Padaco - An open source software tool for exploring accelerometer data

Hyatt Moore IV, PhD

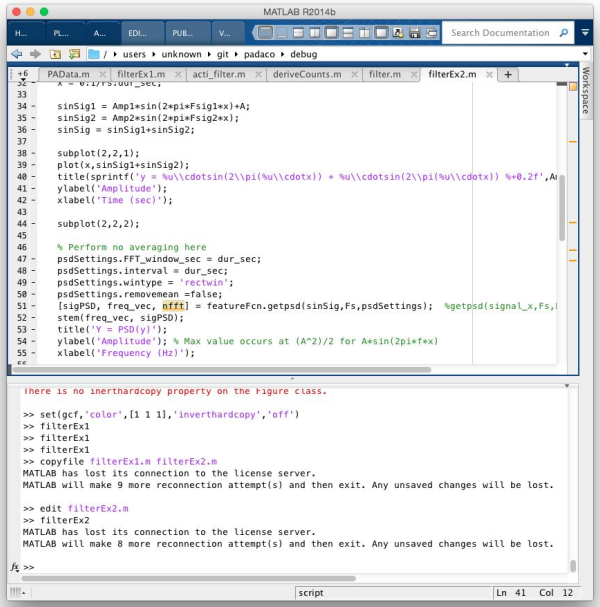
Developed at Stanford University's
Solution Science Lab and Quantitative Science Unit

Outline

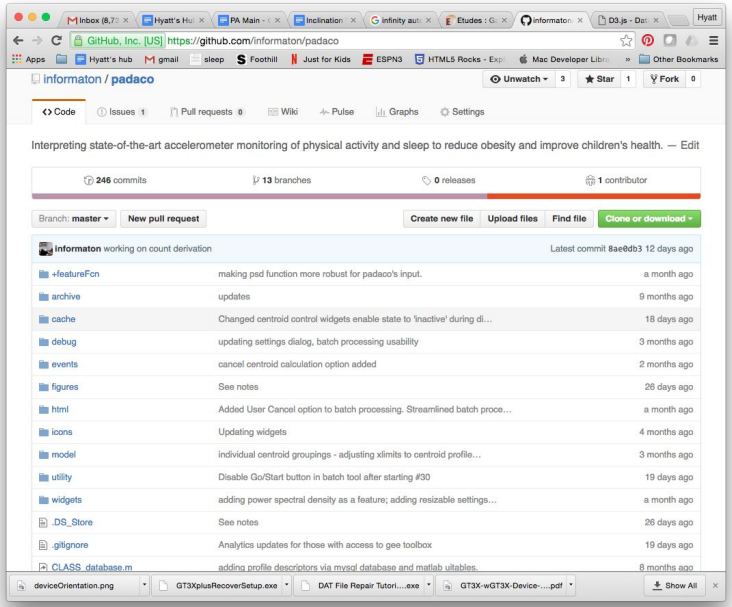
1. Overview
2. Components
   - Single study view
   - Batch mode processing
   - Clustering and results views
3. Demo
4. Questions

Padaco : Physical activity data analysis for assessing childhood obesity

Open source program written in MATLAB for data visualization and analysis of Actigraph GT3X sensor data and exploring patterns in accelerometry data.



MATLAB IDE



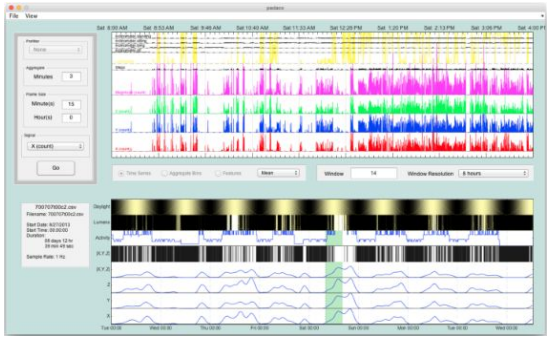
Github open source repository

www.github.com/informaton/padaco

Components

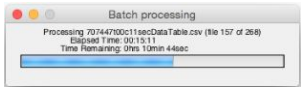
1. Single study

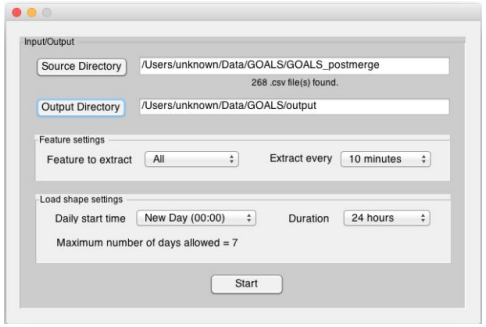
- Visualize one Actigraph file
  - Top figure shows time series view of a selected region of the study
  - Bottom figure shows overall view of the study from start to stop
- Configure parameters and observe features



2. Batch processing

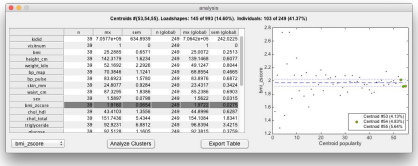
- Process multiple Actigraph files
  - Select directory of files
  - Configure feature extraction parameters
- Processing takes ~ 25 minutes for 268 files (counts)

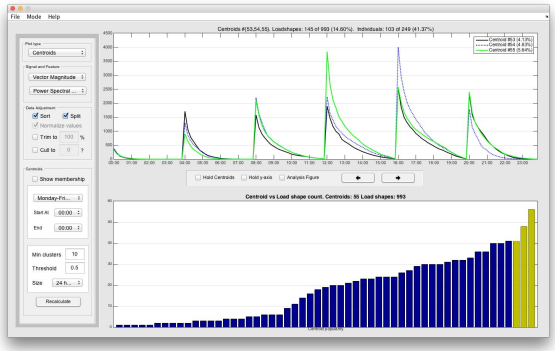




3. Group analysis

- Visualize collective features
  - Feature clustering used to define activity shapes
  - User interface used to adjust evaluation time frames
- Subject variables to feature profile comparisons





