Machine Learning applied to Planetary Sciences

PTYS 595B/495B

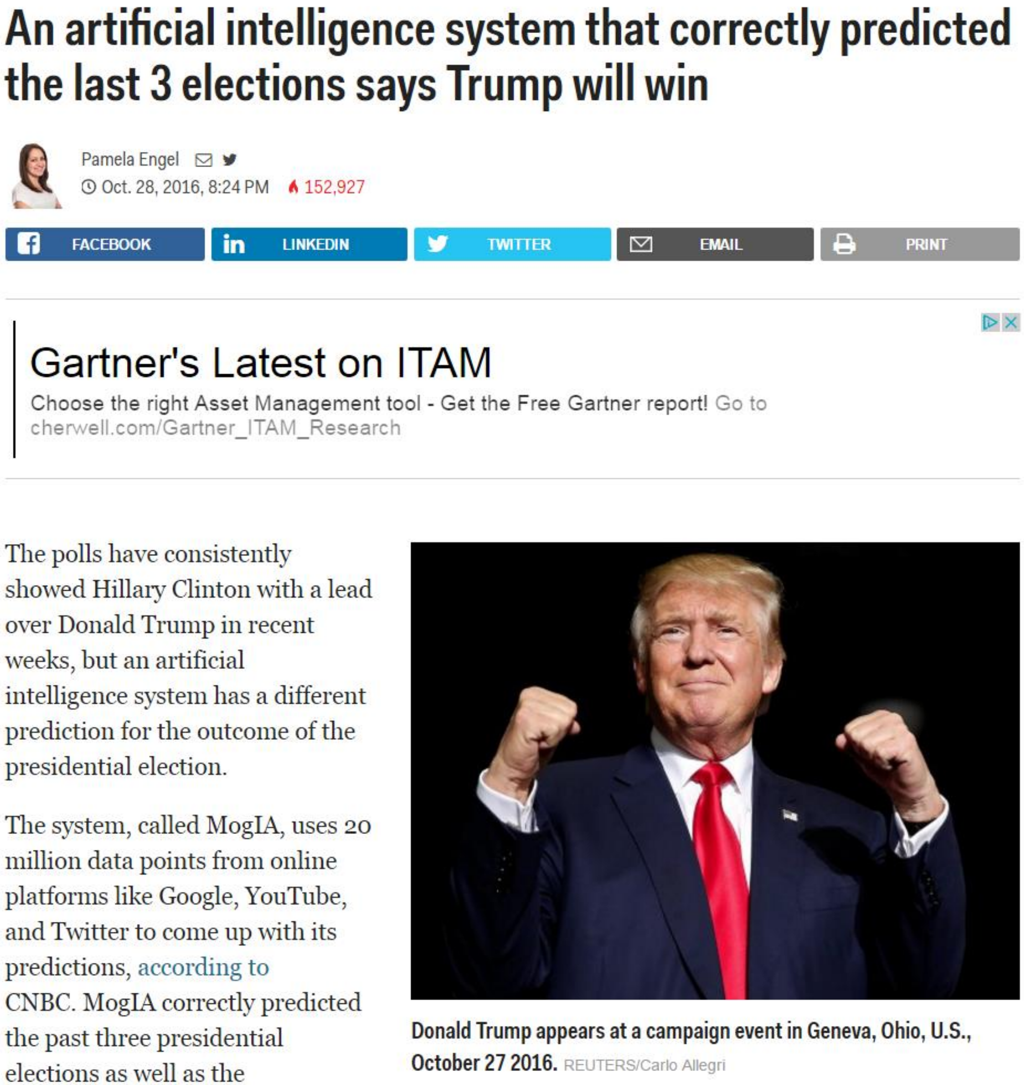
Leon Palafox

https://leonpalafox.github.io/MLClass/

Experts can be wrong

RAGÚ America's #1 Pasta Sauce Brand
2 Servings of VEGGIES in Every ½ Cup of Sauce
Old World Style
RAGÚ
Traditional
Since 1937

In the news!



