

Deep Dark Fantasy on MathWork Challenge

Yuzhe Yang*, Siyuan Liu, Laiyuan Zhang, and Sicheng Yao

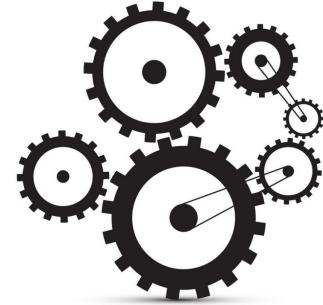
University of Southampton

School of Electronic and Computer Science

23/02/2020



What we have done so far?



- 1. We have finished challenges.
- 2. We collect our own training and testing data via MATLAB mobile.
- 3. We have visualized our newly build-up data and analyze potential features.
- 4. We use the build-in machine learning algorithm to classify human activities.
- 5. We adapt our model into MATLAB mobile to analyze the real time data.
- 6. Have eaten many delicious food and drinks and played many games!

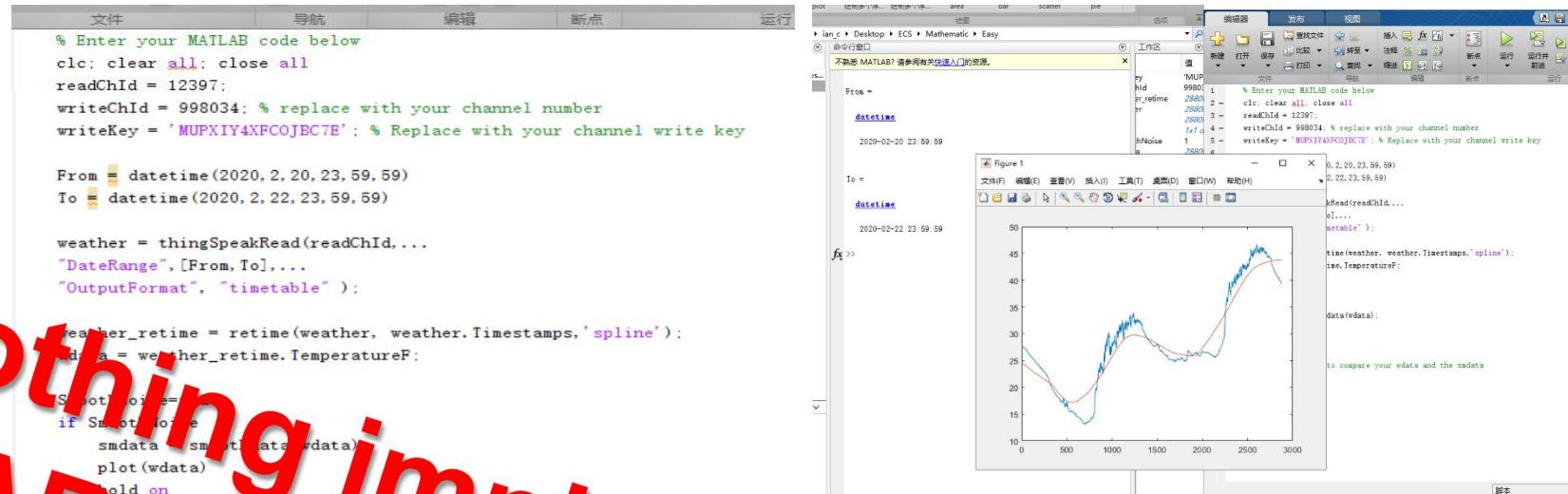
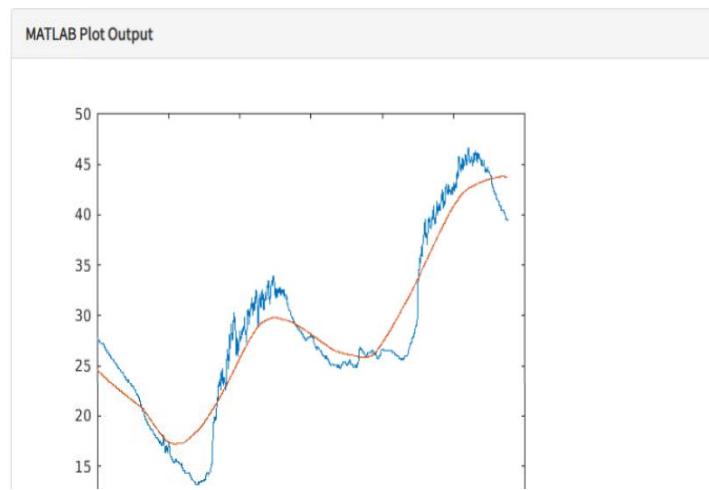
Data smooth in Easy Challenge