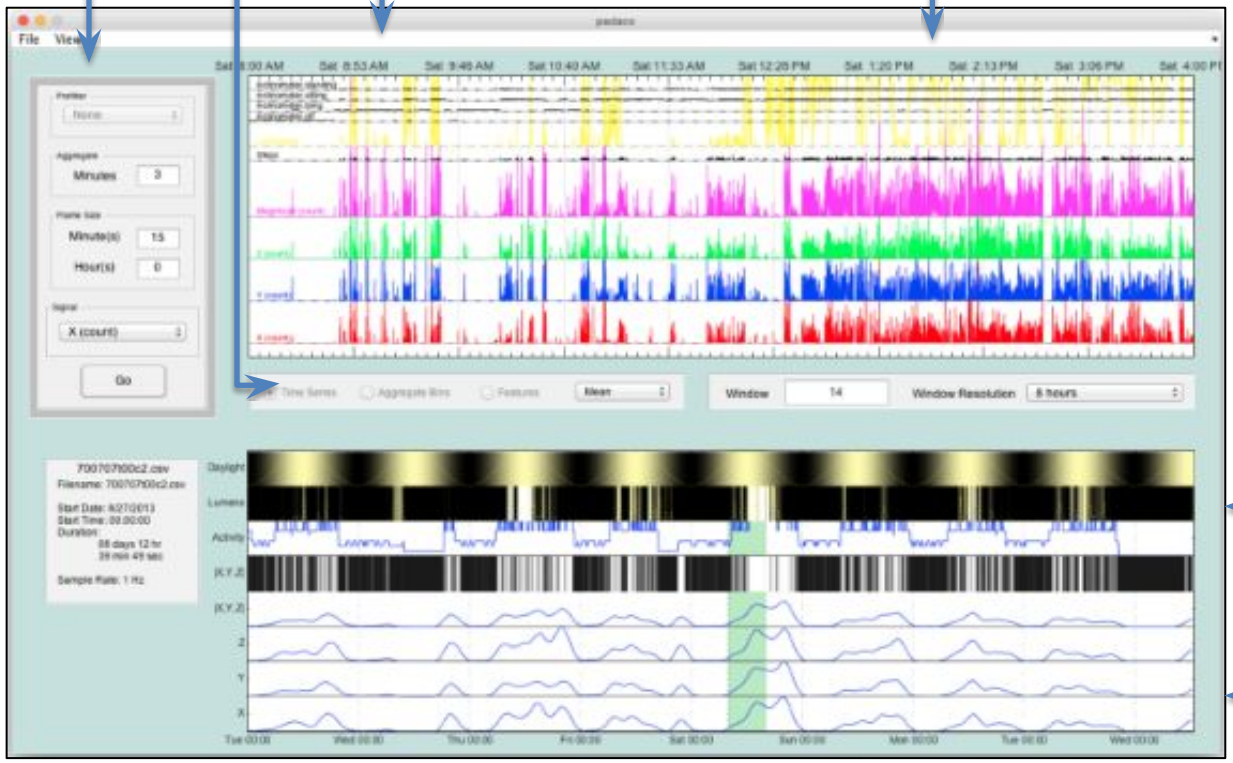
Single study view

Feature/frame controls

Window view controls

Local view (two options)

1. Actigraph signals: acceleration, steps, inclination, luminance
2. Feature function results for a selected Actigraph signal.

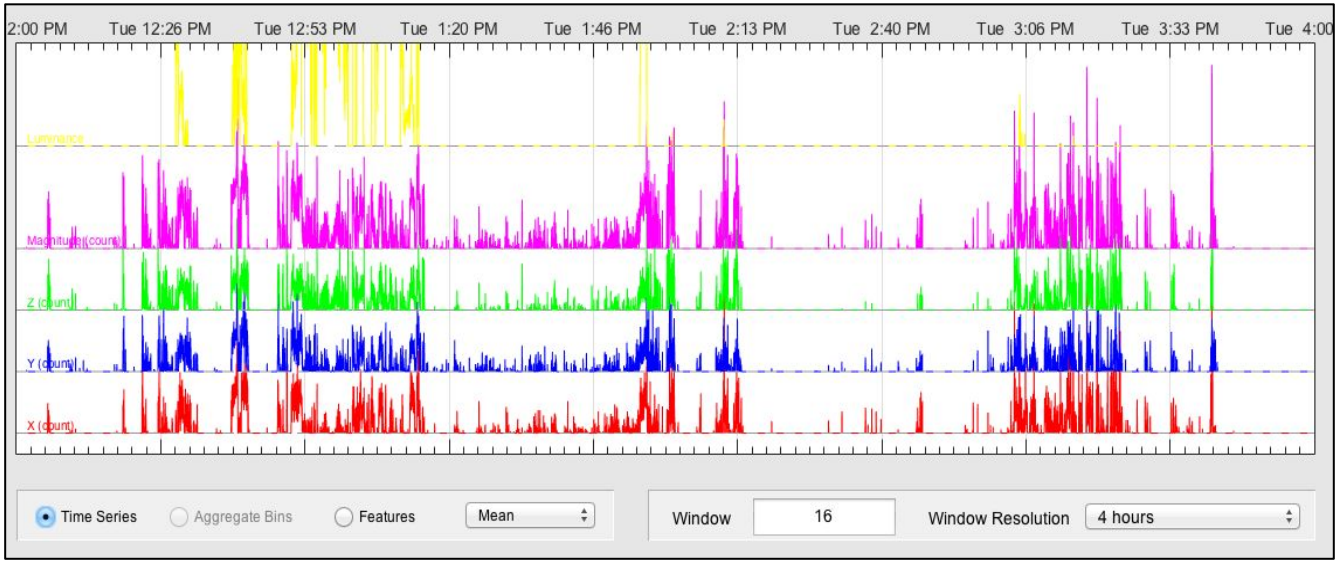


Global view summarizes:

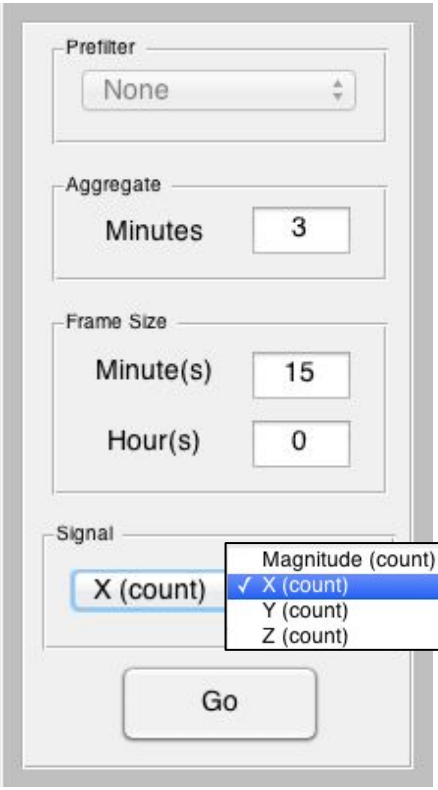
1. Estimated sunlight
2. Luminance
3. Estimated activity state
4. Selected feature function:
    1. Vector magnitude (intensity)
    2. Vector magnitude
    3. X-axis
    4. Y-axis
    5. Z-axis
5. Local view reference (green bar)

Features and frames

Frame – Consecutive, non-overlapping interval of accelerometer samples.
Feature function – a function that takes a frame as input and returns a feature value as its output.
Feature value – the single, numeric value returned by a feature function.
Feature vector – collection of consecutive features.
Profile – feature vector for a 24-hour period.

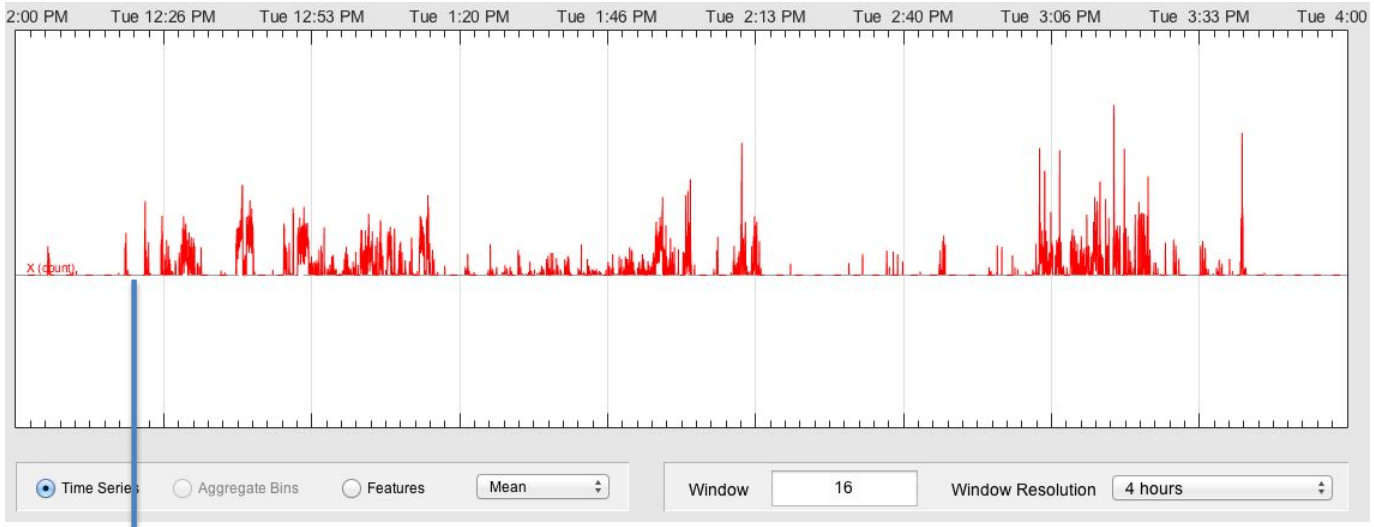


4 hours of GT3X+ data (14,400 seconds) produces feature vectors with 16 values when using a 15 minute frame.

