

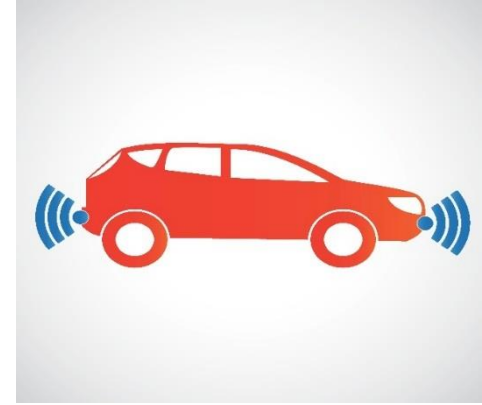
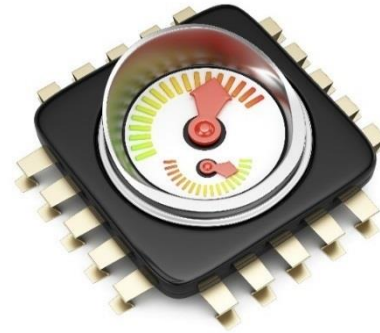
The background features a dark blue field on the left and a grey field on the right, separated by a diagonal line. In the upper right, there are white, stylized waveforms. In the lower right, there is a 3D wireframe mesh with a color gradient from yellow at the top to blue at the bottom. On the far right, there are faint blue circuit-like diagrams.

MATLAB EXPO 2017

Preprocessing & Feature Extraction in Signal Processing Applications

Rick Gentile
Product Manager
Signal Processing and Communications

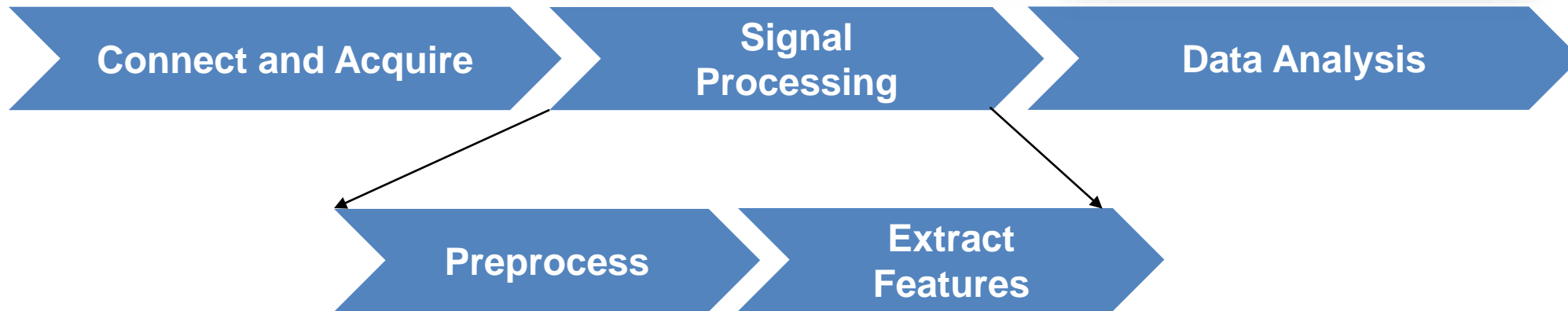
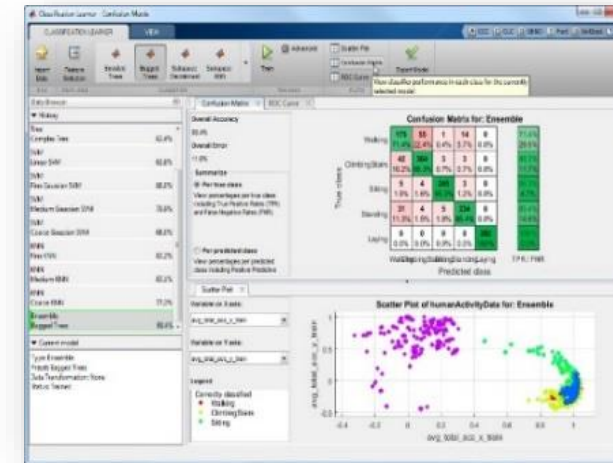
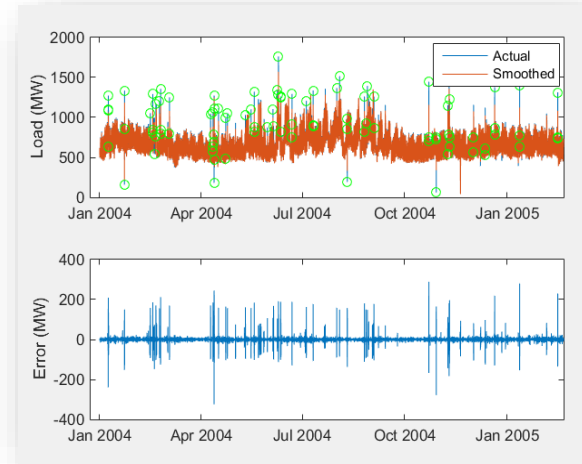
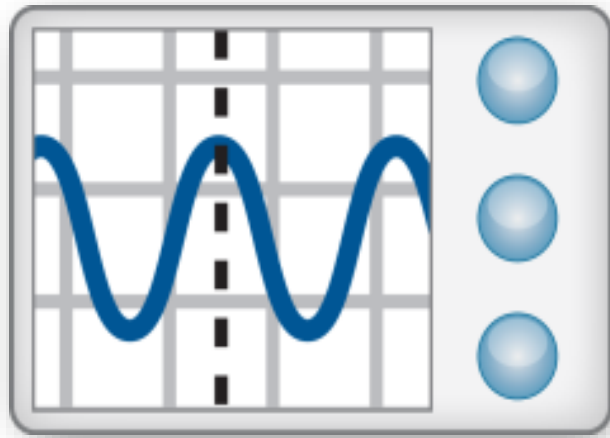
Signals and Data are Everywhere



