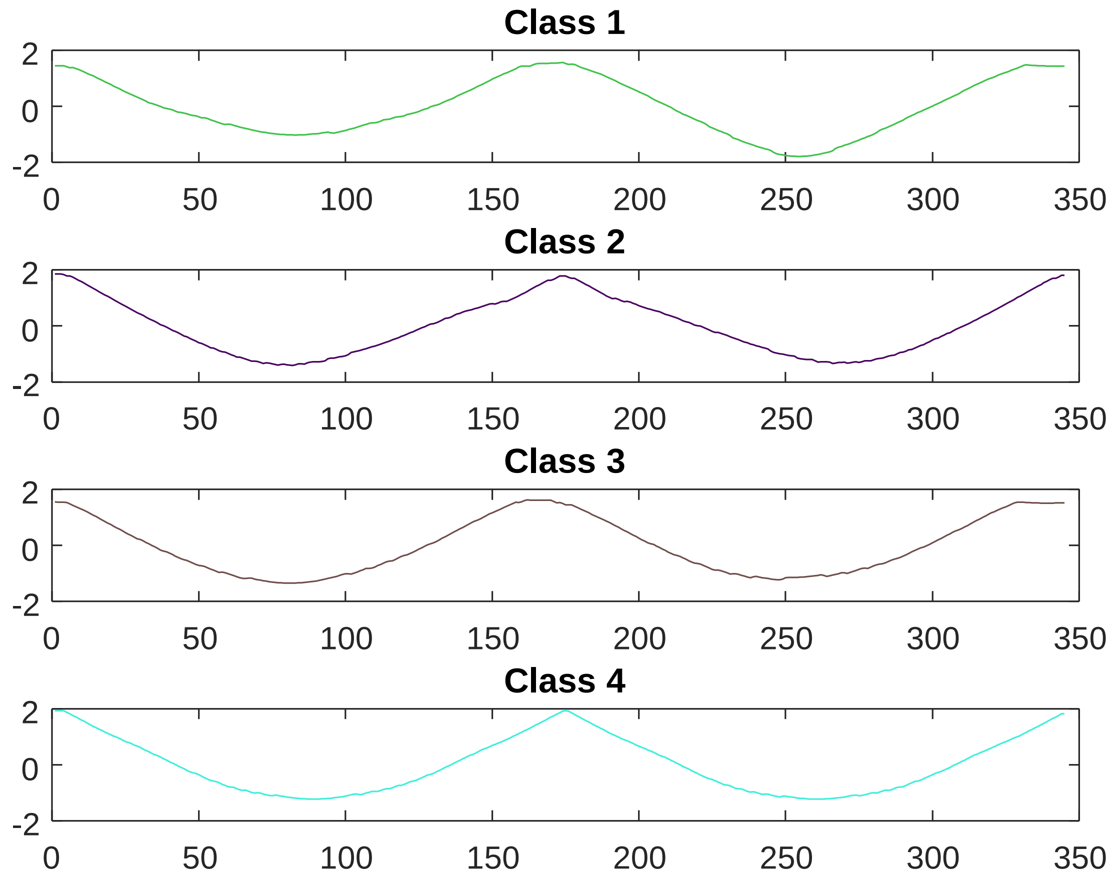
CricketZ

One exemplar per class,
with z-normalization



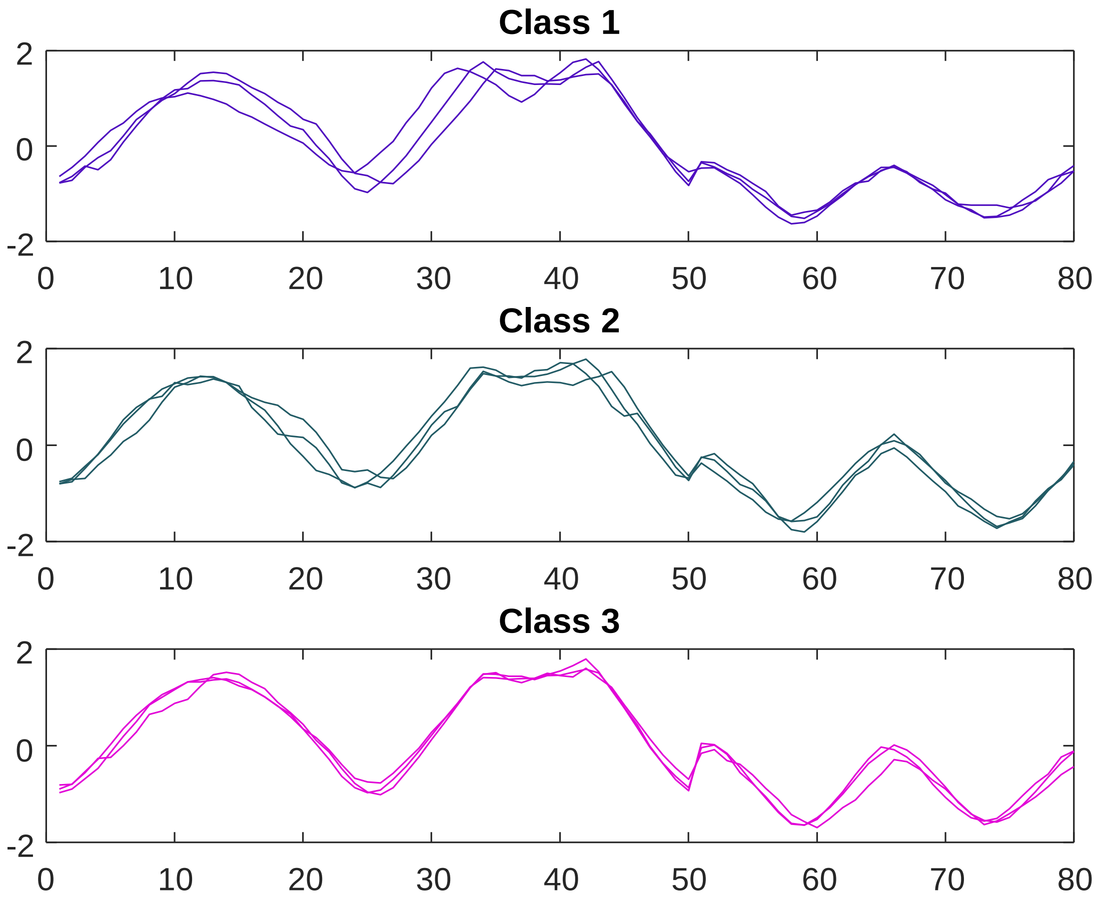
DiatomSizeReduction

One exemplar per class,
with z-normalization



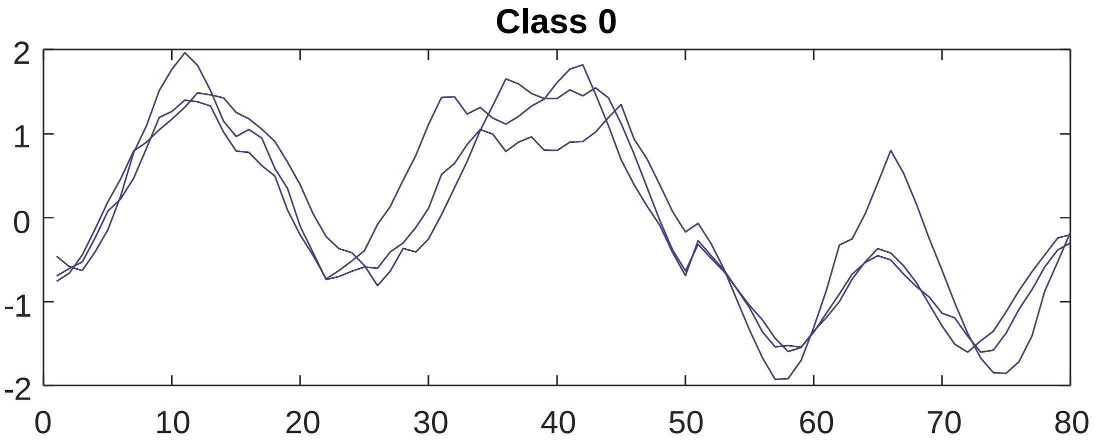
DistalPhalanxOutlineAgeGroup

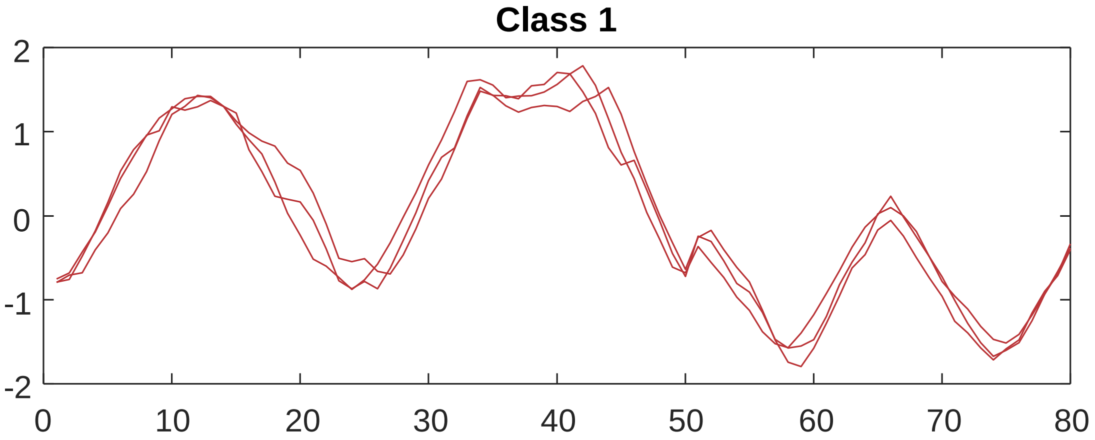
Three exemplars per class,
with z-normalization



DistalPhalanxOutlineCorrect

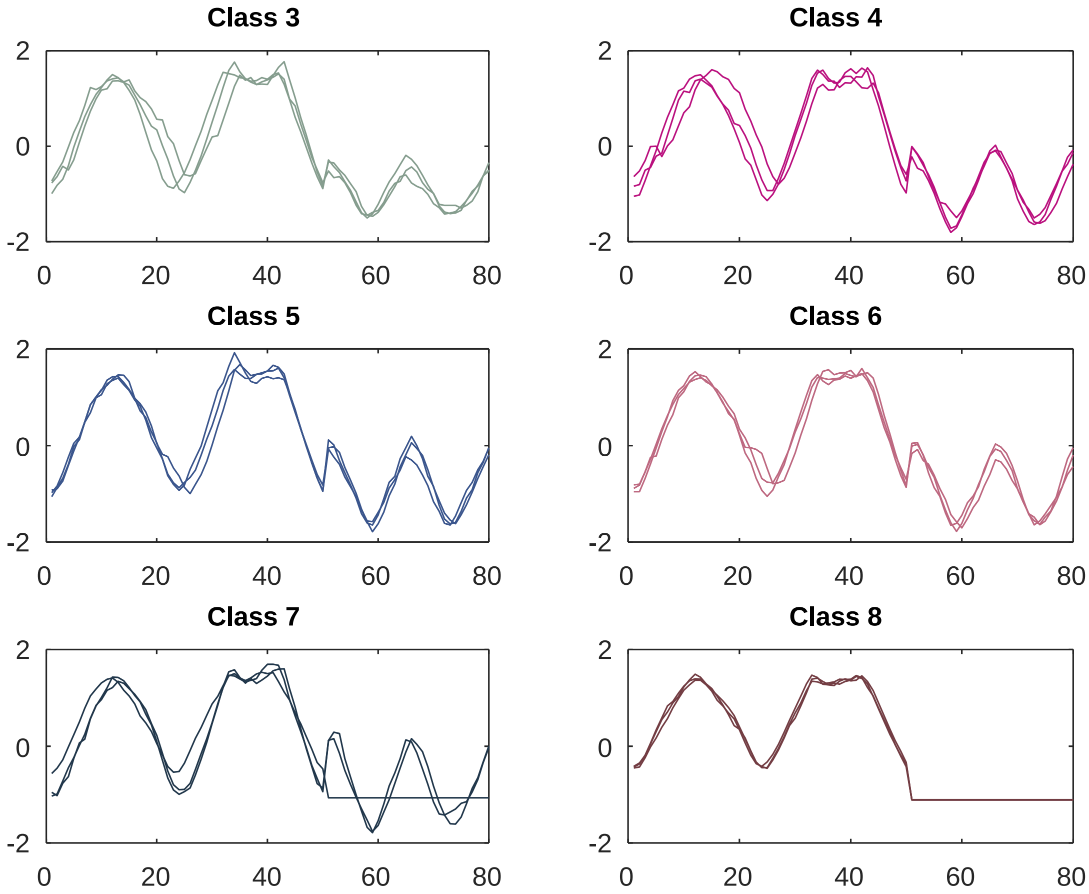
Three exemplars per class,
with z-normalization