An artificial intelligence system that correctly predicted the last 3 elections says Trump will win

Pamela Engel
Oct. 28, 2016, 8:24 PM 152,927

The polls have consistently showed Hillary Clinton with a lead over Donald Trump in recent weeks, but an artificial intelligence system has a different prediction for the outcome of the presidential election.

The system, called MogIA, uses 20 million data points from online platforms like Google, YouTube, and Twitter to come up with its predictions, according to CNBC. MogIA correctly predicted the past three presidential elections as well as the

Donald Trump appears at a campaign event in Geneva, Ohio, U.S., October 27 2016. REUTERS/Carlo Allegri

Application to time series data

Problem Statement

- Both in astronomy and planetary science time series data is important.
- Even if it’s not a time series, we can use these techniques in other datasets.
- Most of the techniques used for time series are rather old.
  - We won’t be talking about those.
- Traditional Machine Learning approaches are very effective.

Datasets

- Sunspots dataset (people just love this one)

- Light curves.

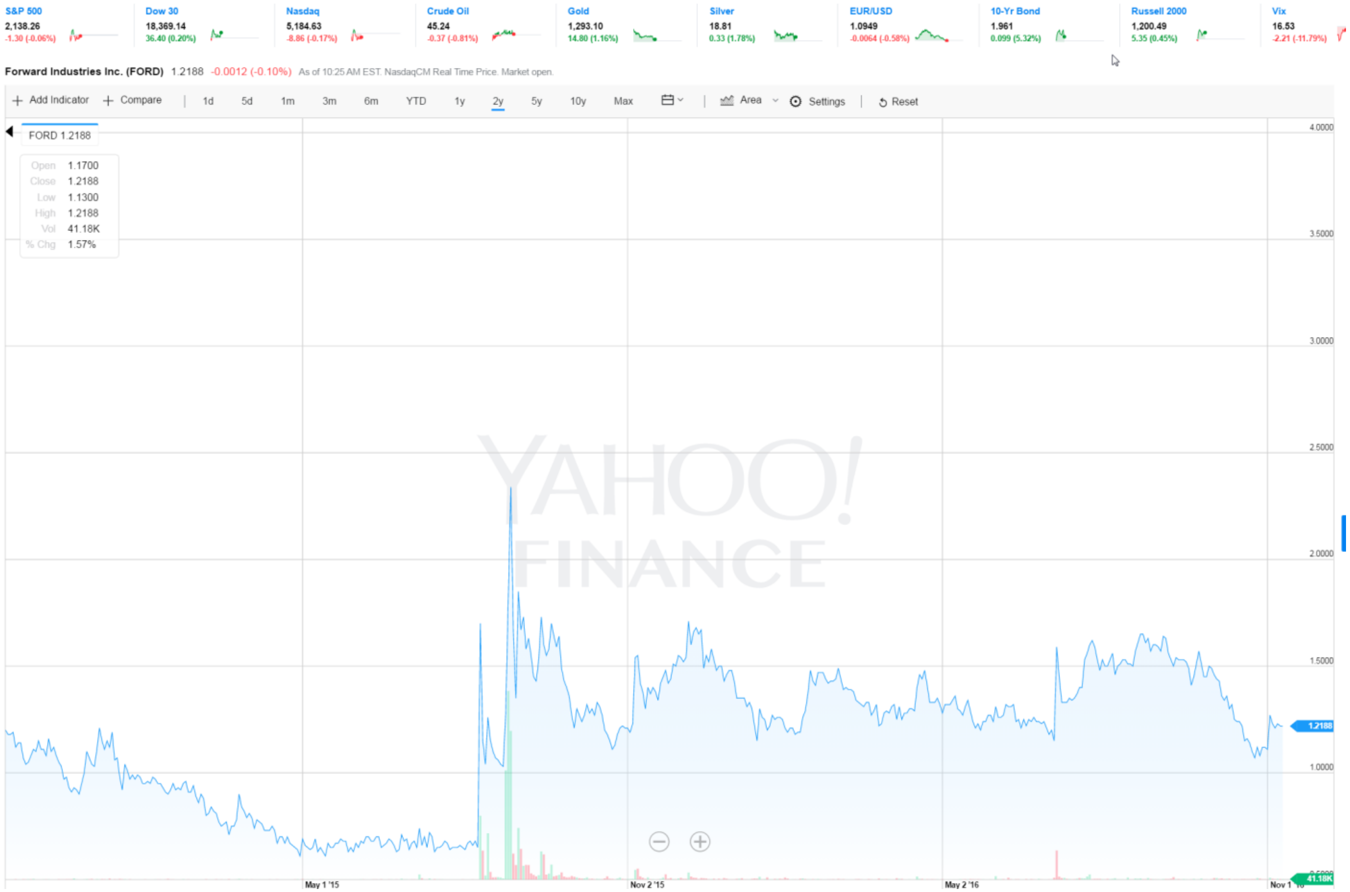
- Terrain Profiles (not strictly time-series, but works!)