phase noise **motion**
acceleration position
 pressure vibration strain
temperature tilt rotation



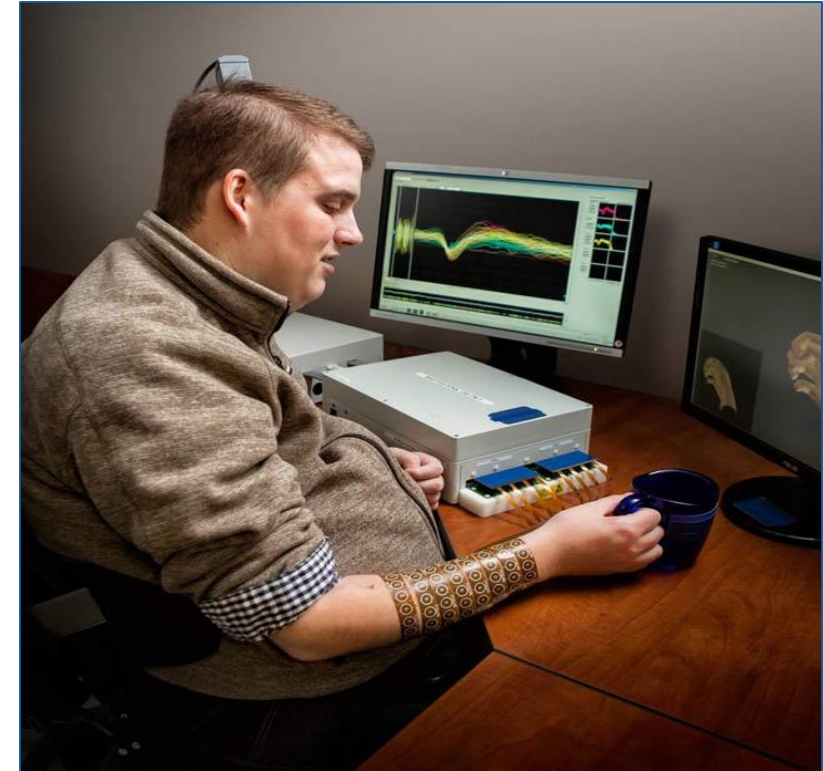
Preprocess and Extract Features for Data Analysis