ThingSpeak™ Channels Apps Support Comm

```
2clc; clear all; close all
3readChId = 12397;
4writeChId = 998034; % replace with your channel number
5writeKey = 'MUPXIY4XFCOJECTE'; % Replace with your channel write key
6
7From = datetime(2020, 2, 20, 23, 59, 59)
8To = datetime(2020, 2, 22, 23, 59, 59)
9
10weather = thingSpeakRead(readChId, ...
11"DateRange", [From, To], ...
12"OutputFormat", "timetable");
13
14weather_retime = retime(weather, weather.Timestamps, 'spline');
15wdata = weather_retime.TemperatureF;
16
17SmoothNoise=true;
18if SmoothNoise
19    smdata = smoothdata(wdata);
20    plot(wdata)
21    hold on
22    plot(smdata)
23    % Include plot to compare your wdata and the smdata
24end
```

Save and Run Save!

Create a public URL: https://thingspeak.com/apps/matlab_visualizations/332079

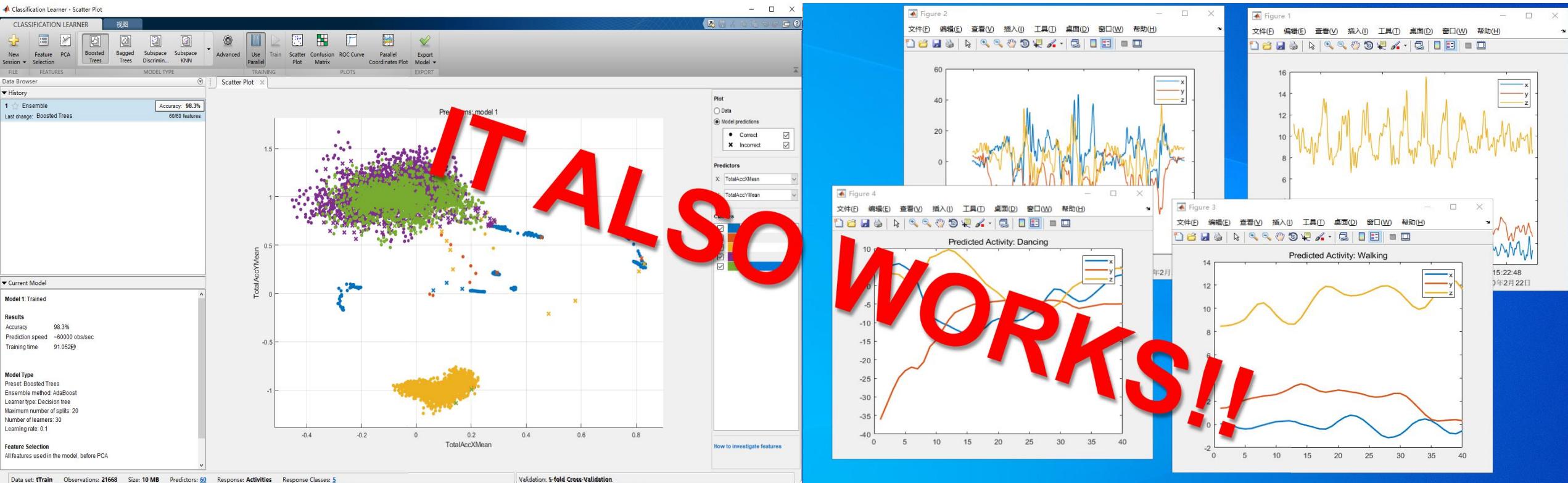


ThingSpeak for IoT Projects

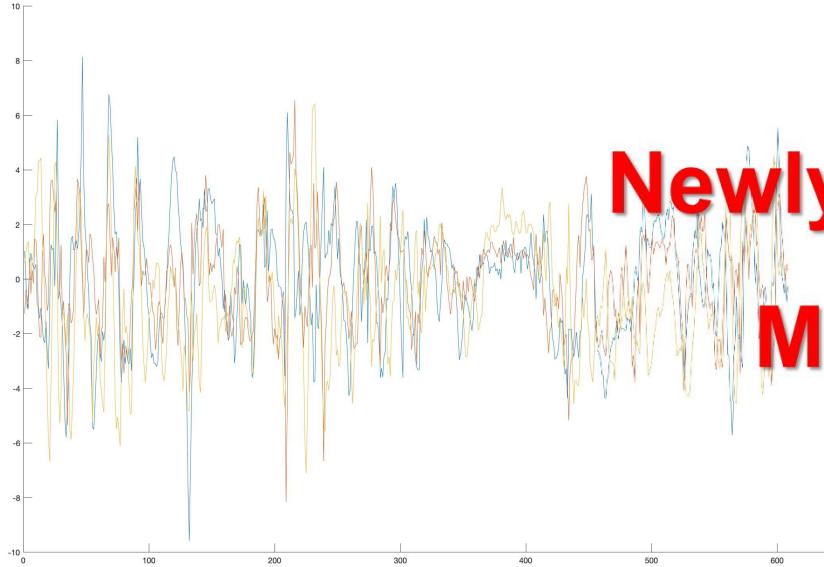
Data collection in the cloud with advanced data analysis using MATLAB