Problems

- Classification:
  - We would like to know whether there is a signal at one particular time.
  - We would like to know if the time series is informative of different effects.
- Regression:
  - We would like to predict the time series in the future.

Problems

- Time Series data is usually 1D, noisy and incomplete.



S&P 500	Dow 30	Nasdaq	Crude Oil	Gold	Silver	EUR/USD	10-Yr Bond	Russell 2000	Vix
2,138.26	18,369.14	5,184.63	45.24	1,293.10	18.81	1.0949	1.961	1,200.49	16.53
-1.30 (-0.06%)	36.40 (0.20%)	-8.86 (-0.17%)	-0.37 (-0.81%)	14.80 (1.16%)	0.33 (1.78%)	-0.0064 (-0.58%)	0.099 (5.32%)	5.35 (0.45%)	-2.21 (-11.79%)

Forward Industries Inc. (FORD) 1.2188 -0.0012 (-0.10%) As of 10:25 AM EST. NasdaqCM Real Time Price. Market open.

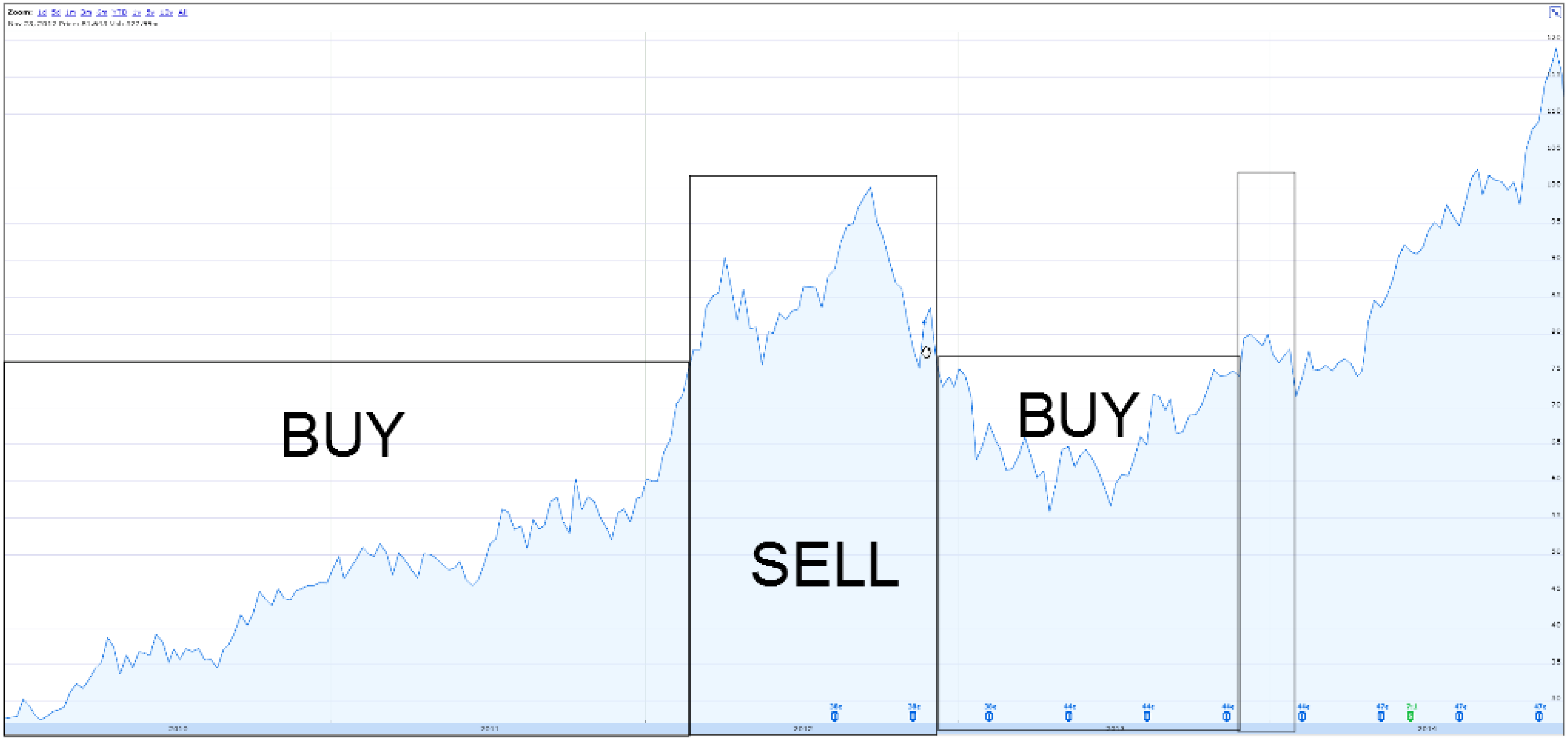
FORD 1.2188

Open	1.1700
Close	1.2188
Low	1.1300
High	1.2188
Vol	41.18K
% Chg	1.57%

Feature selection

- If we have multiple time series, I can get easier, or harder.
- Time series are remarkably non-stationary.
- Non-Stationarity:
  - The signal does not remain constant over time
  - Single data points are very uninformative