Challenge: Gain insights to improve data analysis

Feature Extraction Techniques Help to Restore Arm Movement

- Multichannel electrode implanted in the brain to record brain signals
- Wavelet techniques isolate frequency bands of brain signals that govern movement
- Wavelets help transform 3000 features per channel into a single value

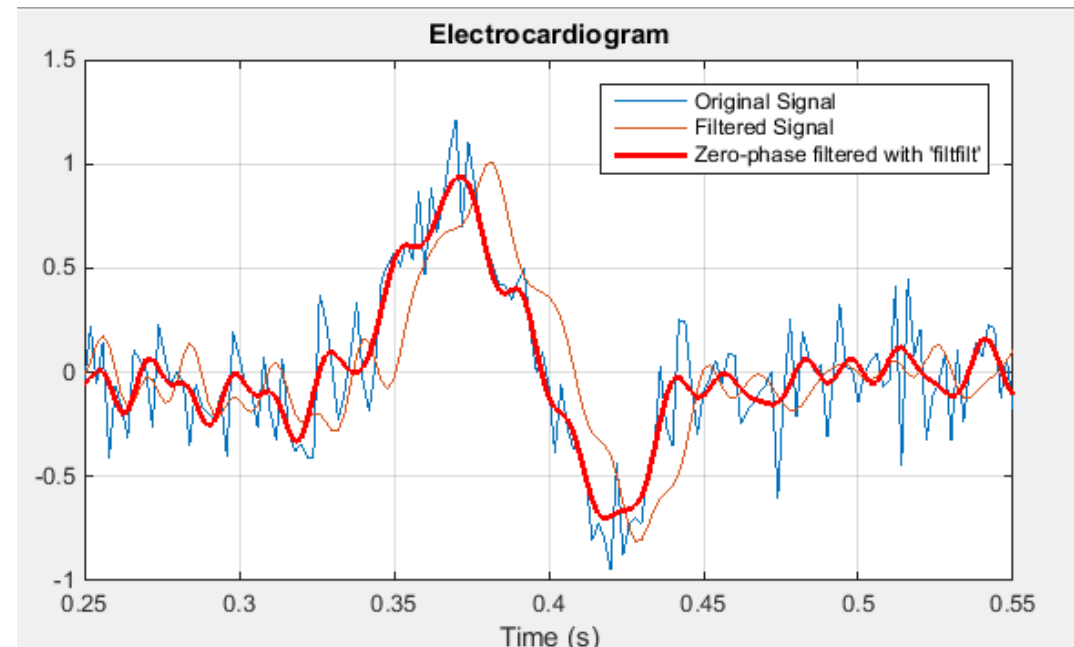
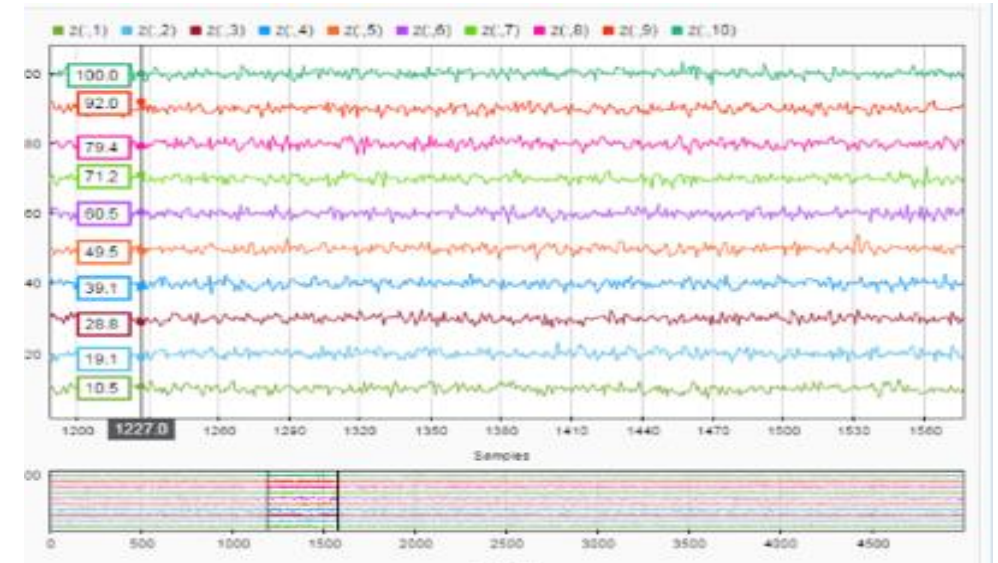