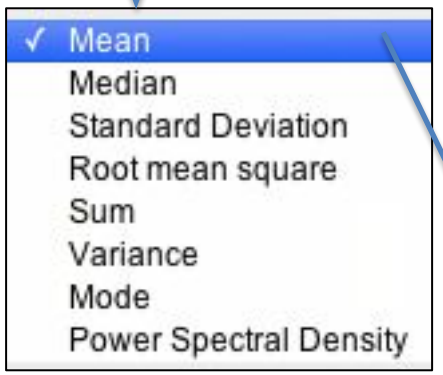


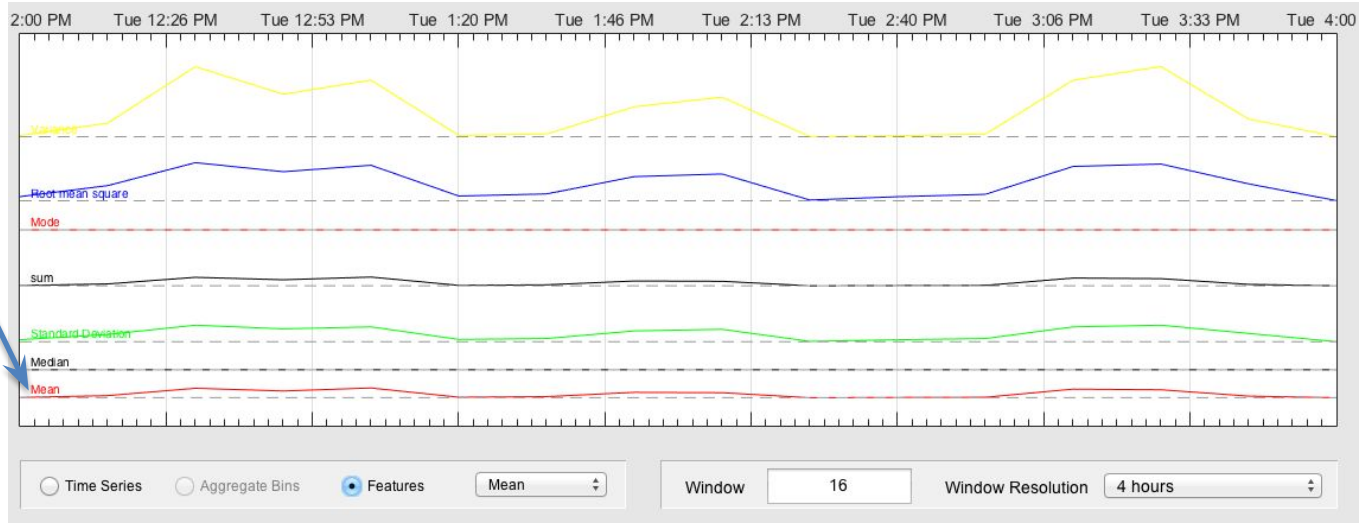
A 15 minute frame size will result in 1 feature every 900 seconds.

Example – mean as feature function



4 hours of x-axis accelerometer counts (14,400 counts)

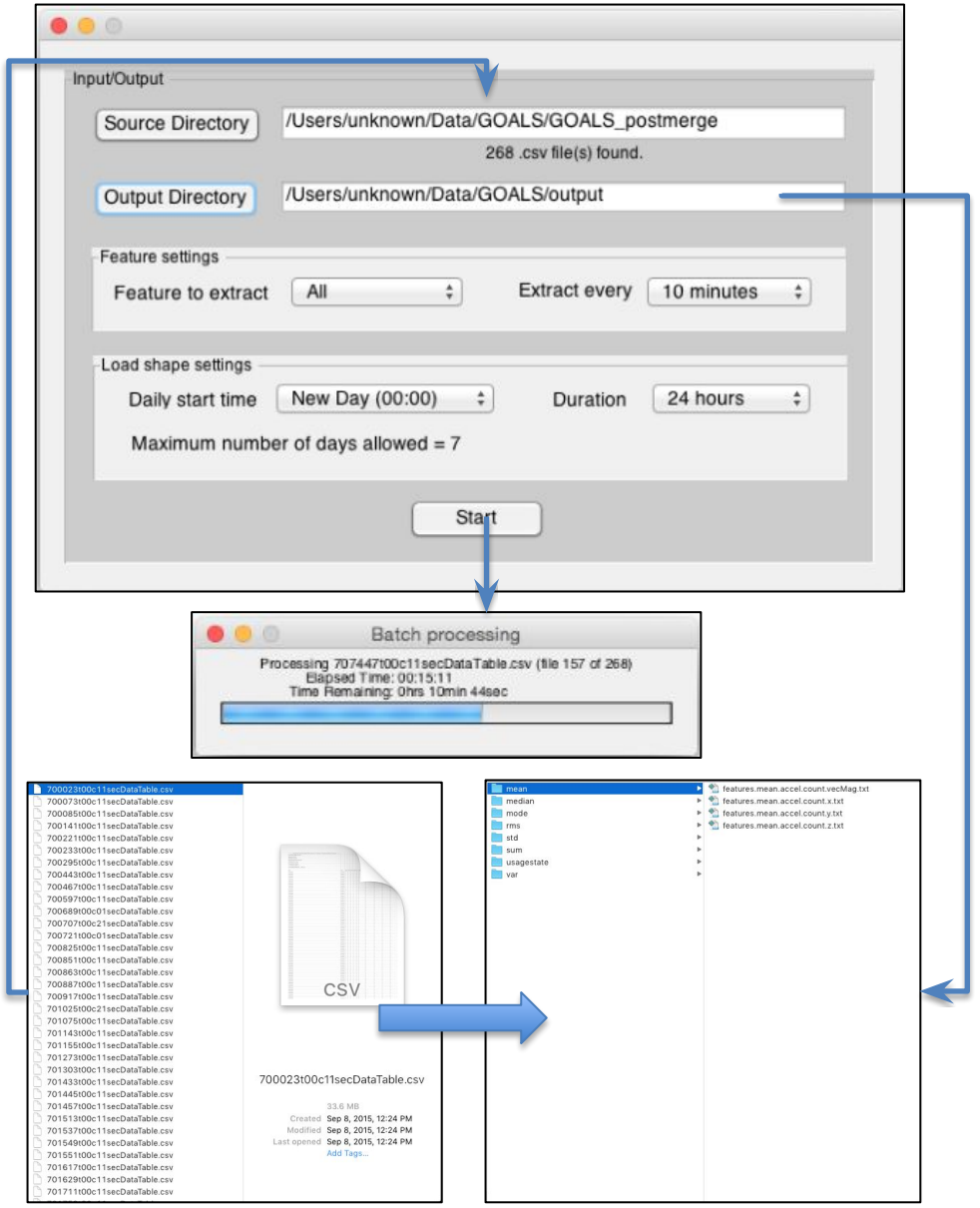




Various corresponding feature vectors (each of length 16)

