

# INTRO TO DATA SCIENCE

## TIME SERIES

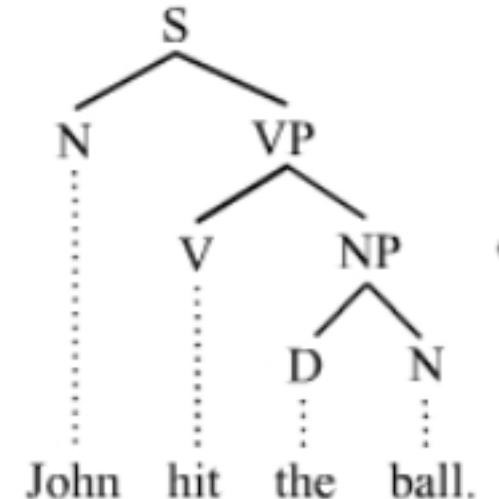
## LAST TIME:

I. WHAT IS NATURAL LANGUAGE PROCESSING

II. NLP APPLICATIONS

III. BASIC NLP PRACTICE

IV. NLP LABS



---

INTRO TO DATA SCIENCE

---

# QUESTIONS?

**WHAT WAS THE MOST INTERESTING THING YOU LEARNT?**

**WHAT WAS THE HARDEST TO GRASP?**

**INTRO**  
**MACHINE LEARNING WITH TIME SERIES**  
**TIME SERIES SIMILARITY**  
**TS GLOSSARY**  
**OTHER TOOLS AND TRICKS**  
**REAL WORLD EXAMPLES**

- LEARN TERMINOLOGY FOR TIME SERIES
- UNDERSTAND WHY ML ON TS IS DIFFERENT
- KNOW THE DIFFERENCE BETWEEN ONLINE AND OFFLINE ALGORITHMS
- BE ABLE TO APPLY EXPLORATORY VISUALIZATION TO TIME SERIES

---

**INTRO TO DATA SCIENCE**

---

**INTRO**

# WHAT IS A TIME SERIES

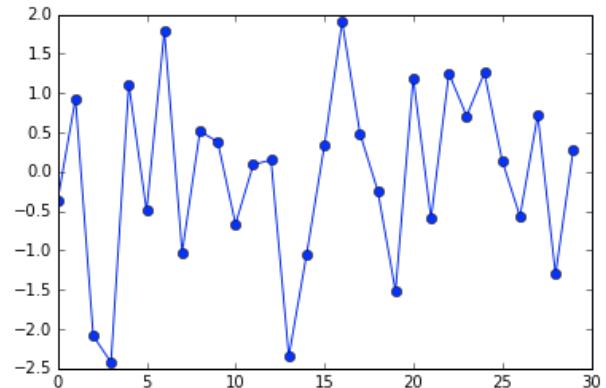
- you tell me....



<http://www.nationalimages.com/132460002506/bluecong101.jpg>

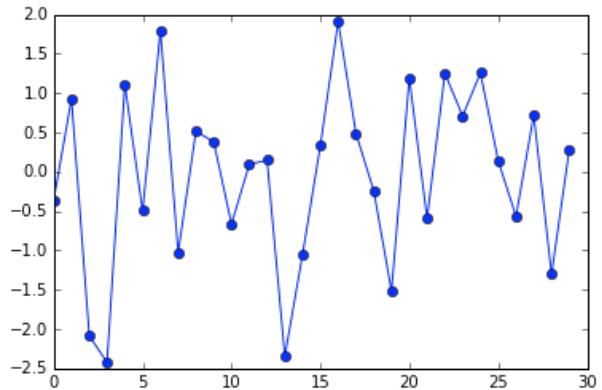
# WHAT IS A TIME SERIES

- Sequence of points with timestamps i.e. an ORDERED collection of numbers



# WHAT IS A TIME SERIES

- Sequence of points with timestamps i.e. an ORDERED collection of numbers
- Timestamps could be
  - regular (sampling at fixed frequency)
  - irregular (events with associated timestamp)



# TSS ARE EVERYWHERE

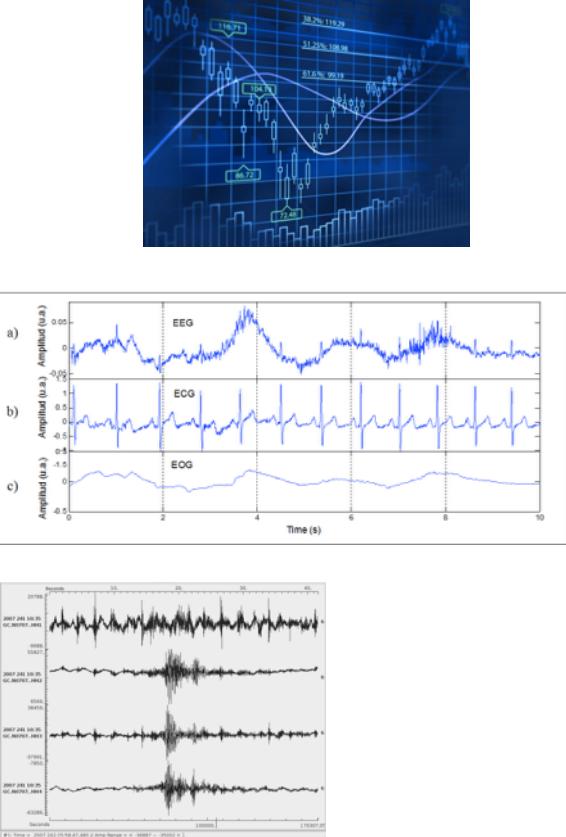
- you tell me where....



<http://www.rsi.it/di/images/1324480900066biwmconiglio1.jpg>

## TSS ARE EVERYWHERE

- Stock market
- Music
- Biosensors, biosignals (EEG, EKG, ECG, Wearables, etc...)
- Website monitoring and analytics
- IoT
- Energy Monitoring
- Traffic Signs
- Earthquakes
- Tides
- Sunspots
- ...



# A TIME SERIES DIFFERENT IS...



# TSS ARE DIFFERENT FROM FLAT DATA

- you tell me why....

	General government			Public non-financial corporations			Non-financial public sector		
	Receipts	Payments	Cash surplus	Receipts	Payments	Cash surplus	Receipts	Payments	Cash surplus
1987-88	32.6	32.4	0.2	4.0	5.9	-0.6	35.5	37.2	-0.4
1988-89	31.9	30.2	1.7	3.8	5.4	-0.1	34.8	34.6	1.6
1989-90	31.6	30.4	1.2	3.7	6.7	-1.6	34.1	36.0	-0.4
1990-91	31.6	32.1	-0.6	3.8	5.9	-0.7	34.2	36.8	-1.2
1991-92	30.3	34.1	-3.9	3.6	5.4	0.0	32.6	38.2	-3.8
1992-93	29.8	34.4	-4.7	3.6	5.0	0.3	32.0	38.1	-4.3
1993-94	30.6	31.5	-0.9	3.8	4.3	0.9	32.7	37.3	-0.6
1994-95	31.4	34.1	-2.7	3.4	4.5	0.8	33.3	37.2	-2.0
1995-96	32.6	34.0	-1.4	3.0	4.5	0.1	33.9	36.9	-1.4
1996-97	33.5	33.9	-0.4	3.1	4.2	0.3	34.4	35.9	-0.1
1997-98	33.1	32.6	0.5	3.0	3.7	0.6	34.4	34.7	1.1
1998-99	37.9	37.4	0.4	na	na	-0.6	na	na	-0.5
1999-00	38.8	36.3	2.5	na	na	-0.1	na	na	2.4
2000-01	37.8	36.5	1.3	na	na	0.0	na	na	1.3
2001-02	36.4	35.9	0.6	na	na	0.1	na	na	0.6
2002-03	37.8	36.3	1.6	na	na	-0.1	na	na	1.6
2003-04(e)	37.3	36.4	0.8	na	na	-0.3	na	na	0.5
2004-05(e)	36.5	35.9	0.6	na	na	na	na	na	na
2005-06(p)	35.9	35.4	0.5	na	na	na	na	na	na
2006-07(p)	35.6	34.8	0.7	na	na	na	na	na	na

	A	B	C	D	E	F	G	H
1	Ph #	Ph Name		Height	Weight	M	M Wt	B Wt
2	123	Smith		150	7803	60.3	3.4	
3	123	Smith		64930	8005	62.3	3.7	
4	123	Smith		64930	7902	58.7	2.9	
5	123	Smith	64930	64930	8101	57.9	3.1	
6	123	Smith	64930	64930	8205	55.2	1.4	
7	123	Smith	2384	2384	7511	61.8	2.5	
8	123	Smith	2384	2384	7801	64.1	2.7	
9	220	Jones		177	7906	59.2	2.2	
10	220	Jones		177	7512	57.4	3.6	
11	220	Jones		177	706	58.2	3.4	
12					59.51	2.89		

# TSS ARE DIFFERENT

- Ordered events
- Correlation in Time
- Periodicity
- Past .... Future ...



# MACHINE LEARNING WITH TIME SERIES LEARN YOU WILL....



---

INTRO TO DATA SCIENCE

---

# MACHINE LEARNING WITH TIME SERIES

# **MACHINE LEARNING WITH TSS**

- example problems....

---

# MACHINE LEARNING WITH TSS

- Prediction of future values (regression)
- Pattern recognition & segmentation (classification, clustering, anomaly detection, dim reduction)
- Compression, noise reduction (preprocessing)

# PREDICTION

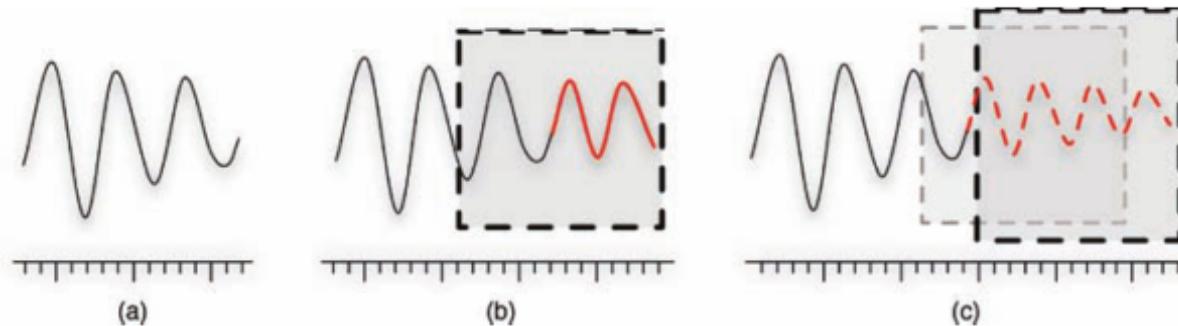


Fig. 5. A typical example of the time-series prediction task. (a) The input time series may exhibit a periodical and thus predictable structure. (b) The goal is to forecast a maximum number of upcoming datapoints within a prediction window. (c) The task becomes really hard when it comes to having *recursive prediction*, that is, the long-term prediction of a time series implies reusing the earlier forecast values as inputs in order to go on predicting.