[User Story: Battelle Neural Bypass Technology Restores Movement to a Paralyzed Man's Arm and Hand](#)

Developed by Battelle Memorial Institute entirely in MATLAB and Wavelet Toolbox

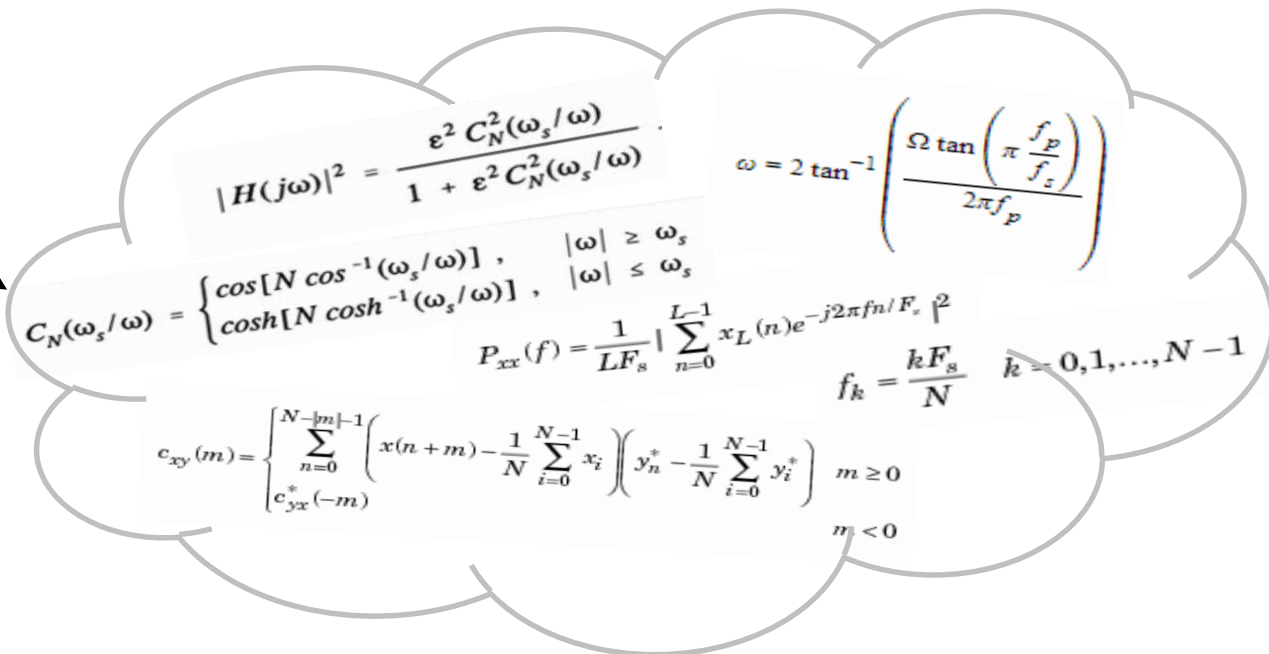
Real-World Signals are Challenging to Analyze

- Large amounts of data
 - Wide data → multiple streams, many sensors
 - Tall data → long signals
- “Messy” time series
 - Noise
 - Non-uniform sampling
 - Lack of alignment between signals
 - Missing data
 - Data outliers



Signal Processing for Engineers and Scientists

How do I compare signals? Is this a signal or just noise? How do I align different signals?
 Are these signals related? How do I measure a delay between signals?