Batch processing

- Determine frame size and feature function
- Reduce - process feature vectors for each accelerometer axes, of each study, and save to disk. Four files per feature (x, y, z, and vector magnitude)



Count data to features vectors

Subject 1 - Actigraph count data (.csv)

---------- Data Table File Created By ActiGraph GT3XPlus ...
Serial Number: CLE1B42120019
Start Time 00:00:00
Start Date 6/15/2013
Epoch Period (hh:mm:ss) 00:00:01
Download Time 16:11:52
Download Date 6/28/2013
Current Memory Address: 0
Current Battery Voltage: 3.87 Mode = 61

Date, Time, Axis1,Axis2,Axis3,Steps,Lux,Inclinometer Off, ...
6/15/2013,00:00:00 0 0,0,0,0,1,0,0,0,0
6/15/2013,00:00:01 0 0,0,0,0,1,0,0,0,0
6/15/2013,00:00:02 0 0,0,0,0,1,0,0,0,0
6/15/2013,00:00:03 0 0,0,0,0,1,0,0,0,0
…
6/28/2013,16:11:49 0 0,0,0,35,1,0,0,0,0
6/28/2013,16:11:50 0 0,0,0,35,1,0,0,0,0
6/28/2013,16:11:51 0 0,0,0,35,1,0,0,0,0

0 0 0 ….
0
0
… 0 0

Signals presented:
Axis 1, 2, 3 and vector magnitude
Luminance
Steps
Inclination (standing, sitting, lying, off)

1. Select a signal
2. Set duration of frames
3. Place the data into consecutive, equal length, frames.
4. Select a feature function: e.g. mean value $y(k)=1/N \cdot \sum x(n)$
5. Apply the feature function to each frame to produce one feature value per frame
6. Organize feature values into consecutive day and save to disk.

Each Actigraph file is transformed to feature vectors

Subject 258 - Actigraph count data (.csv)

---------- Data Table File Created By ActiGraph GT3XPlus ActiLife v6.10.2 Firmware v2.2.1 date format M/d/yyyy Filter Normal

Subject 257 - Actigraph count data (.csv)

---------- Data Table File Created By ActiGraph GT3XPlus ActiLife v6.10.2 Firmware v2.2.1 date format M/d/yyyy Filter Normal

Subject … - Actigraph count data (.csv)

---------- Data Table File Created By ActiGraph GT3XPlus ActiLife v6.10.2 Firmware v2.2.1 date format M/d/yyyy Filter Normal ----------

Subject 2 - Actigraph count data (.csv)

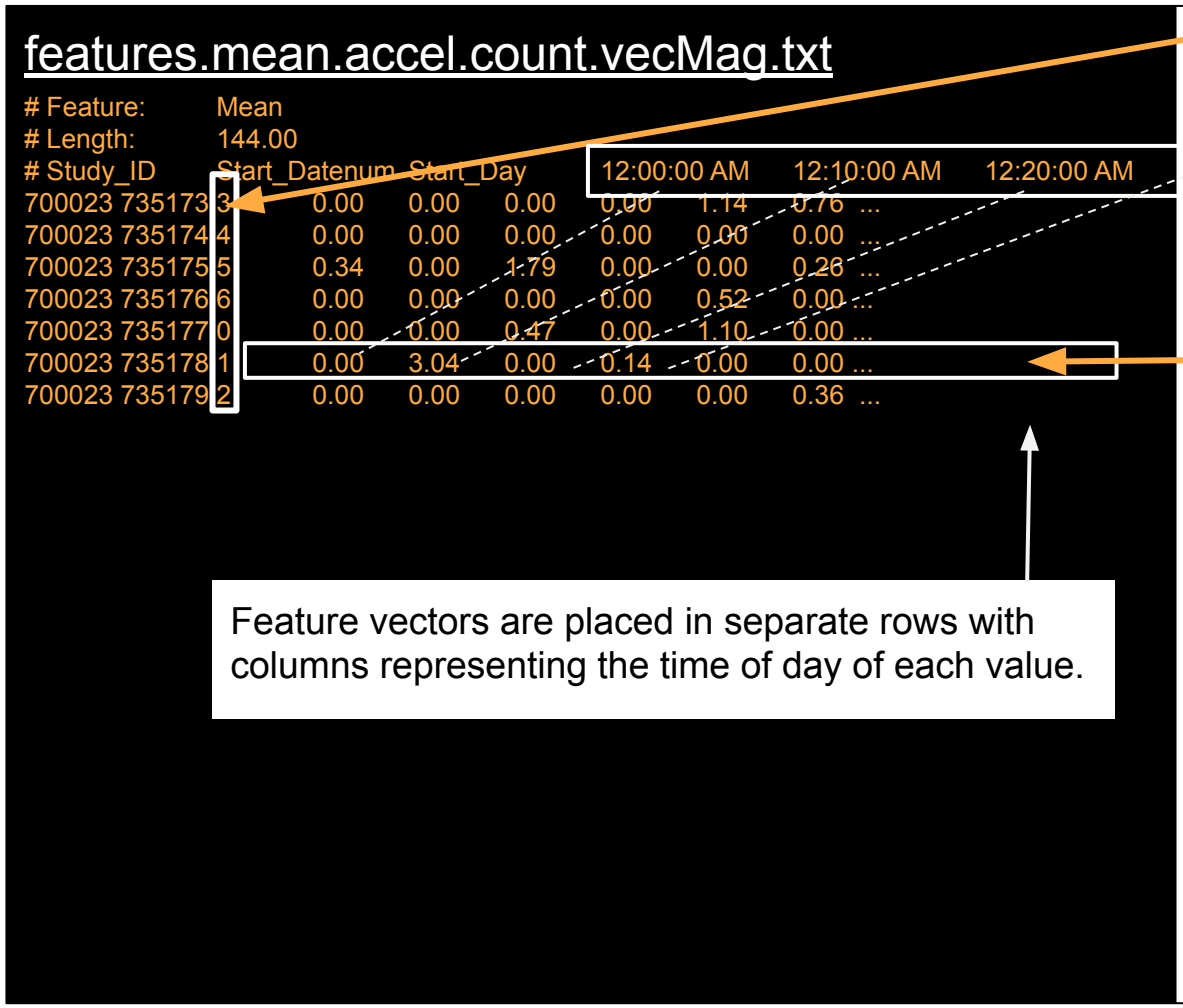
---------- Data Table File Created By ActiGraph GT3XPlus ActiLife v6.10.2 Firmware v2.2.1 date format M/d/yyyy Filter Normal ----------
Serial Number: CLE1B42120019

Subject 1 - Actigraph count data (.csv)

---------- Data Table File Created By ActiGraph GT3XPlus ActiLife v6.10.2 Firmware v2.2.1 date format M/d/yyyy Filter Normal ----------
Serial Number: CLE1B42120019
Start Time 00:00:00
Start Date 6/15/2013
Epoch Period (hh:mm:ss) 00:00:01
Download Time 16:11:52
Download Date 6/28/2013
Current Memory Address: 0
Current Battery Voltage: 3.87 Mode = 61