# ANOMALY DETECTION

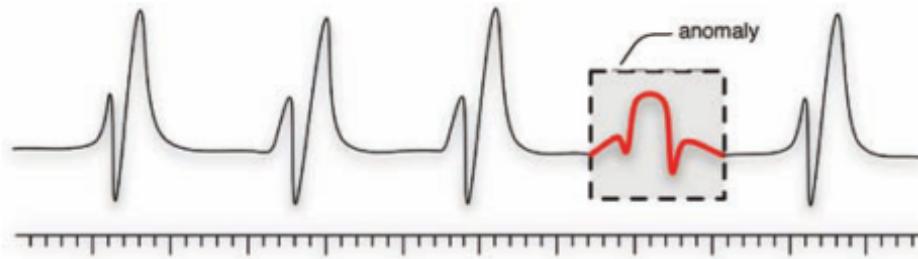


Fig. 6. An idealized example of the anomaly detection task. A long time series which exhibits some kind of periodical structure can be modeled thanks to a reduced pattern of “standard” behavior. The goal is thus to find subsequences that do not follow the model and may therefore be considered as anomalies.

# SEGMENTATION

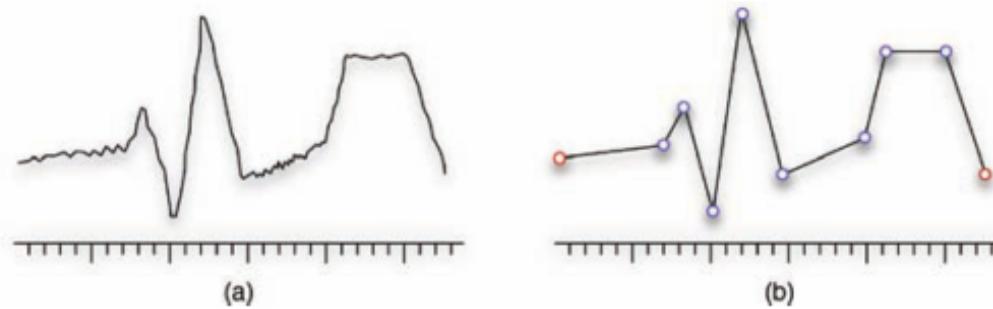


Fig. 4. Example of application of a segmentation system. From (a) usually noisy time series containing a very large number of datapoints, the goal is to find (b) the closest approximation of the input time series with the maximal dimensionality reduction factor without losing any of its essential features.

# MOTIF DISCOVERY

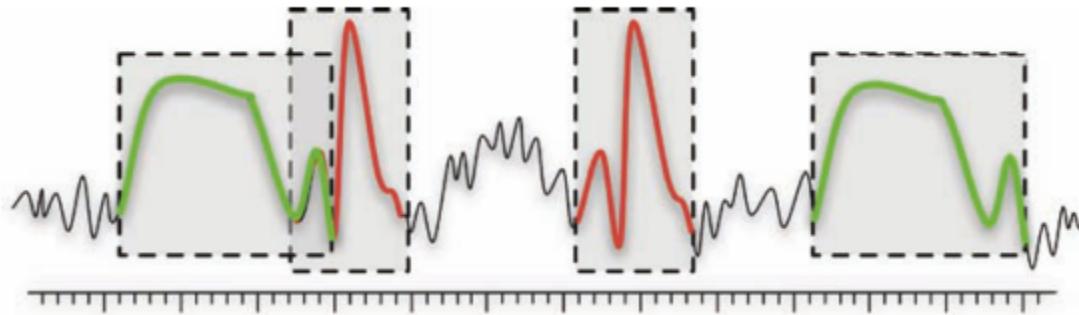


Fig. 7. The task of motif discovery consists in finding every subsequence that appears recurrently in a longer time series. These subsequences are named motifs. This task exhibits a high combinatorial complexity as several motifs can exist within a single series, motifs can be of various lengths, and even overlap.

---

# ML PIPELINE

- Data Munging, preprocessing
- Feature Extraction, feature engineering
- Model Building and Validation
- Results visualization

# PREPROCESSING

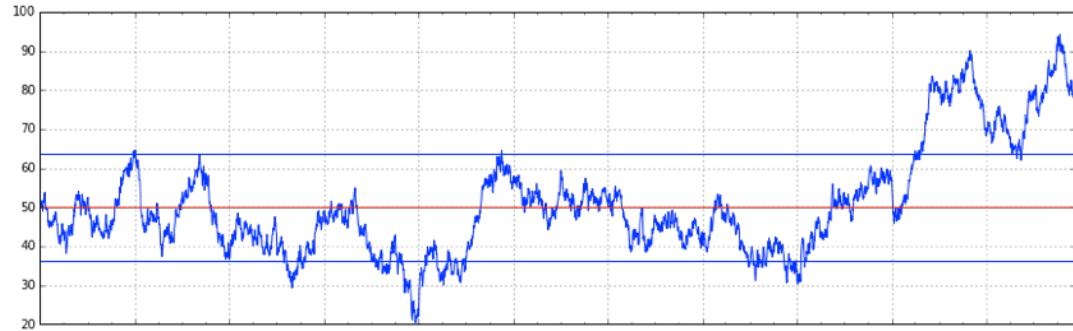
- Normalization
- Filtering



Let's normalize 'em TSS

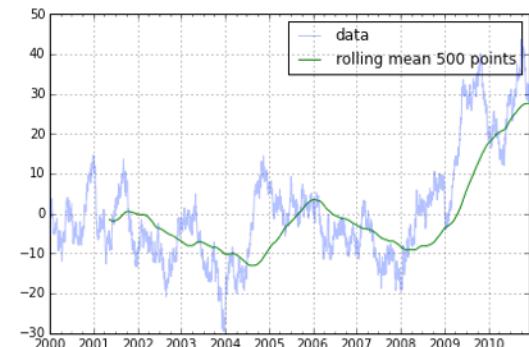
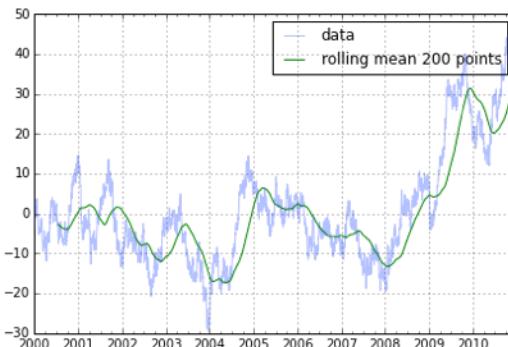
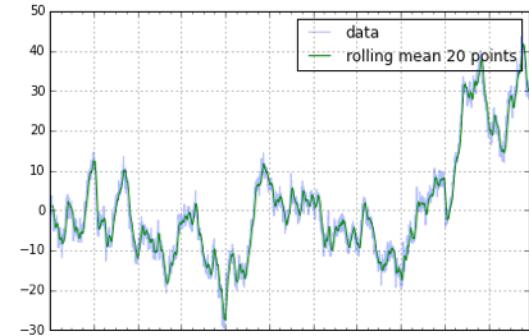
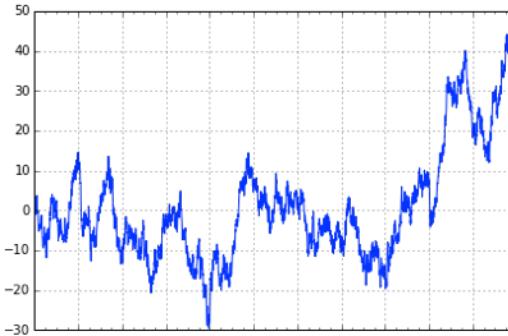
## PREPROCESSING

- Normalization



# PREPROCESSING

- noise reduction



# FEATURE EXTRACTION

- Raw data
- Differences
- Rolling windows

# FEATURE EXTRACTION

- Raw data
- Differences
- Rolling windows



# FEATURE EXTRACTION

- Raw data
- Differences
- Rolling windows

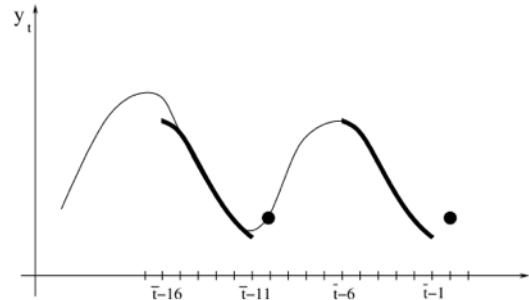


# FEATURE EXTRACTION

- Raw data

$$X = \begin{bmatrix} y_{N-1} & y_{N-2} & \dots & y_{N-n-1} \\ y_{N-2} & y_{N-3} & \dots & y_{N-n-2} \\ \vdots & \vdots & \vdots & \vdots \\ y_n & y_{n-1} & \dots & y_1 \end{bmatrix}$$