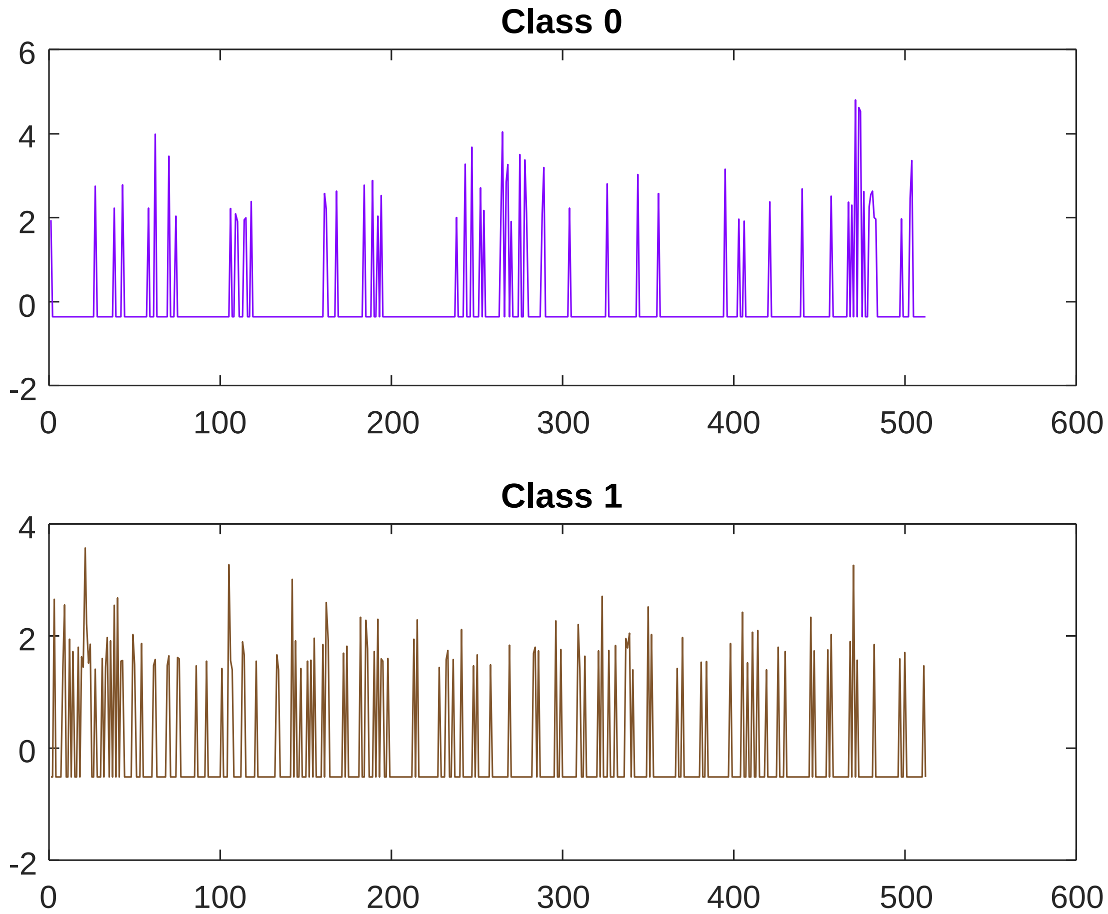
DistalPhalanxTW

Three exemplars per class,
with z-normalization



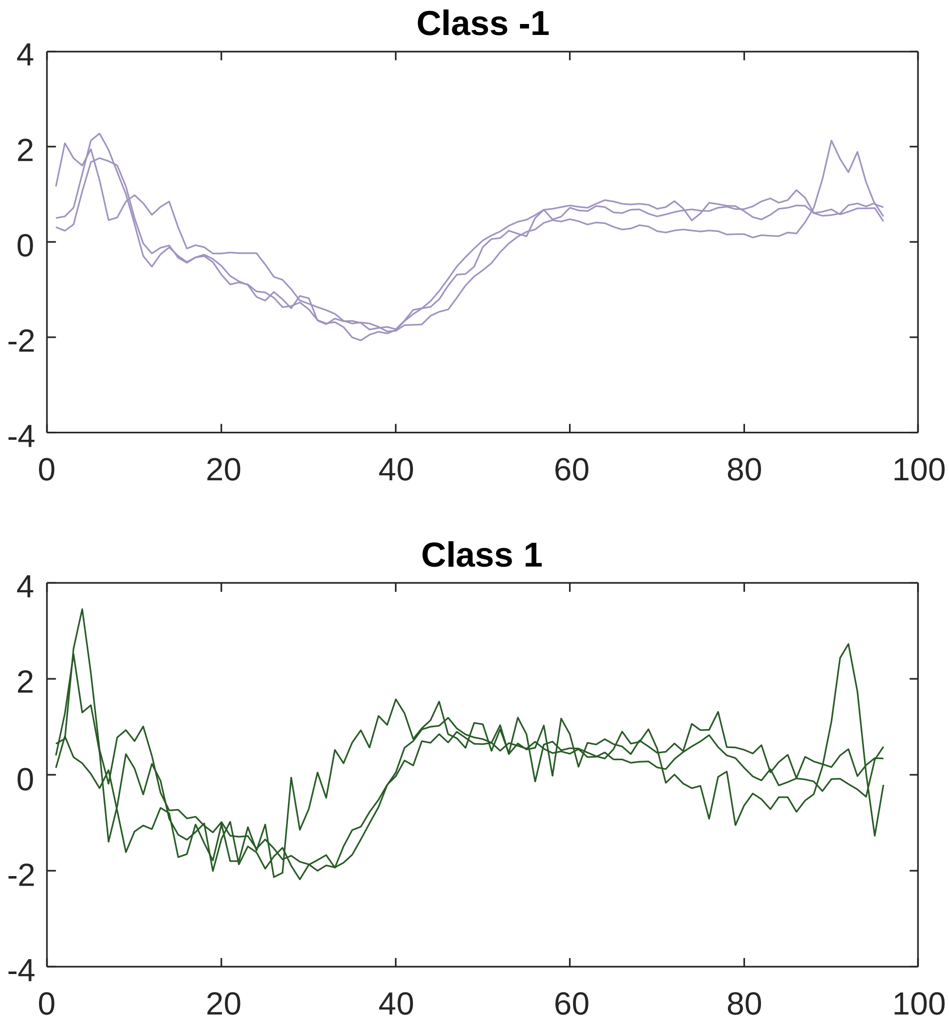
Earthquakes

One exemplar per class,
with z-normalization



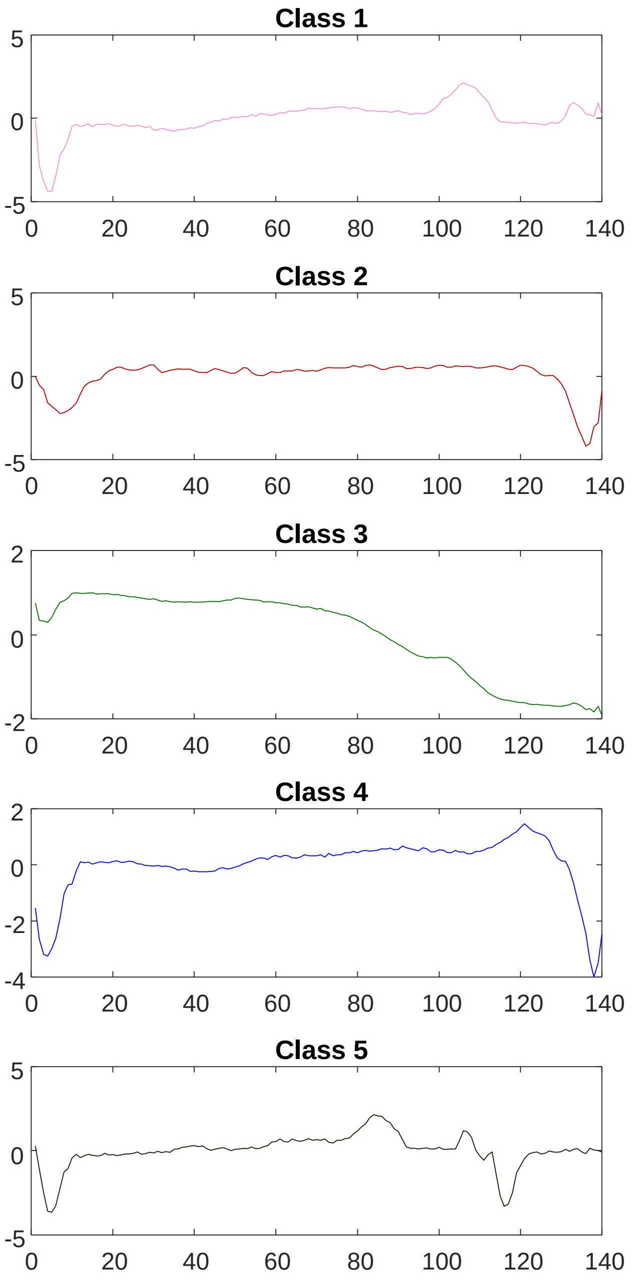
ECG200

Three exemplars per class,
with z-normalization



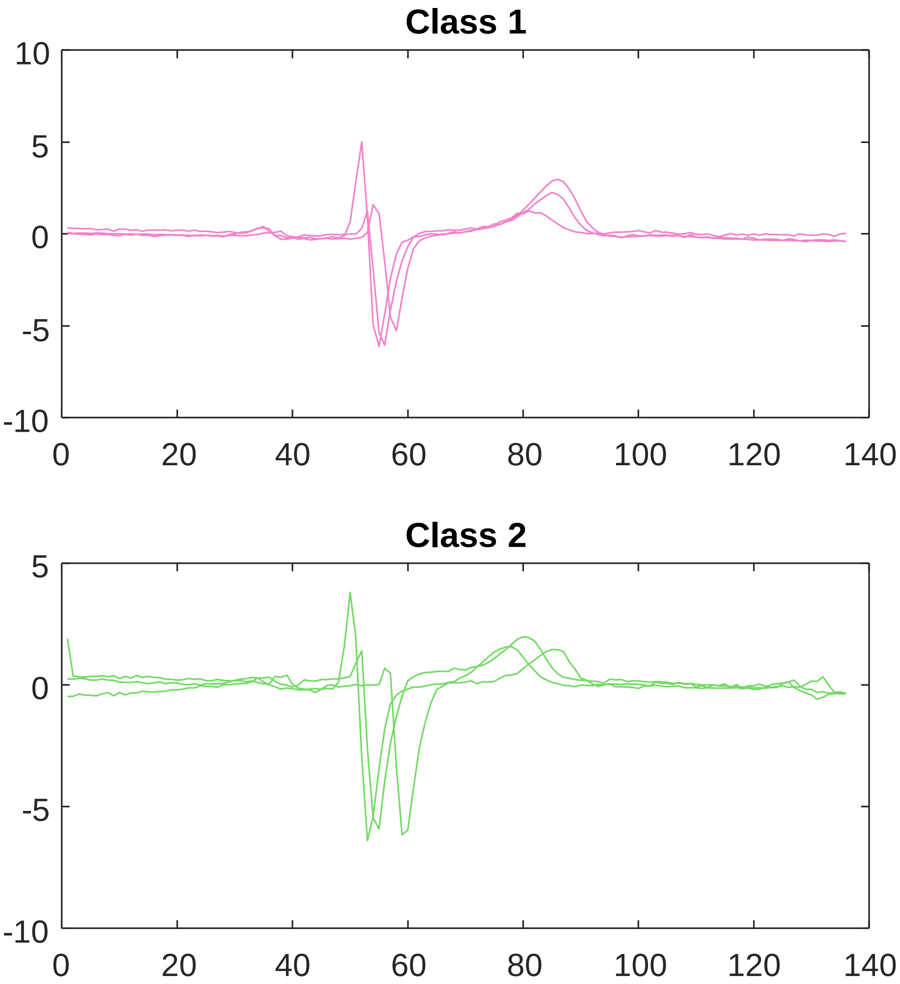
ECG5000

One exemplar per class,
with z-normalization



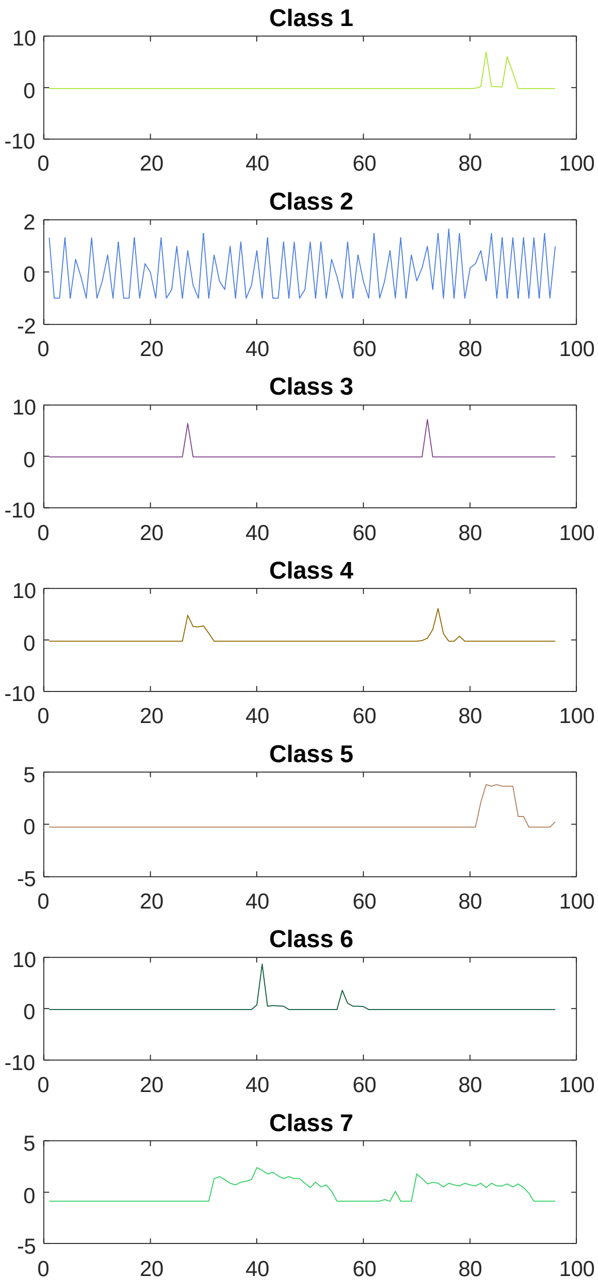
ECGFiveDays

Three exemplars per class,
with z-normalization



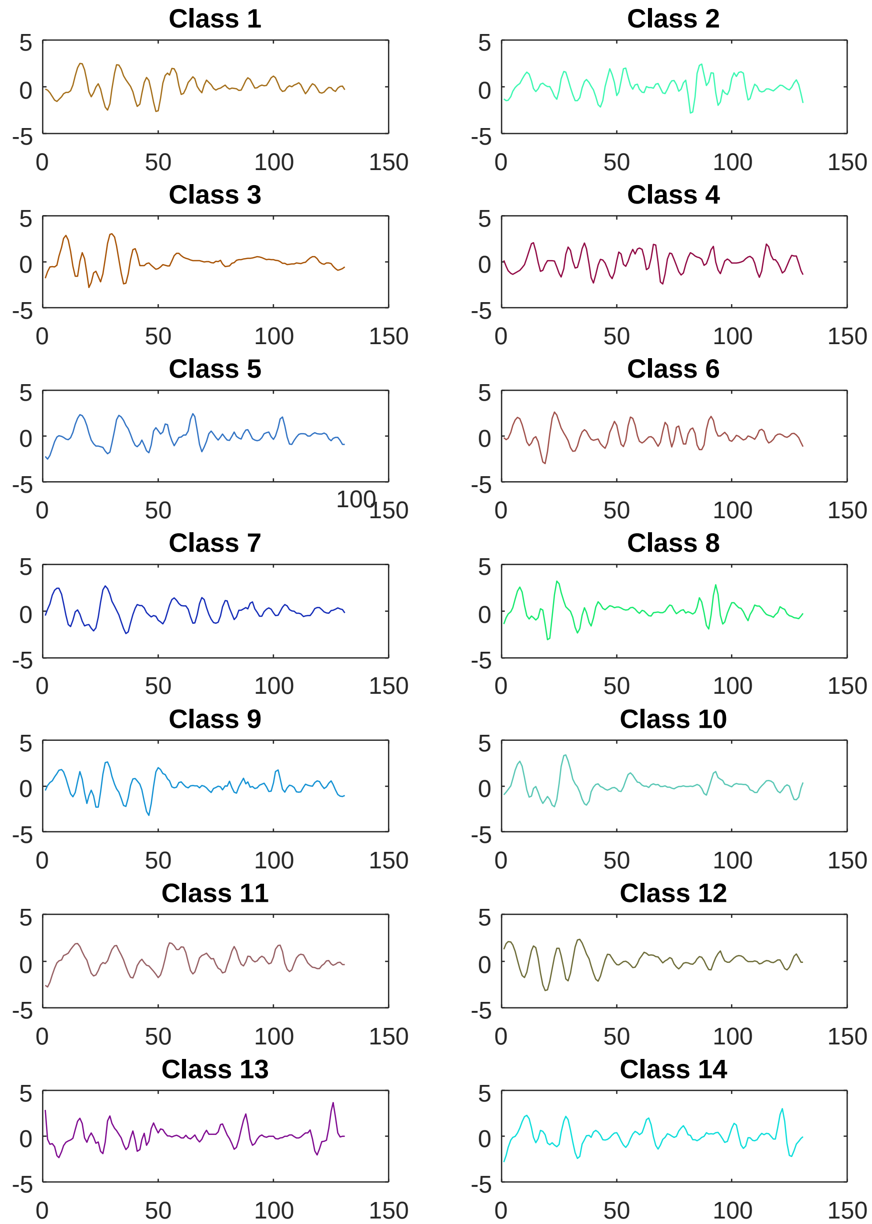
ElectricDevices

One exemplar per class, with z-normalization



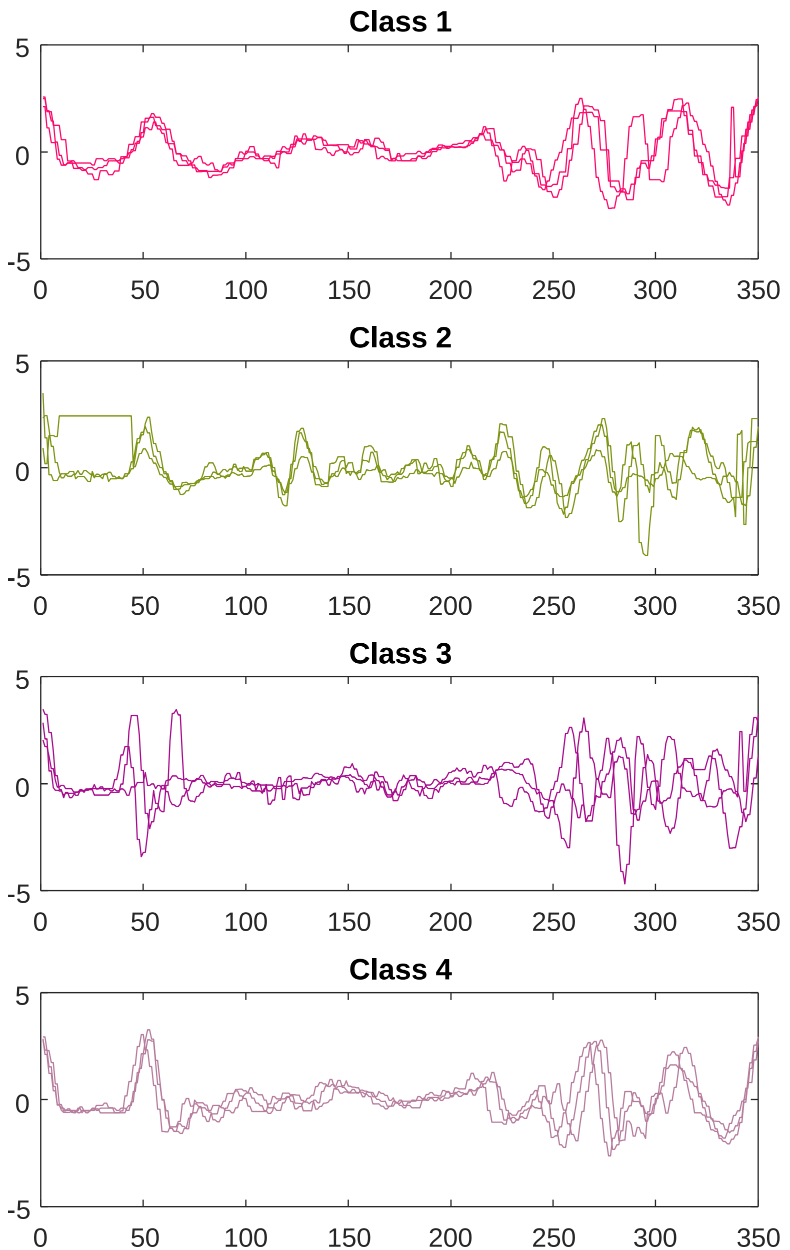
FaceAll

One exemplar per class,
with z-normalization



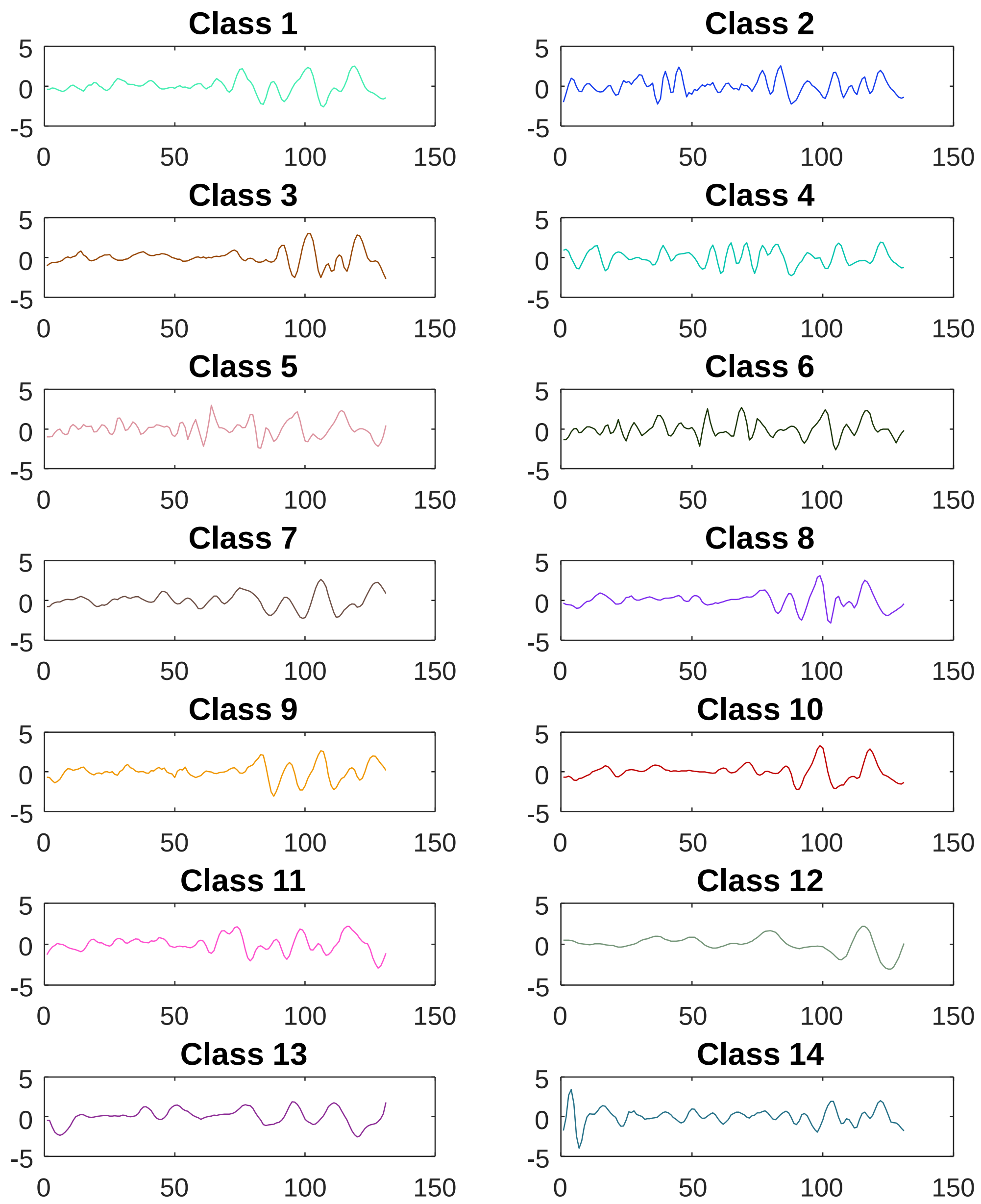
FaceFour

Three exemplars per class,
with z-normalization



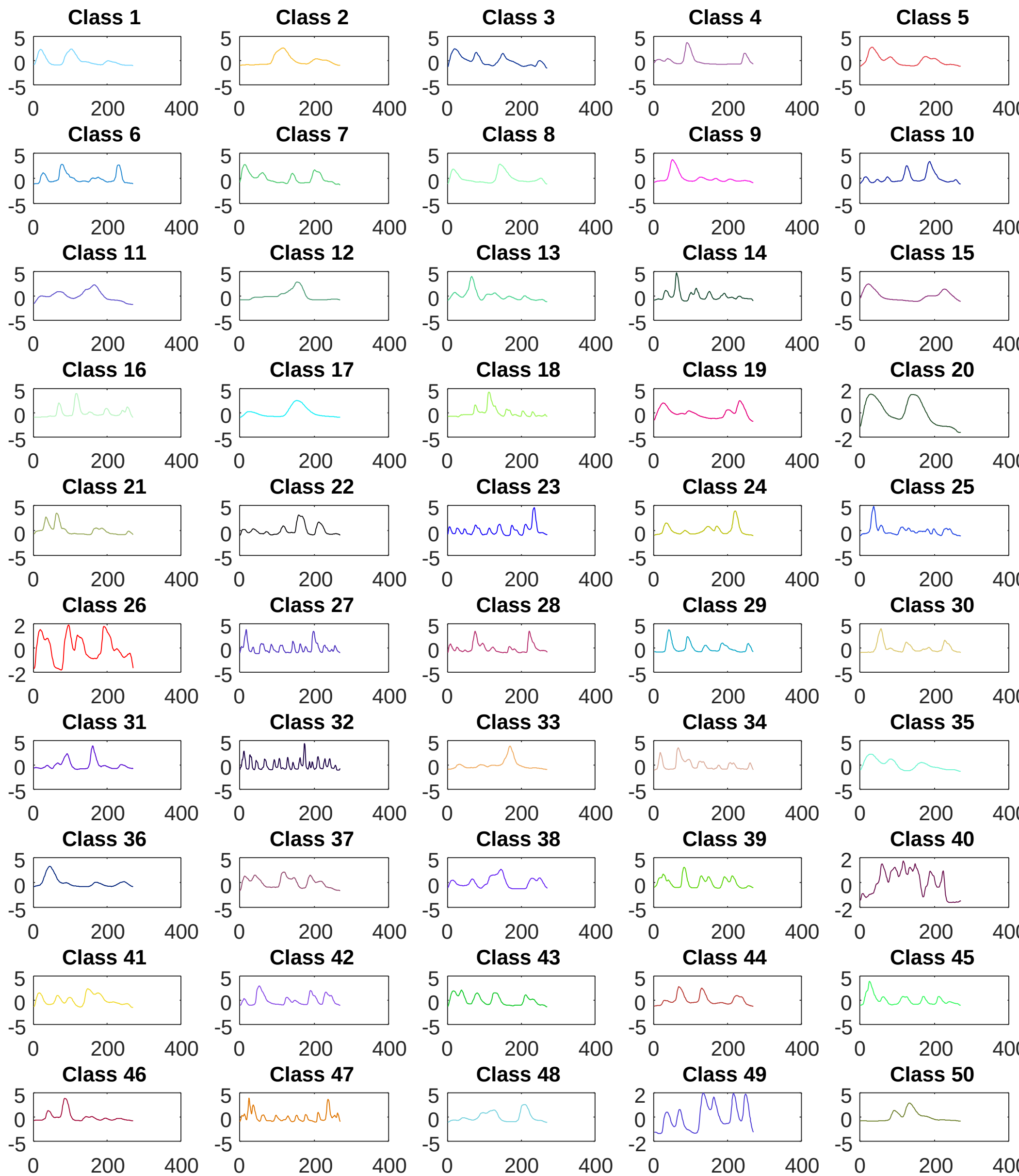
FacesUCR

One exemplar per class,
with z-normalization



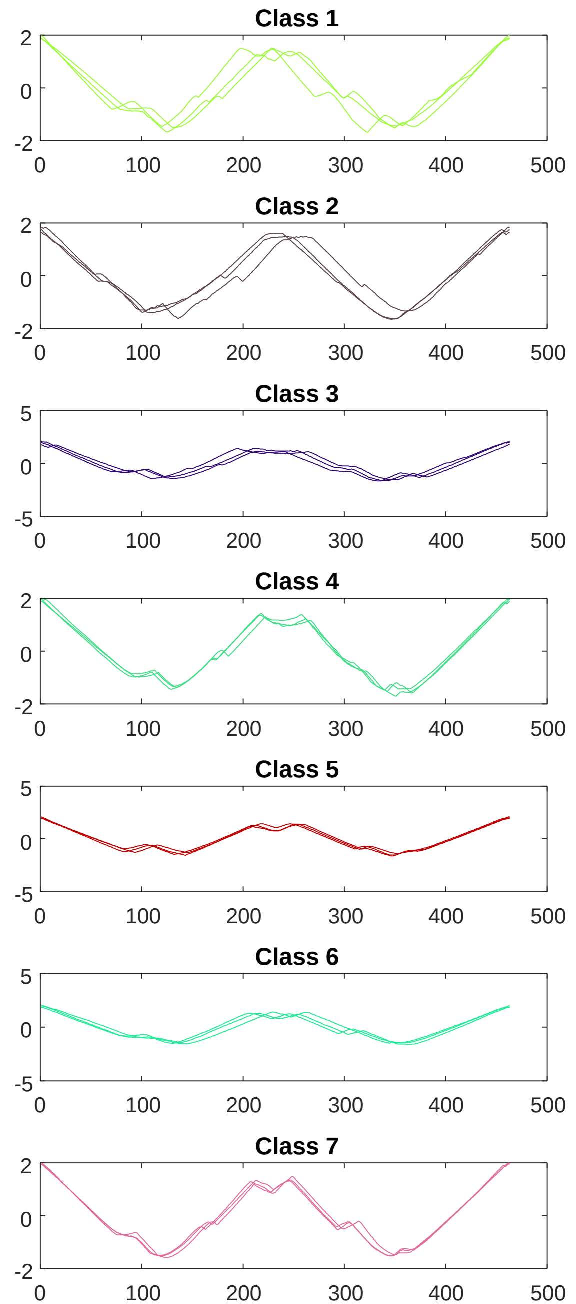
FiftyWords

One exemplar per class,
with z-normalization



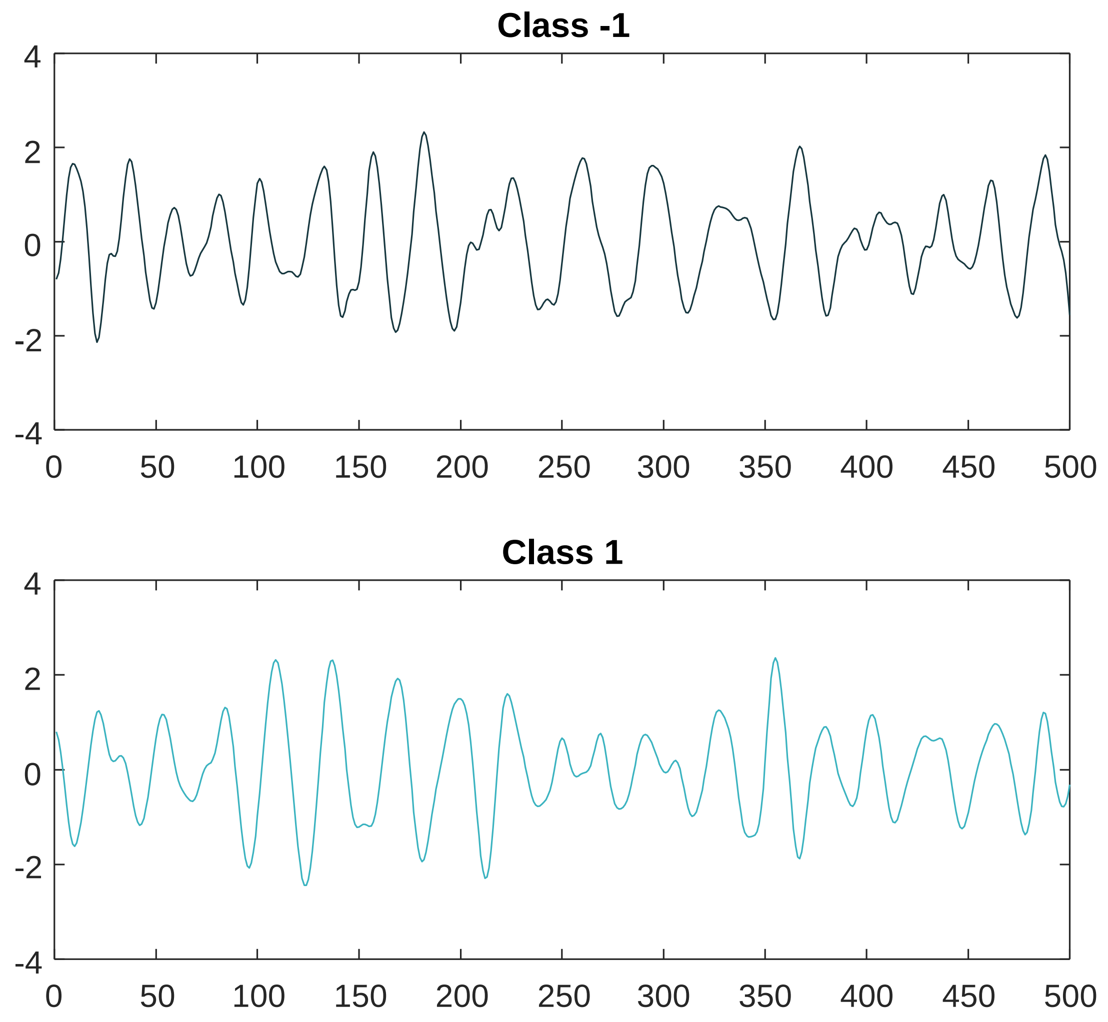
Fish

Three exemplars per class,
with z-normalization



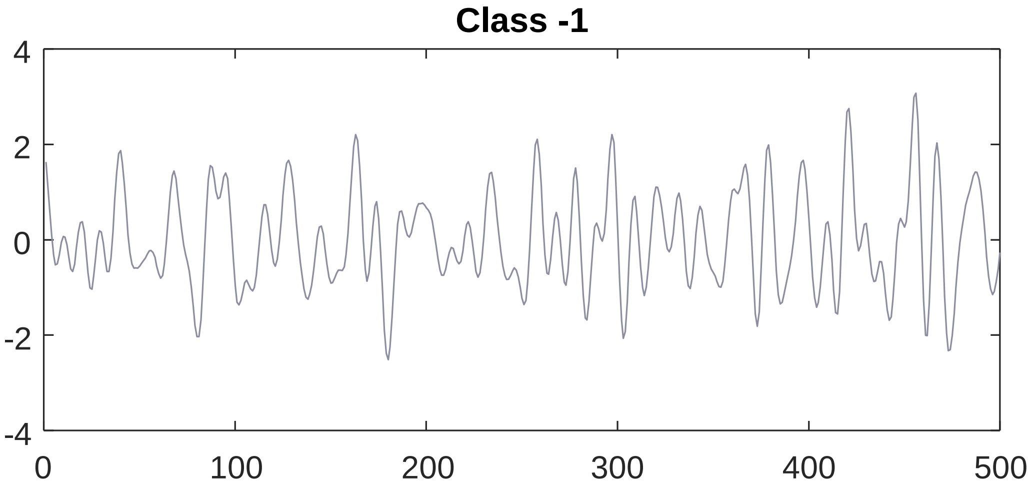
FordA

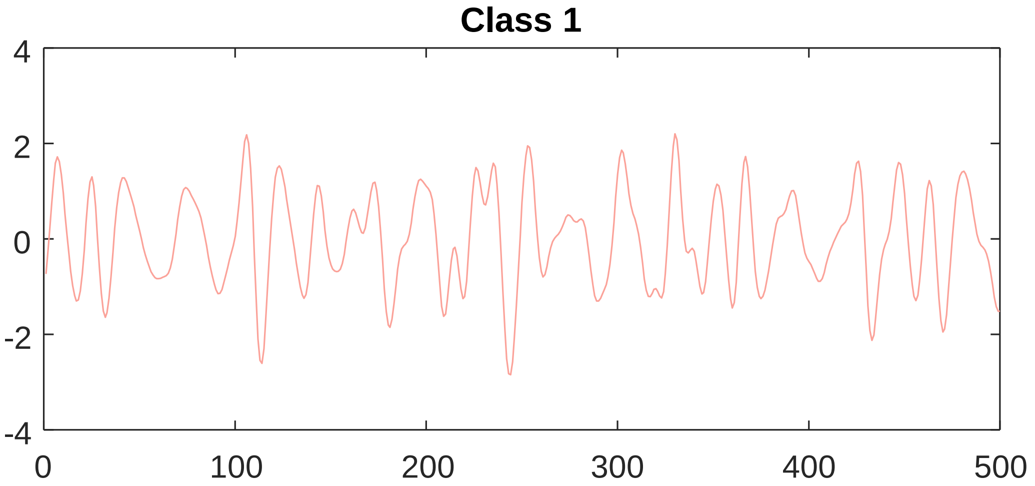
One exemplar per class,
with z-normalization



FordB

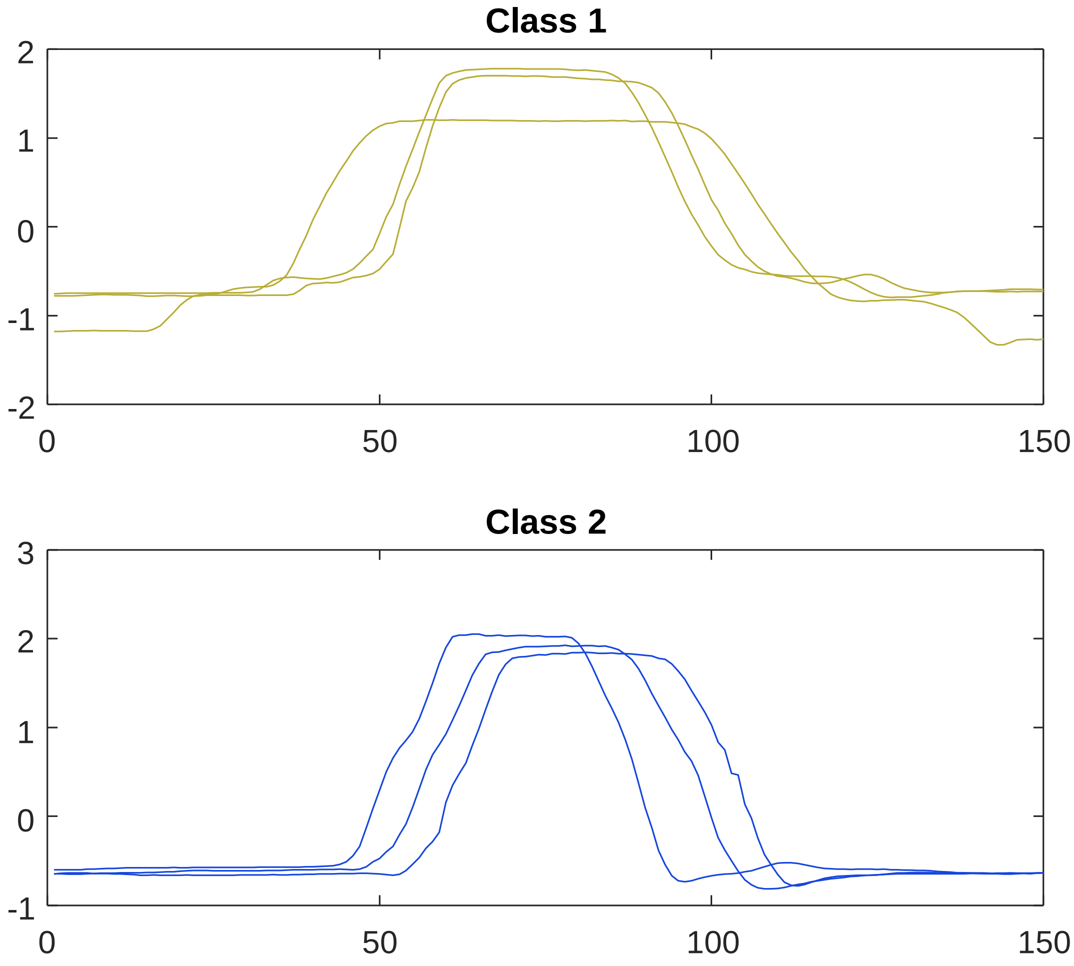
One exemplar per class,
with z-normalization





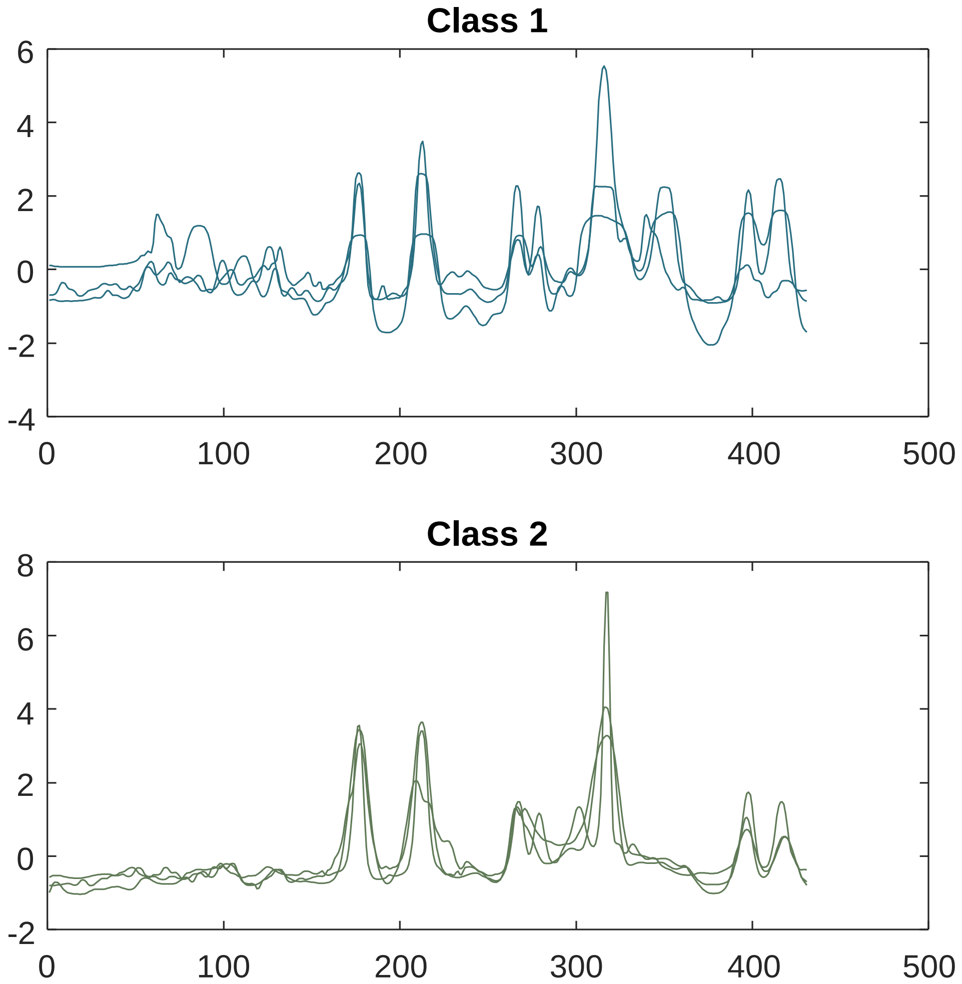
GunPoint

Three exemplars per class,
with z-normalization



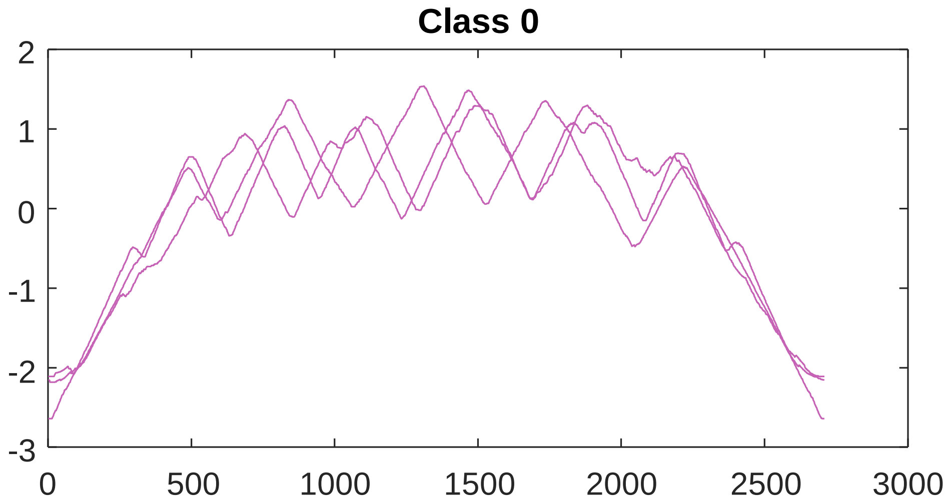
Ham

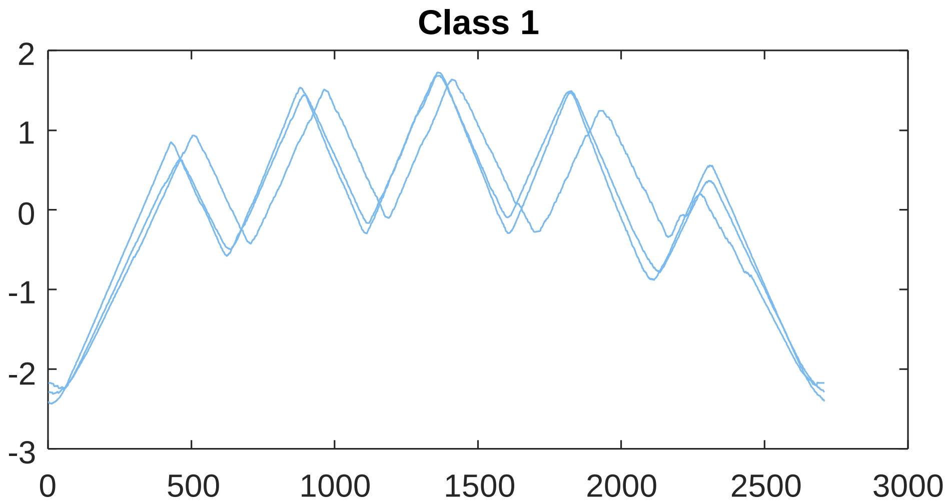
Three exemplars per class,
with z-normalization