Date, Time, Axis1,Axis2,Axis3,Steps,Lux,Inclinometer Off,Inclinometer Standing,Inclinometer Sitting,Inclinometer Lying,Vector Magnitude
6/15/2013,00:00:00,0,0,0,0,0,1,0,0,0,0
6/15/2013,00:00:01,0,0,0,0,0,1,0,0,0,0
6/15/2013,00:00:02,0,0,0,0,0,1,0,0,0,0
6/15/2013,00:00:03,0,0,0,0,0,1,0,0,0,0
6/15/2013,00:00:04,0,0,0,0,0,1,0,0,0,0

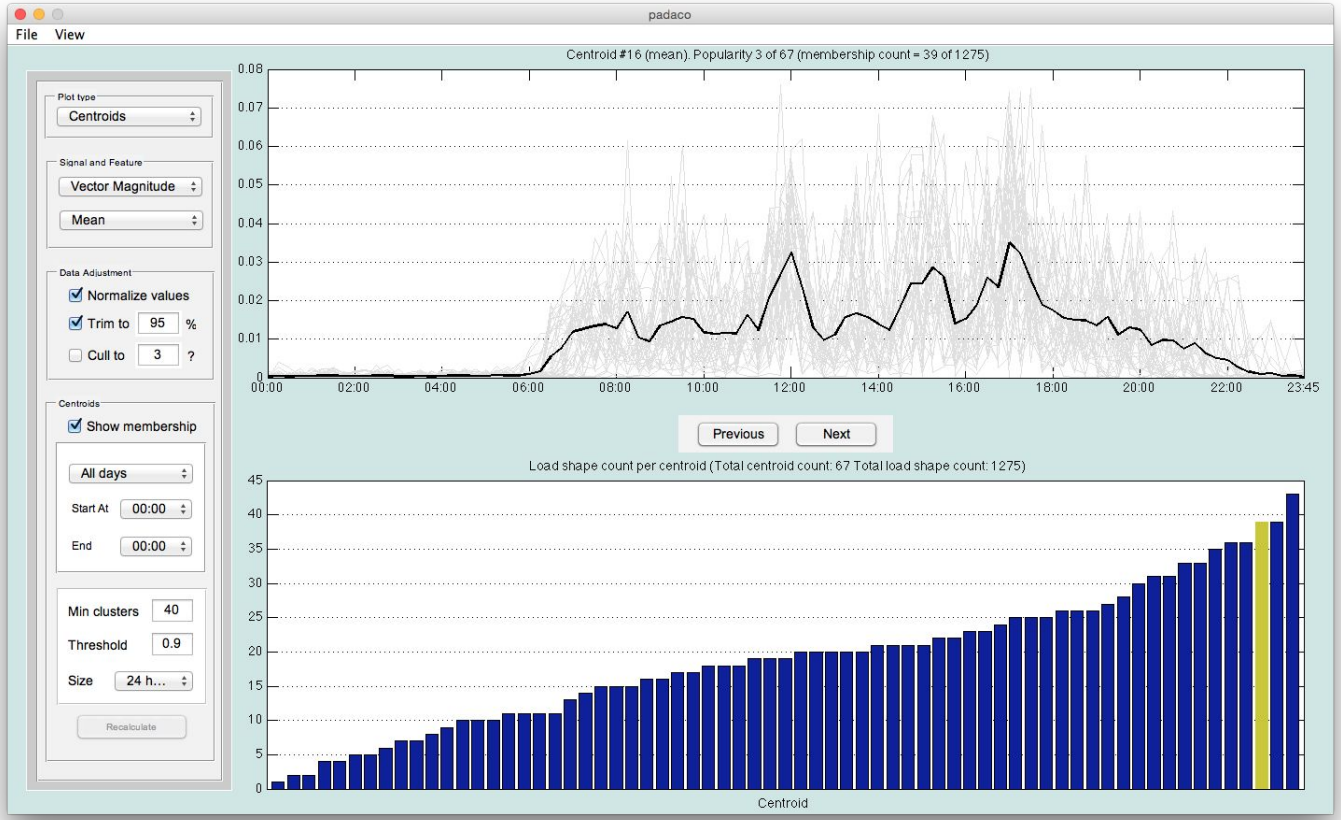
Count data features vectors



1. One week max is stored per subject.
2. Feature vectors are placed on rows, by day. Here, feature vectors are 144 elements in length (1x144), as feature values are taken once every 10 minutes of the day.

Exploring Resultings (demo)

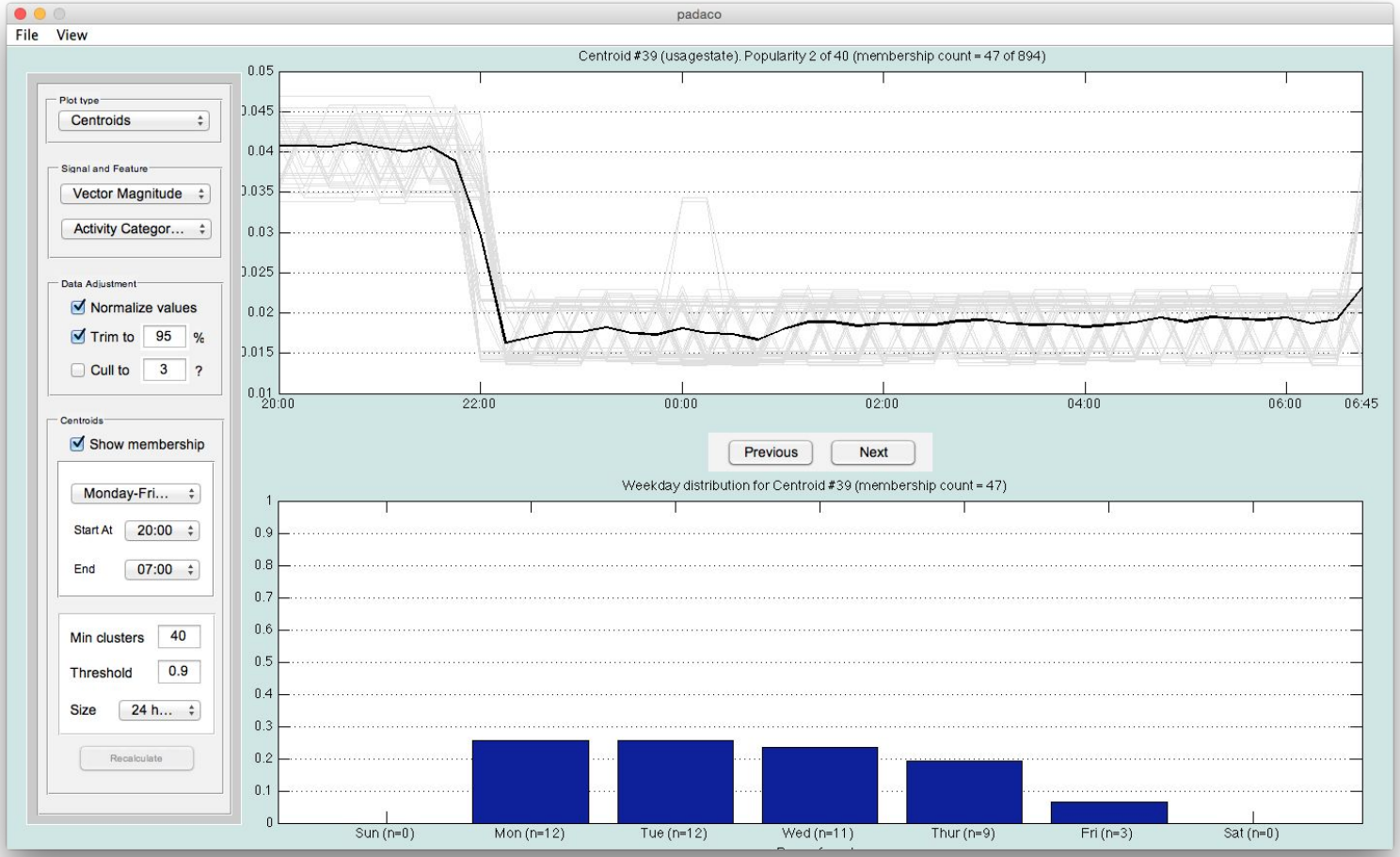
Questions



Left panel contains the interface for defining the signal of interest and schedule reduction for cluster input as well as clustering configuration parameters like the minimum number of clusters and convergence threshold.