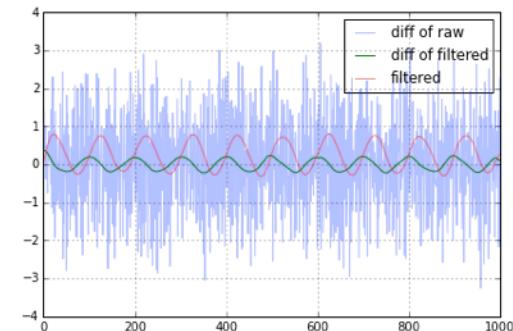
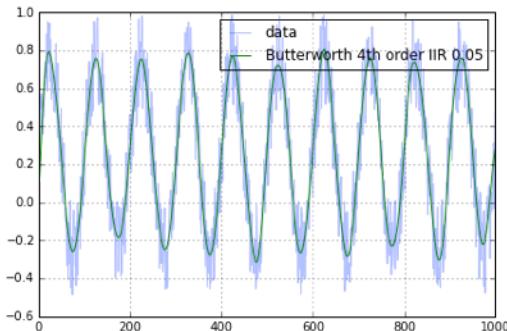
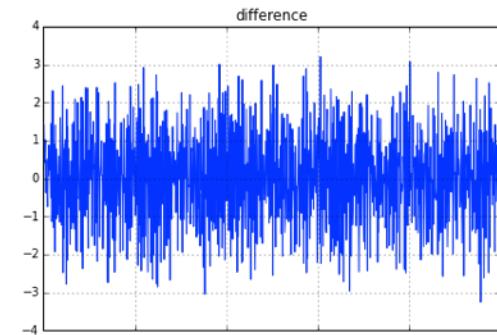
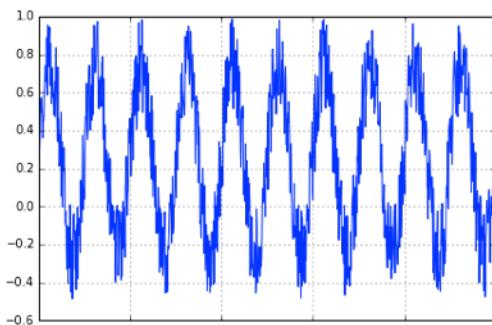
$$Y = \begin{bmatrix} y_N \\ y_{N-1} \\ \vdots \\ y_{n+1} \end{bmatrix}$$



**Fig. 2.** Nearest-neighbor one-step-ahead forecasts. We want to predict at time  $\bar{t} - 1$  the next value of the series  $y$  of order  $n = 6$ . The pattern  $y_{\bar{t}-16}, y_{\bar{t}-15}, \dots, y_{\bar{t}-11}$  is the most similar to the pattern  $\{y_{\bar{t}-6}, y_{\bar{t}-5}, \dots, y_{\bar{t}-1}\}$ . Then, the prediction  $\hat{y}_{\bar{t}} = y_{\bar{t}-10}$  is returned.

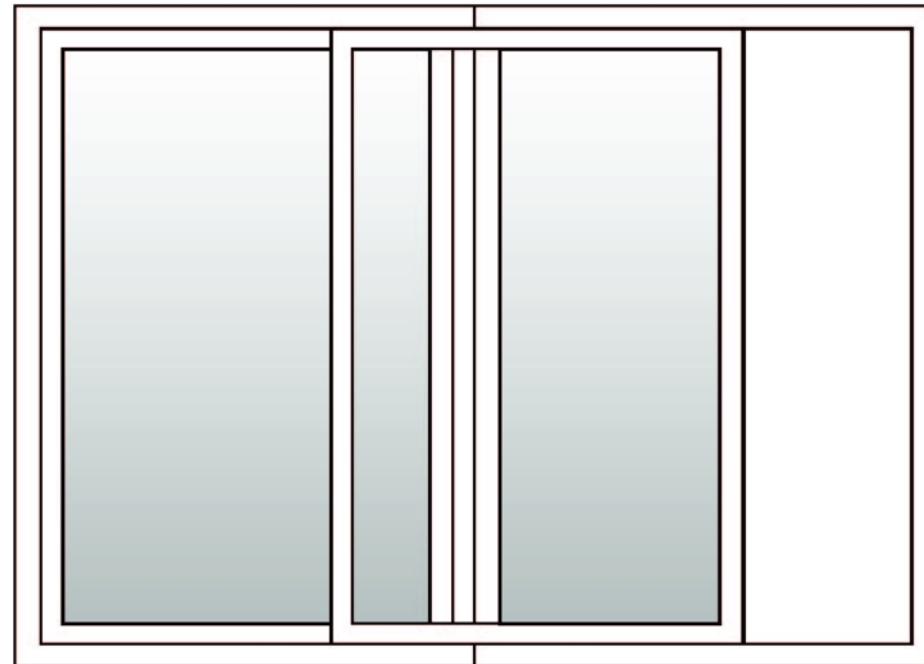
# FEATURE EXTRACTION

- Raw data
- Differences
- $\text{diff}(t) = \{t[i] - t[i-1]\} \text{ for } i = 1 \dots N$



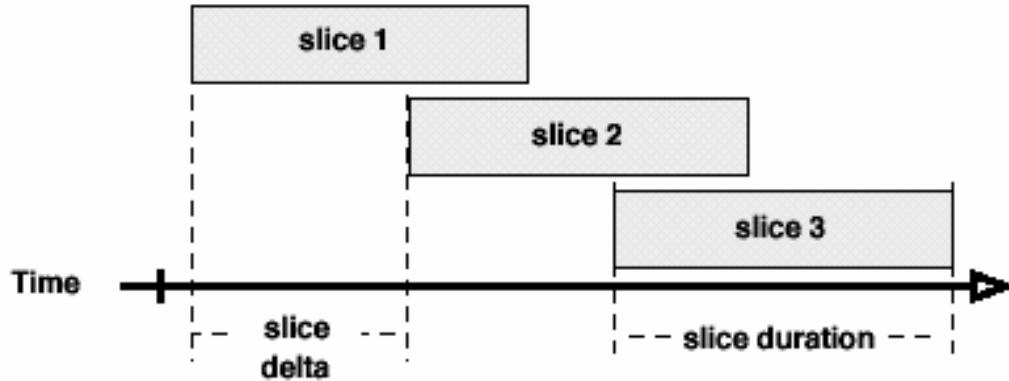
# FEATURE EXTRACTION

- Raw data
- Differences
- Rolling windows



# FEATURE EXTRACTION

- Rolling windows



- For each window calculate features:
  - Stats: Mean, Stdev, Moments
  - Frequency: Spectral Band Power, Entropy
  - Autoregressive model parameters
  - FFT, Hjorth, Mann-Kendall ...
  - Correlation dimension, integral, density, entropy
  - etc....

# FEATURE EXTRACTION

- more complex windows

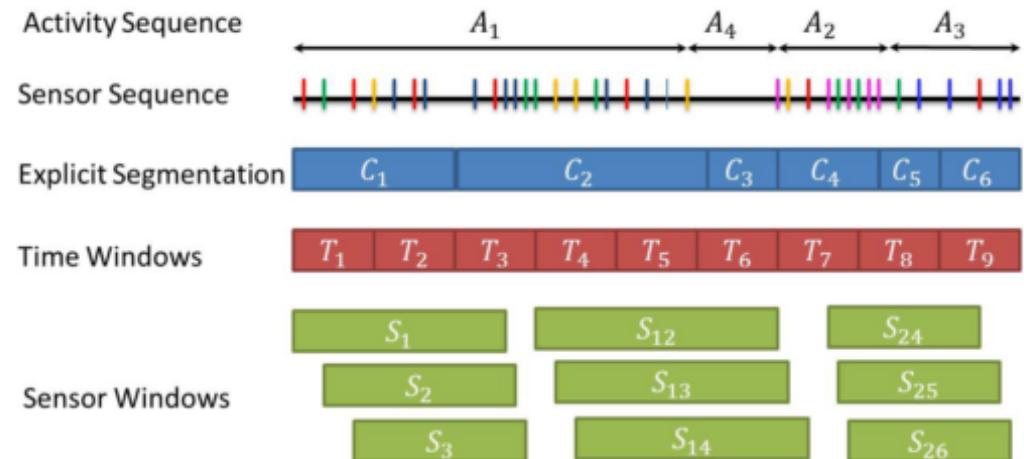


Figure 1: Illustration of the different approaches for stream processing. The different motion/door sensor firings are depicted by the colored vertical lines. The sensor windows are obtained using a sliding window of length 10 sensor events.

# COMPLETE PIPELINE

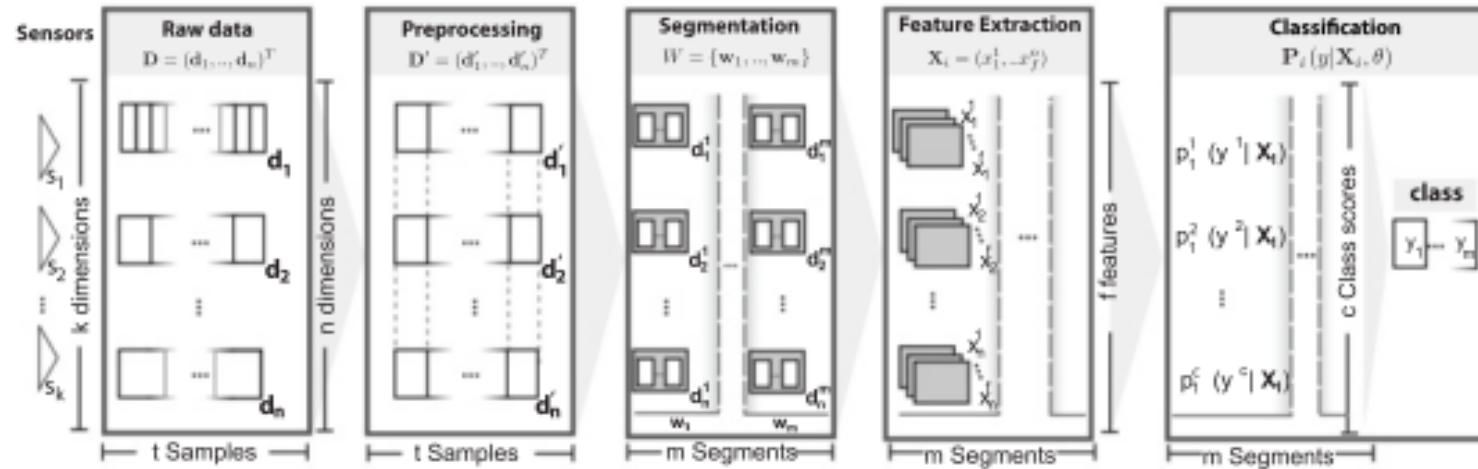


Fig. 1. Typical Activity Recognition Chain (ARC) to recognize activities from wearable sensors. An ARC comprises stages for data acquisition, signal preprocessing and segmentation, feature extraction and selection, training, and classification. Raw signals ( $\mathbf{D}$ ) are first processed ( $\mathbf{D}'$ ) and split into  $m$  segments ( $\mathbf{W}_i$ ) from which feature vectors ( $\mathbf{X}_i$ ) are extracted. Given features ( $\mathbf{X}_i$ ), a model with parameters  $\theta$  scores  $c$  activity classes  $\mathbf{Y}_i = \{y^1, \dots, y^c\}$  with a confidence vector  $\mathbf{p}_i$ .