HandOutlines

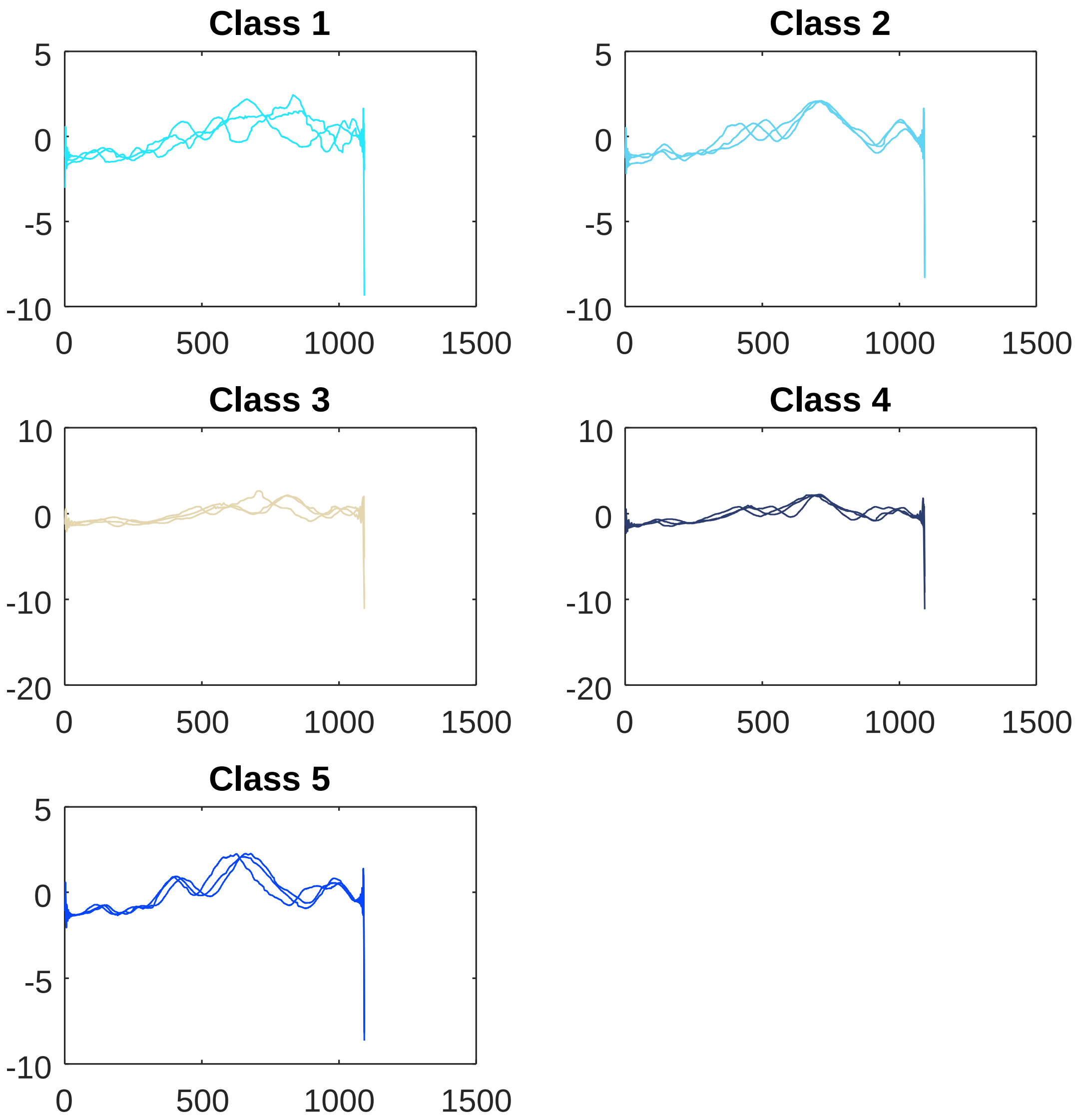
Three exemplars per class,
with z-normalization





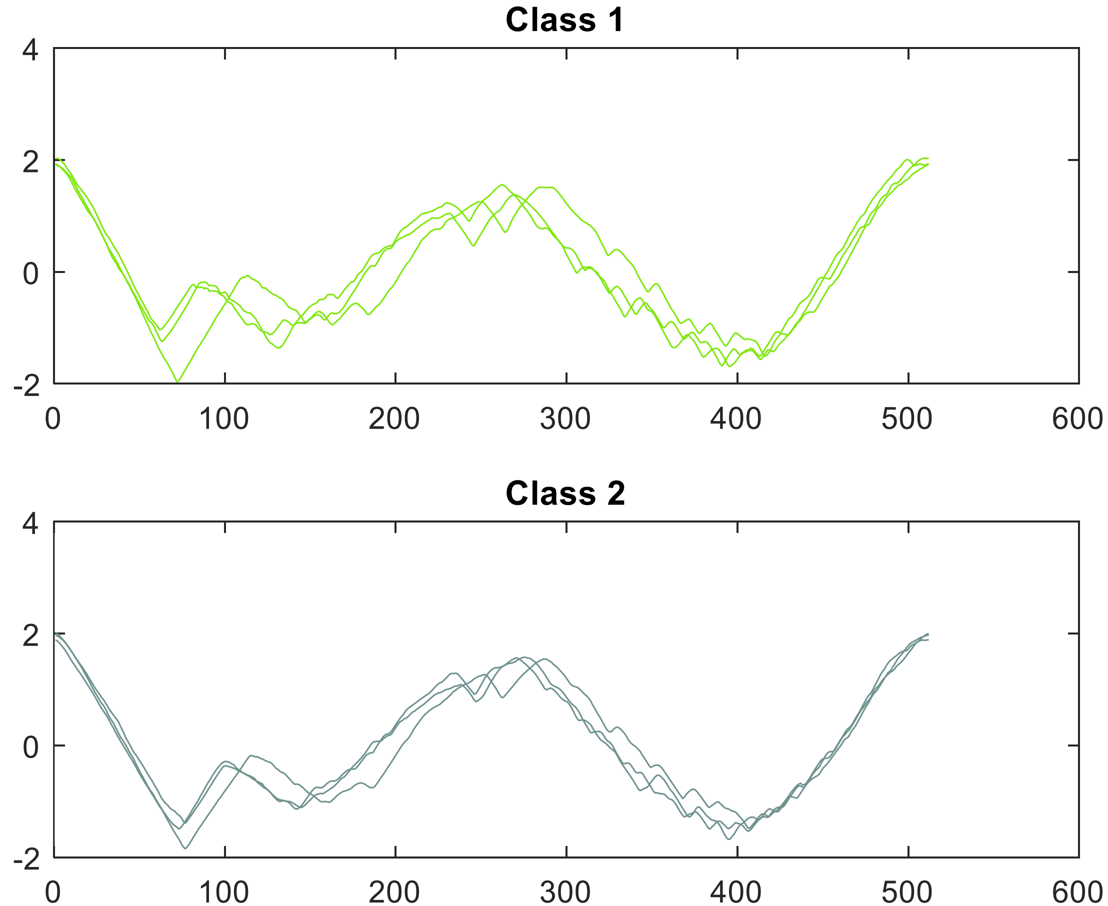
Haptics

Three exemplars per class,
with z-normalization



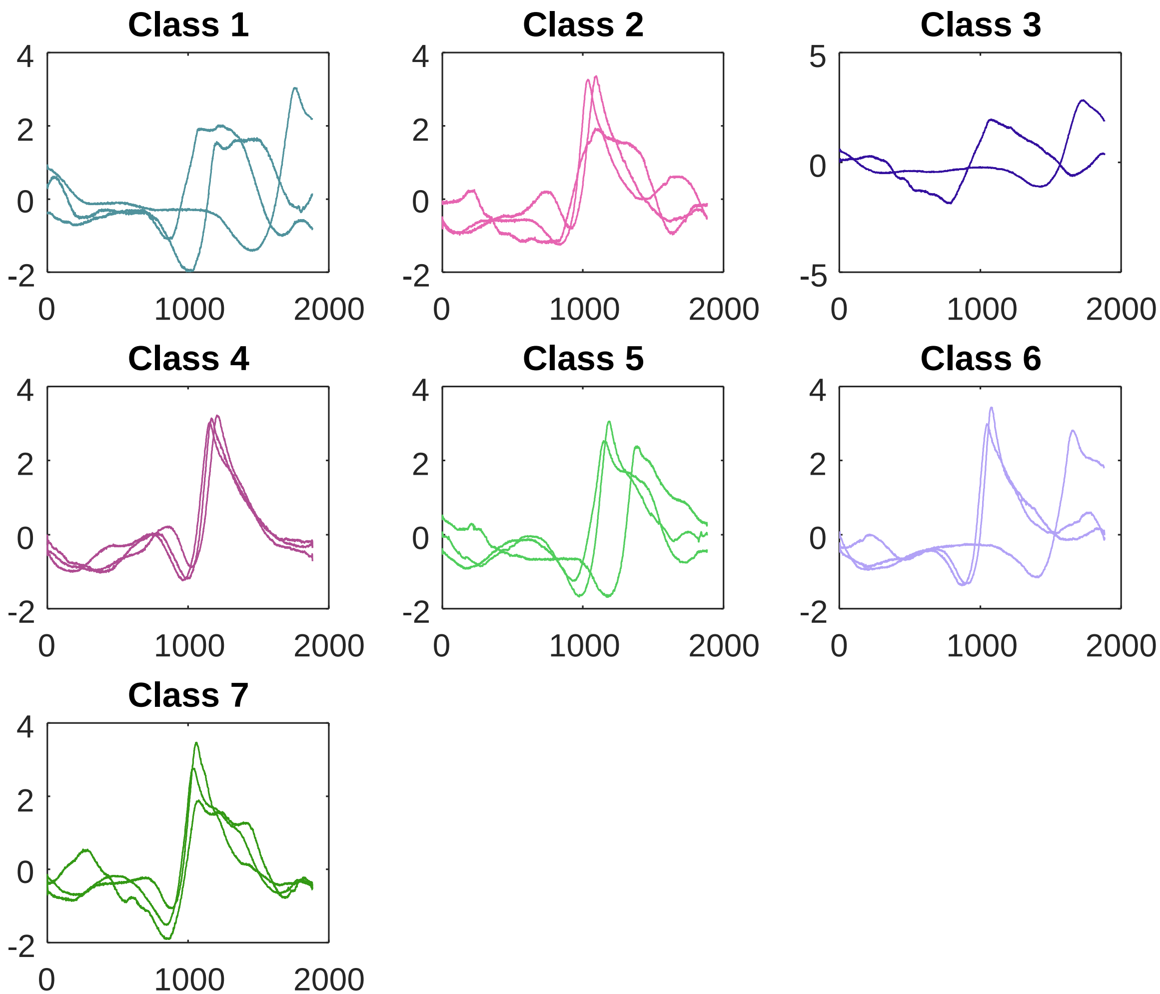
Herring

Three exemplars per class,
with z-normalization



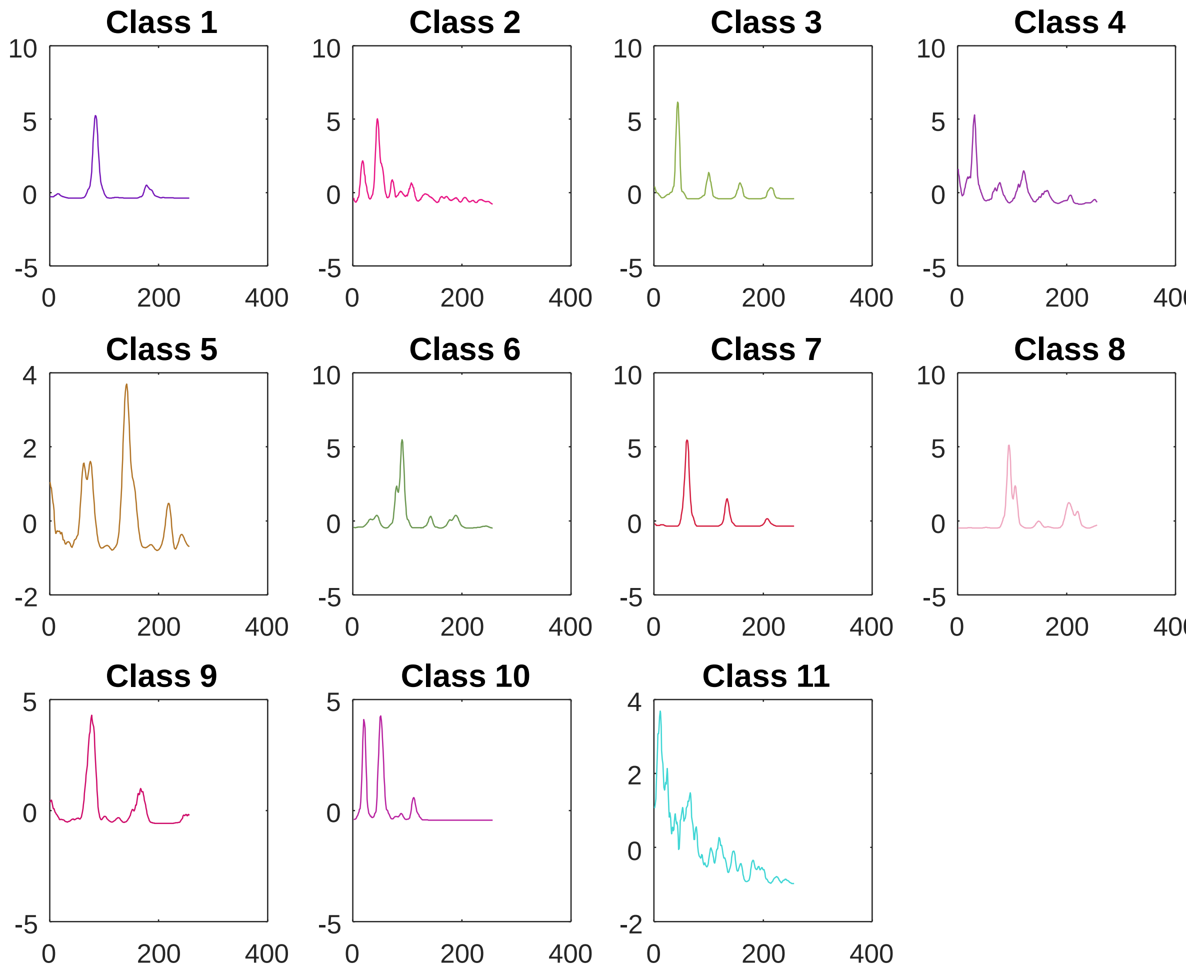
InlineSkate

Three exemplars per class,
with z-normalization



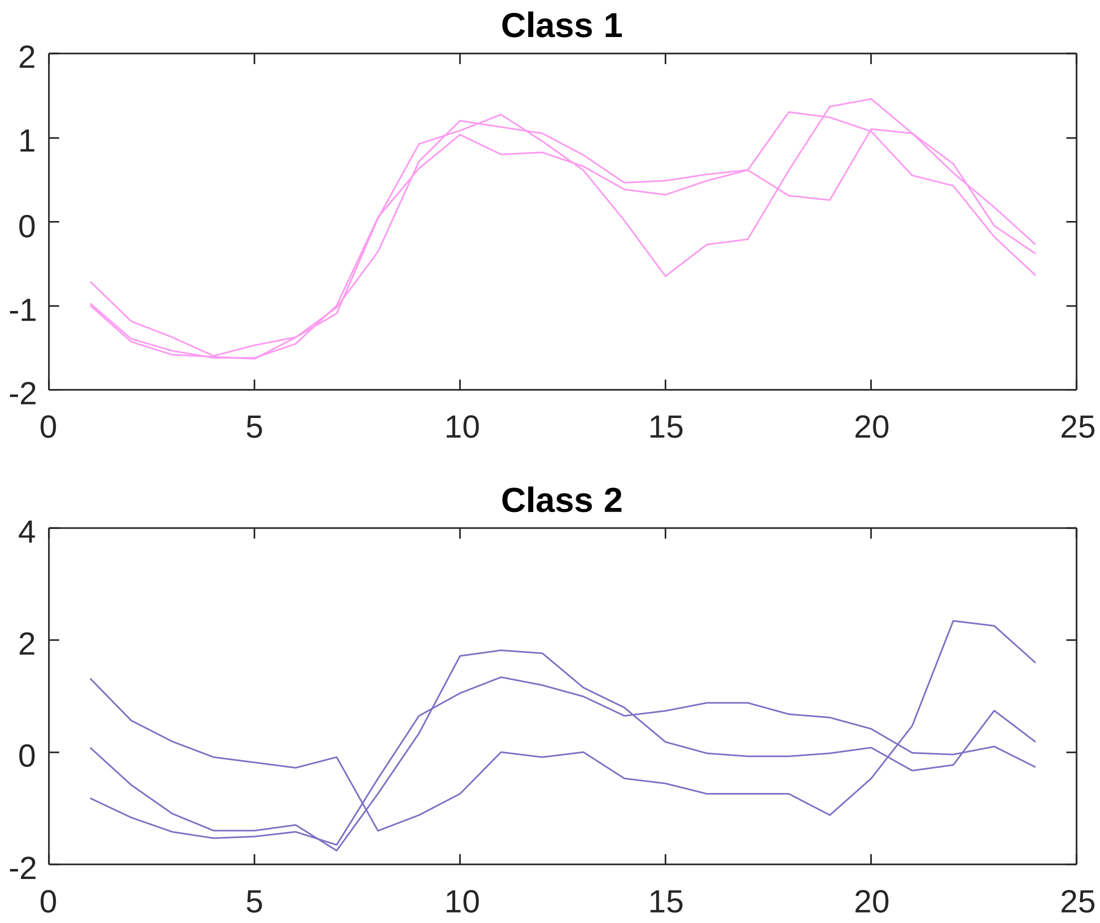
InsectWingbeatSound

One exemplar per class,
with z-normalization



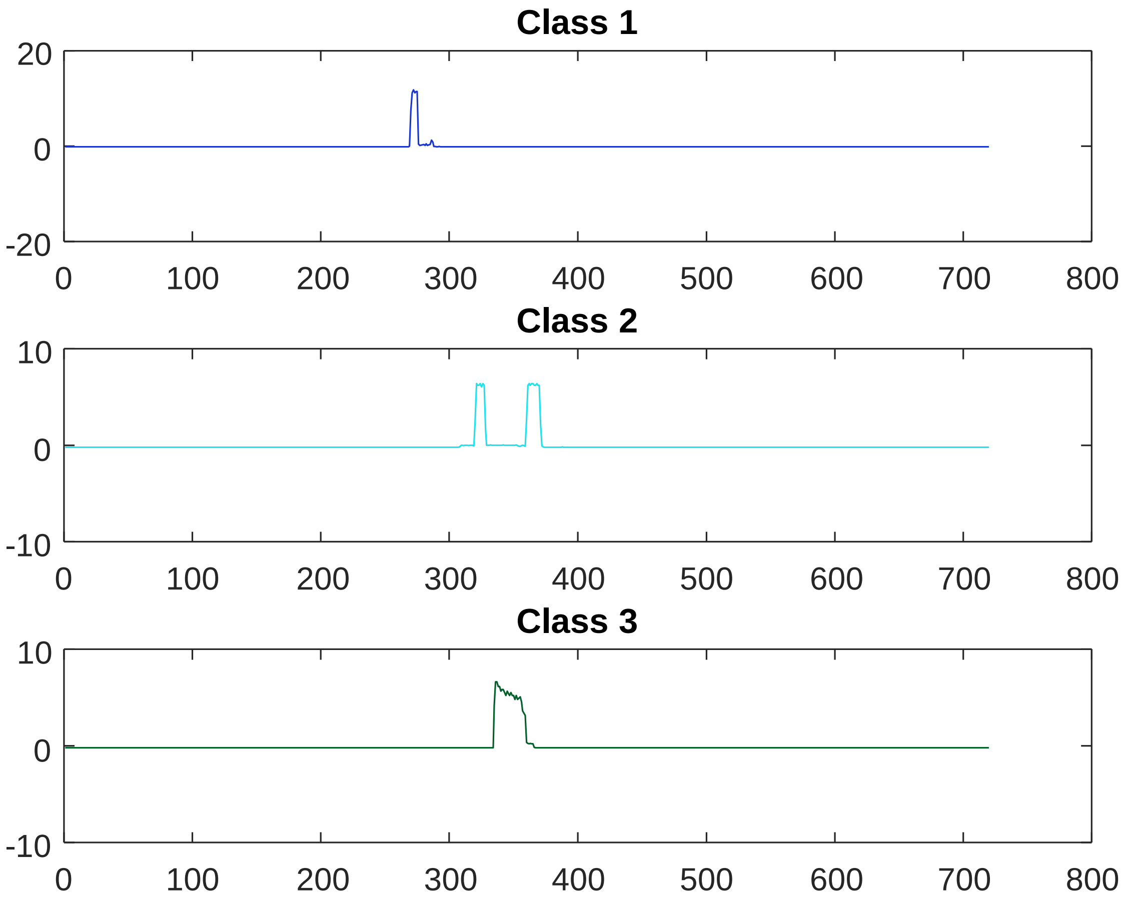
ItalyPowerDemand

Three exemplars per class,
with z-normalization



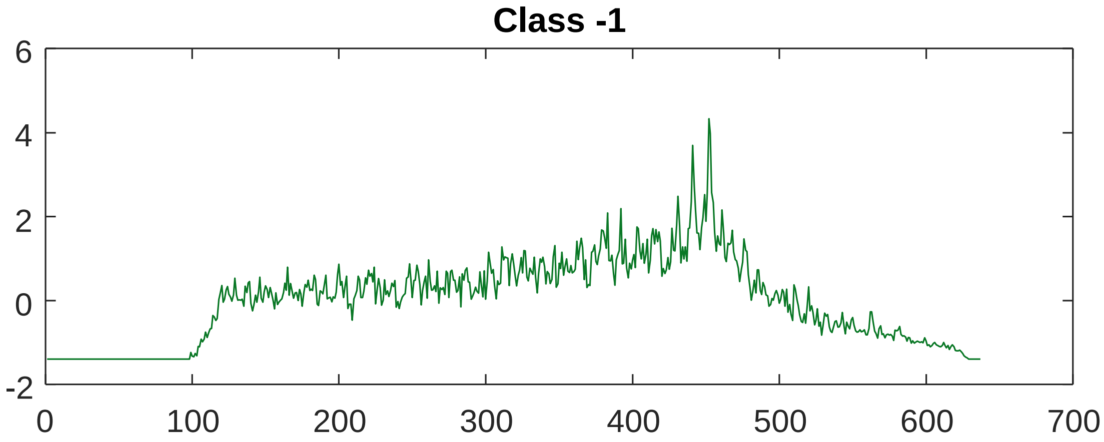
LargeKitchenAppliances

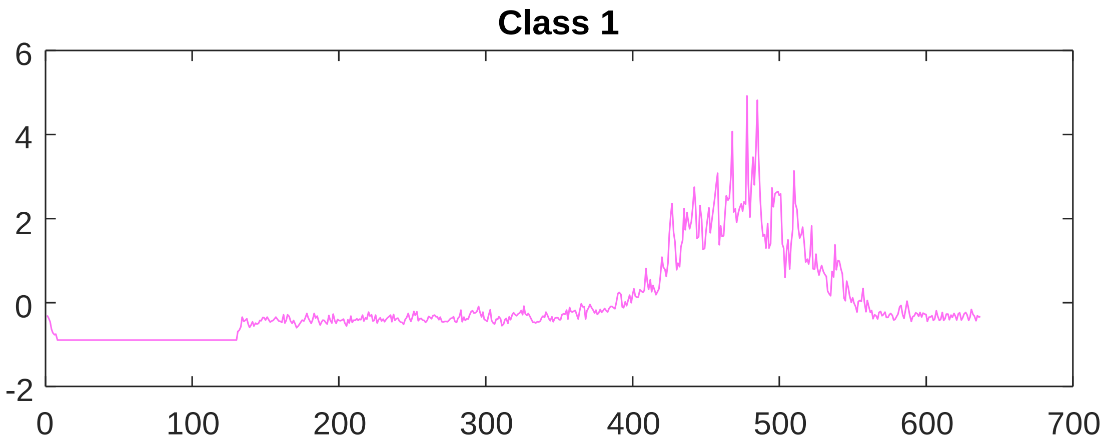
One exemplar per class,
with z-normalization



Lightning2

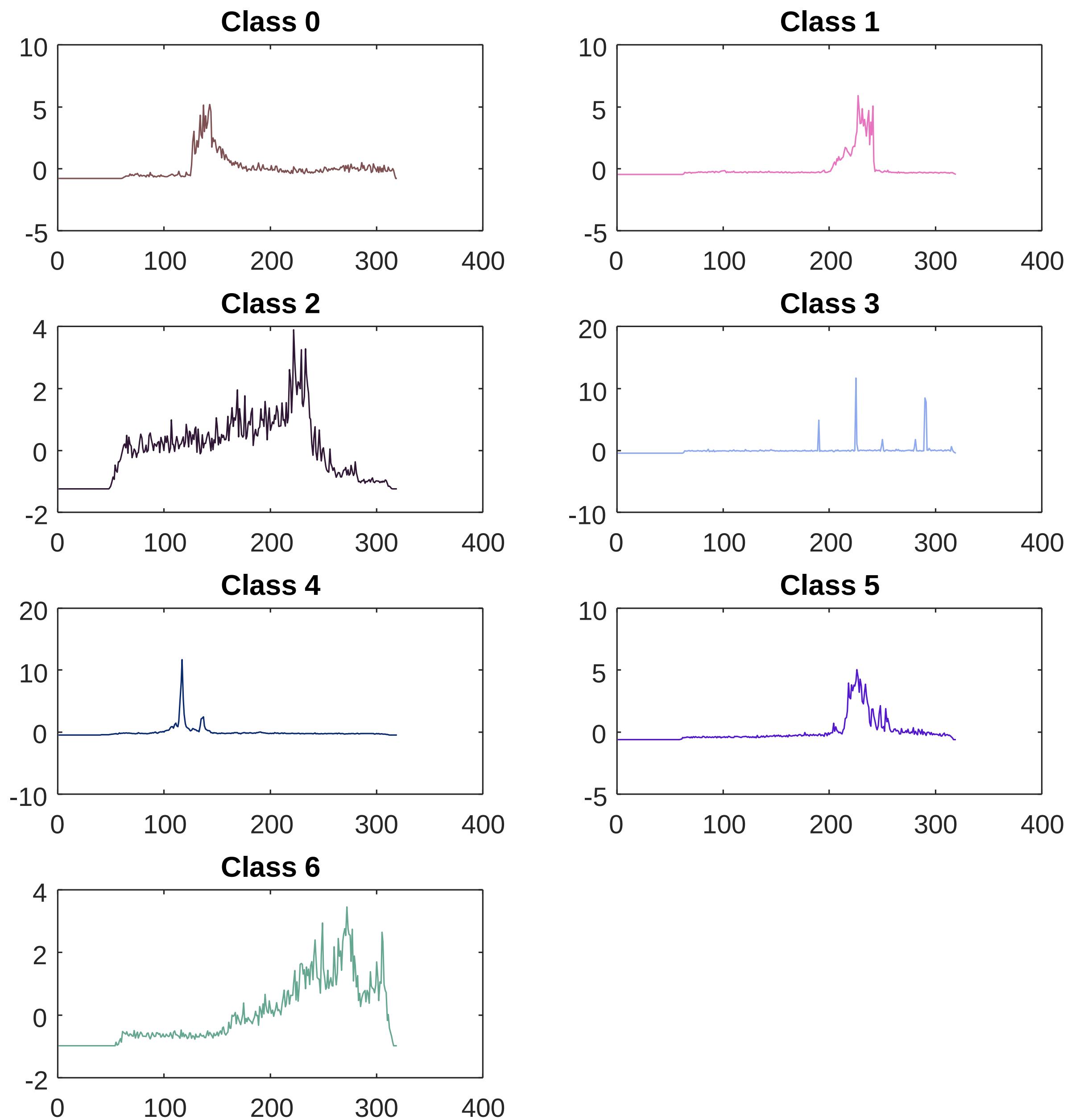
One exemplar per class,
with z-normalization