Channels Learn More

Train on provided dataset and predict on our raw data via MATLAB



Training and Predicting

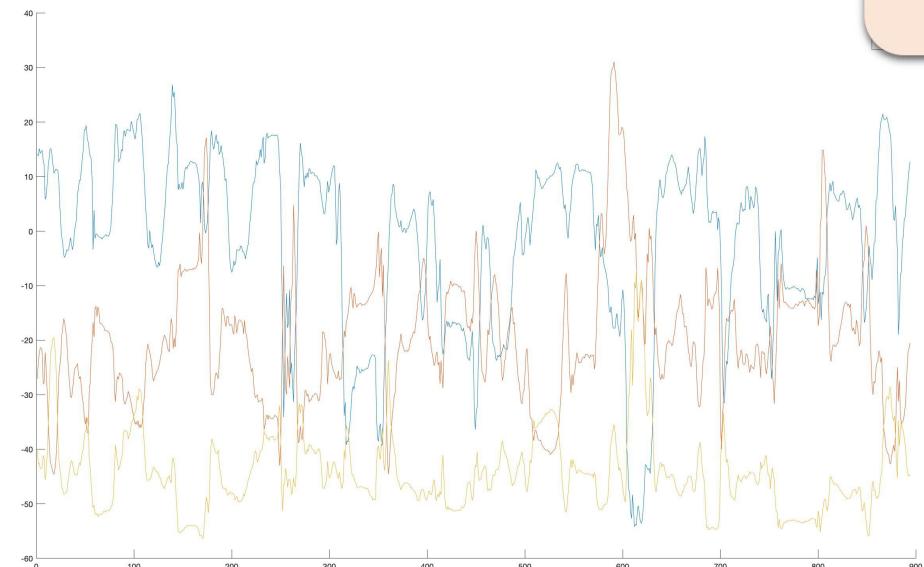


Dancing Magnitude

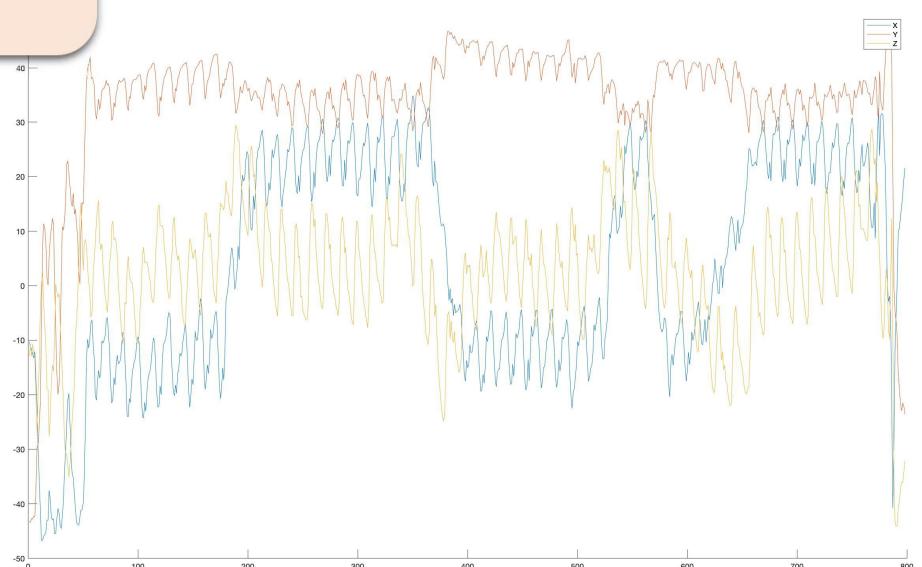


Running Magnitude

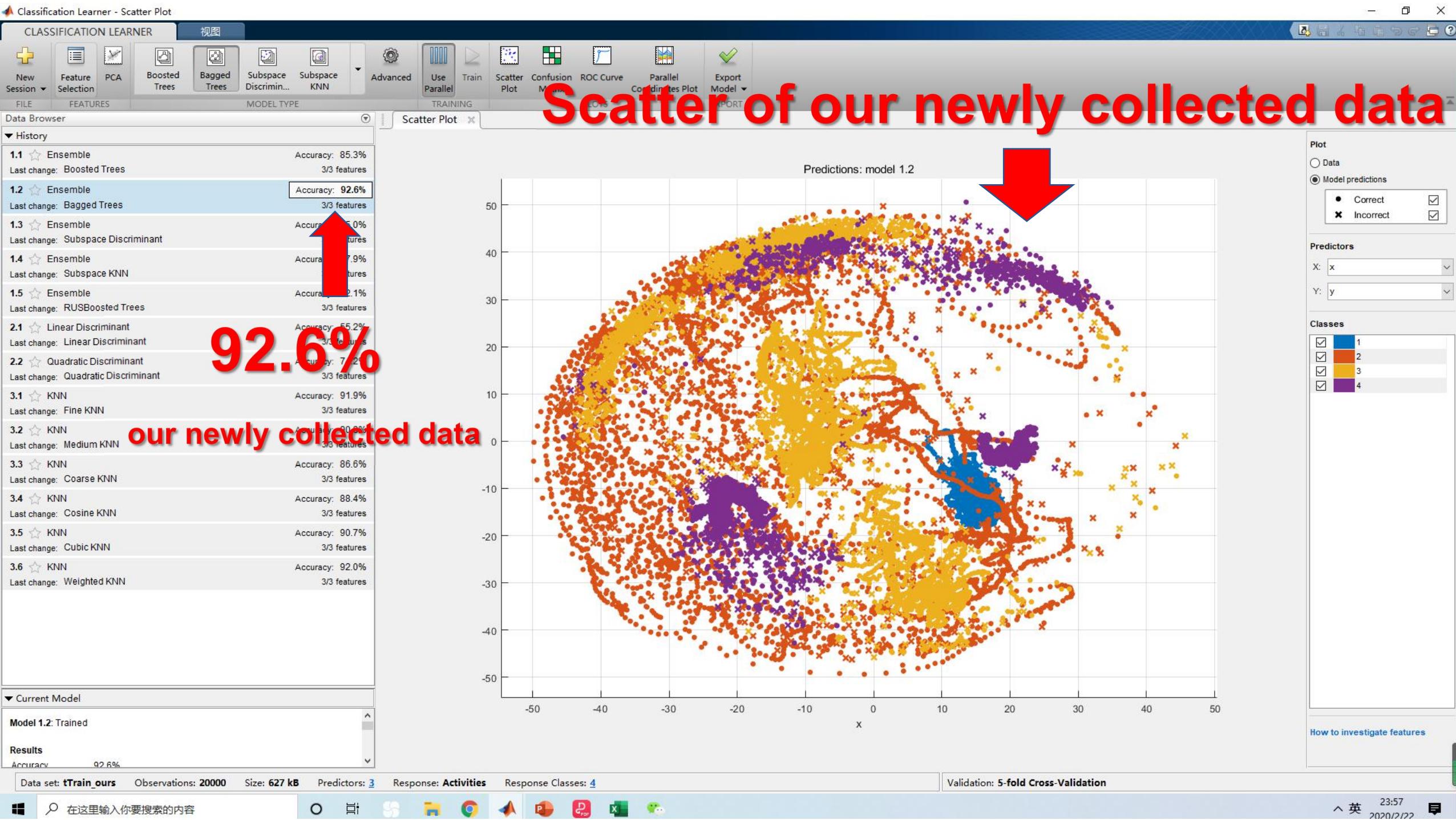
Magnitude



Sitting and Standing Magnitude



Walking Magnitude



Human Action Classification via “Sylvan Gated Bagged Trees”

```
1 - clear all; clc; close all; % Keep super clean.
2 - interval = 8; % How long it is between two detections, in seconds.
3 - decetion_times = 8; % How many times do you want to detect.
4 - trainedClassifier = load('trainedModelBagTree'); % Load .mat file from MATLAB Deive.
5 - for external_loop = 1:1:decetion_times+1
6 -     if external_loop ~= 1 % Skip the first loop for initialisation.
7 -         [a,t] = magfieldlog(m); % a is the input matrix for the model.
8 -         % The size of a should be equal to interval * frequenc,
9 -         % however due to the limited performance on mobile devices,
10 -         % it is apt to be marginally less than the estimation value.
11 -         clear m; % Empty the log variable in every loop for saving memory.
12 -         scattered_coe = abs((max(a)-min(a))/mean(a))
13 -         disp(datestr(now))
14 -         a = array2table(a);