---

INTRO TO DATA SCIENCE

---

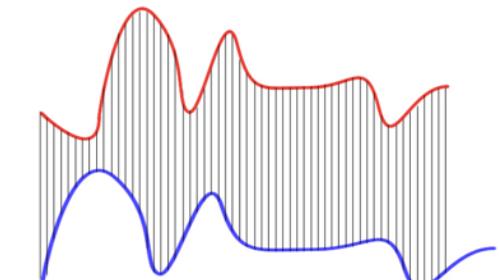
# TIME SERIES SIMILARITY

# SIMILARITY BETWEEN TIME SERIES

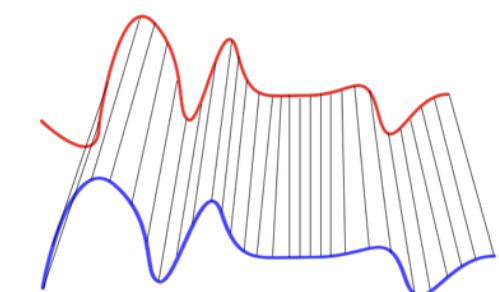
- How did we define distance?



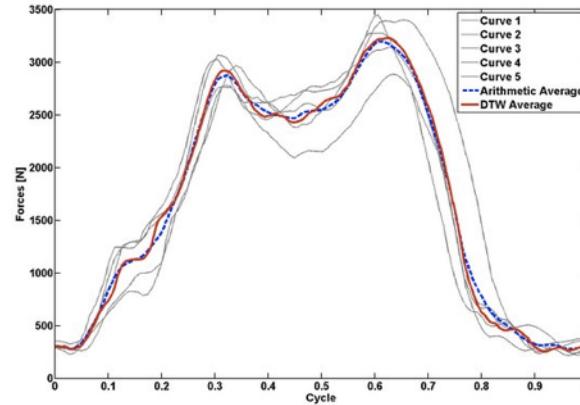
# SIMILARITY BETWEEN TIME SERIES



Euclidean Matching



Dynamic Time Warping Matching



[http://www.orthodex.com/wp-content/uploads/2016\\_6.jpg](http://www.orthodex.com/wp-content/uploads/2016_6.jpg)

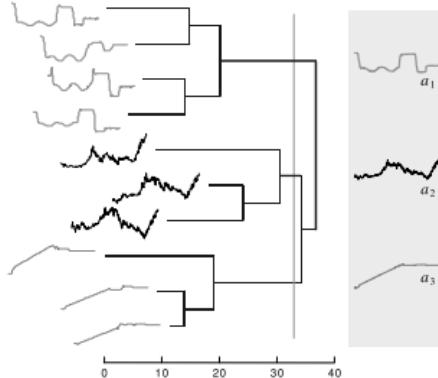
## DISTANCE MEASURES

Table I. Comparison of the Distance Measures surveyed in This Article with the Four Properties of Robustness

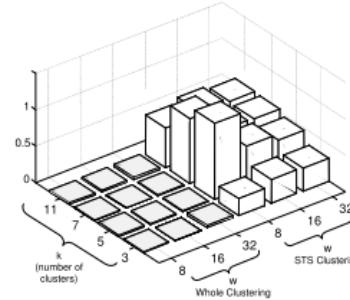
Distance measure	Scale	Warp	Noise	Outliers	Metric	Cost	Param	
<i>Shape-based</i>								
$L_p$ norms				✓	$O(n)$	0		
Dynamic Time Warping (DTW)		✓			$O(n^2)$	1		
LB.Keogh (DTW)		✓	✓		✓	$O(n)$	1	
Spatial Assembling (SpADE)	✓	✓	✓		$O(n^2)$	4		
Optimal Bijection (OSB)	✓	✓		✓	$O(n^2)$	2		
DISSIM	✓	✓			✓	$O(n^2)$	0	
<i>Edit-based</i>								
Levenshtein				✓	✓	$O(n^2)$	0	
Weighted Levenshtein				✓	✓	$O(n^2)$	3	
Edit with Real Penalty (ERP)	✓		✓	✓		$O(n^2)$	2	
Time Warp Edit Distance (TWED)	✓		✓	✓		$O(n^2)$	2	
Longest Common SubSeq (LCSS)	✓	✓	✓			$O(n)$	2	
Sequence Weighted Align (Swale)	✓	✓	✓			$O(n)$	3	
Edit Distance on Real (EDR)	✓	✓	✓		✓	$O(n^2)$	2	
Extended Edit Distance (EED)	✓	✓	✓		✓	$O(n^2)$	1	
Constraint Continuous Edit (CCED)	✓	✓	✓			$O(n)$	1	
<i>Feature-based</i>								
Likelihood			✓	✓		$O(n)$	0	
Autocorrelation			✓	✓	✓	$O(n \log n)$	0	
Vector quantization	✓	✓	✓		✓	$O(n^2)$	2	
Threshold Queries (TQuest)	✓	✓	✓			$O(n^2 \log n)$	1	
Random Vectors	✓	✓	✓			$O(n)$	1	
Histogram			✓	✓	✓	$O(n)$	0	
WARP	✓	✓	✓		✓	$O(n^2)$	0	
<i>Structure-based</i>								
<i>Model-based</i>								
Markov Chain (MC)				✓	✓	$O(n)$	0	
Hidden Markov Models (HMM)	✓	✓	✓	✓		$O(n^2)$	1	
Auto-Regressive (ARMA)				✓	✓	$O(n^2)$	2	
Kullback-Leibler				✓	✓	✓	$O(n)$	0
<i>Compression-based</i>								
Compression Dissimilarity (CDM)	✓	✓	✓			$O(n)$	0	
Parsing-based	✓	✓	✓			$O(n)$	0	

Each distance measure is thus distinguished as *scale* (amplitude), *warp* (time), *noise* or *outliers* robust. The next column shows whether the proposed distance is a metric. The cost is given as a simplified factor of computational complexity. The last column gives the minimum number of parameters setting required by the distance measure.

# SUBSEQUENCE CLUSTERING



**Figure 3.** A hierarchical clustering of ten time series. The clustering can be converted to a  $k$  partitional clustering by “sliding” a cutting line until it intersects  $k$  lines of the dendograms, then averaging the time series in the  $k$  subtrees to form  $k$  cluster centers (gray panel).



**Figure 6.** A comparison of the clustering meaningfulness for whole clustering, and STS clustering, using k-means with a variety of parameters. The two datasets used were buoy\_sensor(1) and ocean.

---

**INTRO TO DATA SCIENCE**

---

# **TS GLOSSARY**

# GLOSSARY

- Online VS Offline algorithms
- Frequency-domain VS time-domain
- Parametric VS non-parametric

## ONLINE

- an online algorithm is one that can process its input piece-by-piece in a serial fashion, i.e., in the order that the input is fed to the algorithm, without having the entire input available from the start.



# OFFLINE

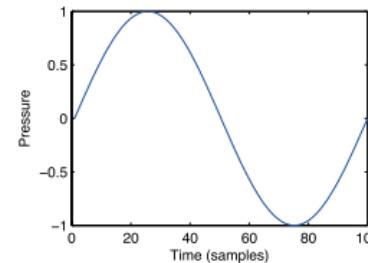
- an offline algorithm is given the whole problem data from the beginning and is required to output an answer which solves the problem at hand.



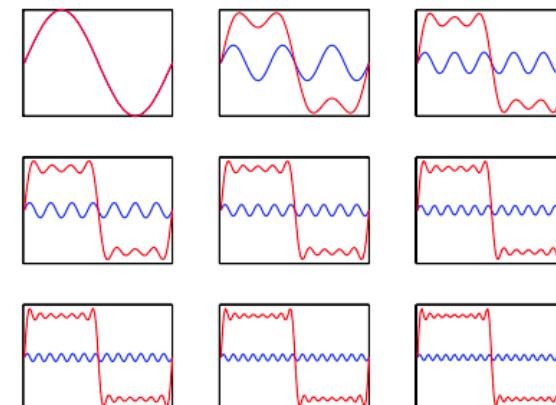
# FREQUENCY DOMAIN

- Sinusoids
  - simple waveform
  - single frequency
- 3 parameters:
  - $s(t) = a * \sin(f * t + p)$ 
    - $a$  = amplitude
    - $f$  = frequency
    - $p$  = phase
- They are excellent building blocks

$$s_N(x) = \frac{A_0}{2} + \sum_{n=1}^N A_n \cdot \sin\left(\frac{2\pi n x}{P} + \phi_n\right), \quad \text{for integer } N \geq 1.$$