Bottom window shows the number of shapes (y-axis) per cluster (x-axis) and highlights the selected clusters, to be shown in the upper window, with yellow bars. Here, the third most popular centroid is shown in black, in the upper window, with its member shapes in light gray.

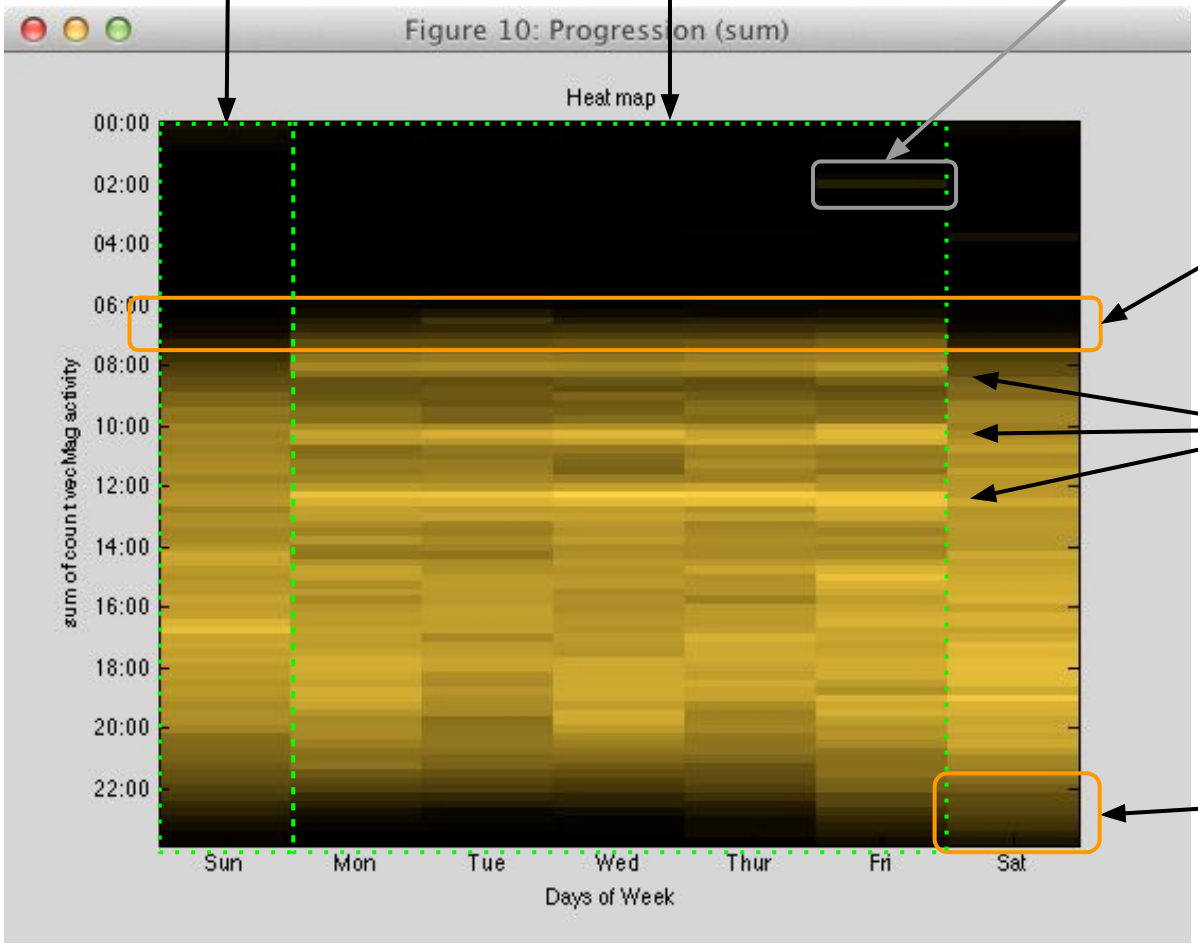
Day of week distribution for different patterns



Weekend activity appears smooth, gradually increasing and then decreasing across the day.

Weekday activity exhibits more grooves. likely due to school schedules share by subjects (e.g. recess, lunch times),

Outlier value, unknown, extremely high feature found in one subject. This value can be discarded by removing the top 95% of values before normalization.

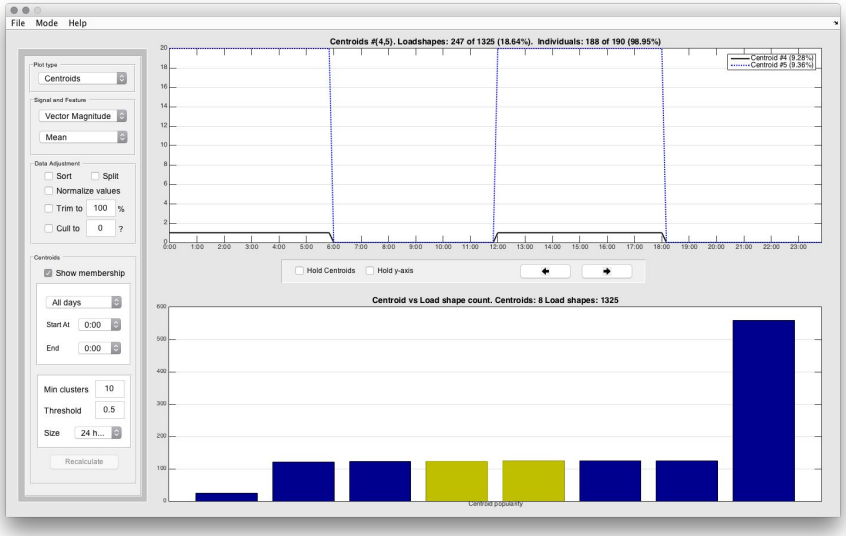


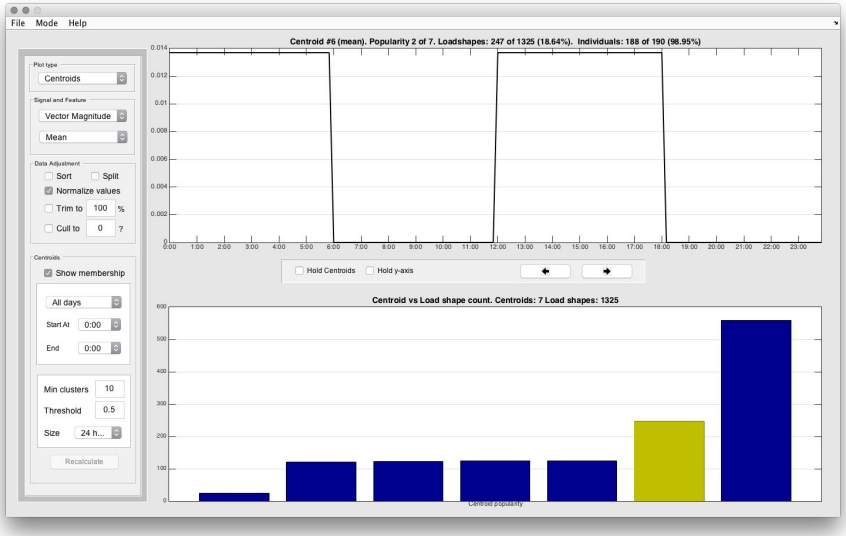
Subjects' activity begins earlier on school days than weekends.