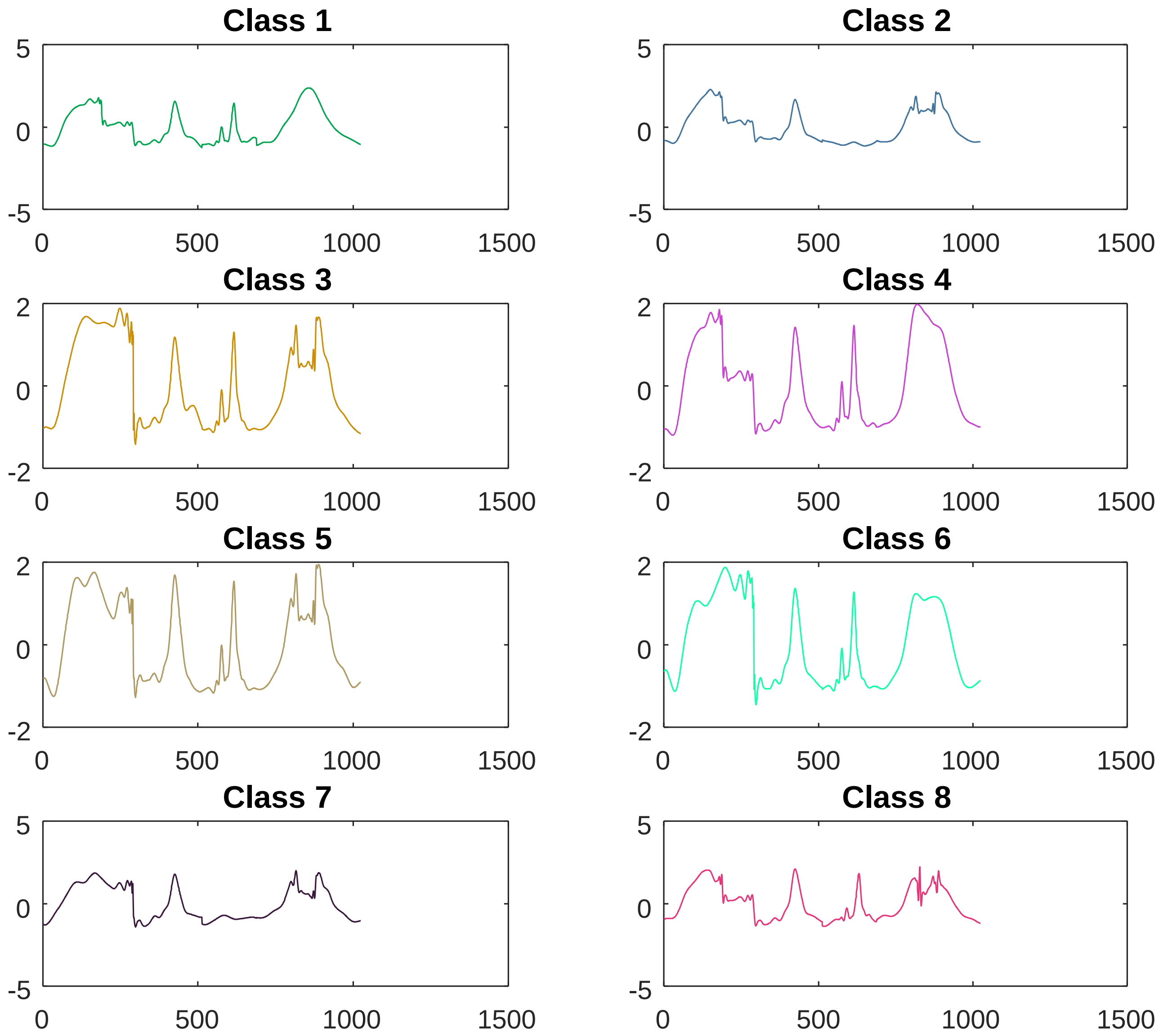
Lightning7

One exemplar per class,
with z-normalization



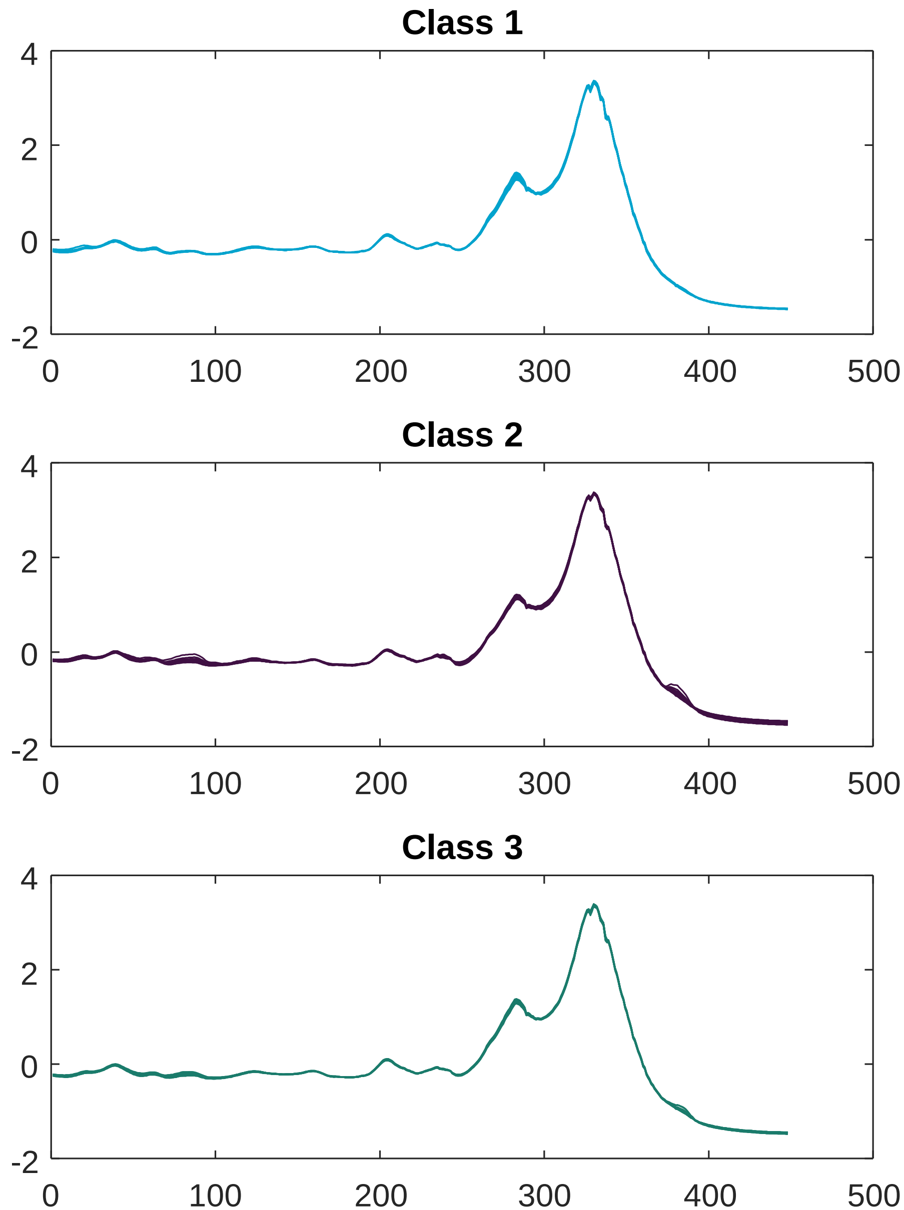
Mallat

One exemplar per class, with z-normalization



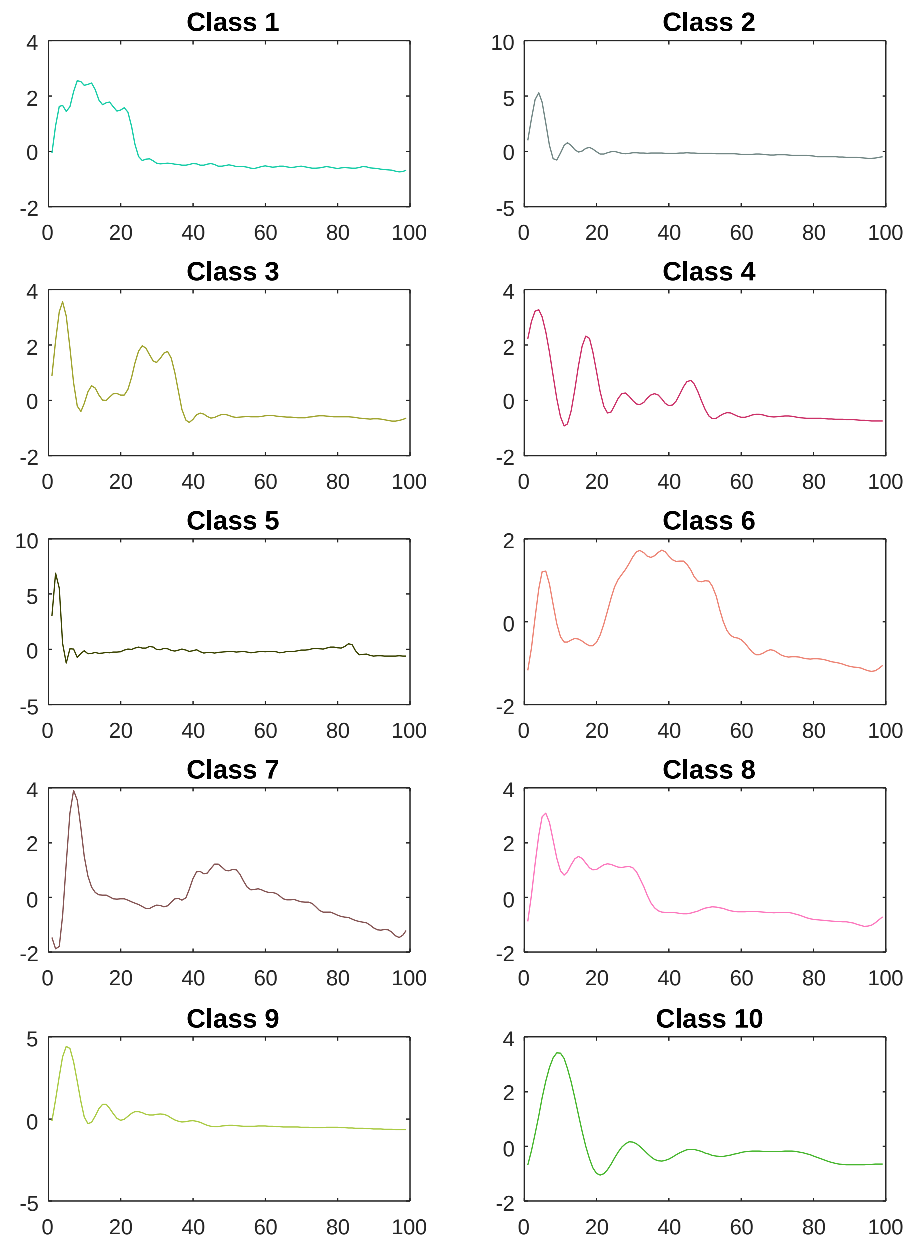
Meat

Twenty exemplars per class, with z-normalization



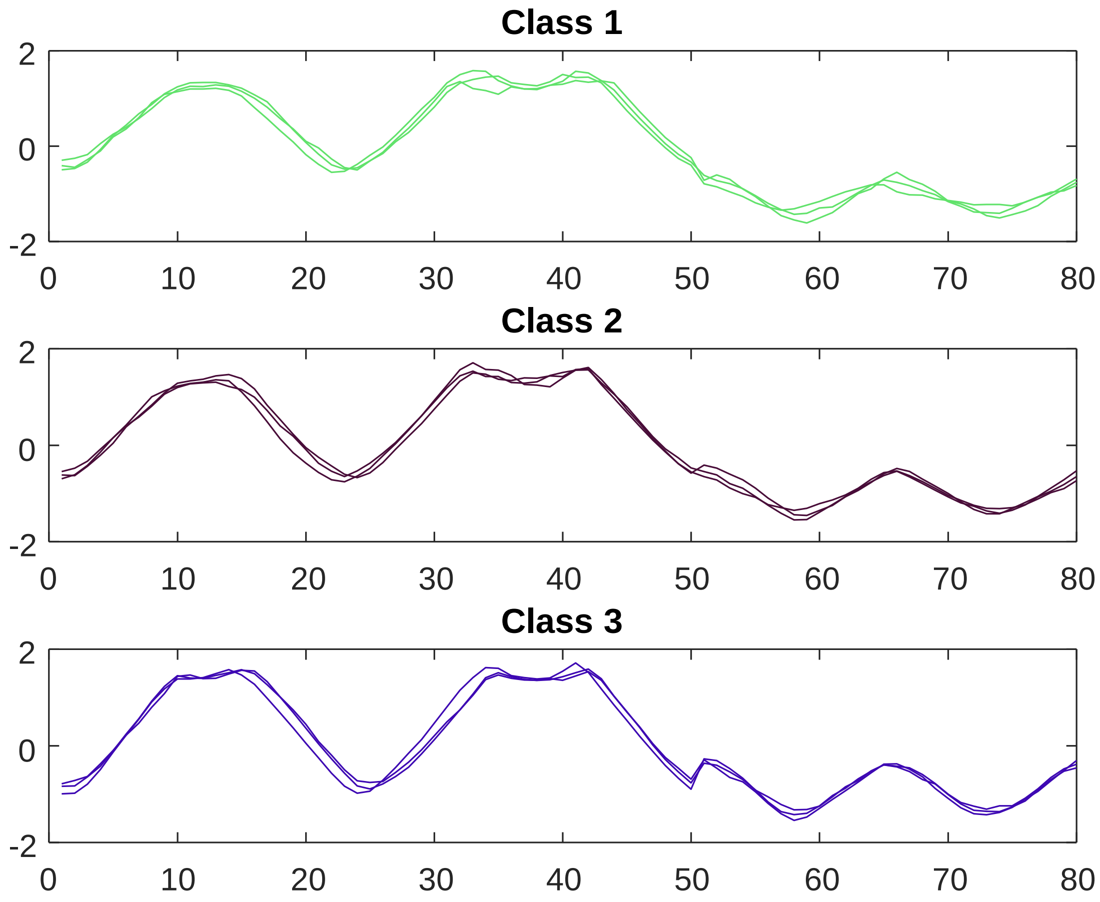
MedicalImages

One exemplar per class,
with z-normalization



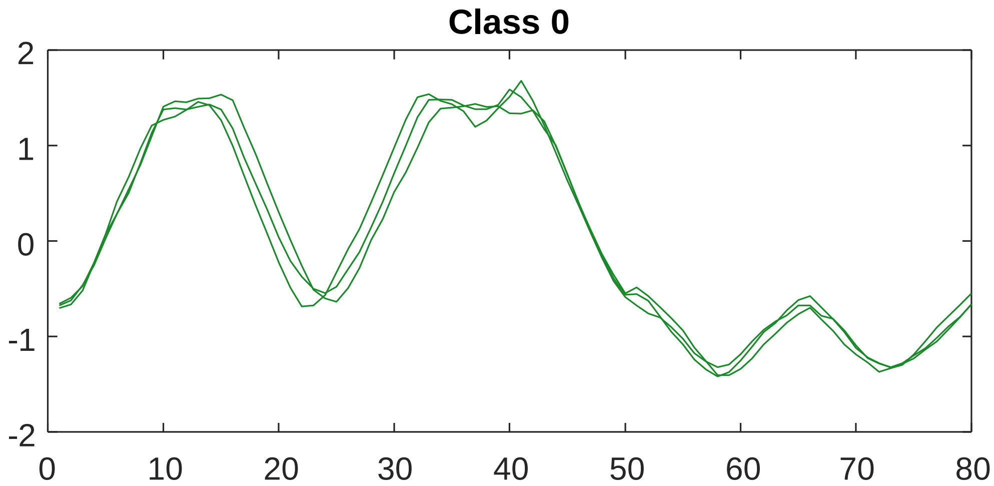
MiddlePhalanxOutlineAgeGroup

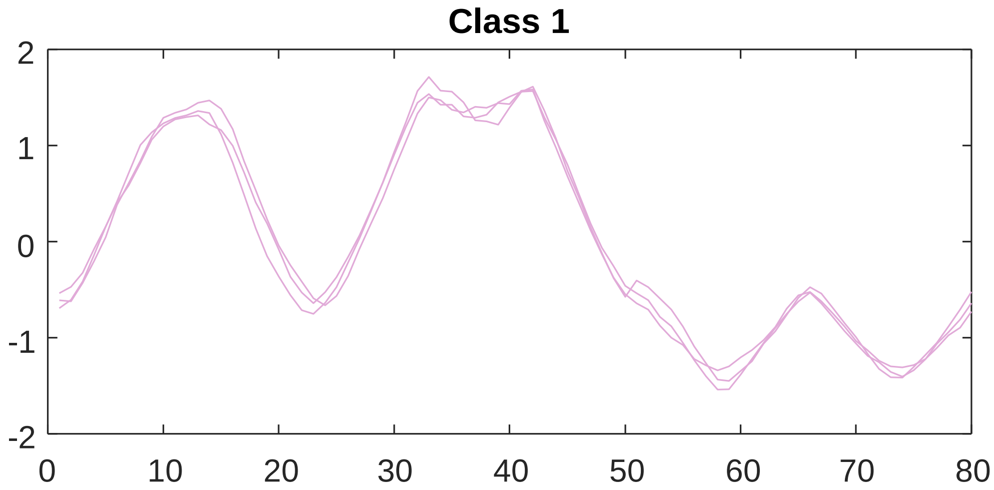
Three exemplars per class,
with z-normalization



MiddlePhalanxOutlineCorrect

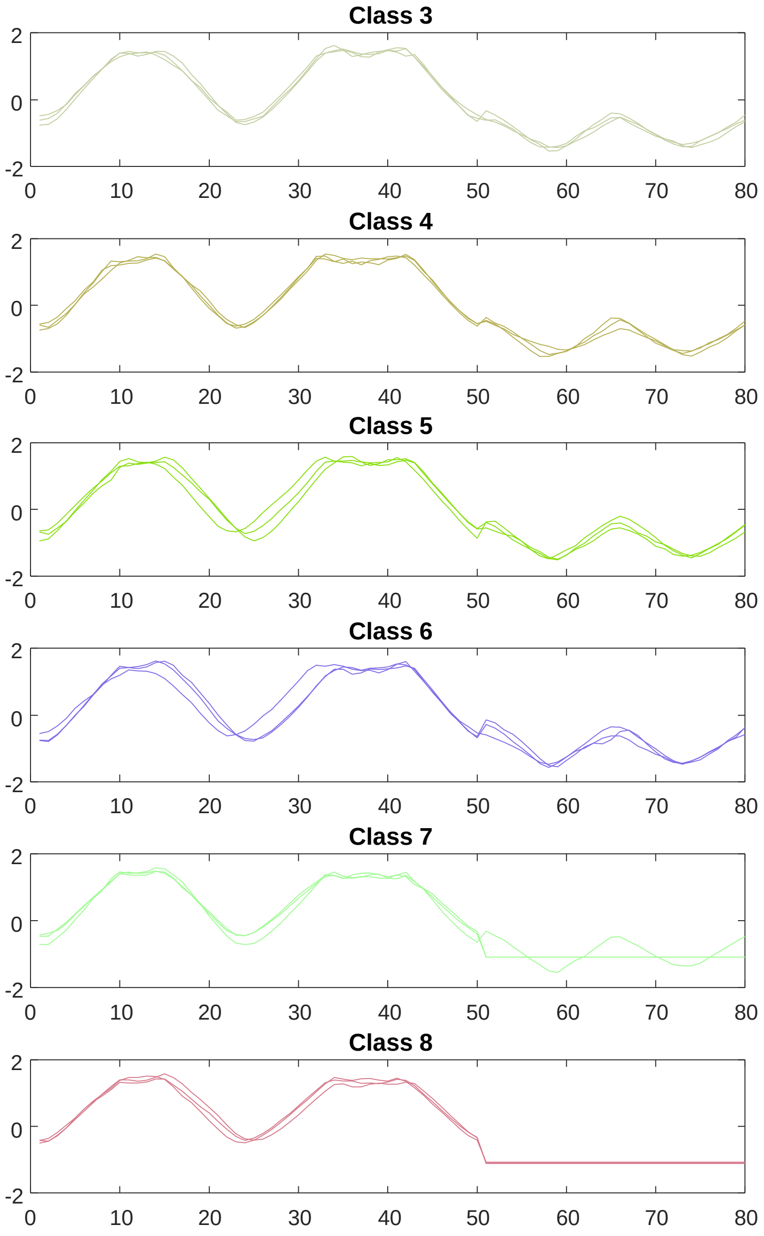
Three exemplars per class,
with z-normalization