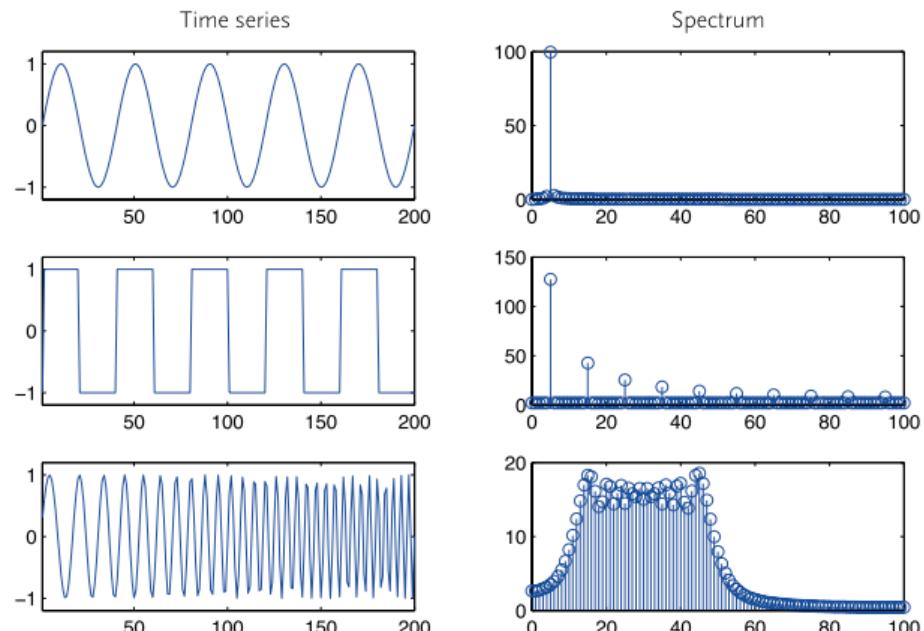


Making a square wave with sines



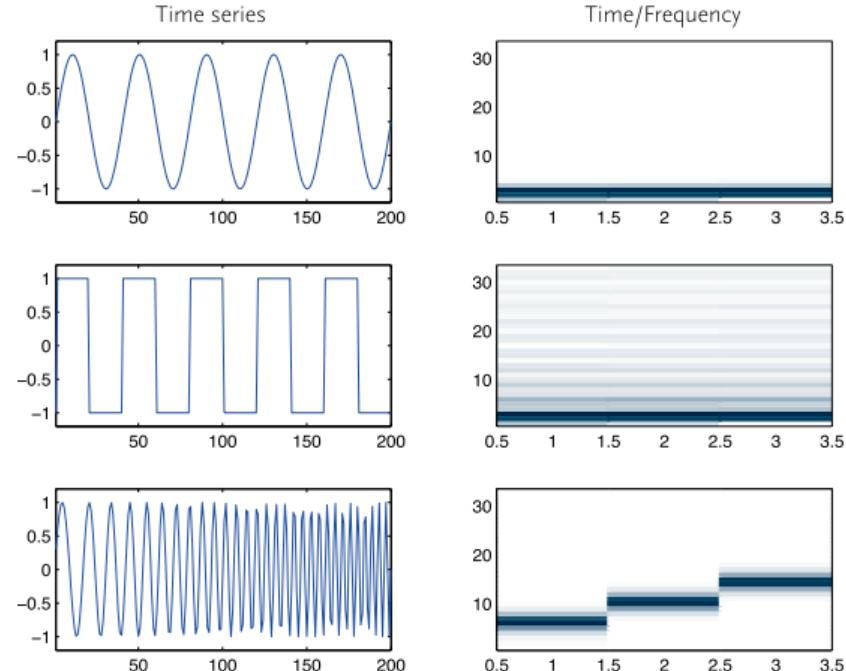
# FREQUENCY SPECTRUM

- Time series can be decomposed in terms of “sinusoid presence”
- That’s the frequency spectrum
- No temporal information in this representation, only frequency
- So a signal with changing frequency becomes a smeared spike

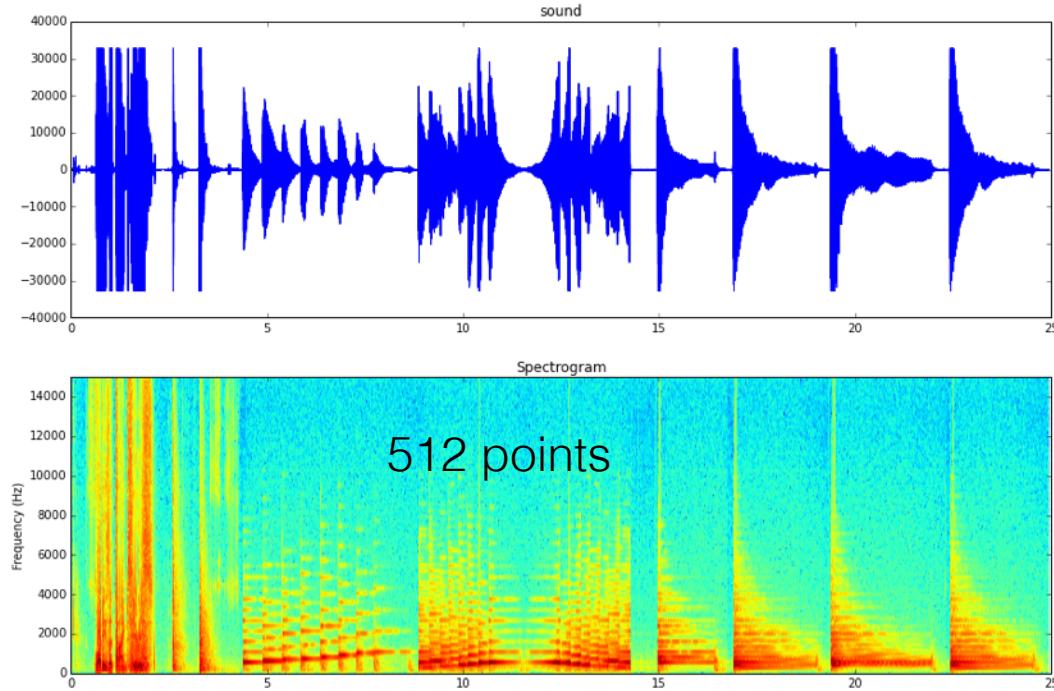


# TIME/FREQUENCY REPRESENTATION

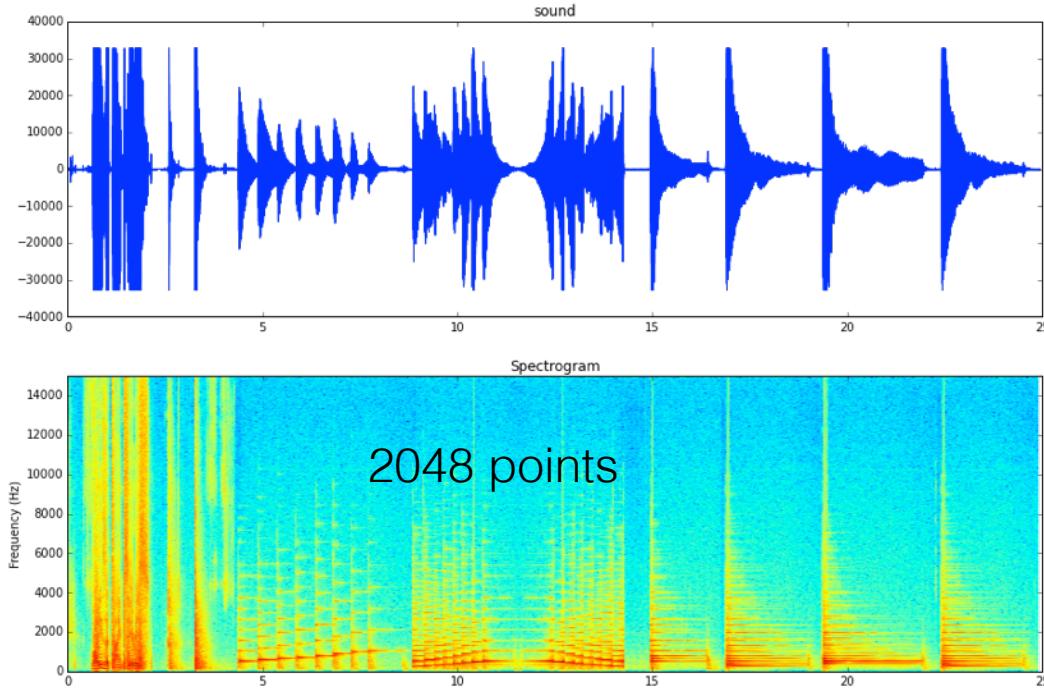
- Time-ordered series of frequency compositions
- Calculate Frequency Spectrum in each window
- Works really well when information is encoded in the frequency, like for example ... in sounds!