Stock Market

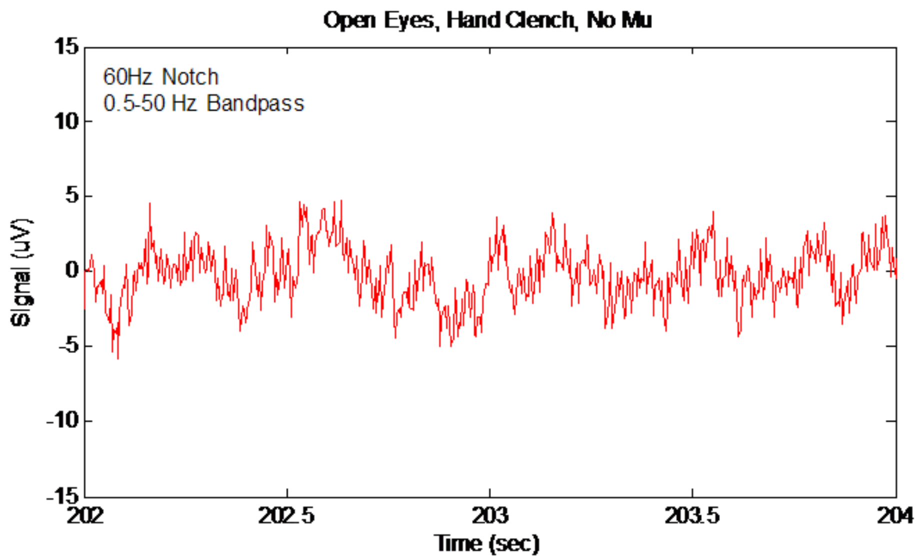


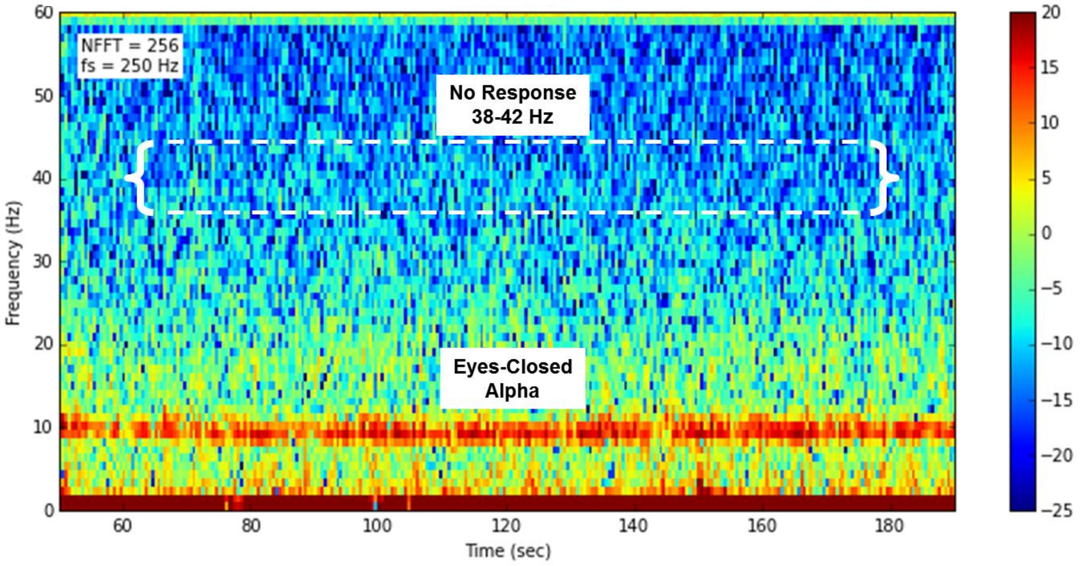
You are hired to create a classifier that will sell or buy.

Features

- We need to create features that will be stationary.
- Spectrogram
  - Use Fourier transforms at certain interval windows.
  - The plot becomes one of Time vs Frequency
  - Single command in Matlab and in Python