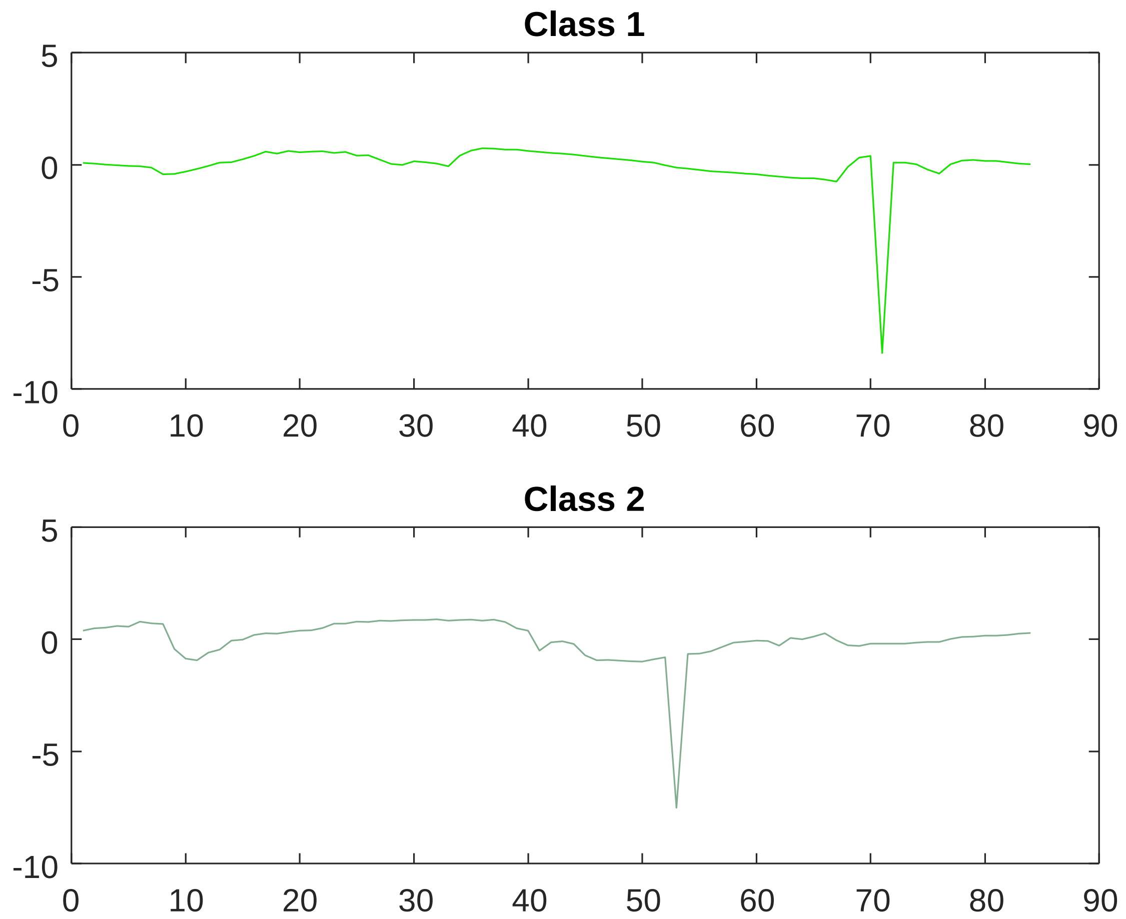
MiddlePhalanxTW

Three exemplars per class,
with z-normalization



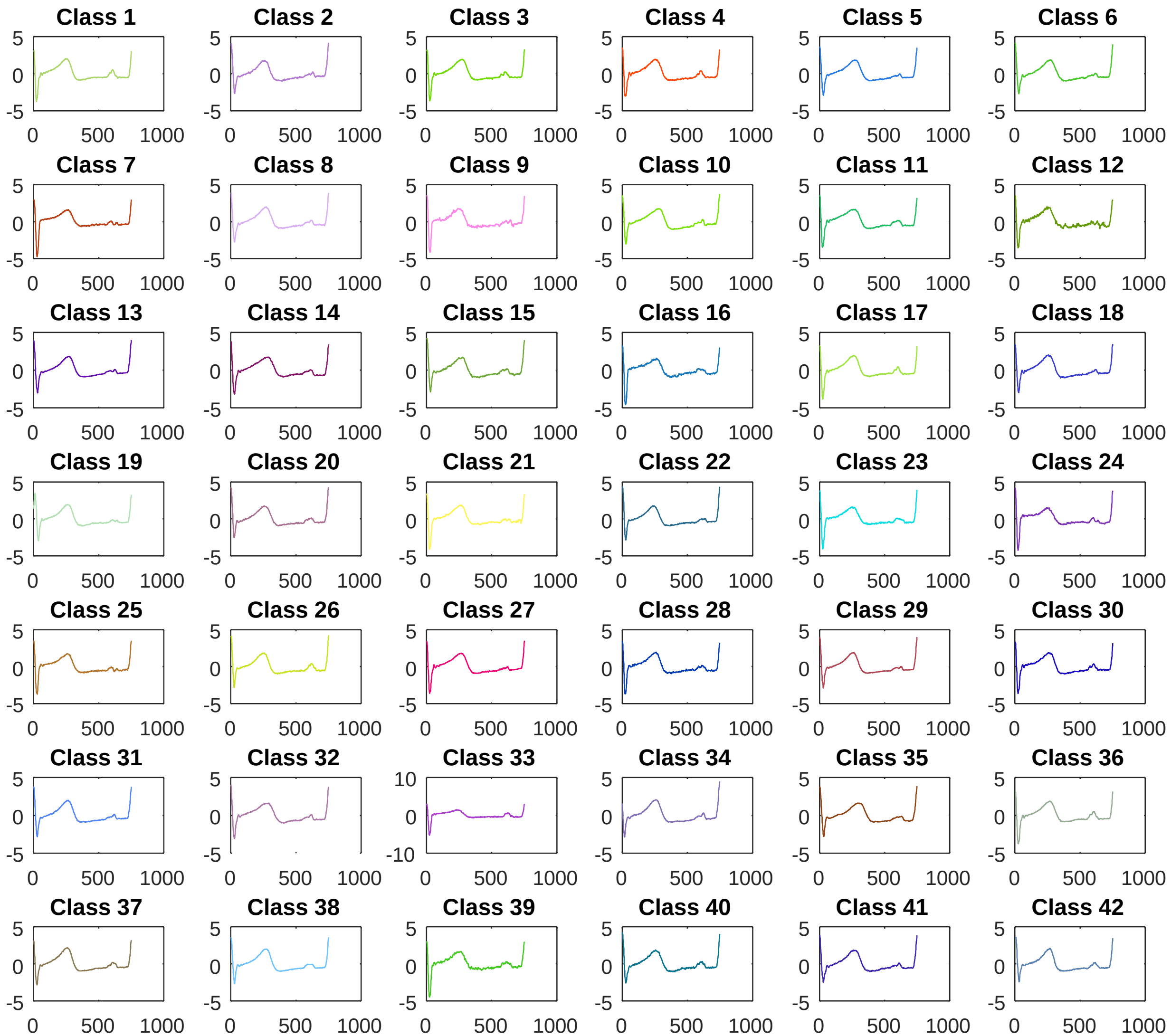
MoteStrain

One exemplar per class,
with z-normalization



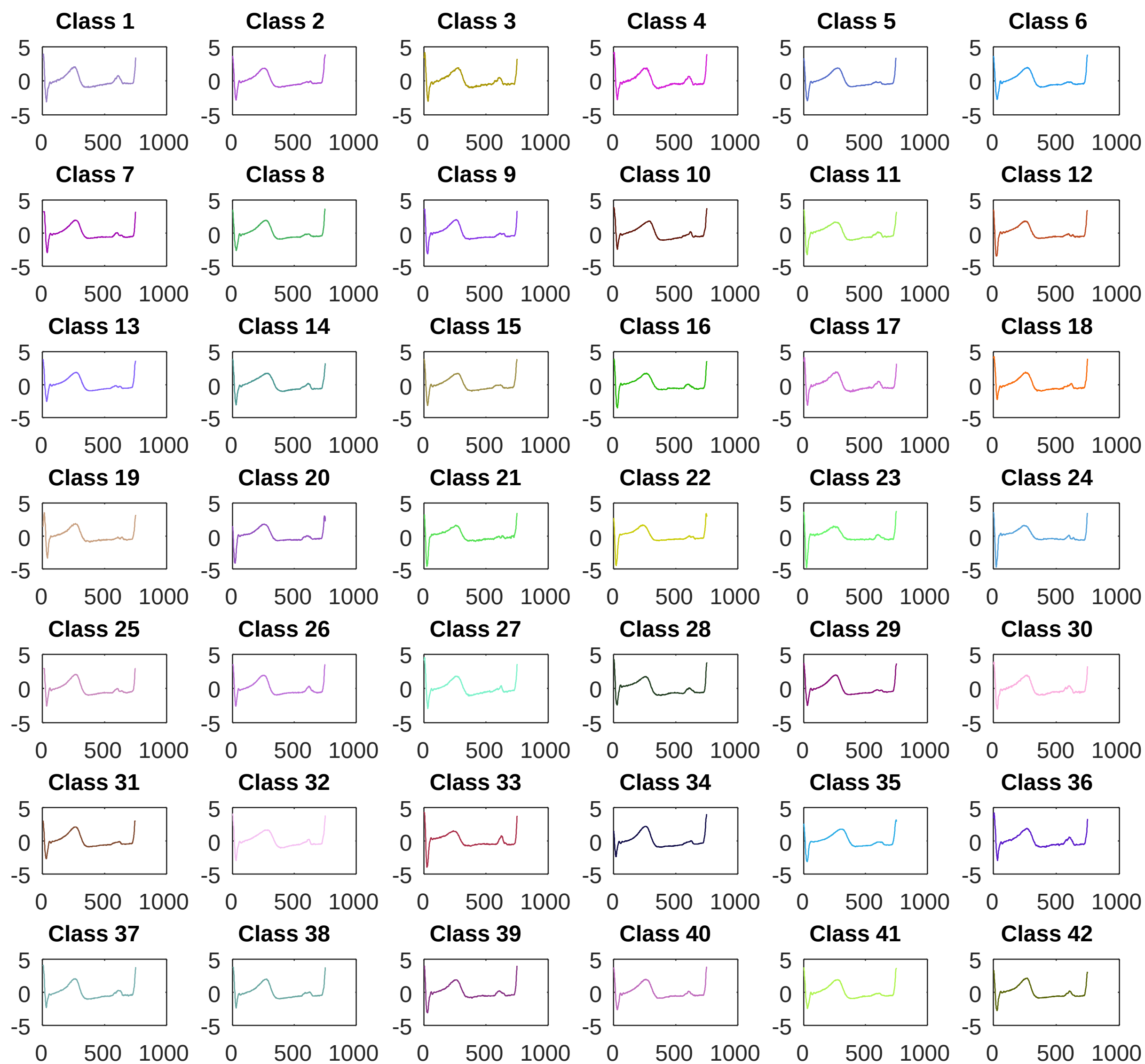
NonInvasiveFetalECGThorax1

One exemplar per class,
with z-normalization



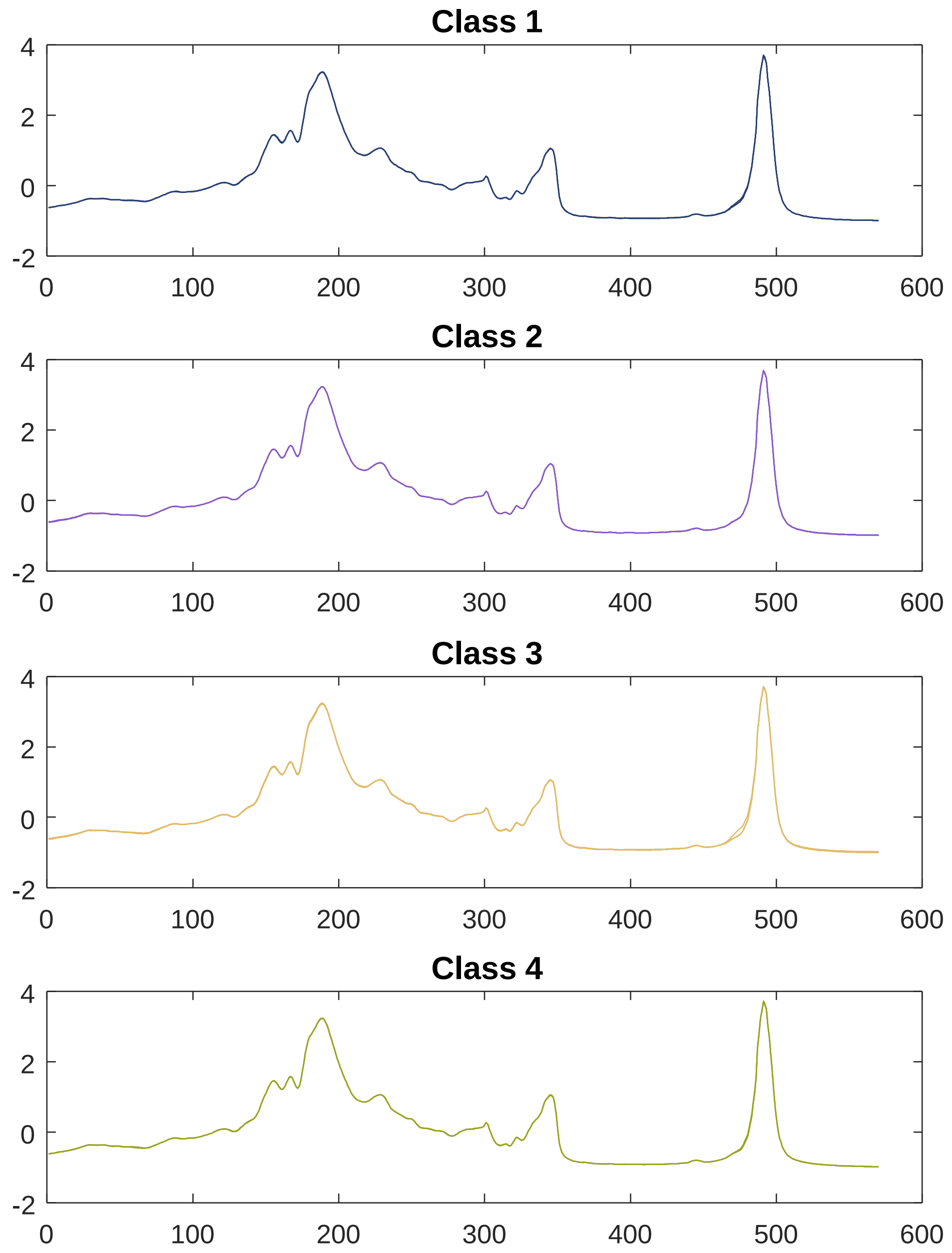
NonInvasiveFetalECGThorax2

One exemplar per class,
with z-normalization



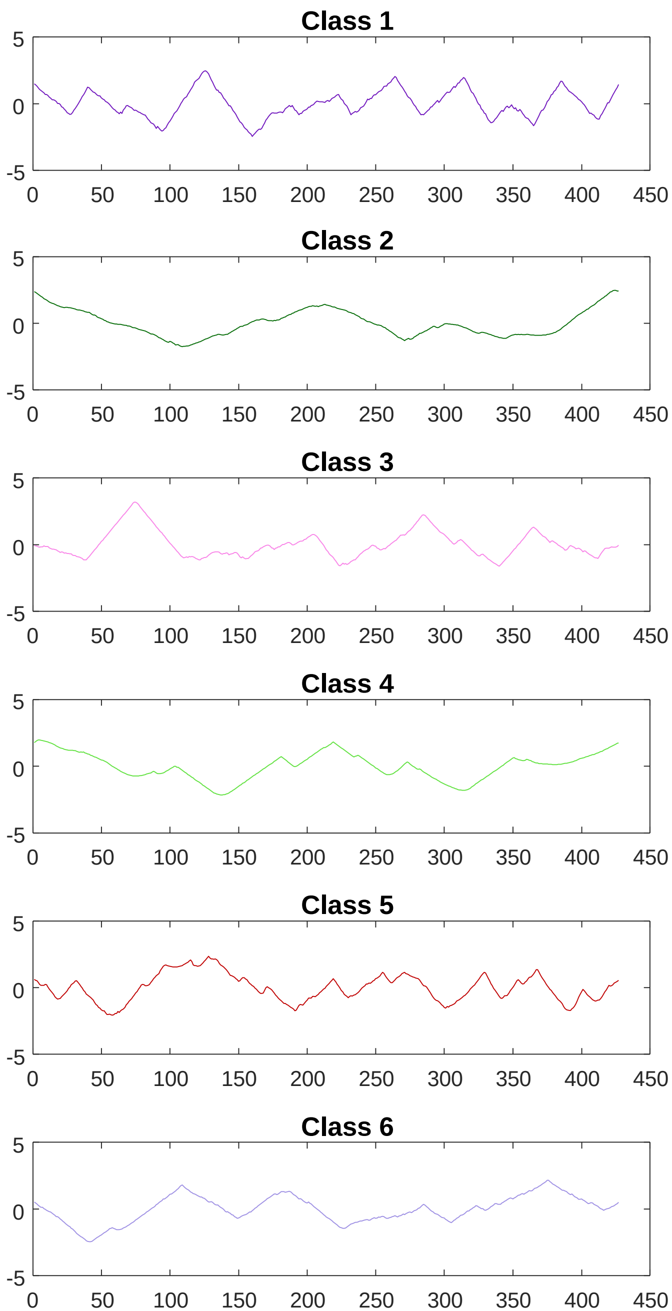
OliveOil

Three exemplars per class,
with z-normalization



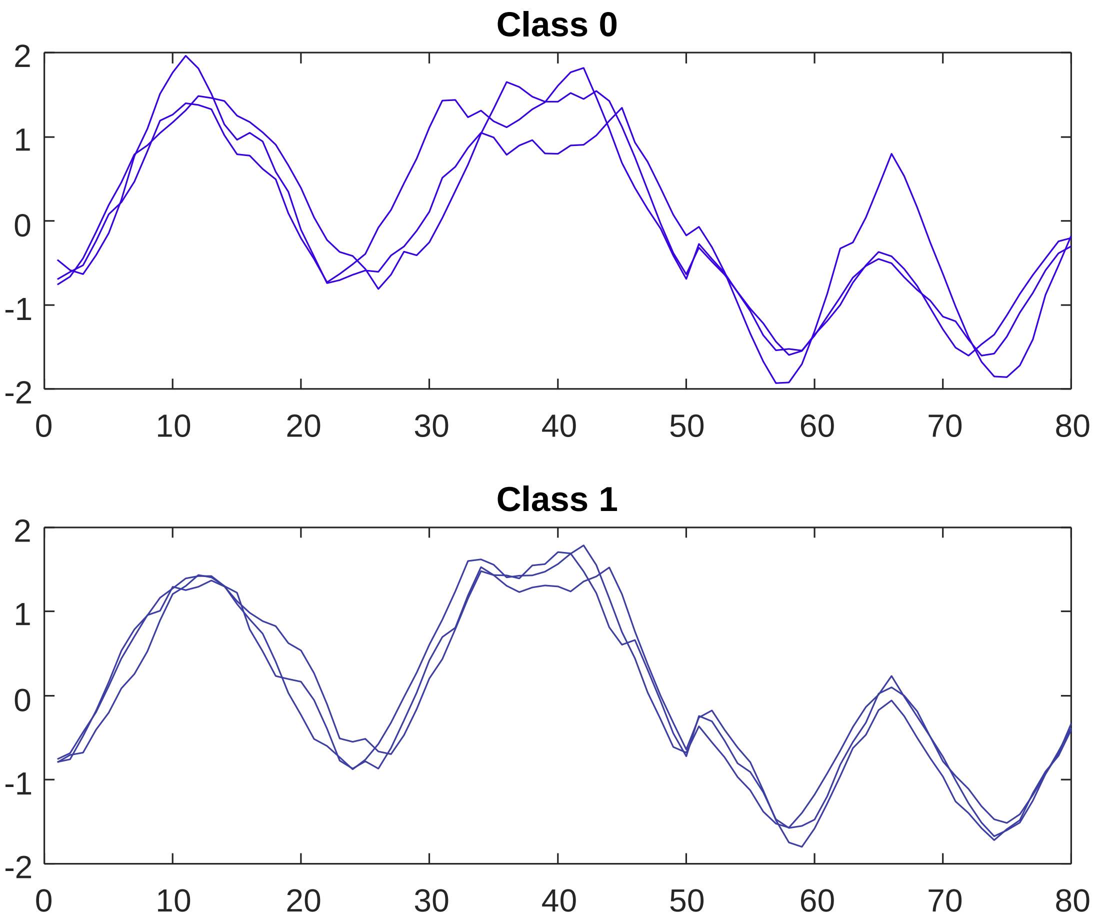
OSULeaf

One exemplar per class,
with z-normalization



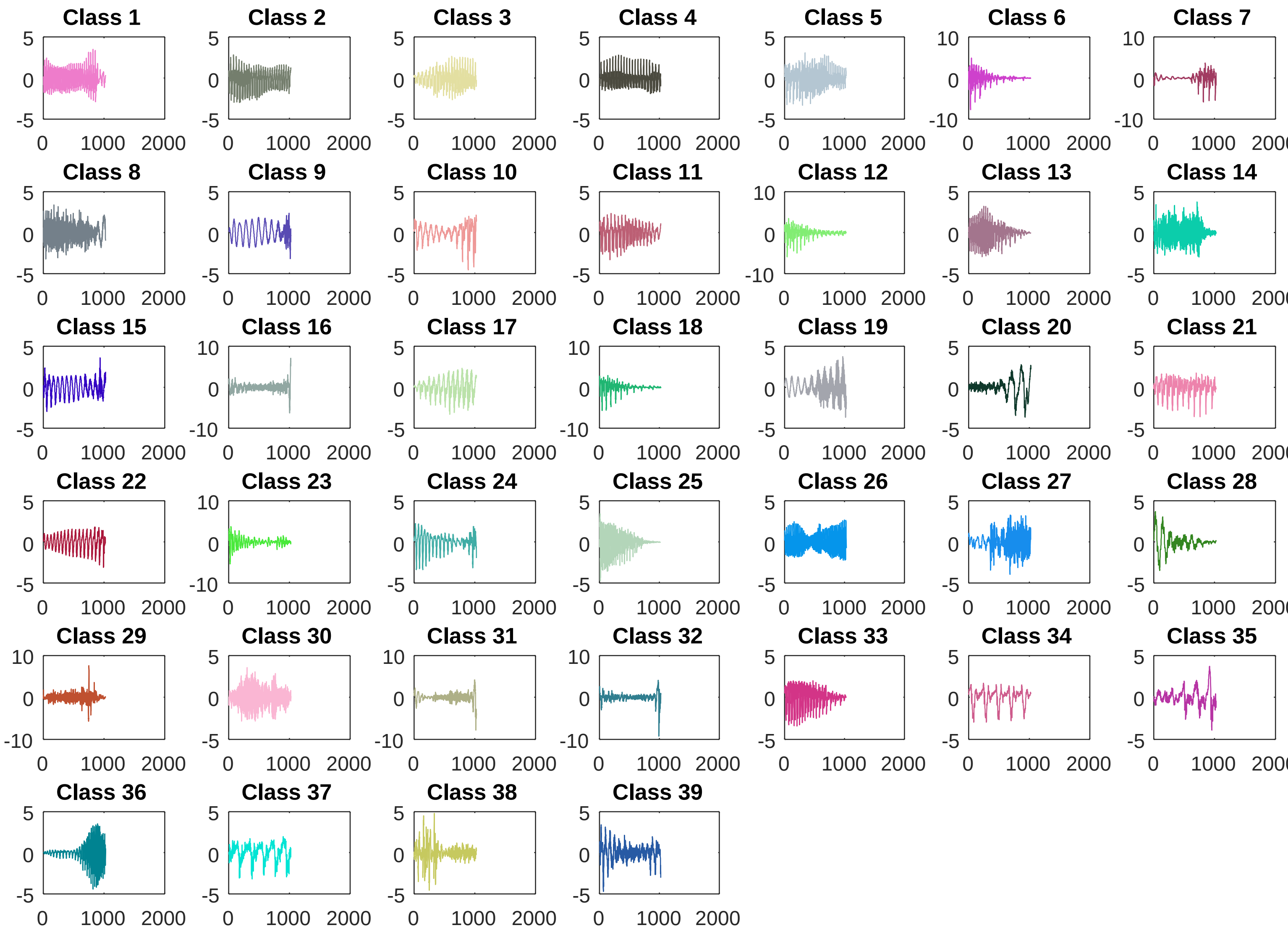
PhalangesOutlinesCorrect

Three exemplars per class,
with z-normalization



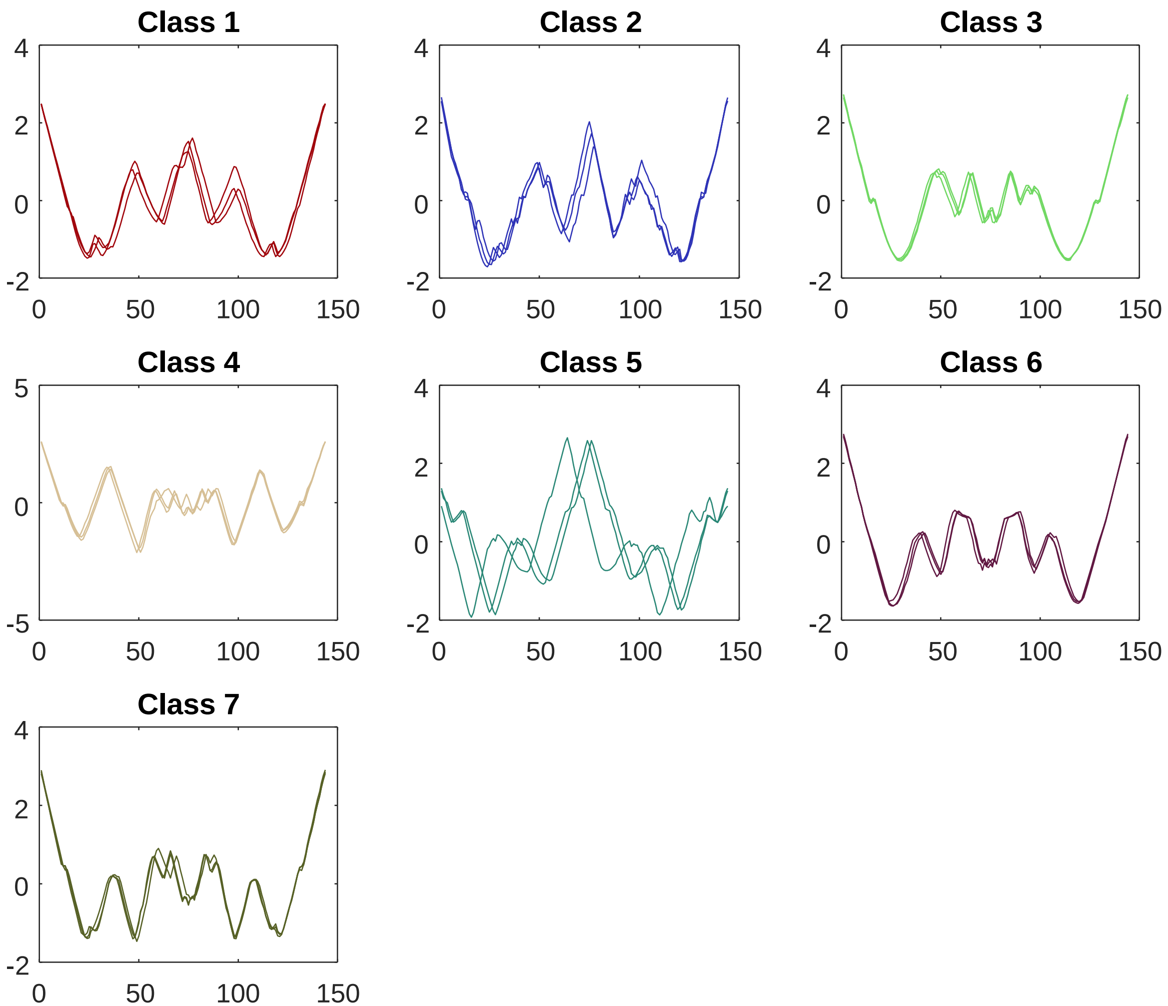
Phoneme

One exemplar per class,
with z-normalization



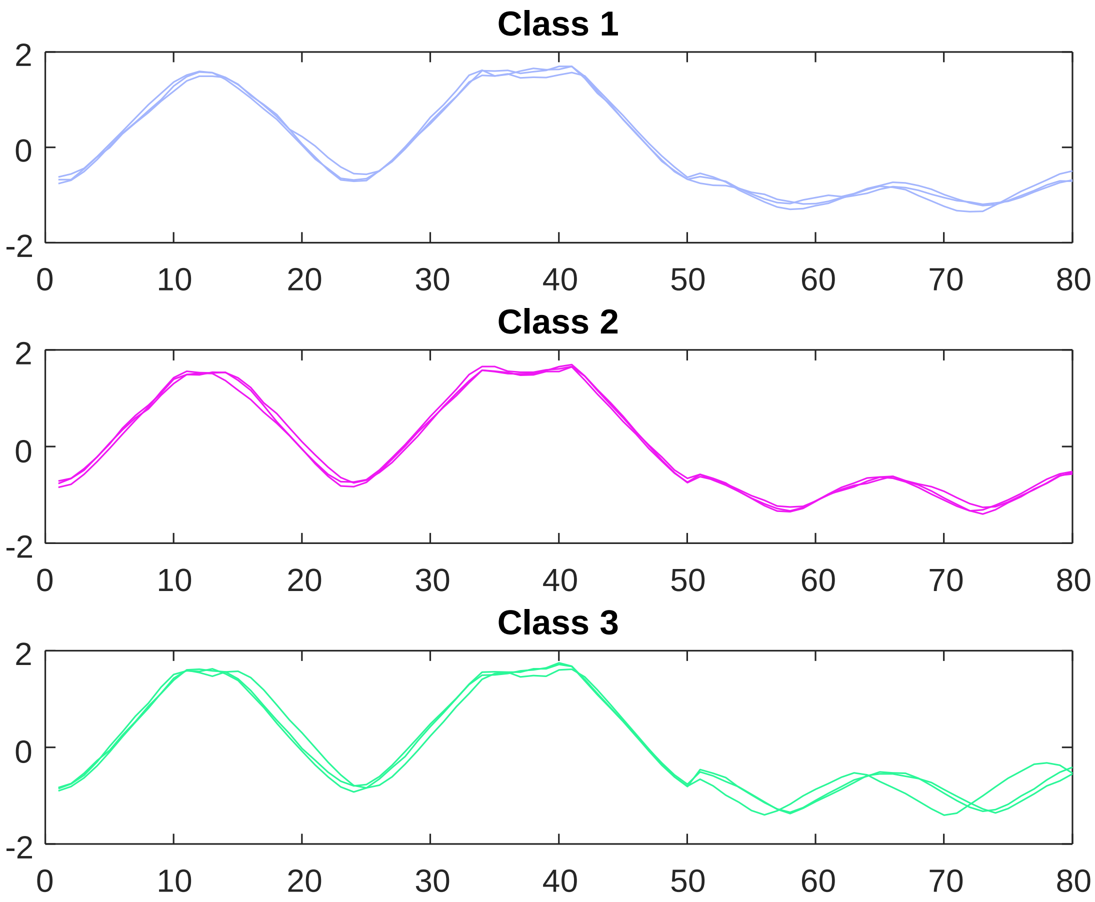
Plane

Three exemplars per class,
with z-normalization



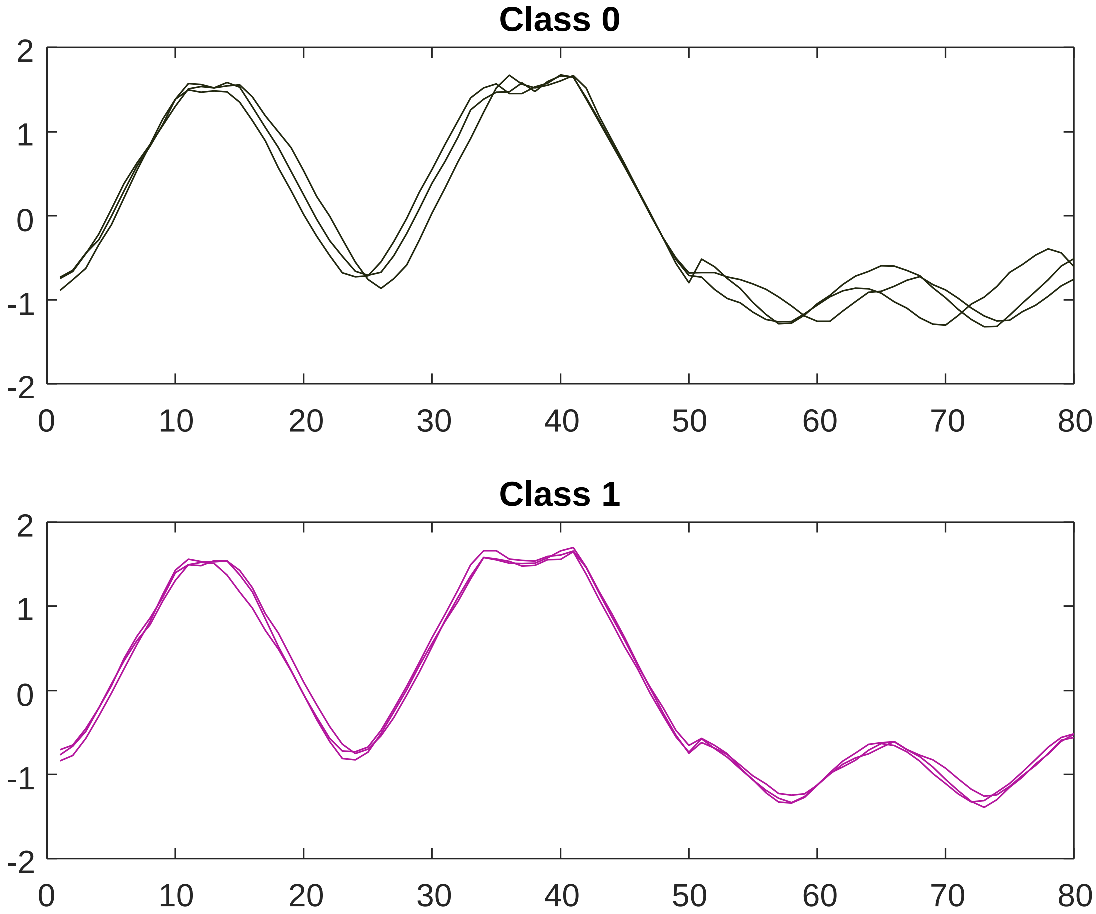
ProximalPhalanxOutlineAgeGroup

Three exemplars per class,
with z-normalization



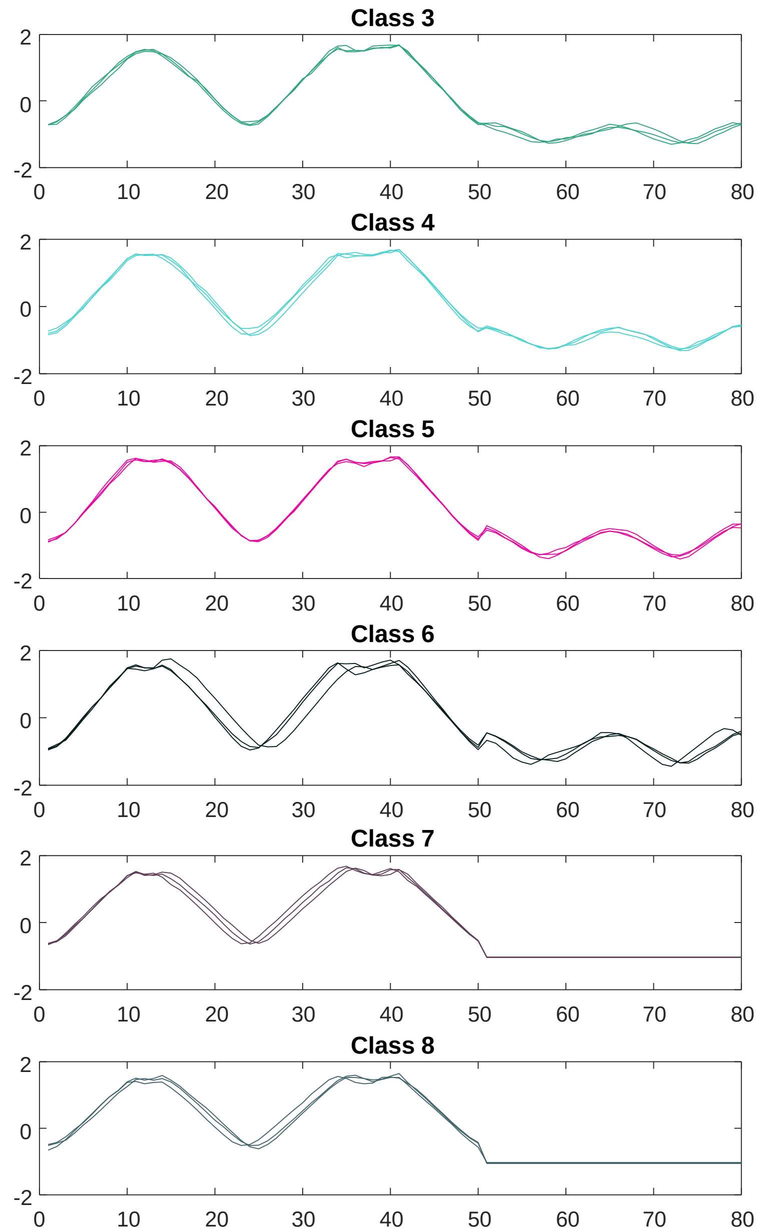