Common activity levels on school days seen at 8:00AM, 10:15AM, and 12:30PM

Subjects show activity later in the evenings with latest activity occurring on Saturday night.

Normalization before clustering allows us to focus on the *shape*.



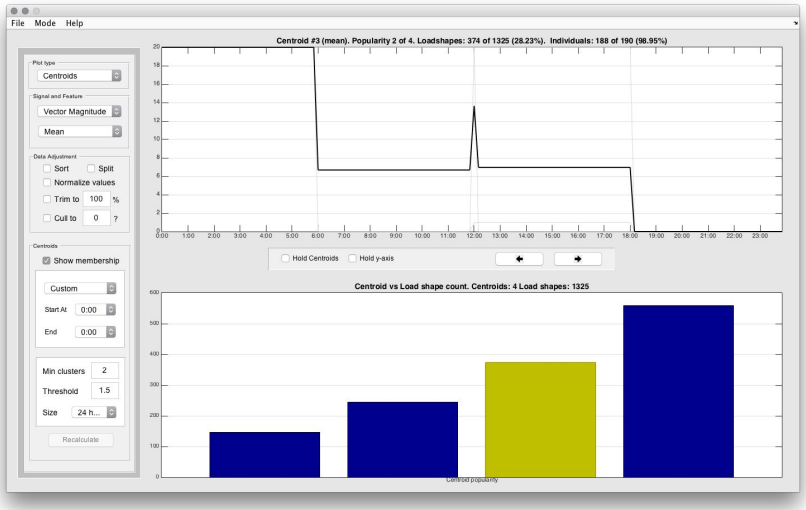


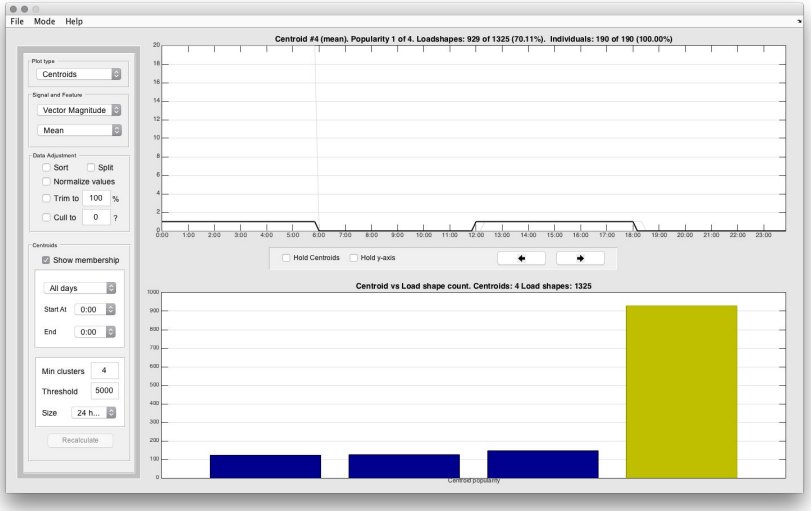
Without normalization. These two profiles (top window) are the same in when they Increase and decrease in value across the day, but differ in how much they increase. The first profile increases to 20, while the second increases to 1.

With normalization. The two profiles from the left figure (top window) are consolidated because the degree by which each profile changed in magnitude (20 vs 1) is no longer taken into account.

Note: Normalization does not make sense with categorical data which does not have a linear relationship or progression between categories.

K-medoids vs K-means



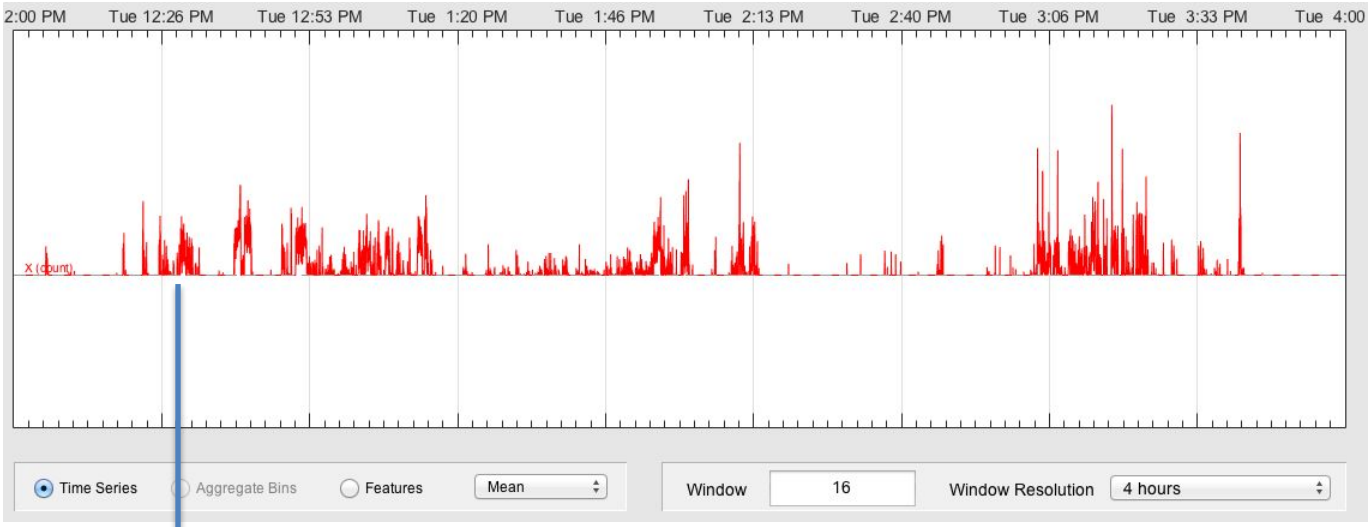


K-means finds centroids or profiles based the minimum mean distance between the profile and other shapes in the data. However, this can easily produce a profile that does not exist.

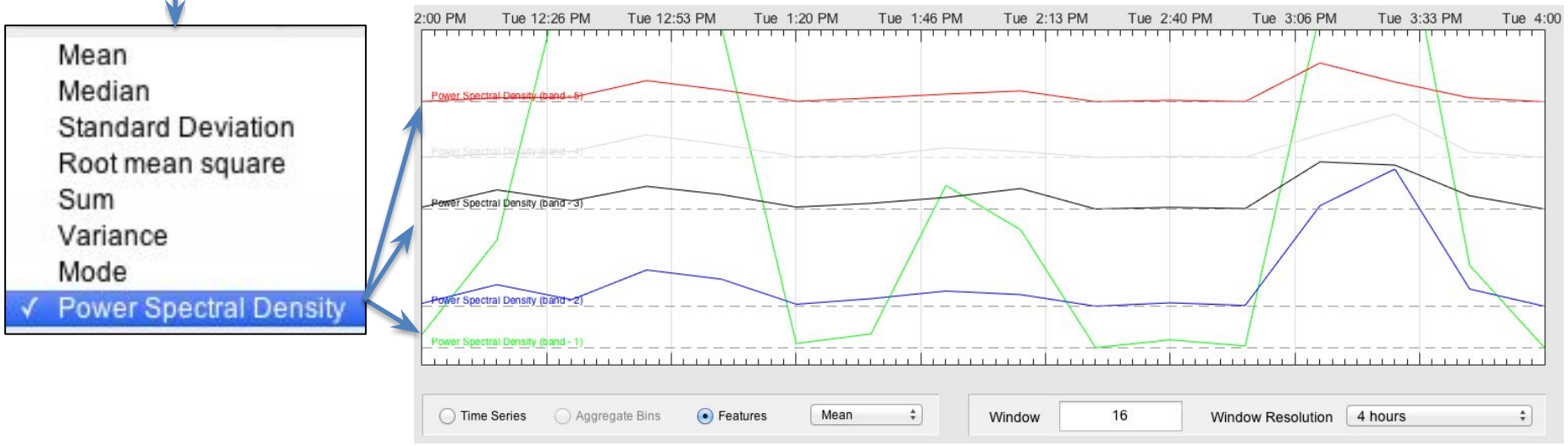
K-medoids finds profiles based on the minimum distance to a medoid, the most central shape in a group. The advantage here is that the profile will be obtained from values in the data.