EEG Example

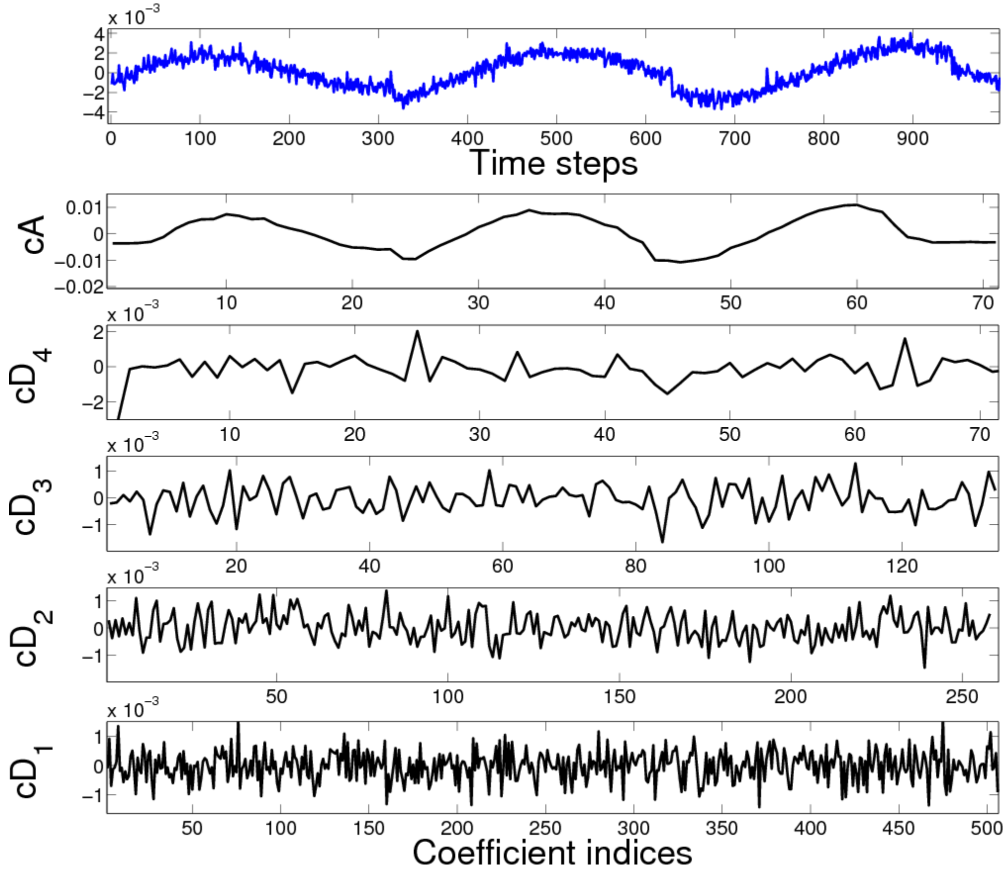

Open Eyes, Hand Clench, No Mu


No Response
38-42 Hz

Eyes-Closed
Alpha

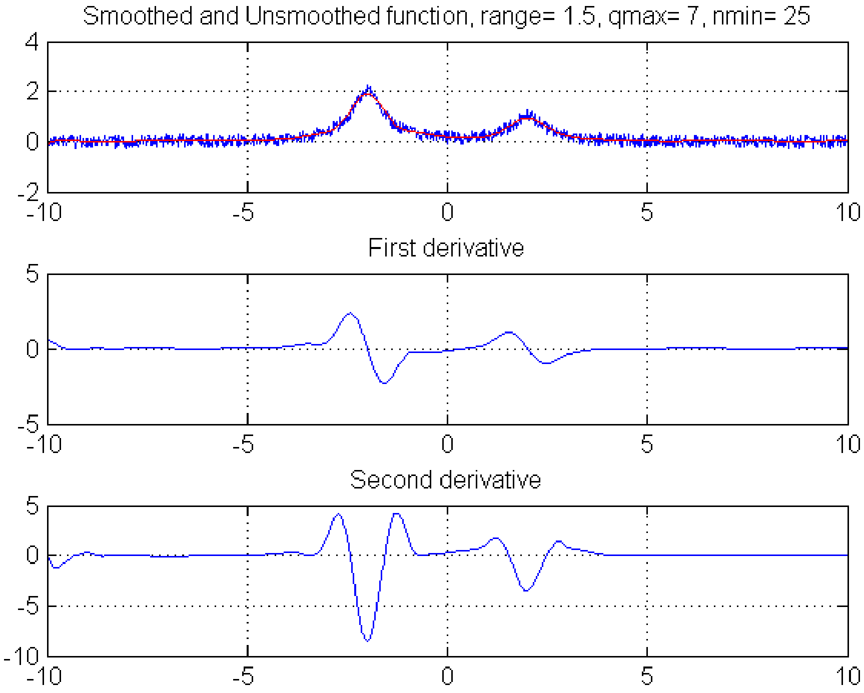
Features

- Wavelet Transform