ProximalPhalanxOutlineCorrect

Three exemplars per class,
with z-normalization



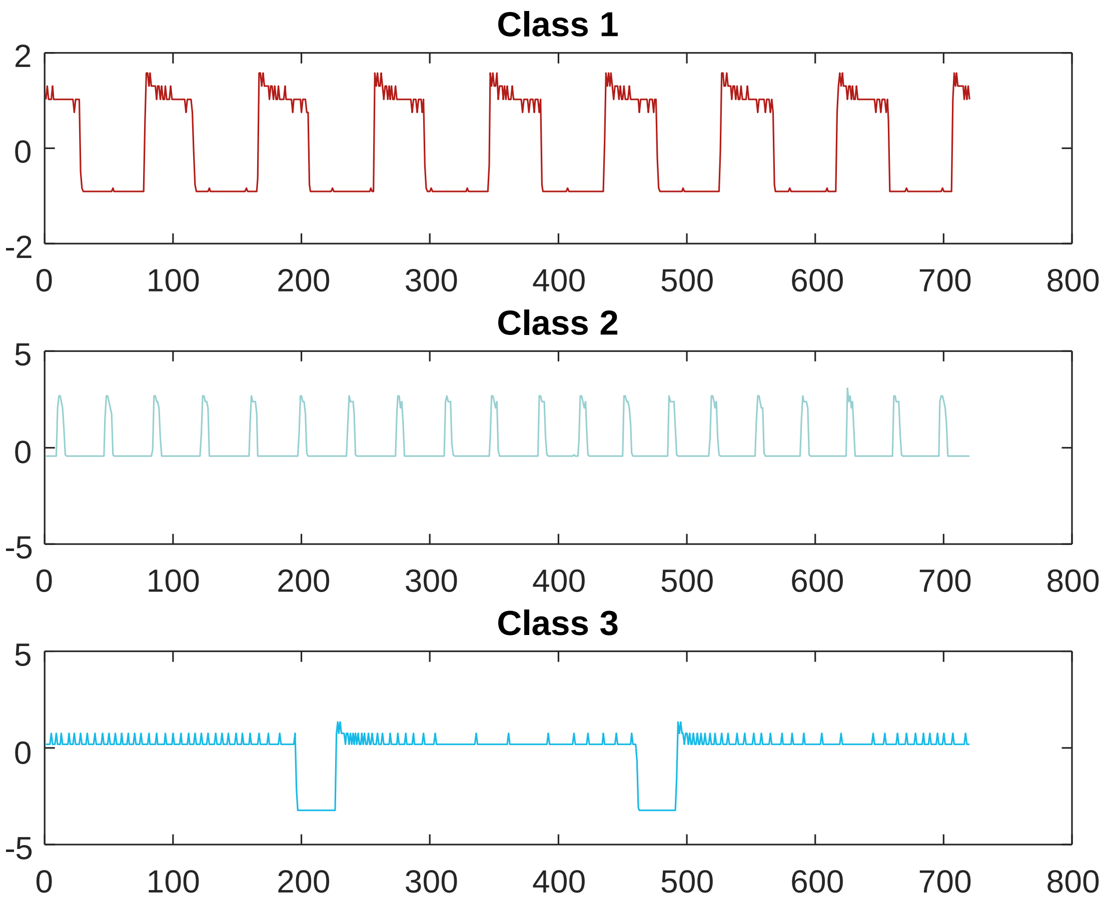
ProximalPhalanxTW

Three exemplars per class,
with z-normalization



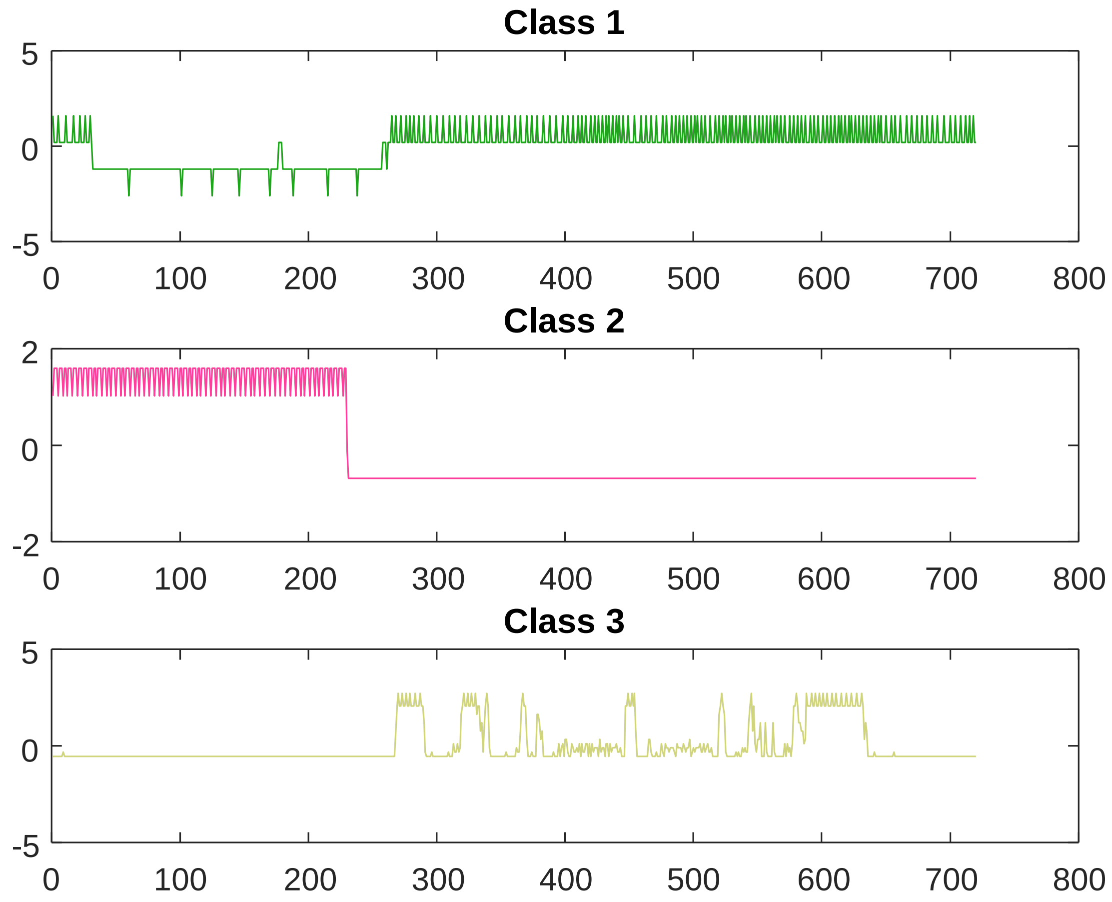
RefrigerationsDevices

One exemplar per class,
with z-normalization



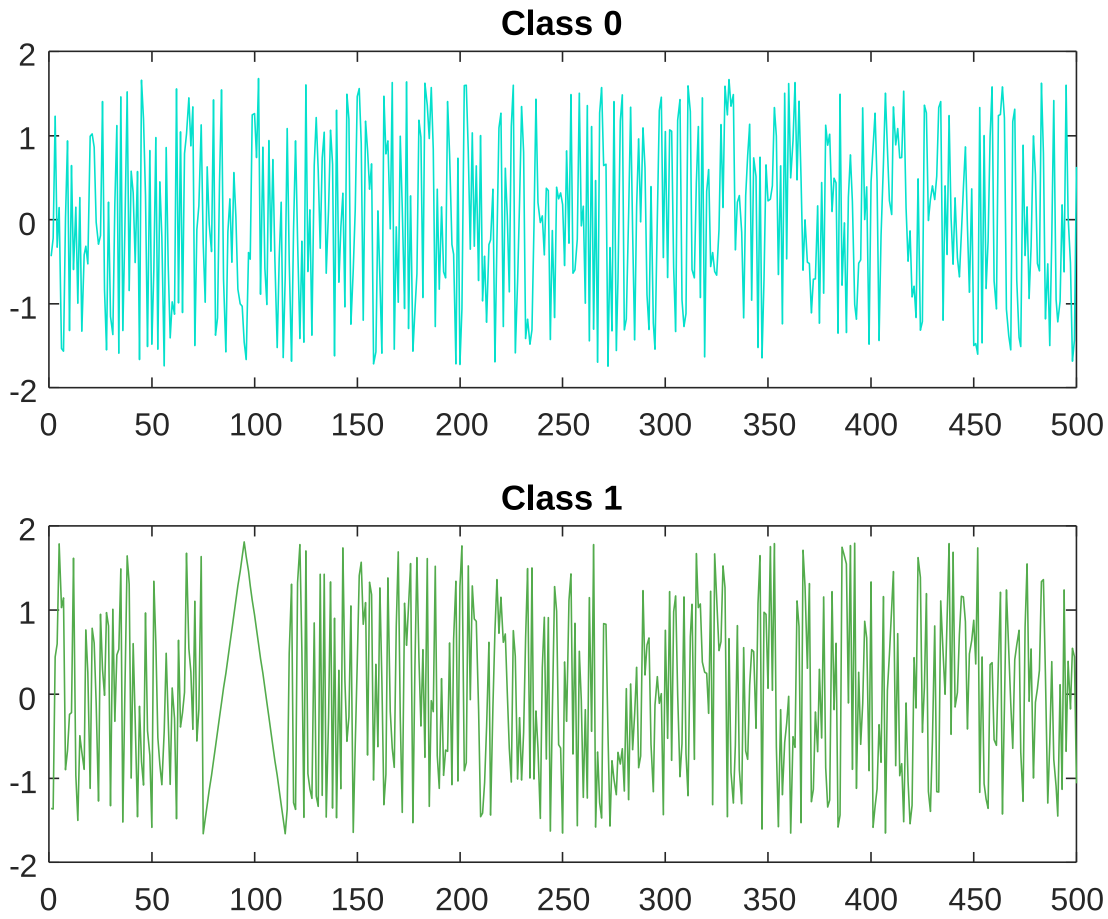
ScreenType

One exemplar per class,
with z-normalization



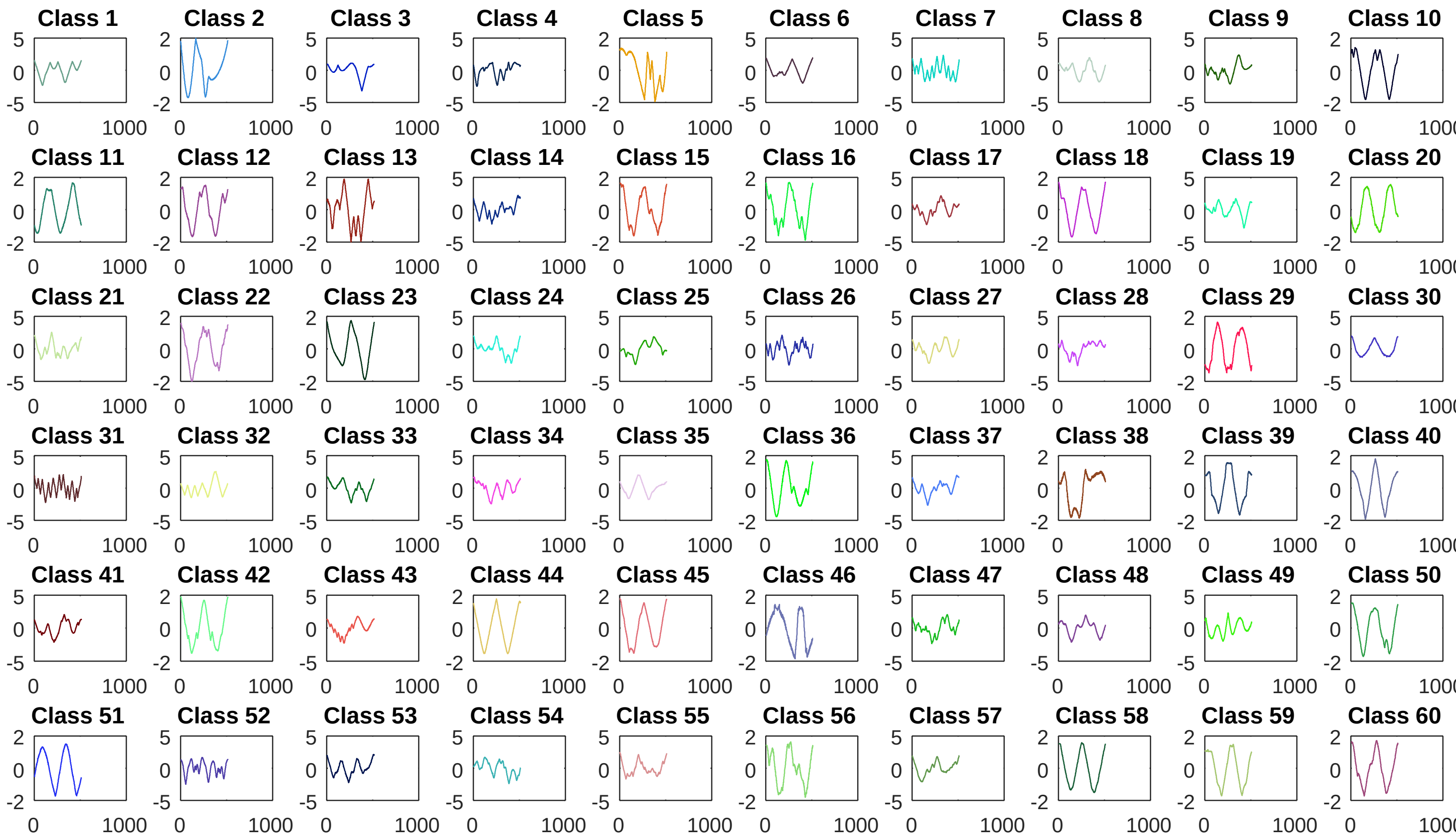
ShapeletSim

One exemplar per class,
with z-normalization



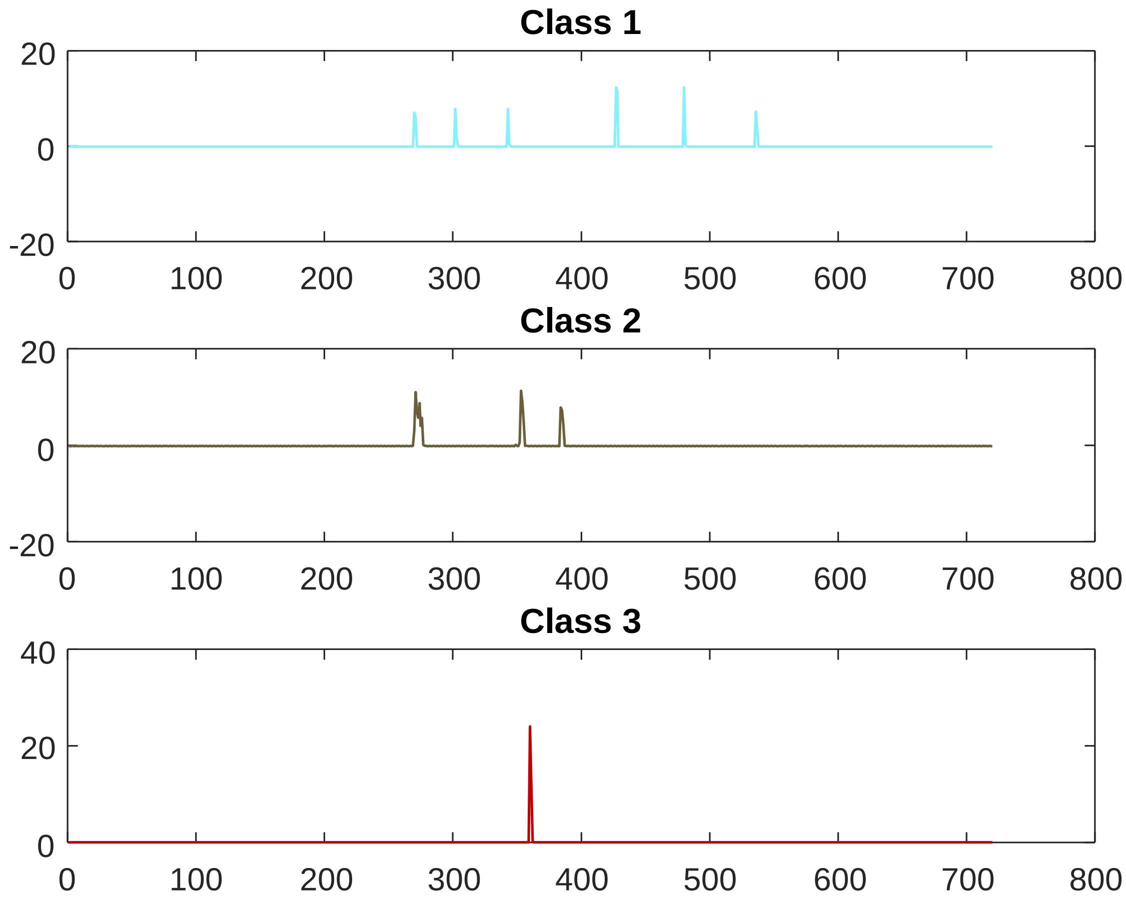
ShapesAll

One exemplar per class,
with z-normalization



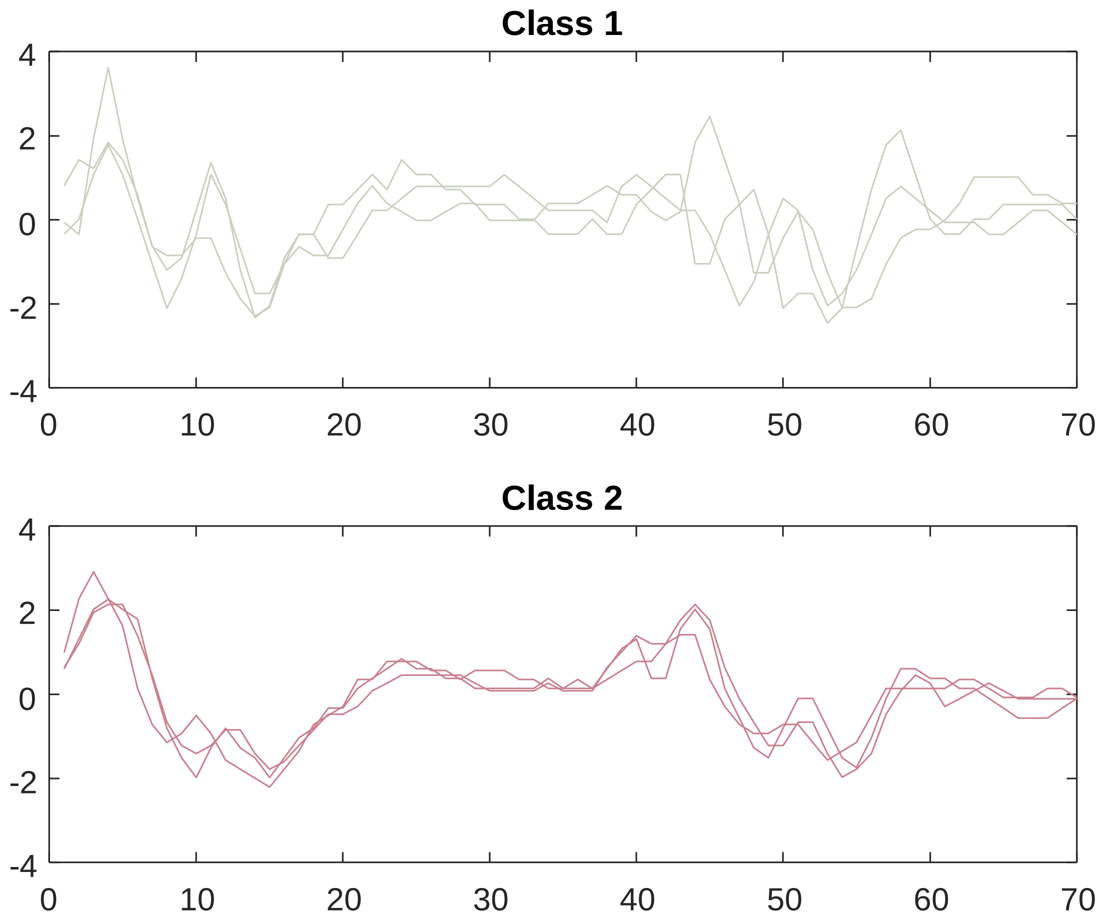
SmallKitchenAppliances

One exemplar per class,
with z-normalization



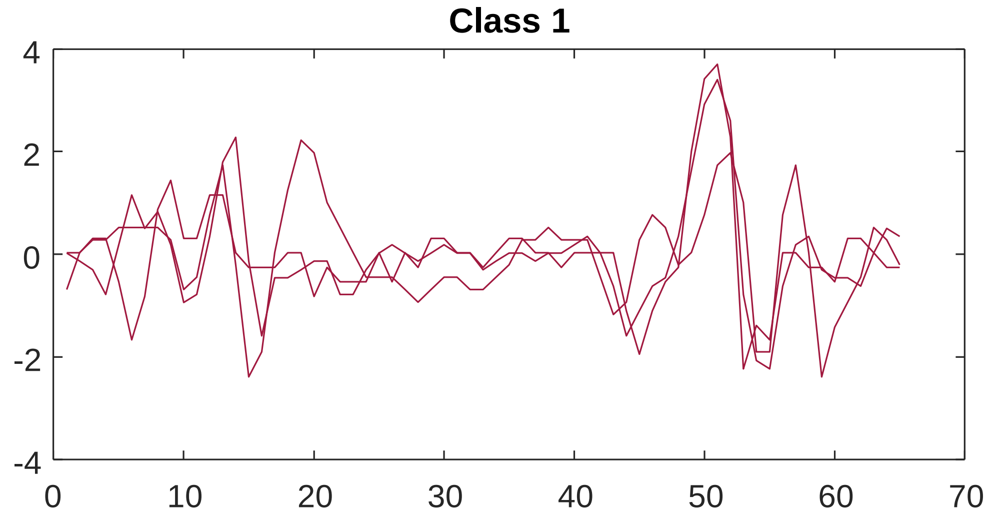
SonyAIBORobotSurface1

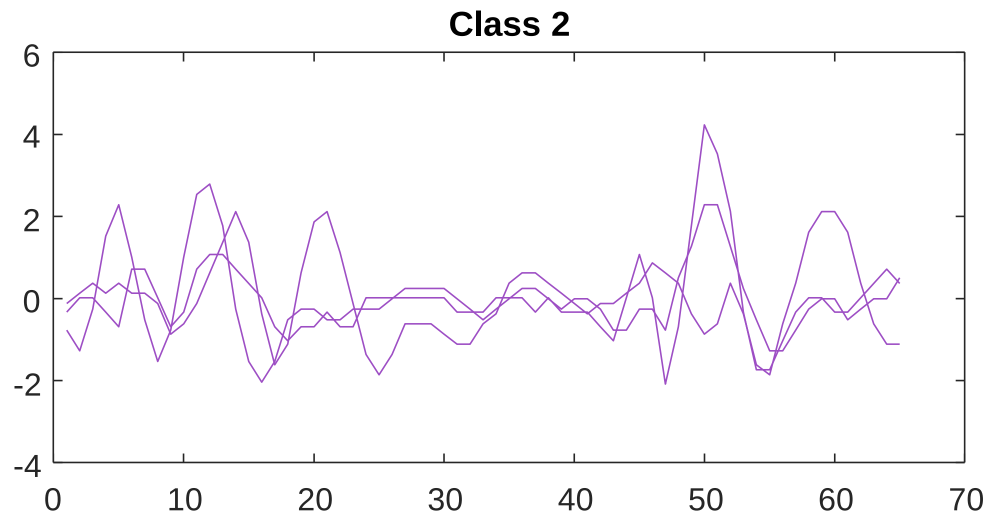
Three exemplars per class,
with z-normalization



SonyAIBORobotSurface2

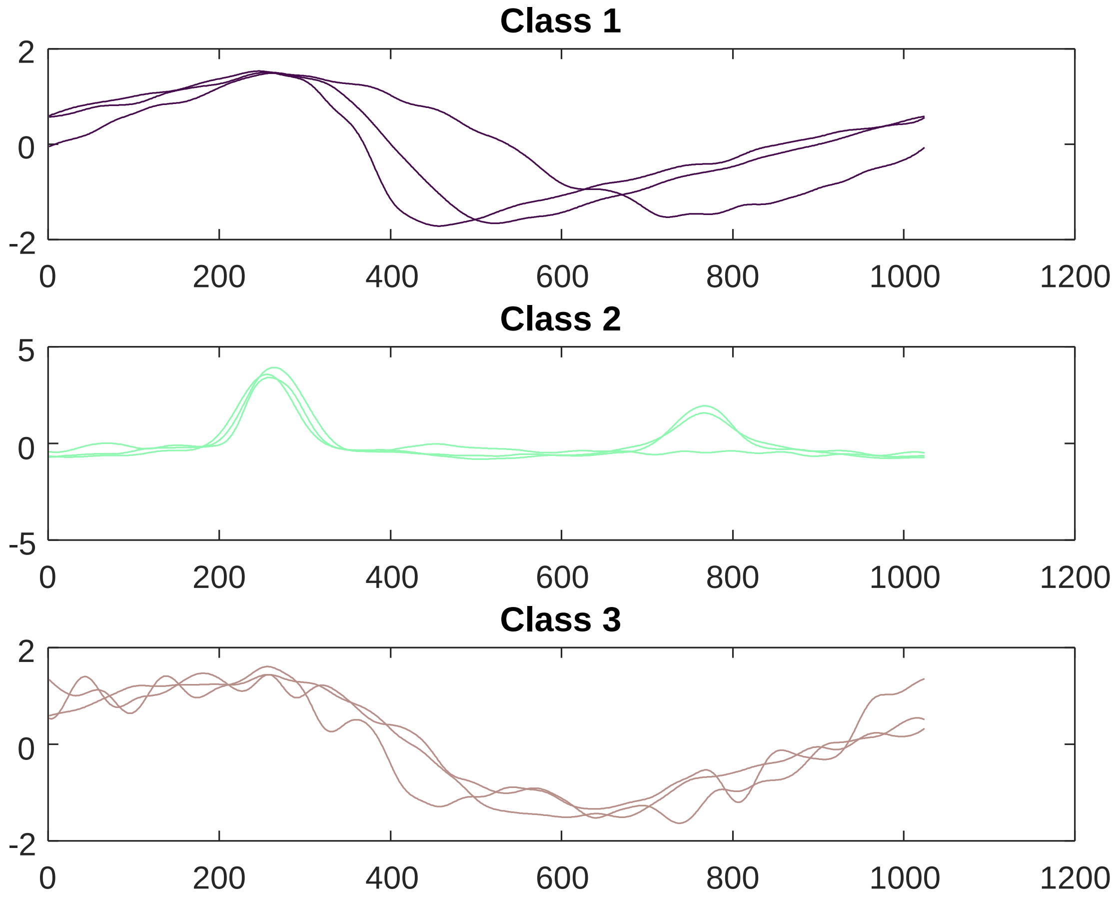
Three exemplars per class,
with z-normalization





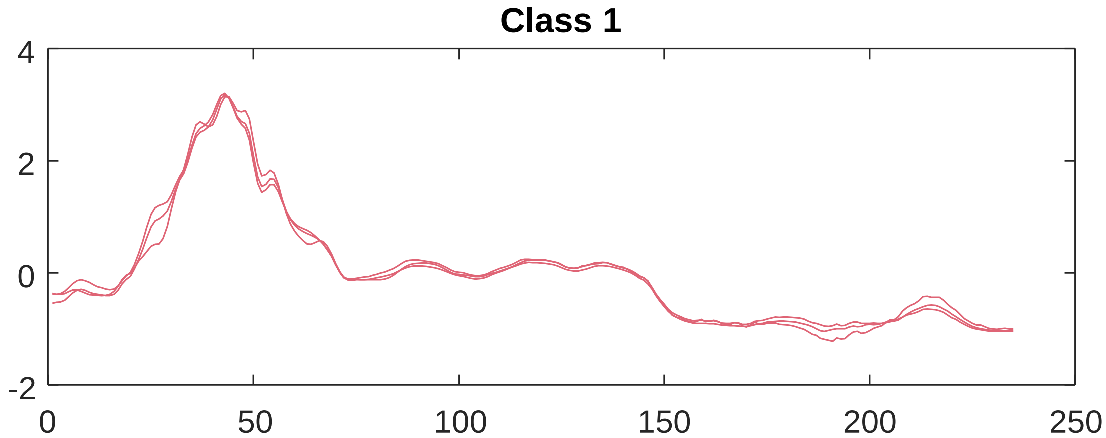
StarLightCurves

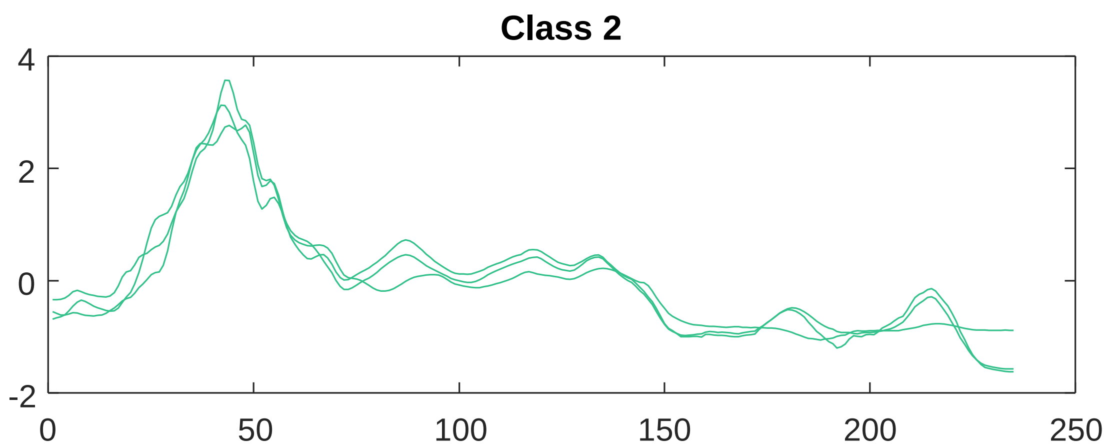
Three exemplars per class,
with z-normalization