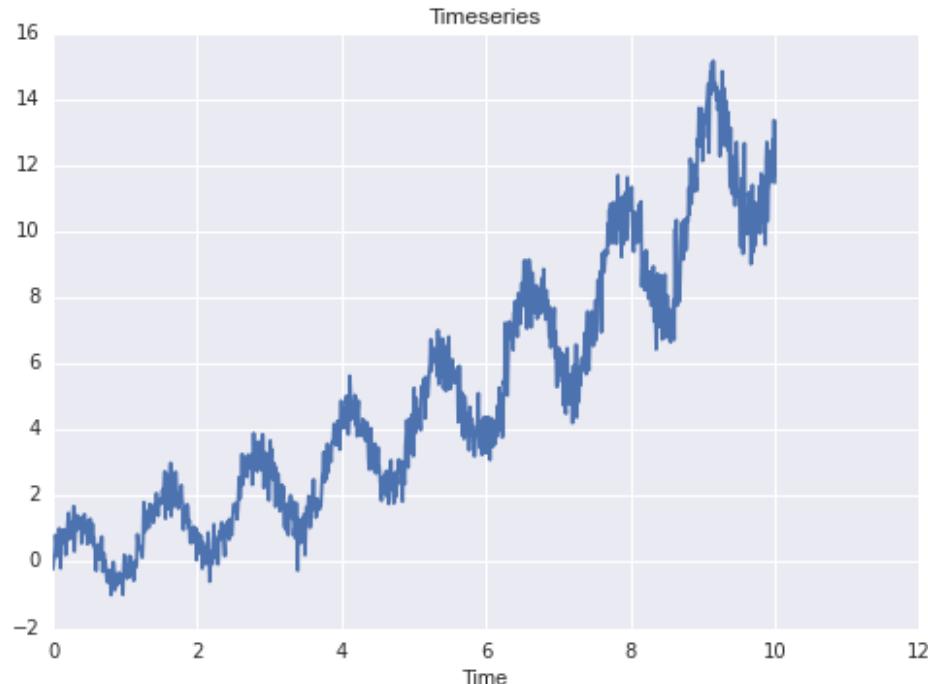
## SPECTROGRAM



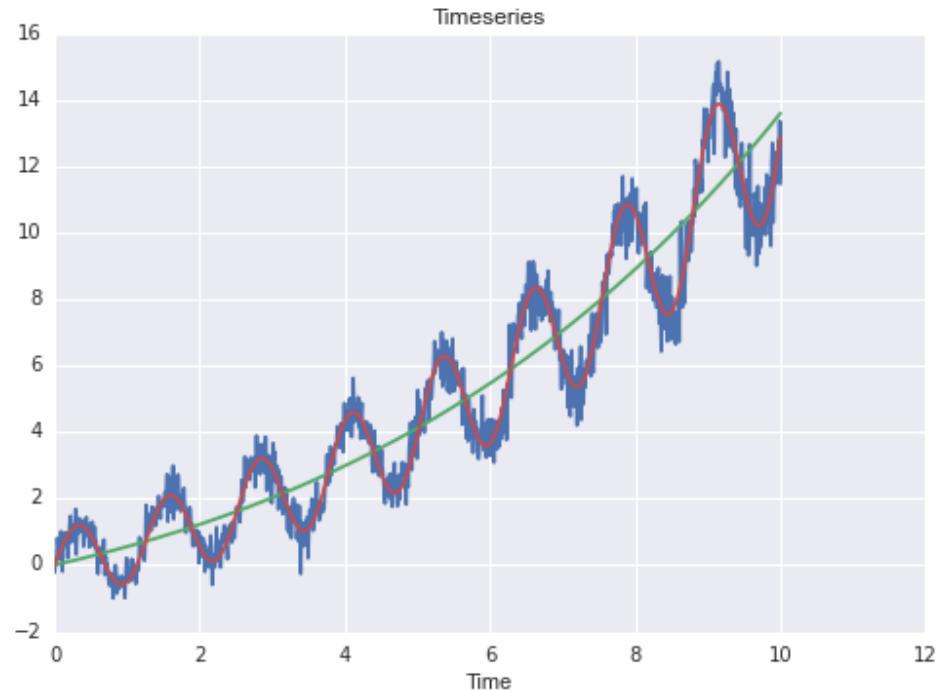
## SPECTROGRAM



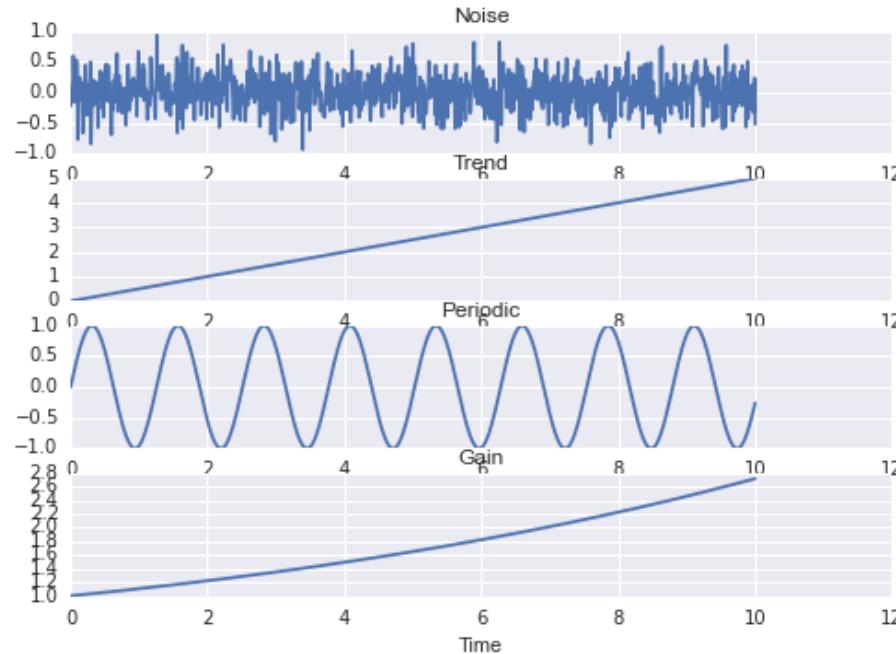
## PARAMETRIC OR NOT?



## PARAMETRIC OR NOT?



## PARAMETRIC OR NOT?



---

**INTRO TO DATA SCIENCE**

---

# **OTHER TOOLS AND TRICKS**

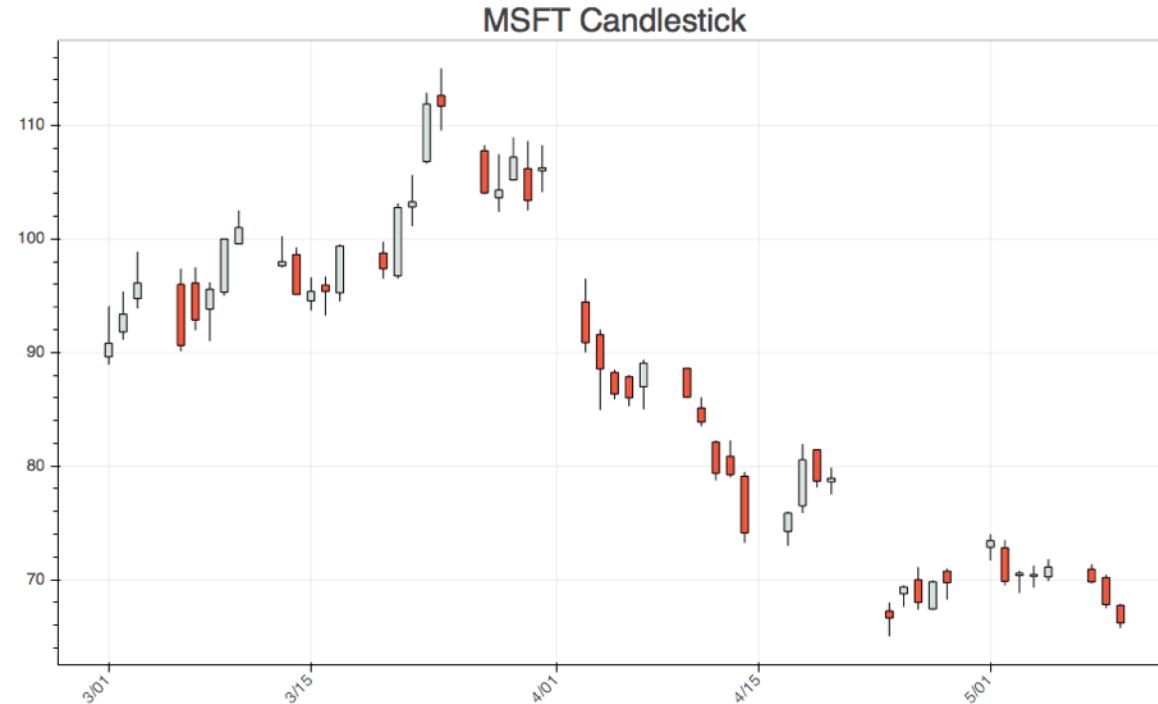
# Other Tools & Tricks



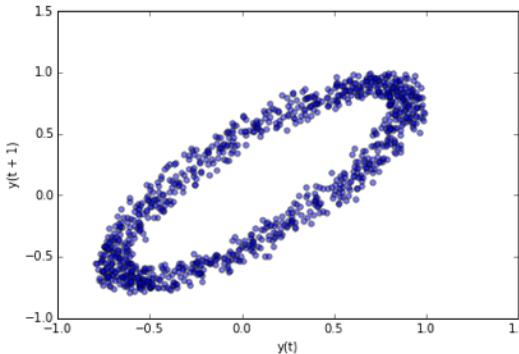
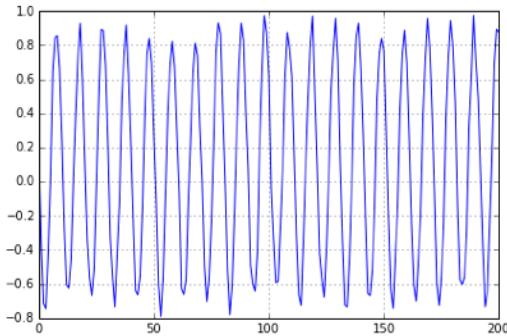
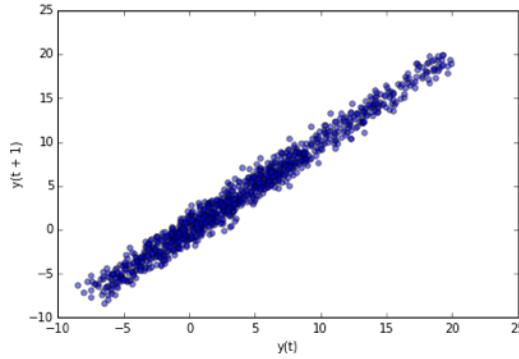
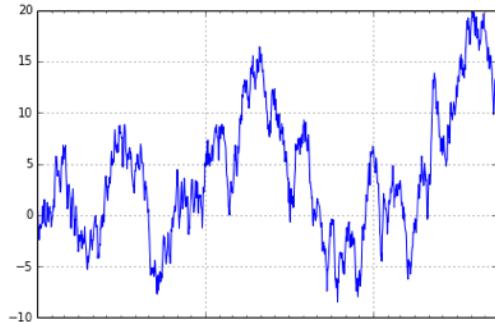
# **OTHER TOOLS AND TRICKS**

- Candlestick plot
- Lag plot
- Autocorrelation plot
- Time Maps

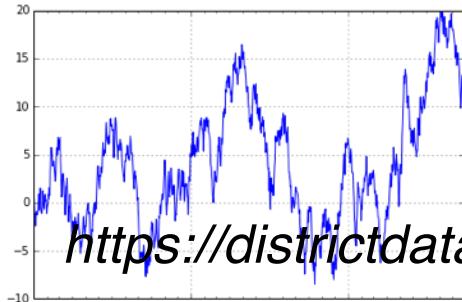
## CANDLESTICK PLOT



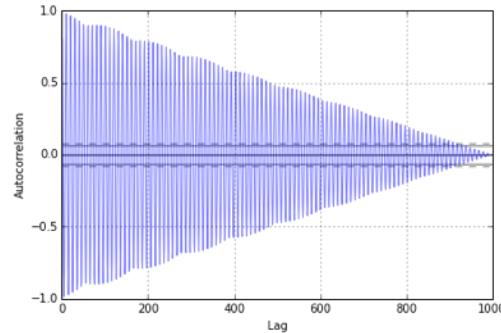
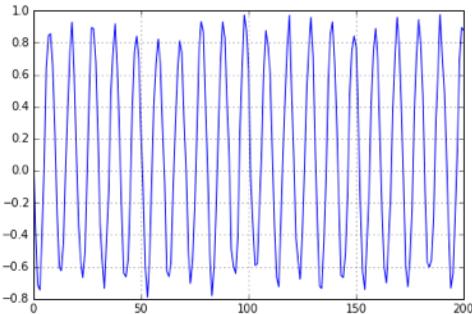
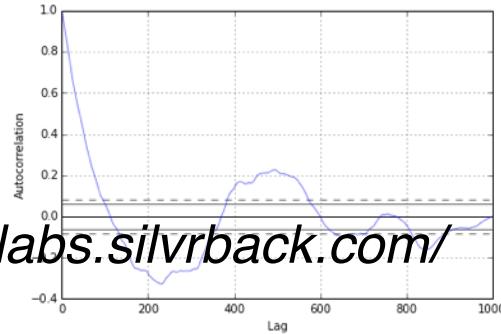
## LAG PLOT