Note: Simulated data shown with an absurdly large threshold (5,000) to show the medoid example on the right; otherwise, the data converges to the exact number of clusters.

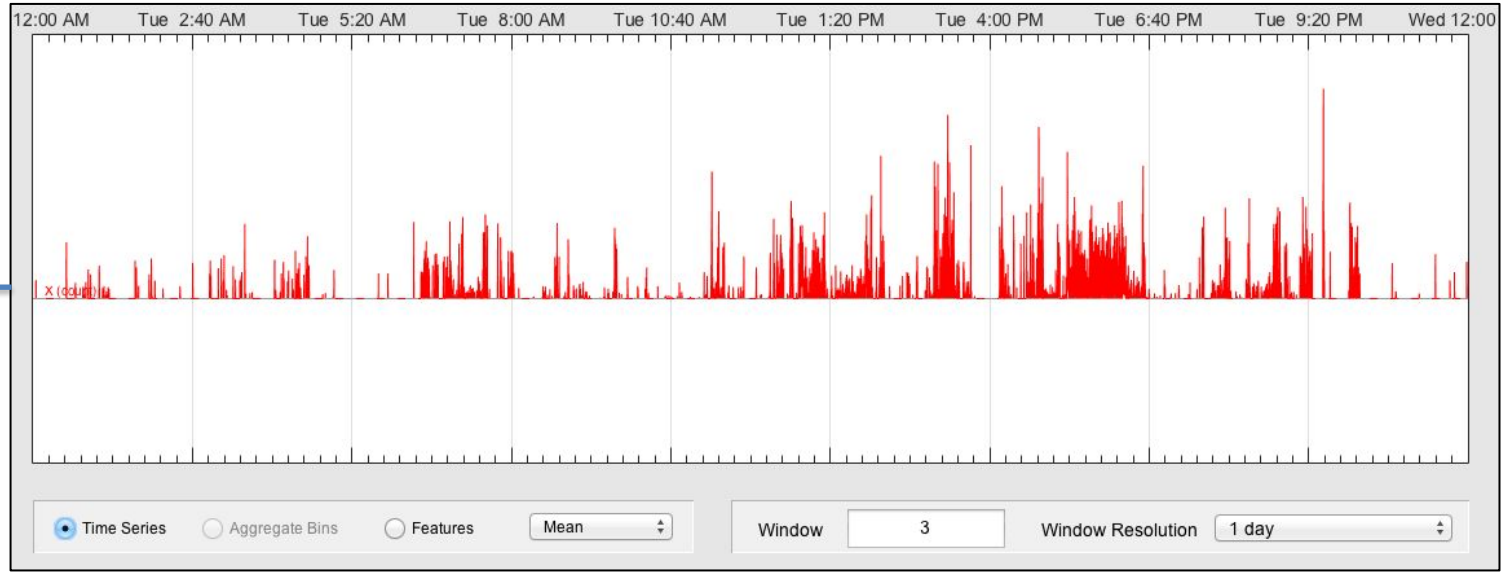
Example – PSD as feature function



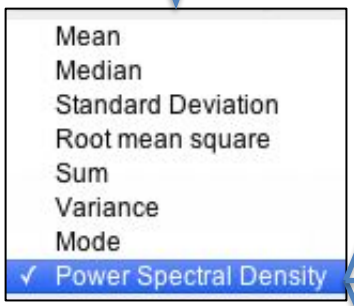
4 hours of x-axis accelerometer counts (14,400 counts)

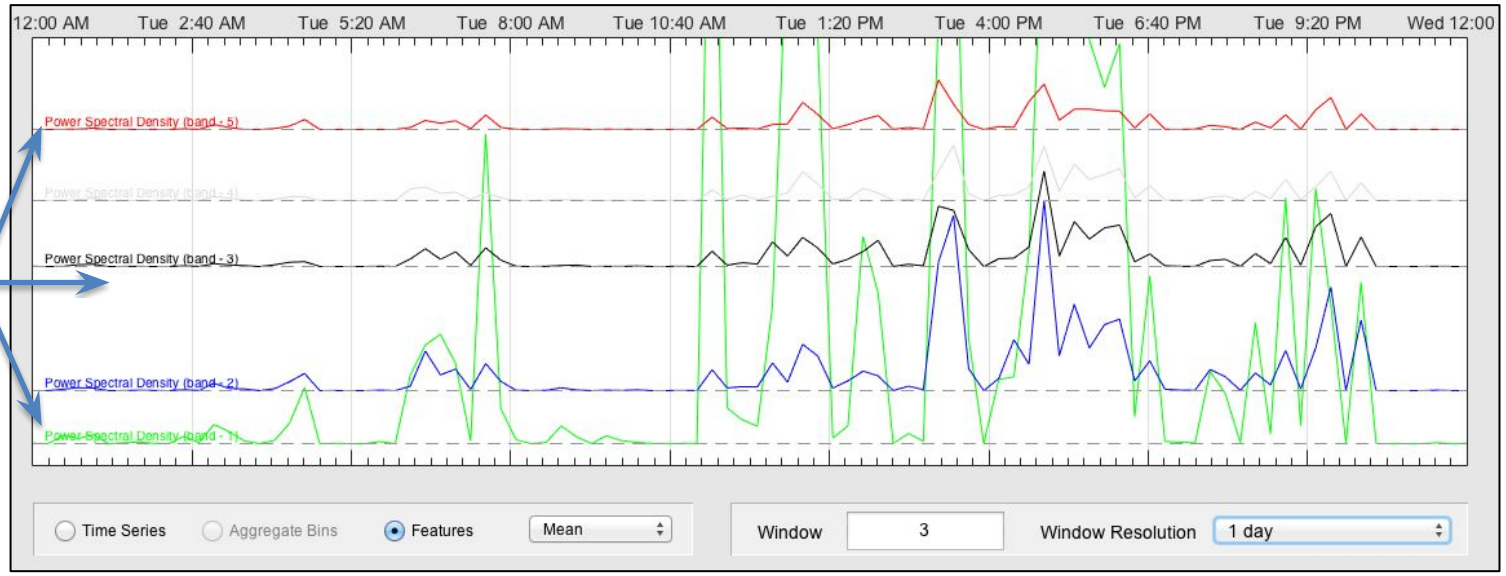


Corresponding power spectrum feature vectors (length 16)



24 hours of x-axis accelerometer counts (86,400 counts)





Corresponding power spectrum band profiles for Tuesday (length 96)