15 -         featlabels_test = {'x', 'y', 'z'};
16 -         a.Properties.VariableNames = featlabels_test;
17 -         pred = trainedClassifier.trainedModelBagTree.predictFcn(a);
18 -         pred = pred';
19 -         pred = pred(:)';
20 -         tab_pred = tabulate(pred);
21 -         if scattered_coe >= exp(-1)
22 -             disp(tab_pred)
23 -         else
24 -             disp('User lost himself.');
25 -         end
26 -     end
27 -     m = mobiledev; % Turn on and initialise the receiver.
28 -     m.MagneticSensorEnabled = 1; % Ensure the magnetic sensor is on.
29 -     m.Logging = 1; % Start logging.
30 -     tinit_time=clock; % Start timing.
31 -     % Timer.
32 -     for inner_loop = 1:1:1073741824 % Very bad design.
33 -         if etime(clock,tinit_time)>interval
34 -             break
35 -         end
36 -     end
37 -     fprintf('\n')
38 - end
```

Gated Function:

if (max_value + min_value)/mean_value >1/e

- Motivation:**
Use nature principle to solve nature problems

- Contribution:**
Eliminate misclassification

Results illustration

```

≡ ⏴ ⏵
0.7163

23-Feb-2020 11:20:13
1.0000 4.0000 5.1948
2.0000 20.0000 25.9740
3.0000 53.0000 68.8312

scattered_coe =
3.4422

23-Feb-2020 11:20:24
1.0000 0 0
2.0000 74.0000 92.5000
3.0000 0 0
4.0000 6.0000 7.5000

scattered_coe =