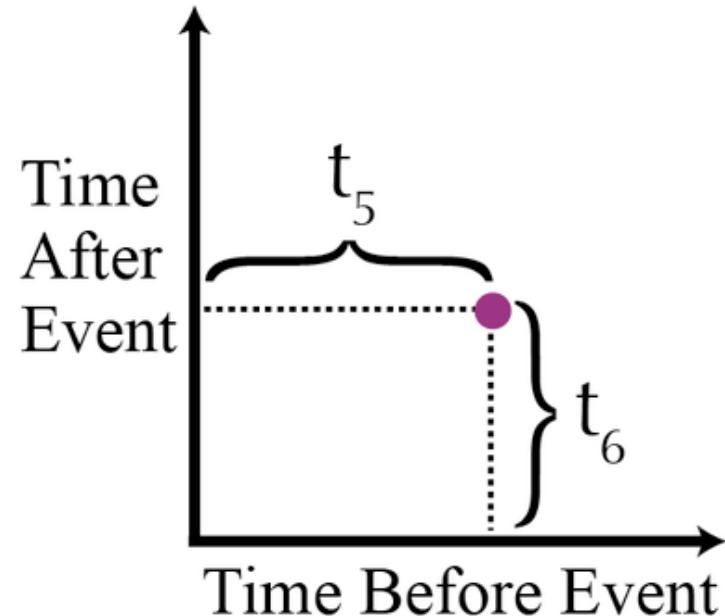
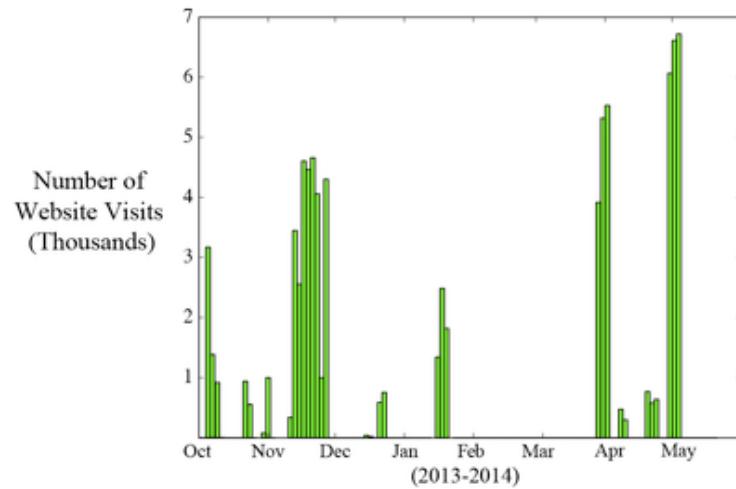
## AUTOCORRELATION FUNCTION



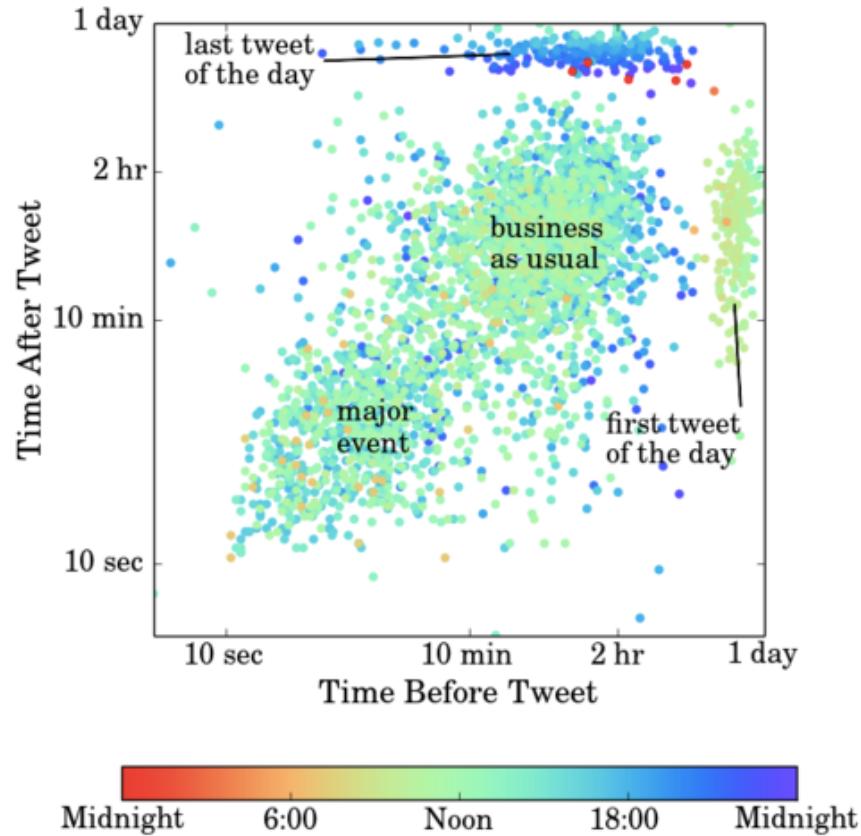
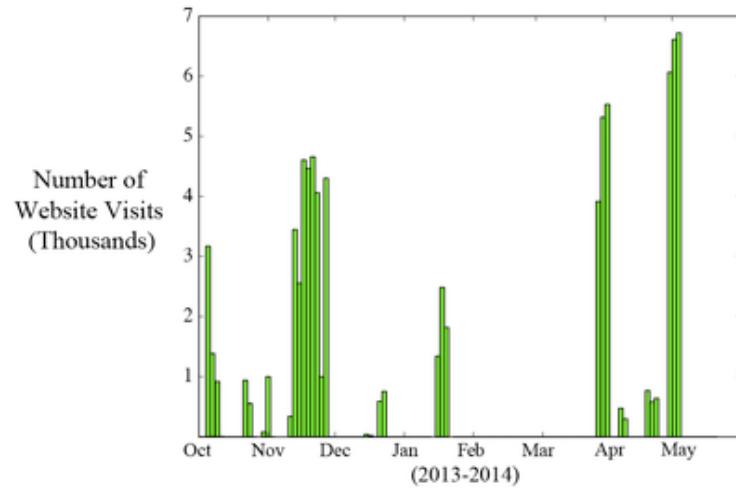
*https://districtdatalabs.silvrback.com/*



## TIME MAPS



## TIME MAPS



# VALIDATION BIAS

- Train - Test split
- Survivor bias
- Labelled data



Table I. Main Characteristics of Human Activity Recognition Systems

Type	Characteristic	Description
Execution	Offline	The system records the sensor data first. The recognition is performed afterwards. Typically used for non-interactive applications such as health monitoring.
	Online	The system acquires sensor data and processes it in real time. Typically used for activity-based computing and interactive applications in human-computer interaction.
Generalisation	User independent	The system is optimised for working with a large number of users.
	User specific	The system is tailored to a specific user. Performance is usually higher than in the user-independent case, but does not generalise as well to other users.
	Temporal	The system should be robust to temporal variations caused by external conditions (sensor displacement, drifting sensor response such as barometers or gyroscopes)
Recognition	Continuous	The system automatically “spots” the occurrence of activities or gestures in the streaming sensor data.
	Isolated (Segmented)	The system assumes that the sensor data stream is segmented at the start and end of a gesture by an oracle. It only classifies the sensor data in each segment into one of the activity classes. The oracle can be an external system (e.g. cross-modality segmentation) or the experimenter when assessing classification performance in the design phase.
Activities	Periodic	Activities or gestures exhibiting periodicity, such as walking, running, rowing, biking, etc. Sliding window segmentation and frequency-domain features are generally used for classification.
	Sporadic	The activity or gesture occurs sporadically, interspersed with other activities or gestures. Segmentation plays a key role to isolate the subset of data containing the gesture.
	Static	The system deals with the detection of static postures or static pointing gestures.
System model	Stateless	The recognition system does not model the state of the world. Activities are recognised by spotting specific sensor signals. This is currently the dominant approach when dealing with the recognition of activity primitives (e.g. reach, grasp).
	Stateful	The system uses a model of the environment, such as the user's context or an environment map with location of objects. This enhances activity recognition performance, at the expense of more design-time knowledge and a more complex recognition system.

# ENGINEERING CHALLENGES

- Time Series databases
- Latency
- Online VS Offline...
- [http://en.wikipedia.org/wiki/Time\\_series\\_database](http://en.wikipedia.org/wiki/Time_series_database)

---

**INTRO TO DATA SCIENCE**

---

# **REAL WORLD EXAMPLES**

---

# REAL WORLD EXAMPLES



# RECOGNIZING MOTION PRIMITIVES

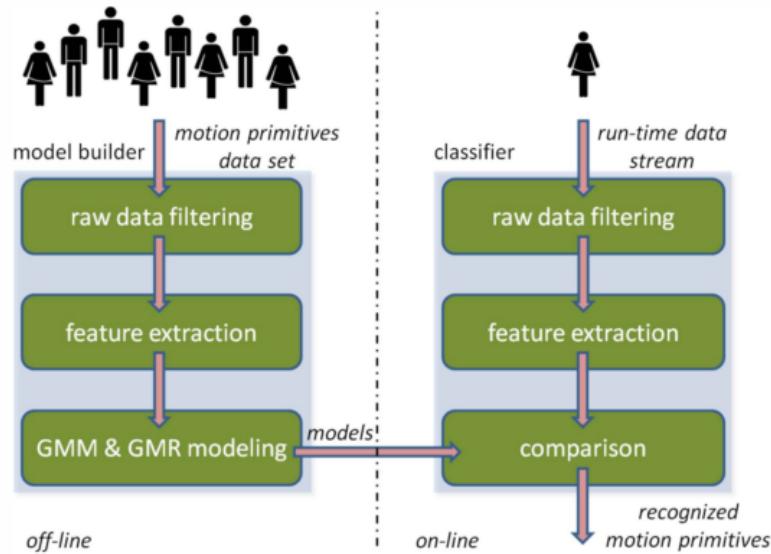


Fig. 1. System architecture.

# RECOGNIZING MOTION PRIMITIVES

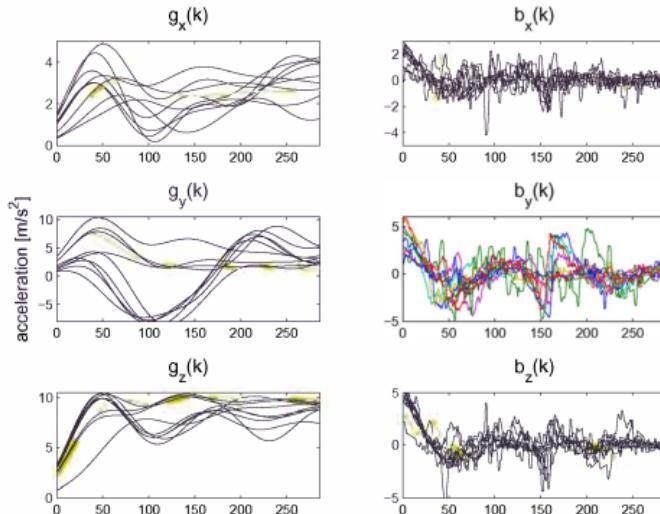


Fig. 2. Feature curves extracted from the trials of the *eating with knife and fork* motion primitive training set.