Strawberry

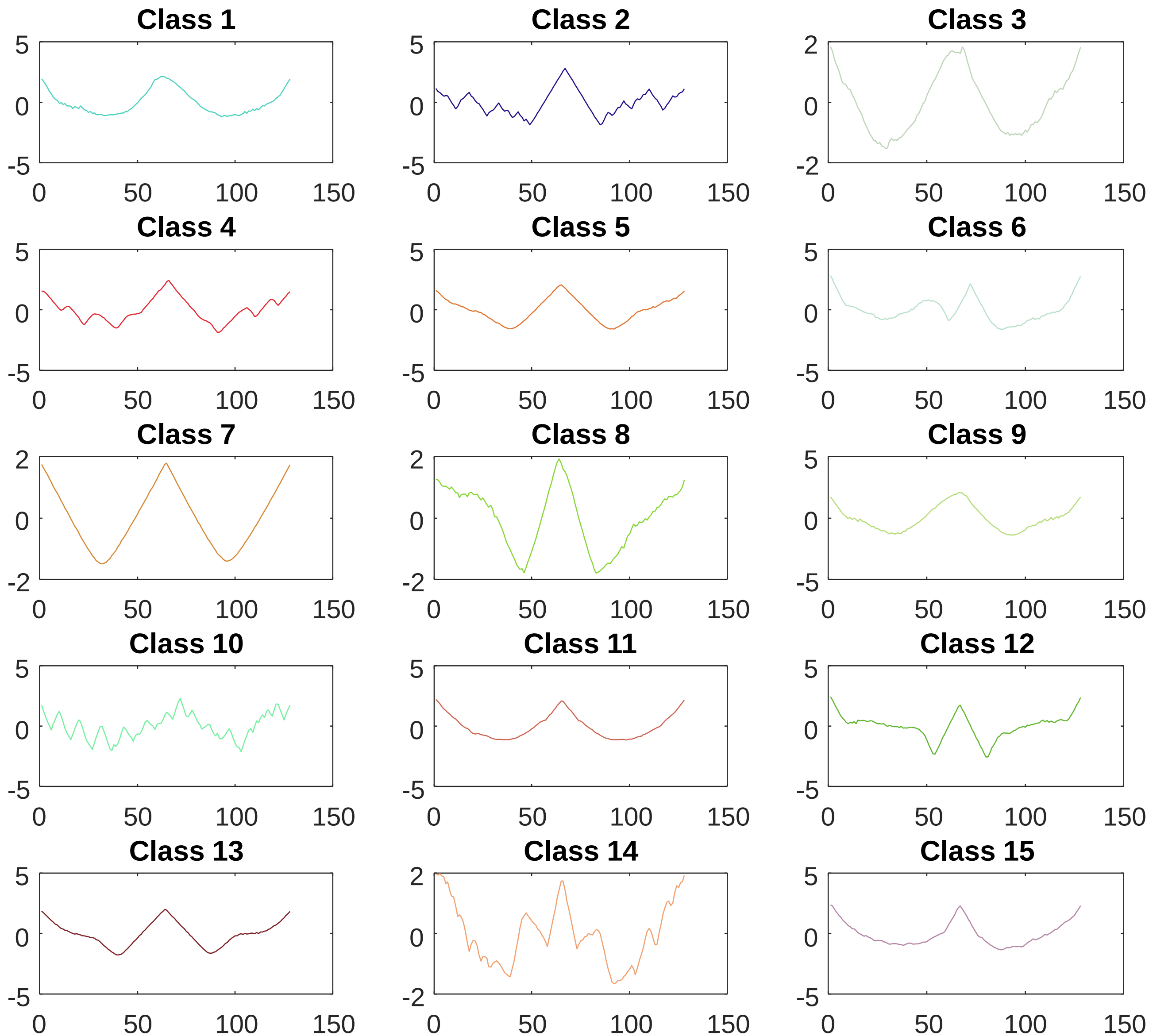
Three exemplars per class,
with z-normalization





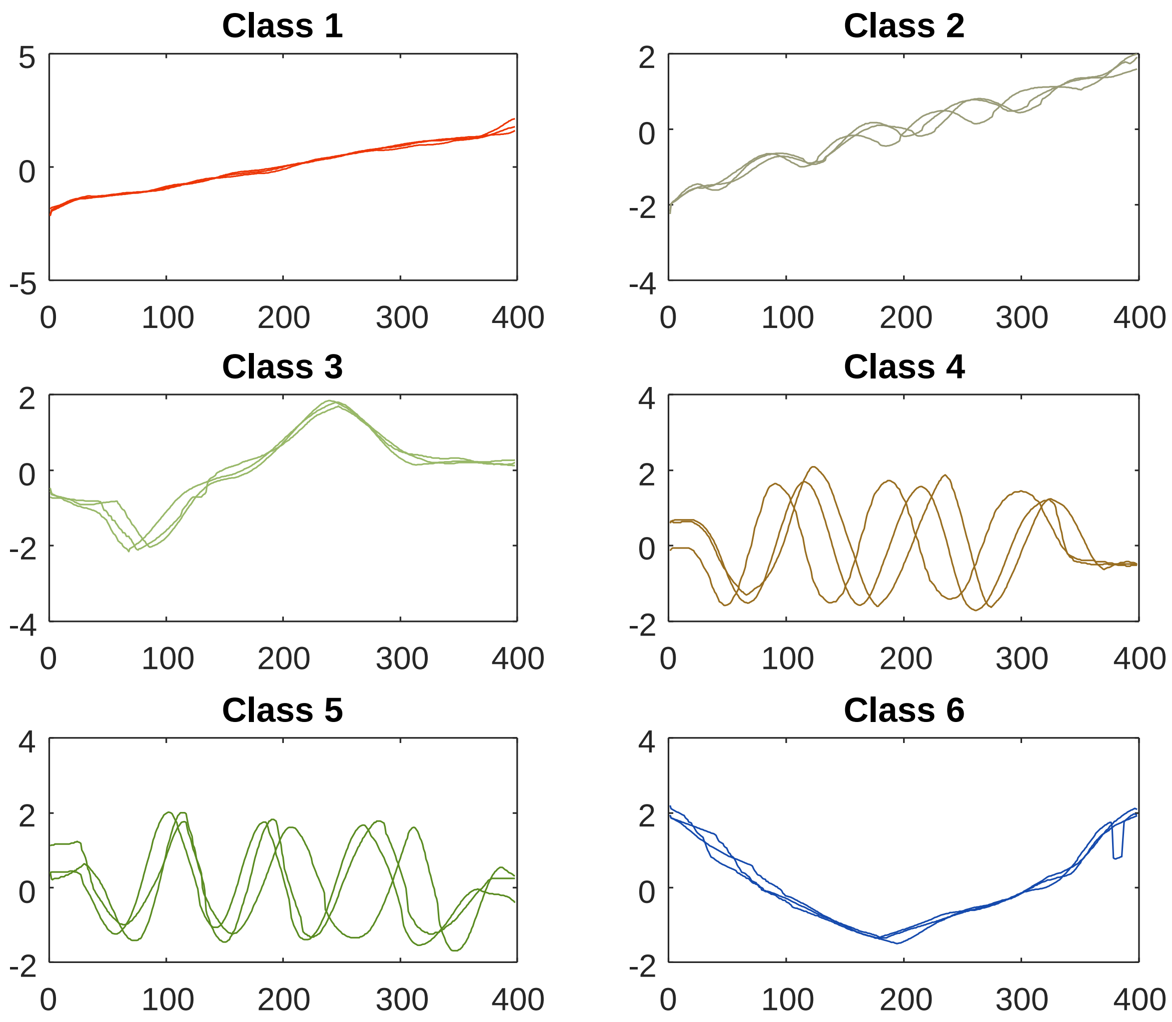
SwedishLeaf

One exemplar per class, with z-normalization



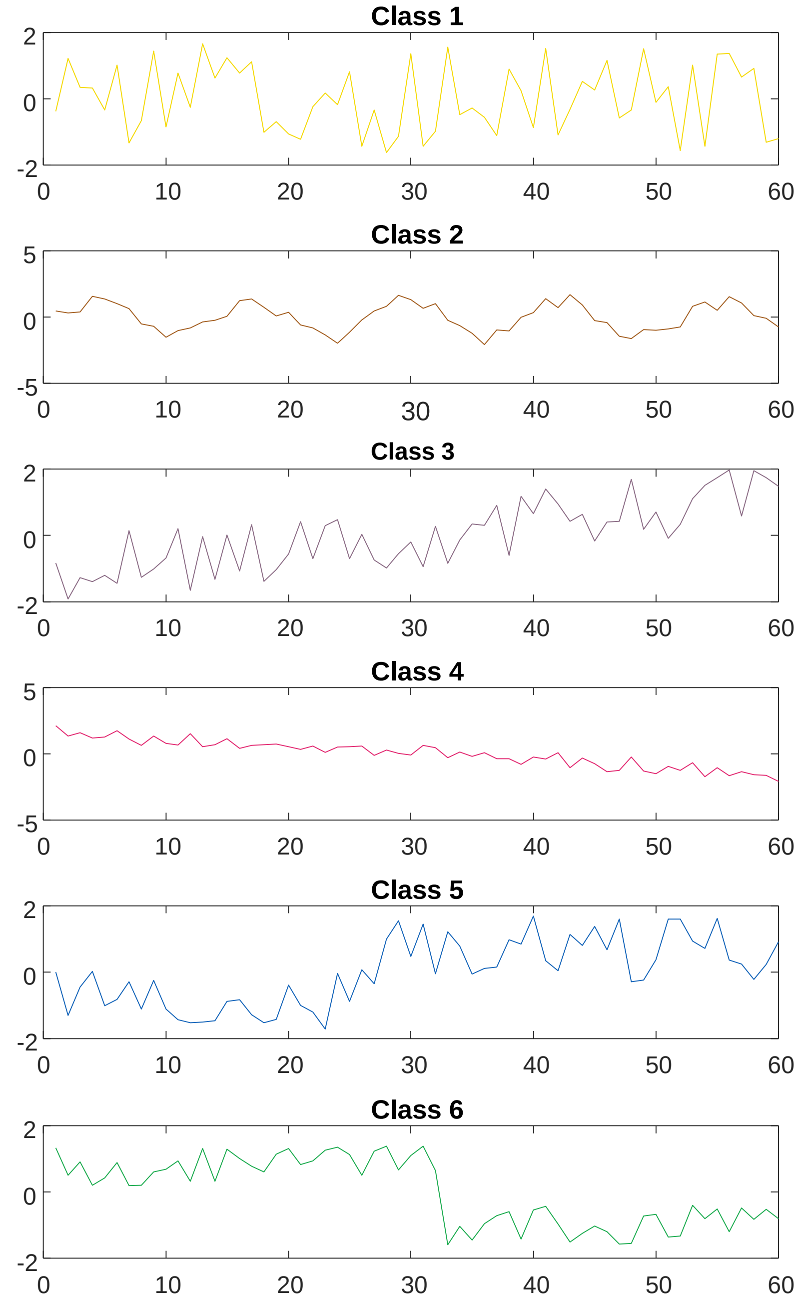
Symbols

Three exemplars per class,
with z-normalization



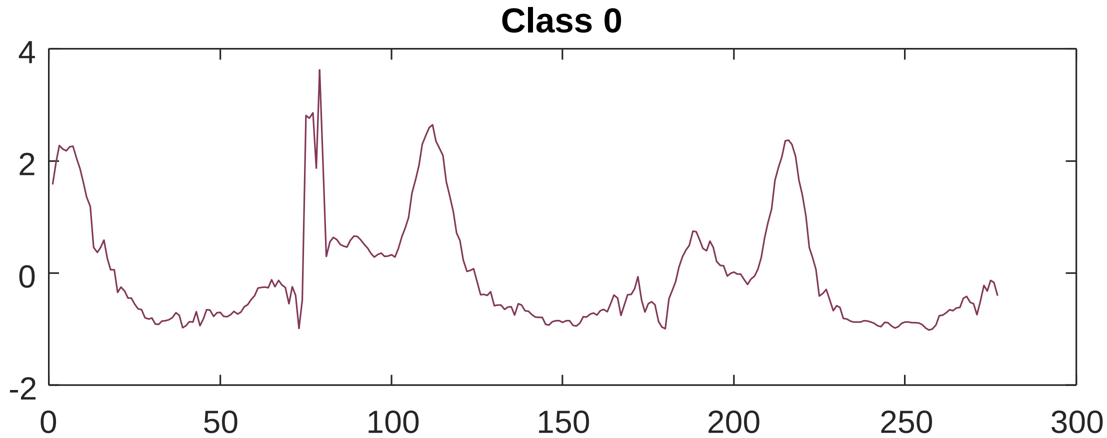
SyntheticControl

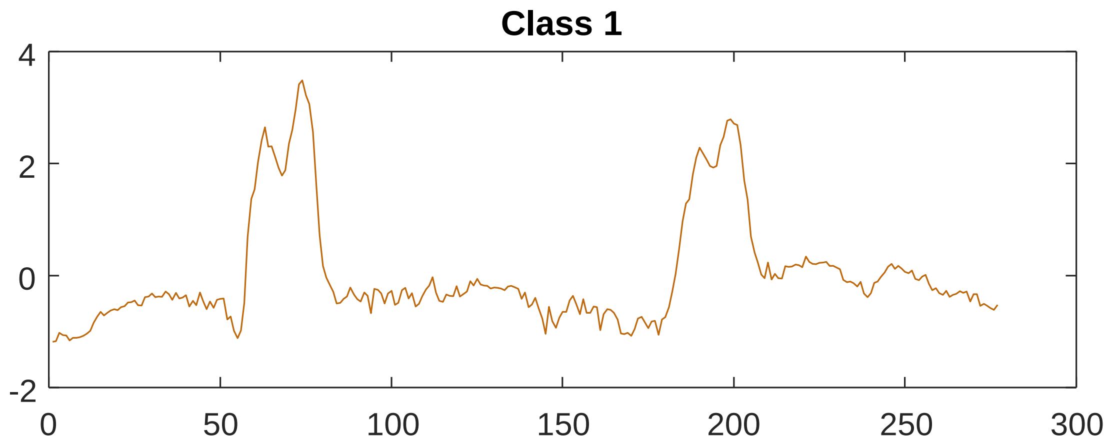
One exemplar per class,
with z-normalization



ToeSegmentation1

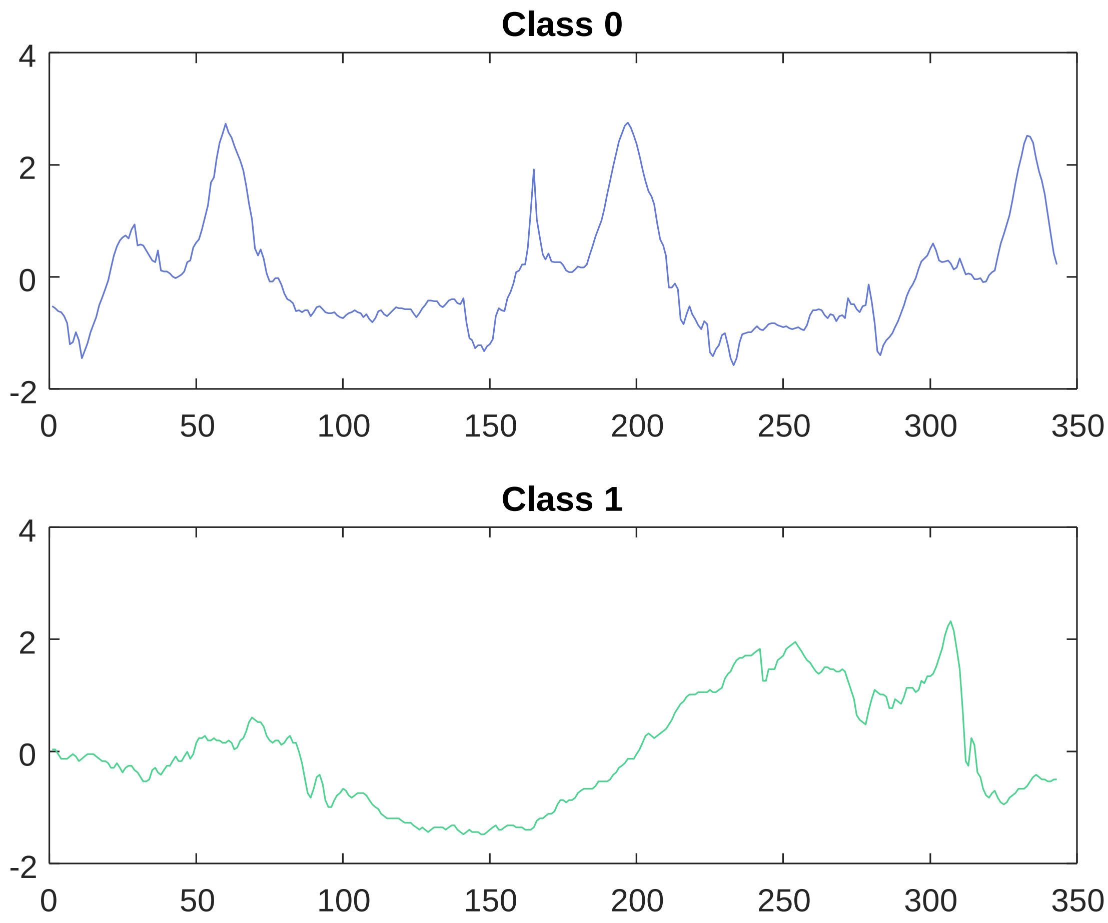
One exemplar per class,
with z-normalization