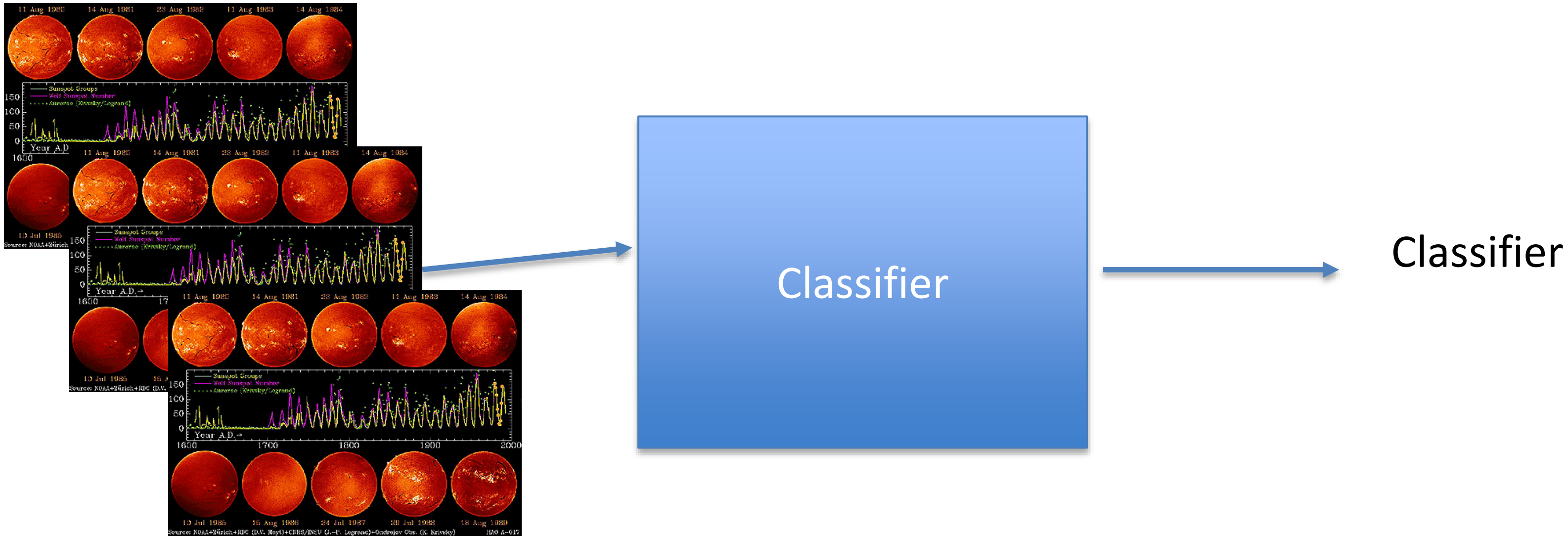
Features

- Trend extraction
  - Differencing of time series.



Features

- Full data extraction
  - Only if we have small, multiple time-series.