# RECOGNIZING MOTION PRIMITIVES

- trick in this case is to know the beginning of the action

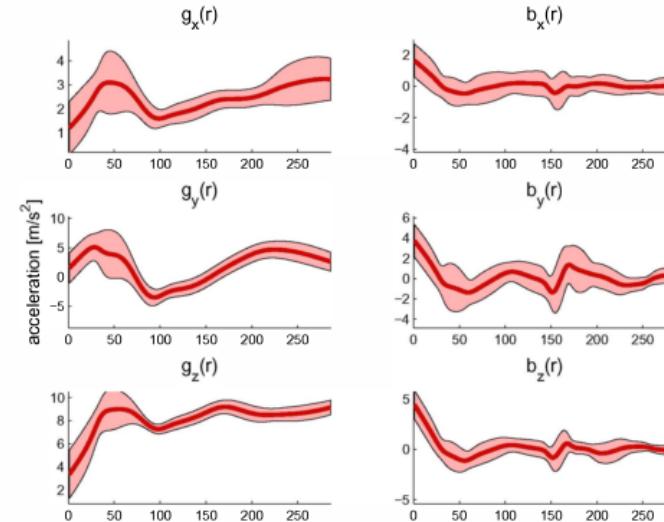
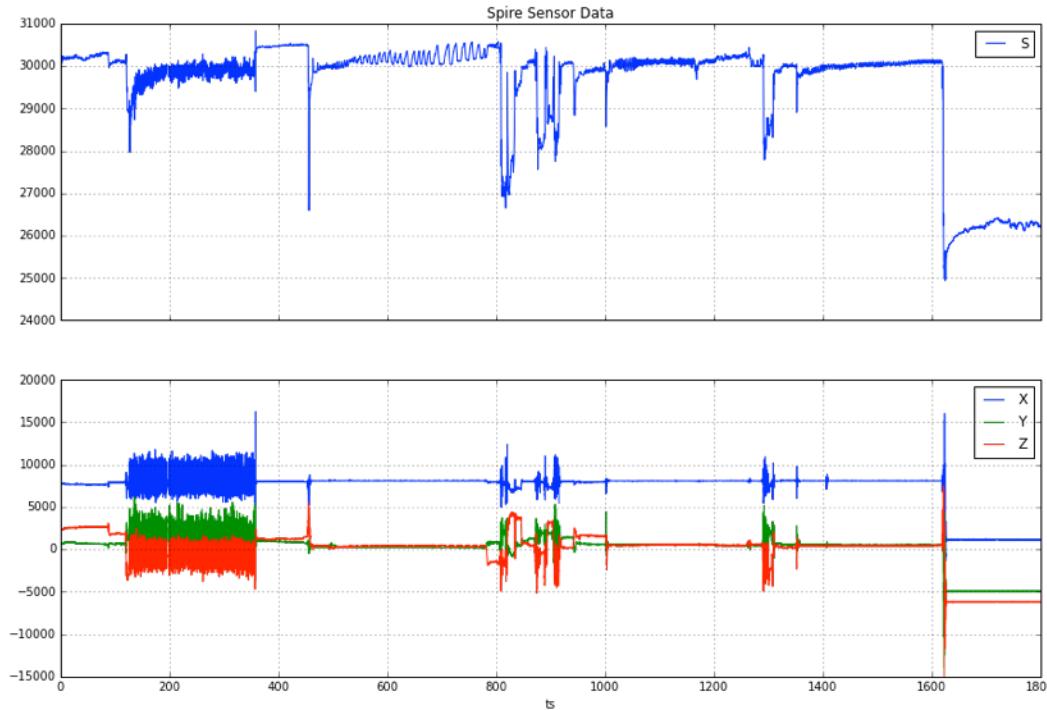


Fig. 3. 2D projections of the *eating with knife and fork* motion primitive model retrieved via GMR.

## SPIRE

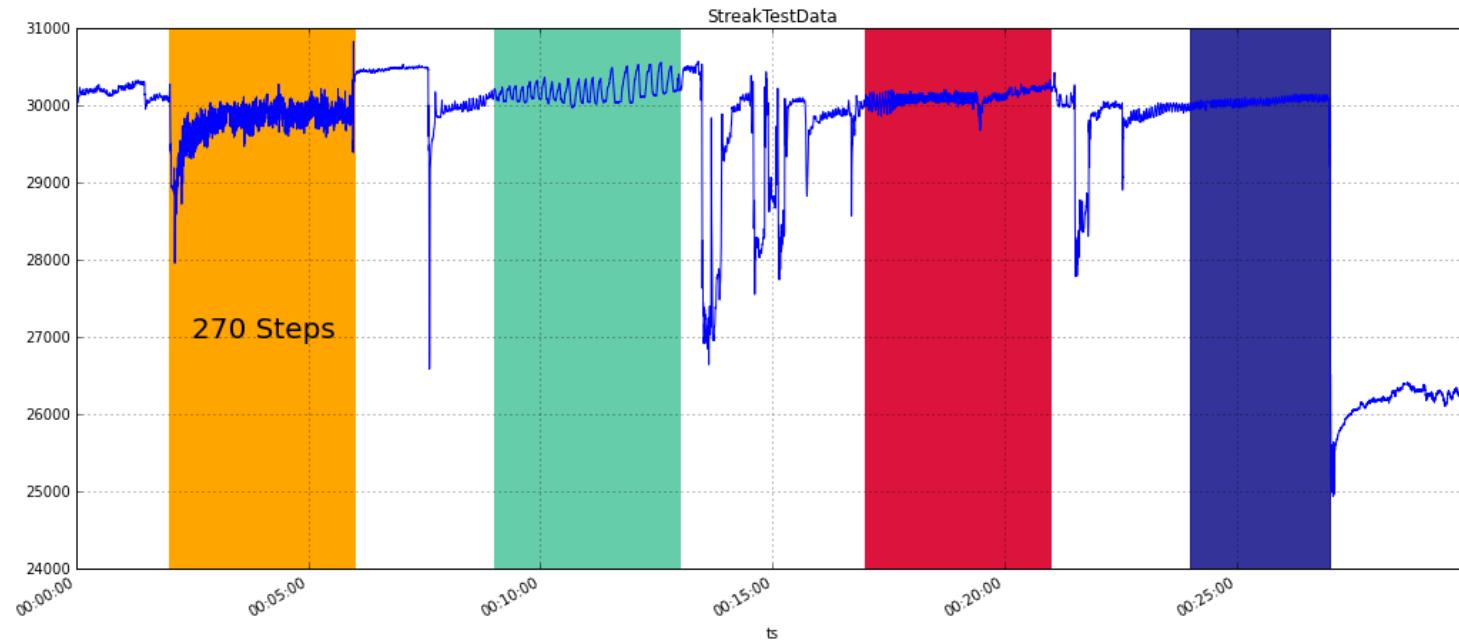


# SPIRE

- Wearable sensor
- Detects physical activity
- Counts steps
- Measures calories
- Detects Breathing Patterns
- Classifies Breathing in different States of Mind



## SPIRE



# DEEP SPEECH

- Recurrent Neural Networks (RNNs)

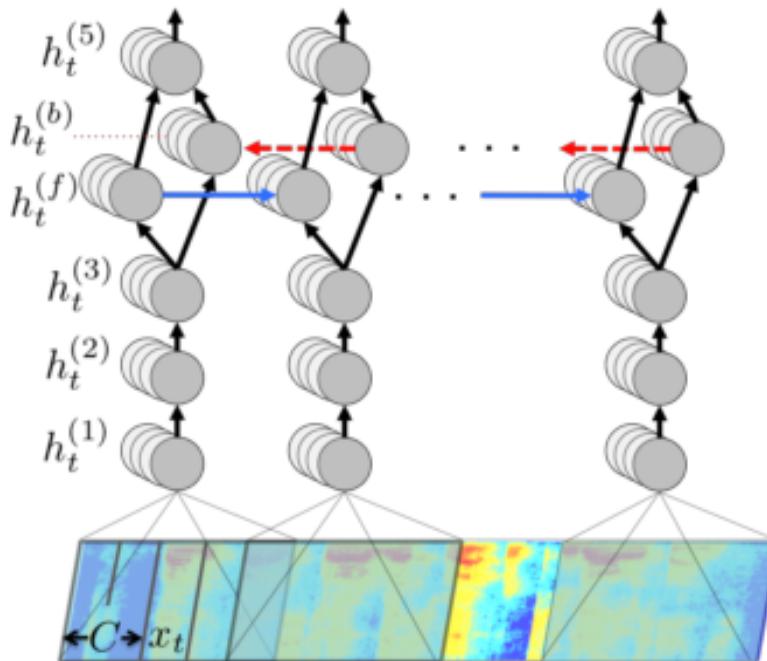


Figure 1: Structure of our RNN model and notation.

# DEEP SPEECH

Model	SWB	CH	Full
Vesely et al. (GMM-HMM BMMI) [43]	18.6	33.0	25.8
Vesely et al. (DNN-HMM sMBR) [43]	12.6	24.1	18.4
Maas et al. (DNN-HMM SWB) [28]	14.6	26.3	20.5
Maas et al. (DNN-HMM FSH) [28]	16.0	23.7	19.9
Seide et al. (CD-DNN) [39]	16.1	n/a	n/a
Kingsbury et al. (DNN-HMM sMBR HF) [22]	13.3	n/a	n/a
Sainath et al. (CNN-HMM) [36]	<b>11.5</b>	n/a	n/a
<b>DeepSpeech SWB</b>	20.0	31.8	25.9
<b>DeepSpeech SWB + FSH</b>	13.1	<b>19.9</b>	<b>16.5</b>

Table 3: Published error rates (%WER) on Switchboard dataset splits. The columns labeled “SWB” and “CH” are respectively the easy and hard subsets of Hub5’00.

# CONCLUSIONS

- Time Series Are different from Flat Datasets
- Preprocessing is very important
- Information can be encoded in time domain or frequency domain or both
- Windows are useful to extract higher order features
- Distance measures between time series useful for clustering
- Choice of tools will depend strongly on the problem to solve

# REFERENCES

- [http://en.wikipedia.org/wiki/Time\\_series](http://en.wikipedia.org/wiki/Time_series)
- [http://www.cs.ucr.edu/~eamonn/selected\\_publications.htm](http://www.cs.ucr.edu/~eamonn/selected_publications.htm)
- <http://www.stat.berkeley.edu/~bartlett/courses/153-fall2010/lectures/1.pdf>