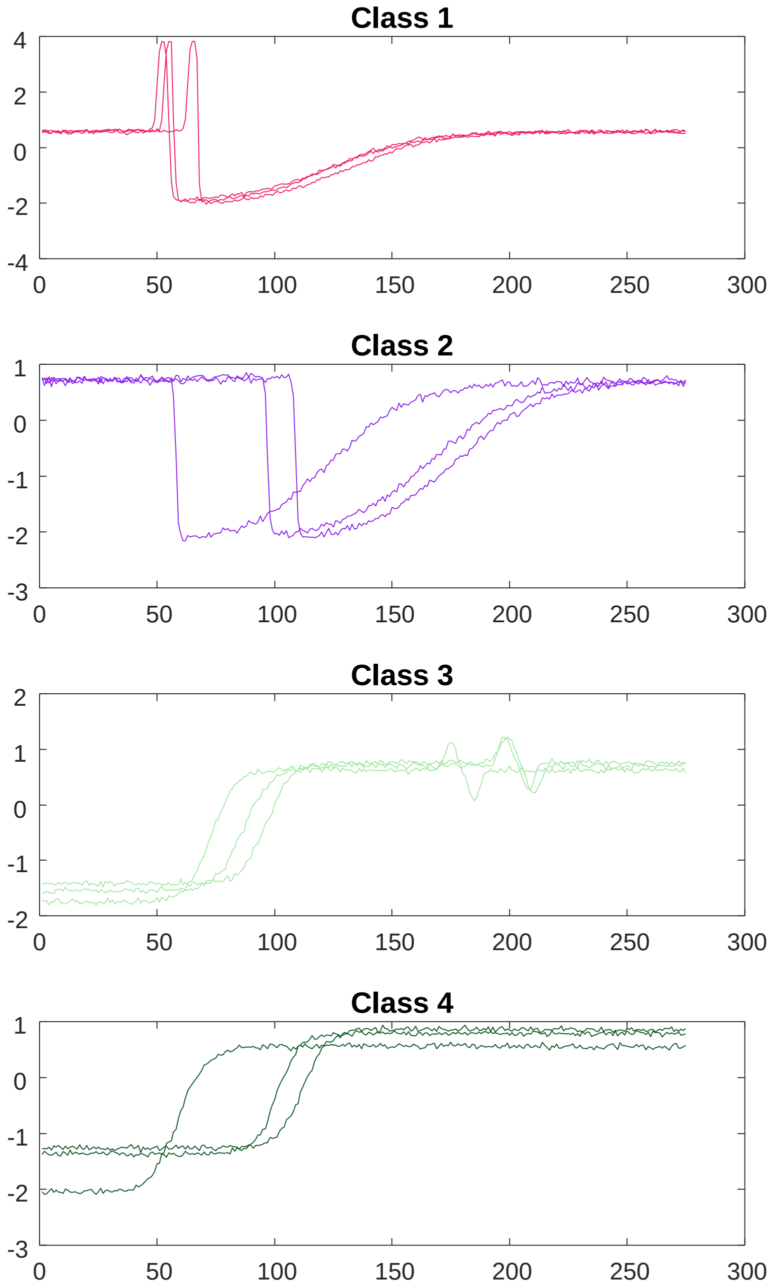
ToeSegmentation2

One exemplar per class,
with z-normalization



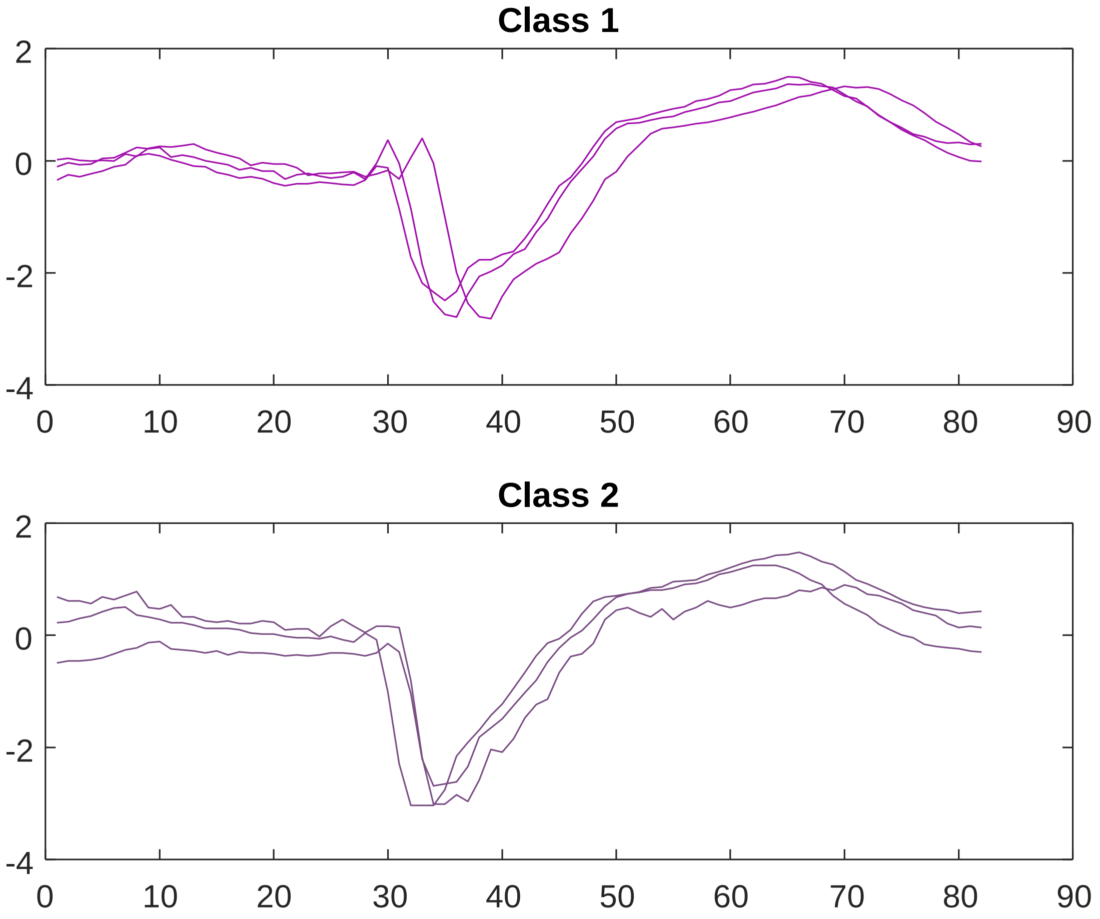
Trace

Three exemplars per class,
with z-normalization



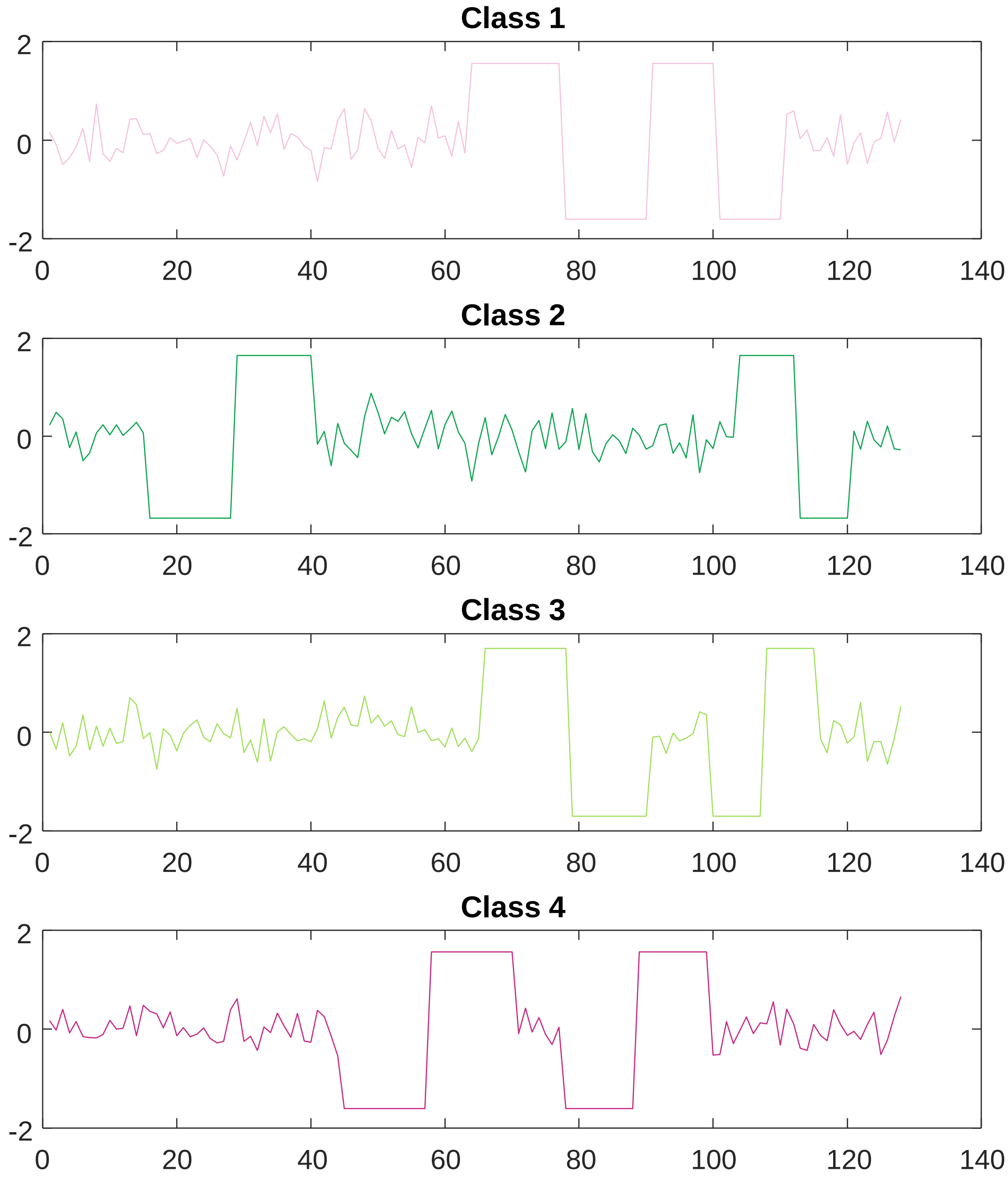
TwoLeadECG

Three exemplars per class,
with z-normalization



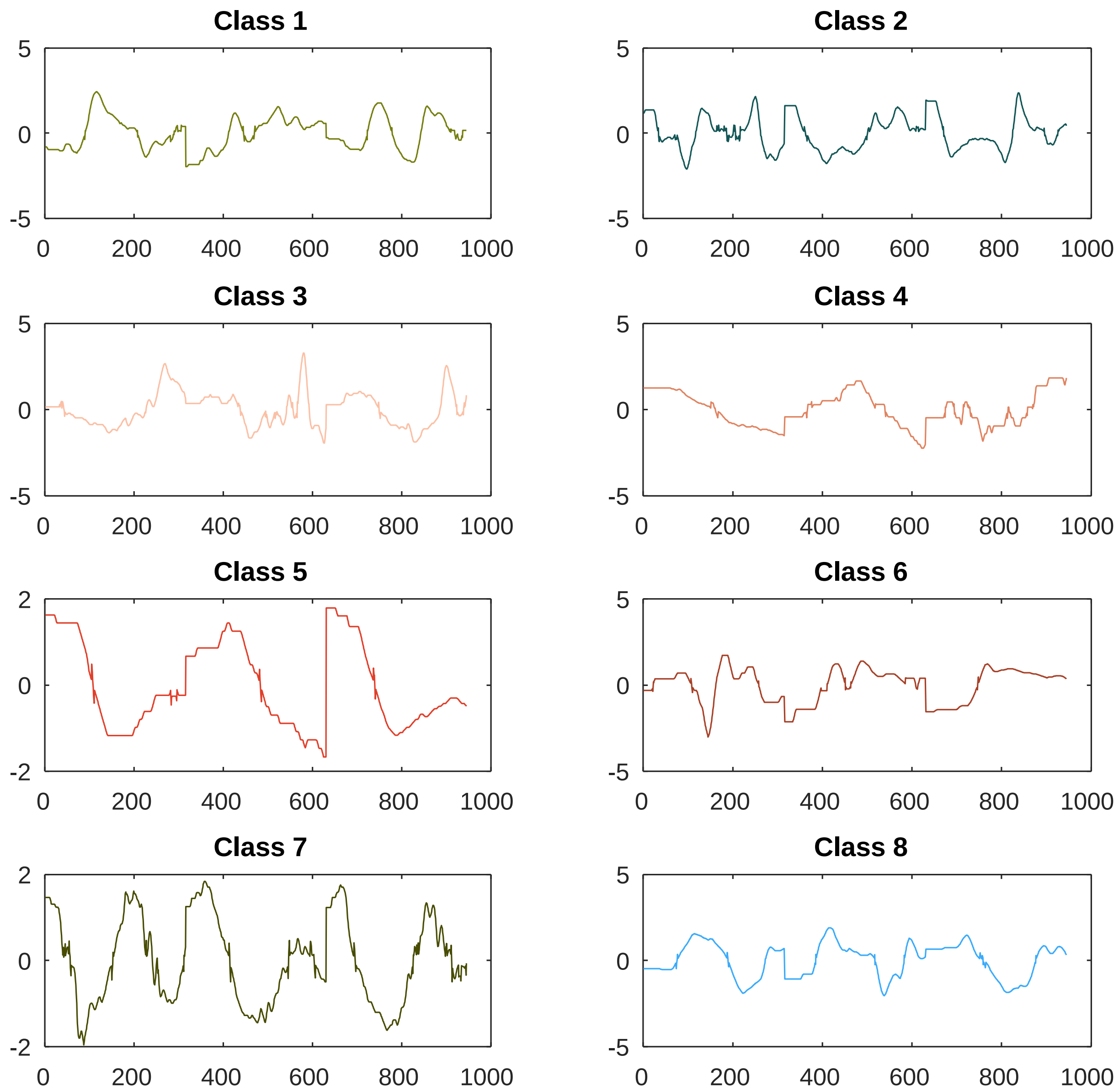
TwoPatterns

One exemplar per class,
with z-normalization



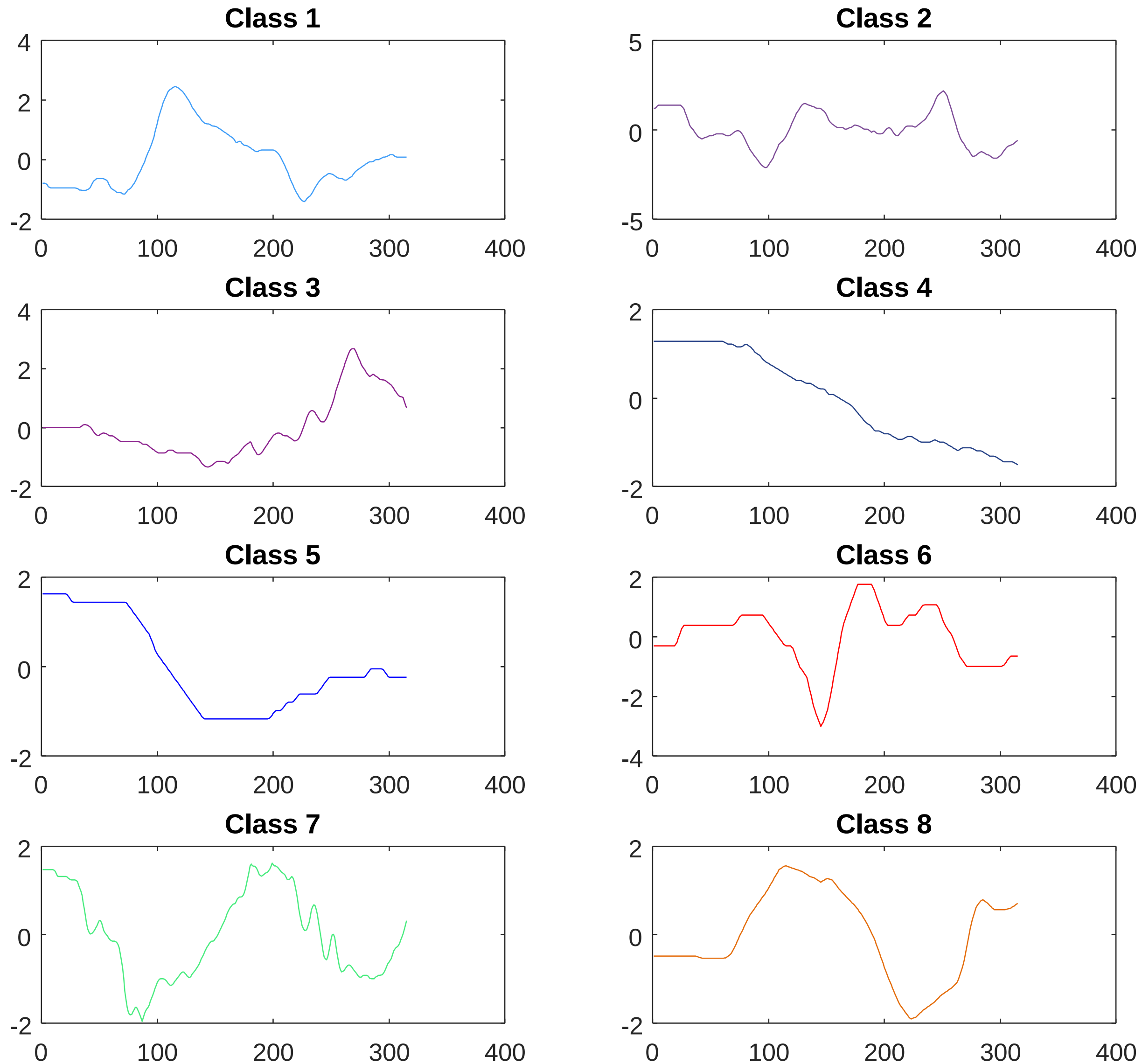
UWaveGestureLibraryAll

One exemplar per class,
with z-normalization



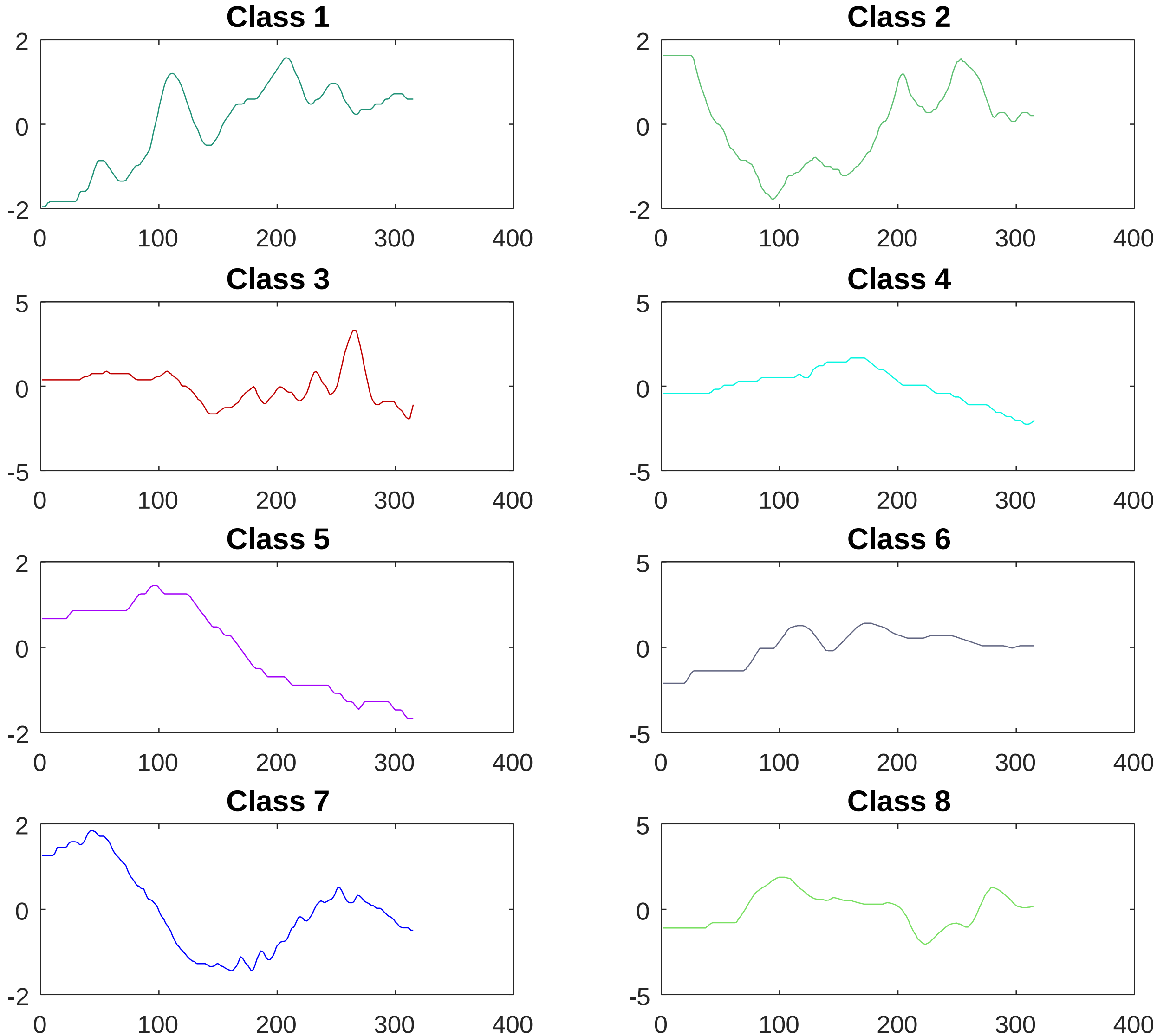
UWaveGestureLibraryX

One exemplar per class, with z-normalization



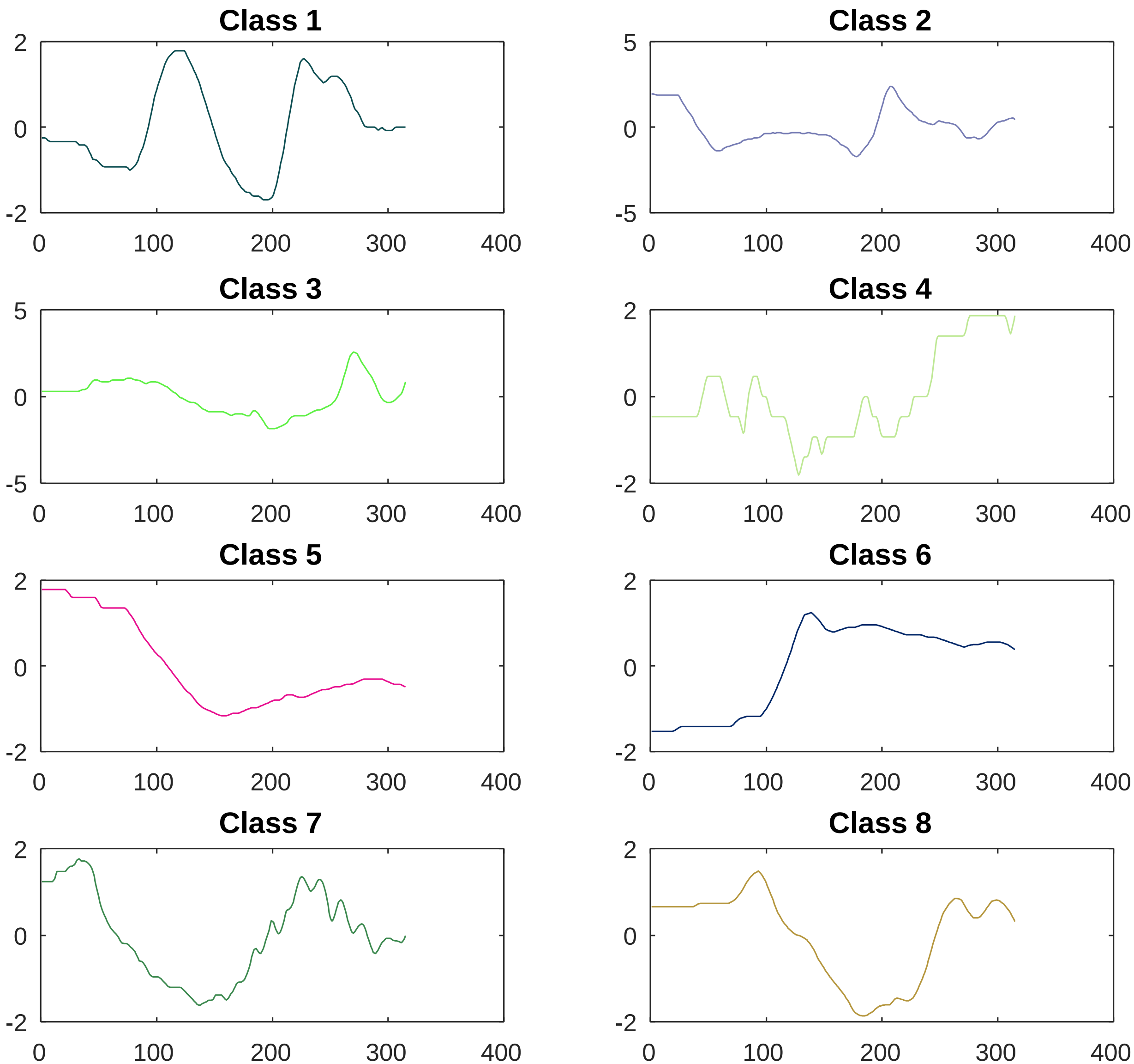
UWaveGestureLibraryY

One exemplar per class,
with z-normalization



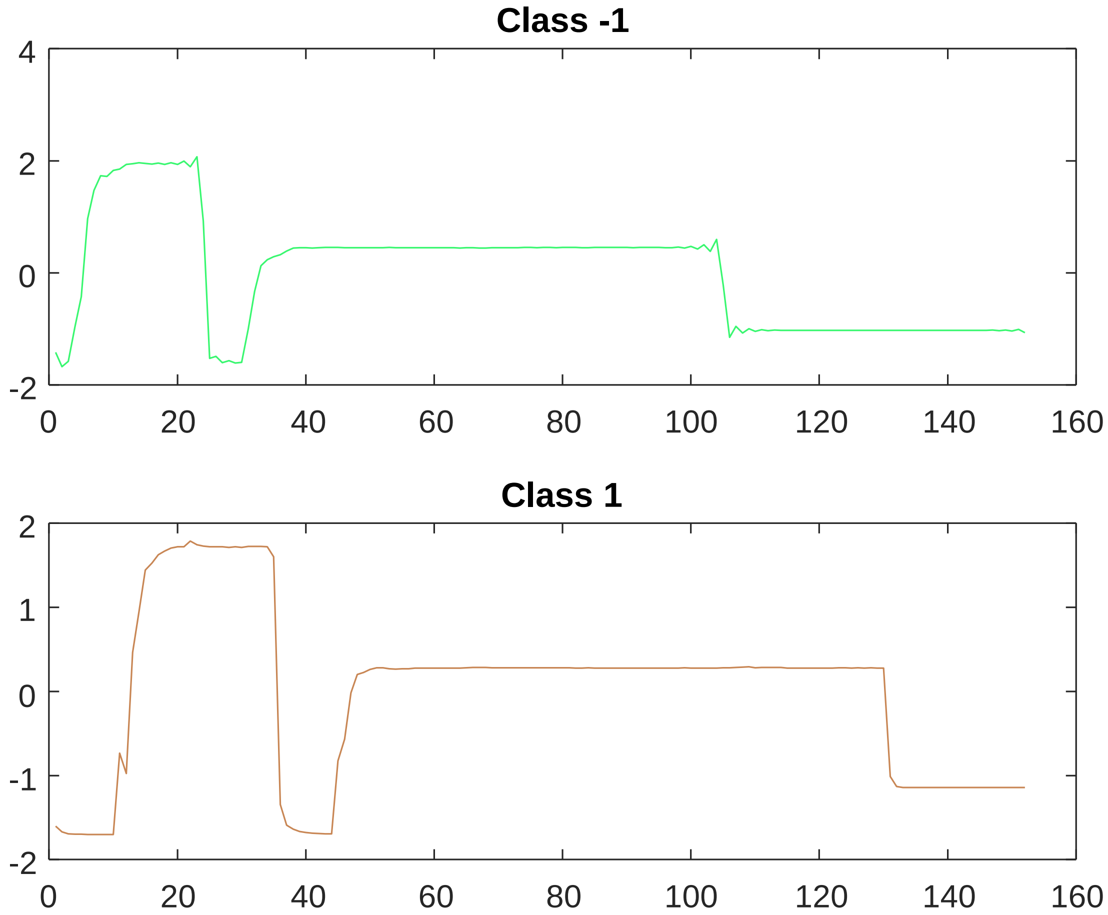
UWaveGestureLibraryZ

One exemplar per class,
with z-normalization



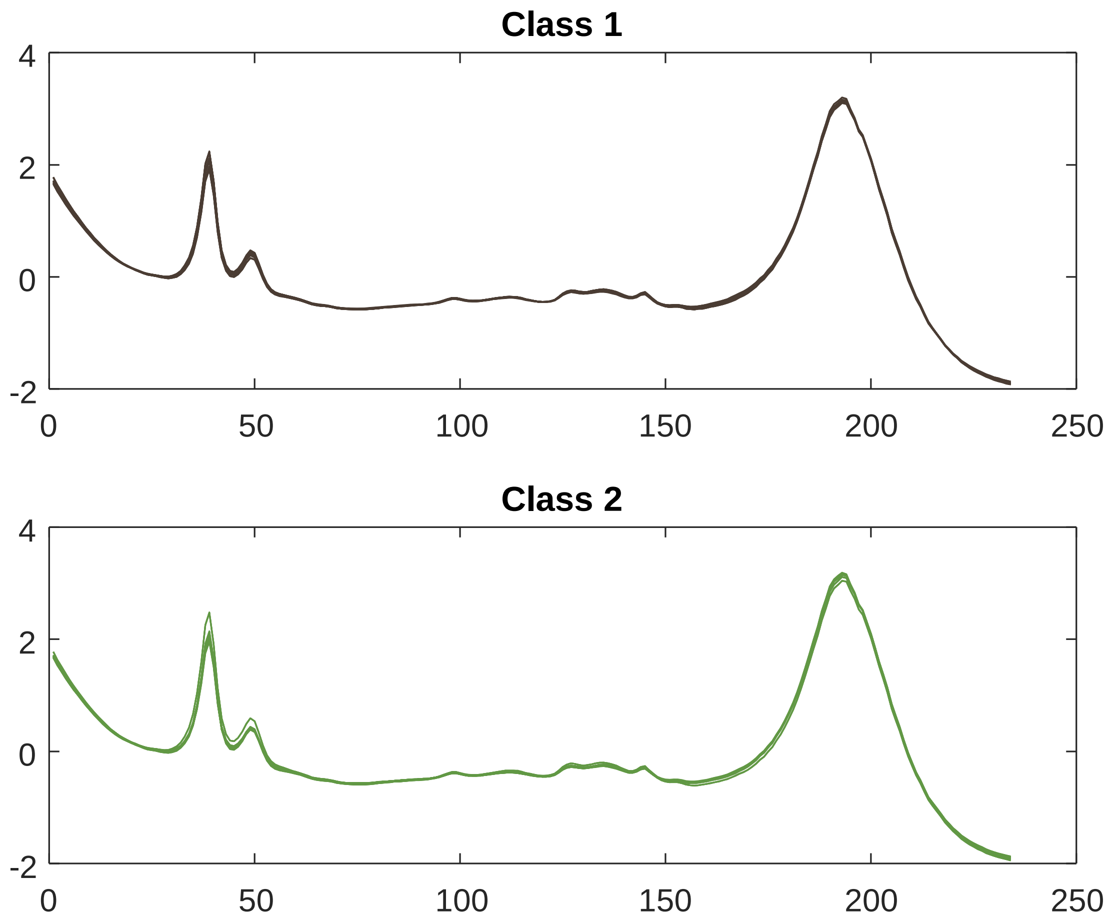
Wafer

One exemplar per class,
with z-normalization



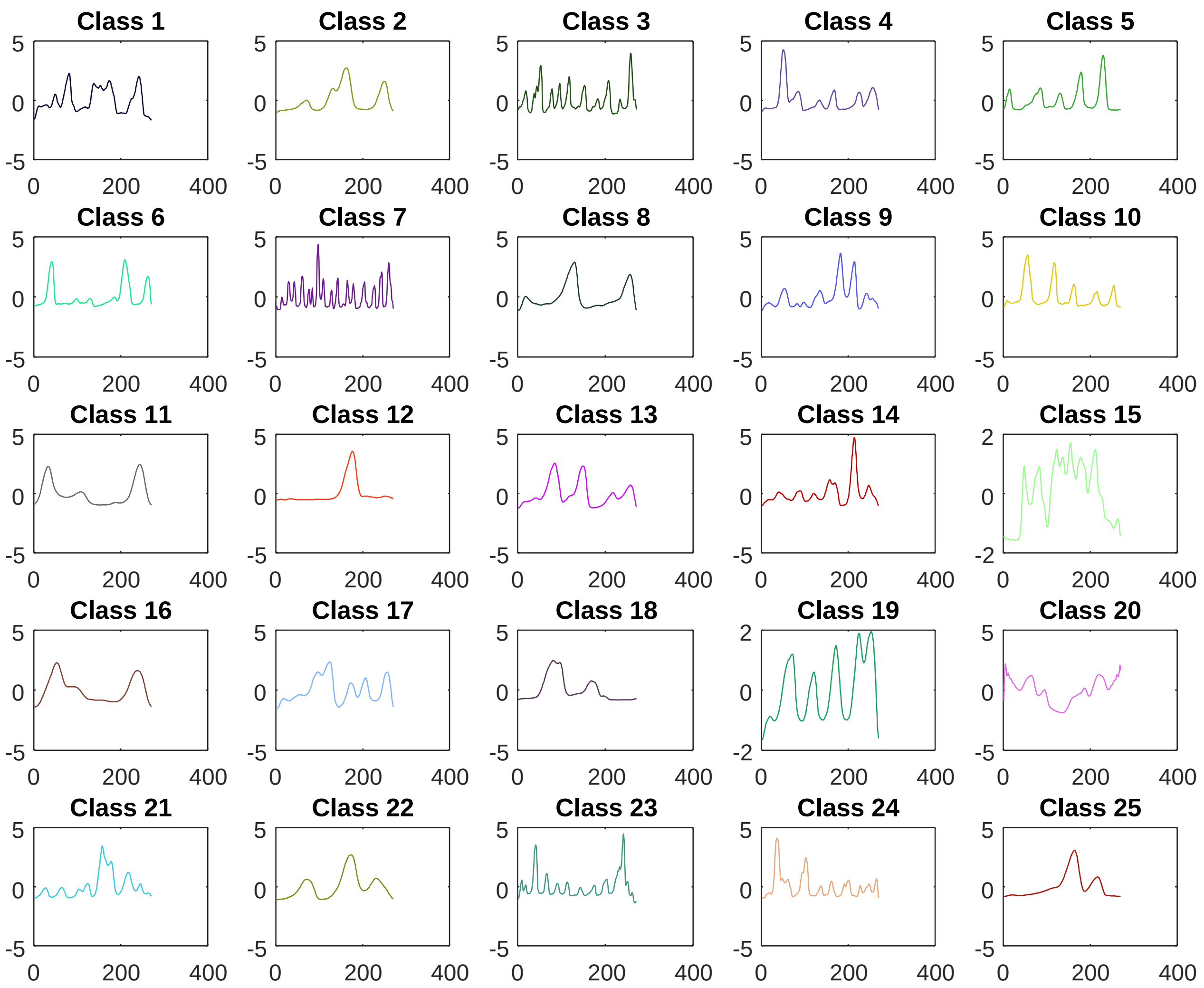
Wine

Twenty exemplars per
class, with z-normalization



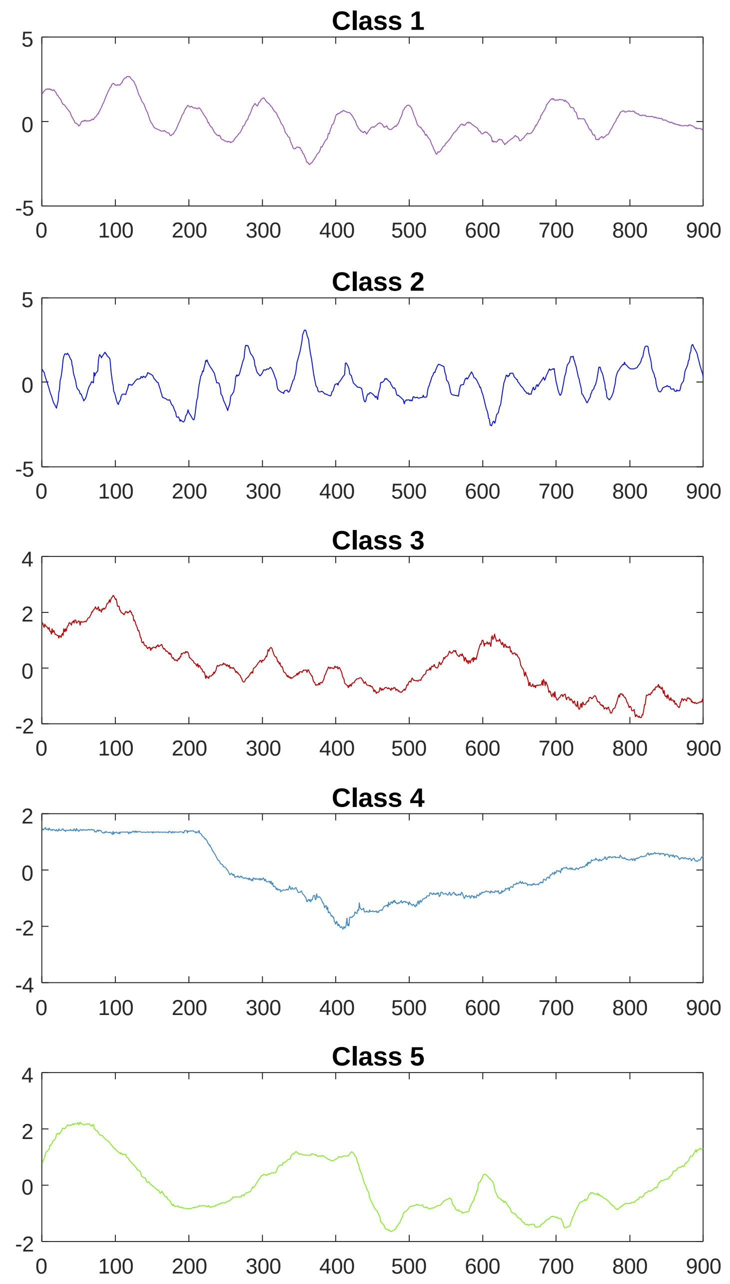
WordSynonyms

One exemplar per class,
with z-normalization



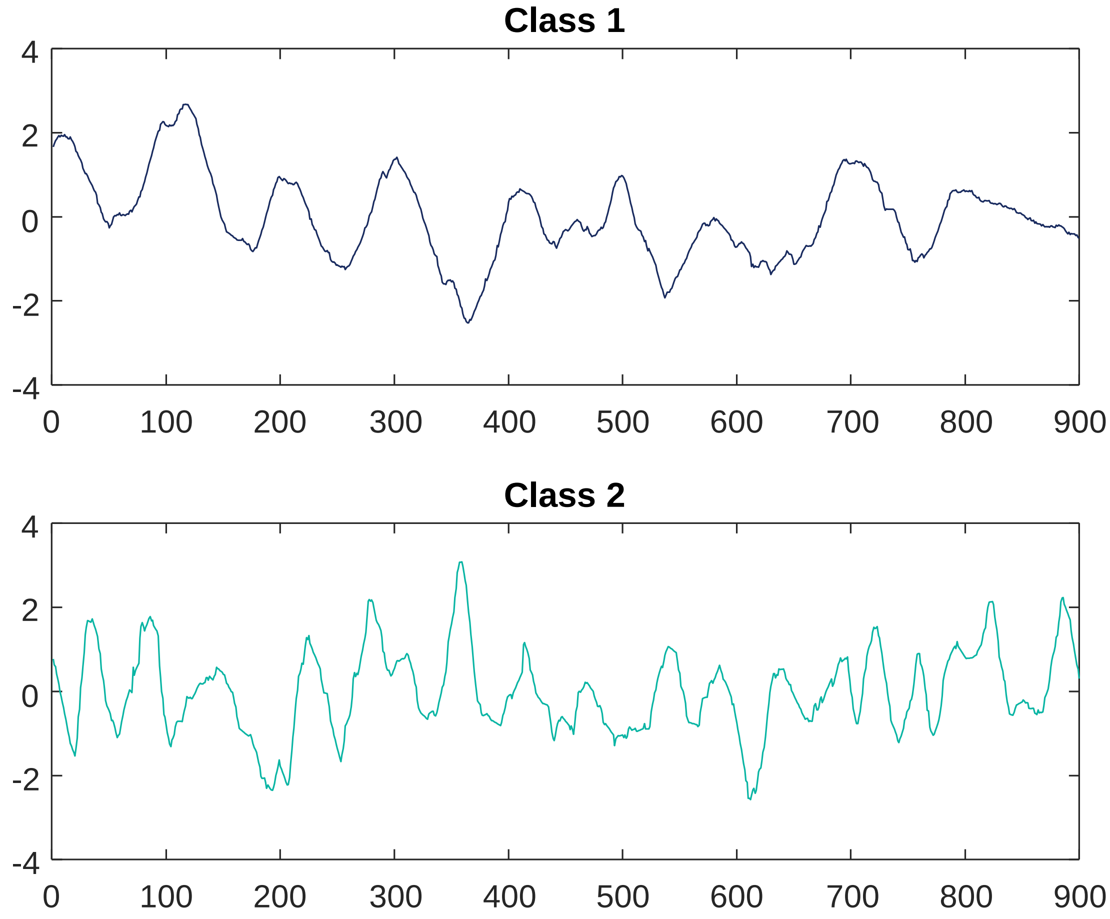
Worms

One exemplar per class,
with z-normalization



WormsTwoClass

One exemplar per class,
with z-normalization



Yoga

One exemplar per class,
with z-normalization