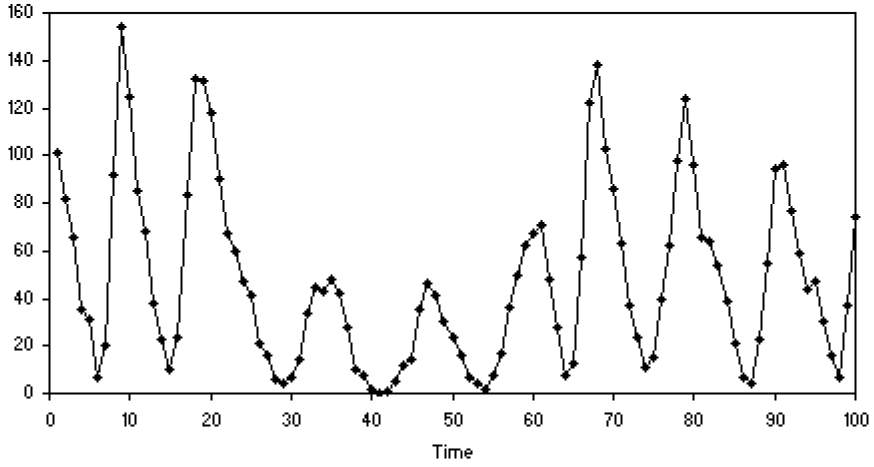


Classifier

Classifier

Testing-Training

- How do we select the training and testing data?
  - Just using our traditional approach of random sampling won’t do anything for us.