0.2766

23-Feb-2020 11:20:32
User lost himself.

scattered_coe =
3.7991

23-Feb-2020 11:20:45
1.0000 0 0
2.0000 63.0000 81.8182
3.0000 11.0000 14.2857
4.0000 3.0000 3.8961

>> Enter command here...

```



Running Results on
MATLAB mobile

Confusion Matrix

Data Browser

▼ History

1.1 ★ Ensemble	Accuracy: 85.3%
Last change: Boosted Trees	3/3 features
1.2 ★ Ensemble	Accuracy: 92.6%
Last change: Bagged Trees	3/3 features
1.3 ★ Ensemble	Accuracy: 55.0%
Last change: Subspace Discriminant	3/3 features
1.4 ★ Ensemble	Accuracy: 87.9%
Last change: Subspace KNN	3/3 features
1.5 ★ Ensemble	Accuracy: 82.1%
Last change: RUSBoosted Trees	3/3 features
2.1 ★ Linear Discriminant	Accuracy: 55.2%
Last change: Linear Discriminant	3/3 features
2.2 ★ Quadratic Discriminant	Accuracy: 74.2%
Last change: Quadratic Discriminant	3/3 features
3.1 ★ KNN	Accuracy: 91.9%
Last change: Fine KNN	3/3 features
3.2 ★ KNN	Accuracy: 90.8%
Last change: Medium KNN	3/3 features
3.3 ★ KNN	Accuracy: 86.6%
Last change: Coarse KNN	3/3 features
3.4 ★ KNN	Accuracy: 88.4%
Last change: Cosine KNN	3/3 features
3.5 ★ KNN	Accuracy: 90.7%
Last change: Cubic KNN	3/3 features
3.6 ★ KNN	Accuracy: 92.0%
Last change: Weighted KNN	3/3 features

Cross-Validation Accuracy from Different Models

J.P.Morgan

arm



Acknowledge!

FACTSET