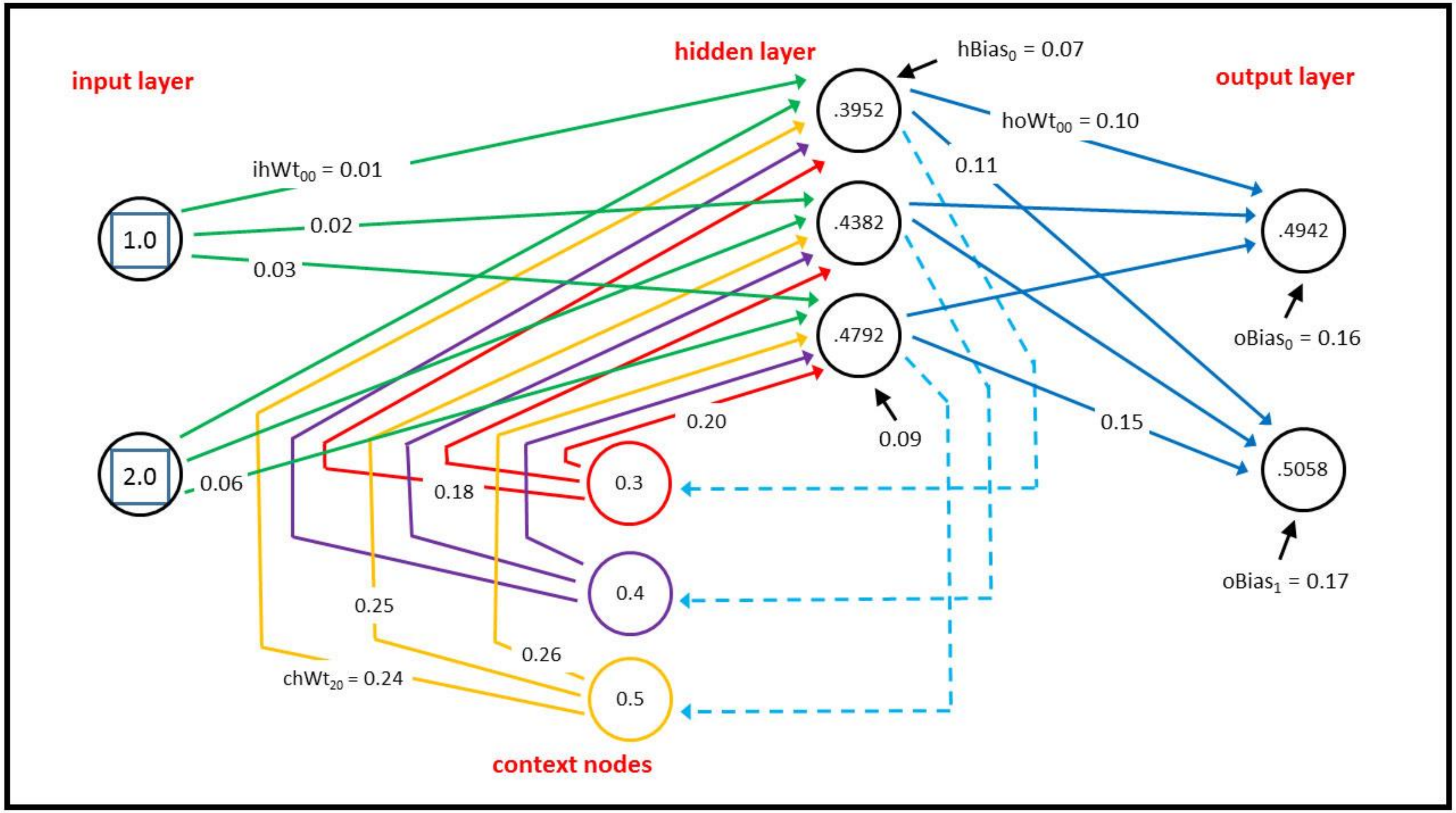
There is no one approach

- Divide the time series in N consecutive folds to do CV.
  - Do not take random points.
- You can train with only half the image and test with the other half.
  - 2-Fold CV
- Testing and Training have to be roughly the same size.
  - You can and will overfit time series data. Regardless of what you do.
    - Course of non-stationarity

Real time data series analysis

- What is needed:
  - If the problem is regression in real time:
    - Gradient descent, and training as we get new data.
    - We can’t have an obscene number of features.
- Which algorithms can do this:
  - SVMs are surprisingly bad to do things in real-time.
  - Neural Nets and Deep Learning are getting there, but they are also not so hot.

Recurrent Neural Networks



input layer
hidden layer
hBias$_0$ = 0.07
output layer
.3952
hoWt$_{00}$ = 0.10
0.11
ihWt$_{00}$ = 0.01
.4382
0.02
1.0
0.03
.4942
.4792
oBias$_0$ = 0.16
0.20
0.09
0.15
2.0
0.06
0.18
0.3
.5058
0.4
0.25
oBias$_1$ = 0.17
0.26
chWt$_{20}$ = 0.24
0.5
context nodes