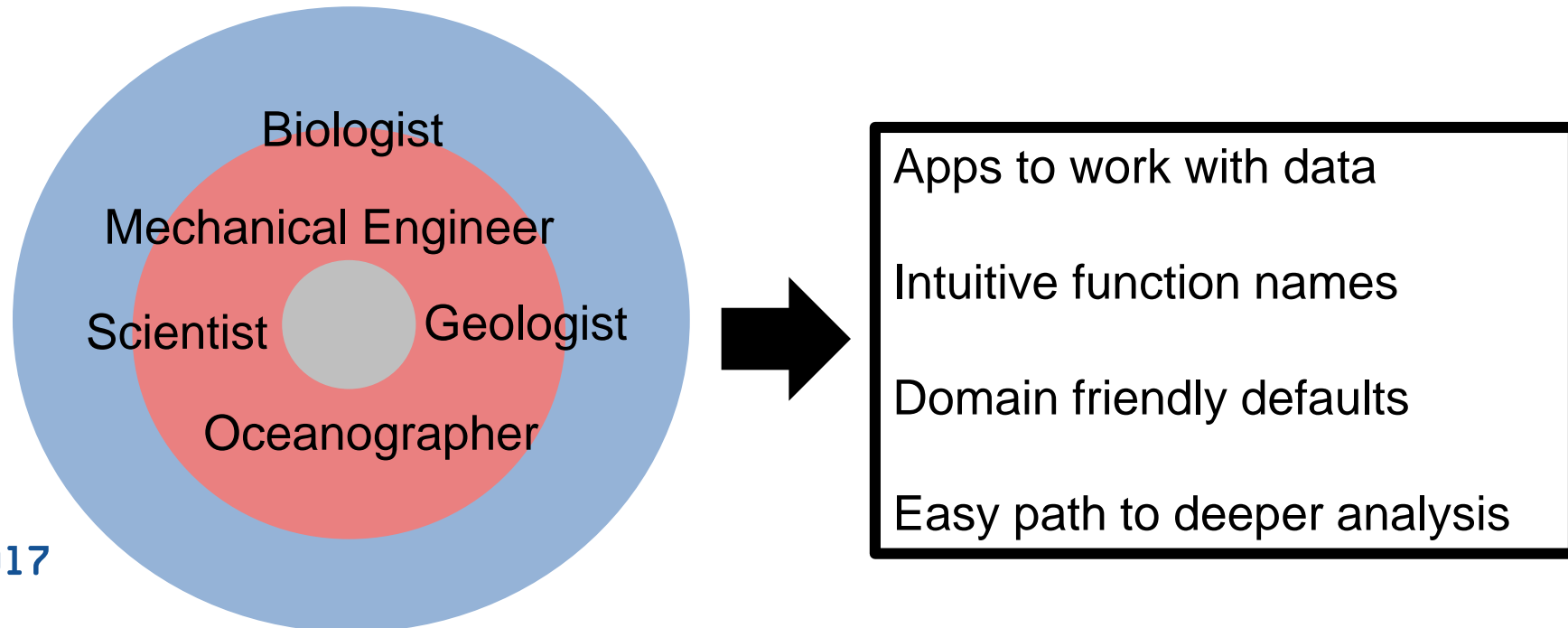


Support for Real-World Applications

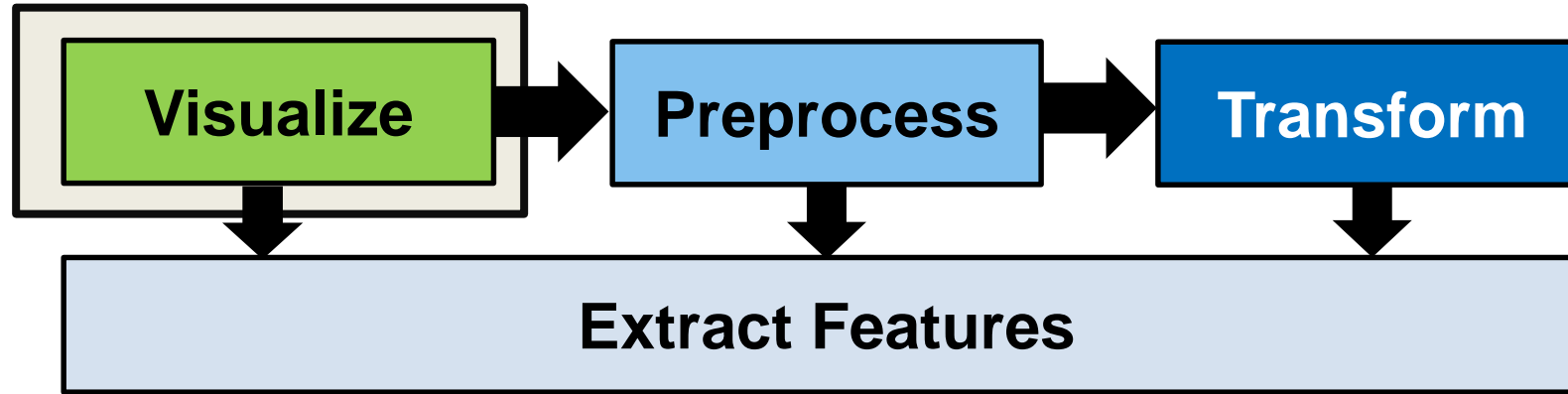
Signal Processing Toolbox

Wavelet Toolbox

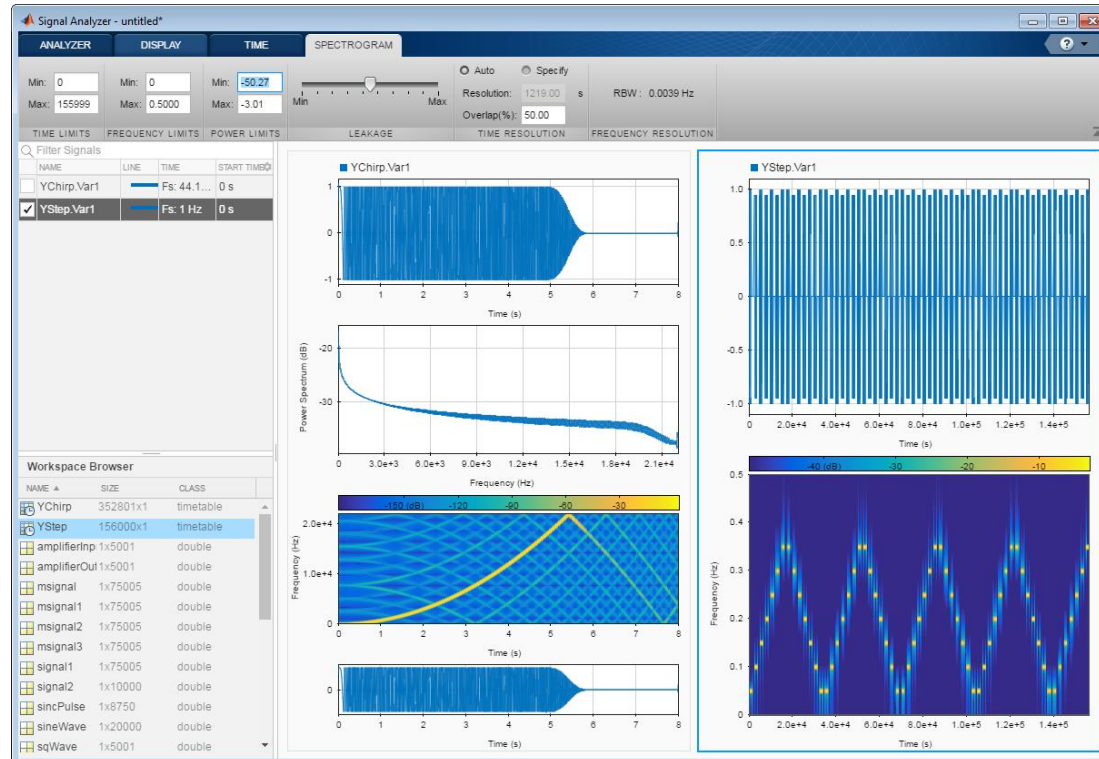
- Traditional users: Electrical Engineer with Signal Processing background
- Expanded focus over recent releases:
 - Scientists require signal processing techniques but may not be proficient in this area



Signal Preprocessing and Feature Extraction



Signal Analyzer App

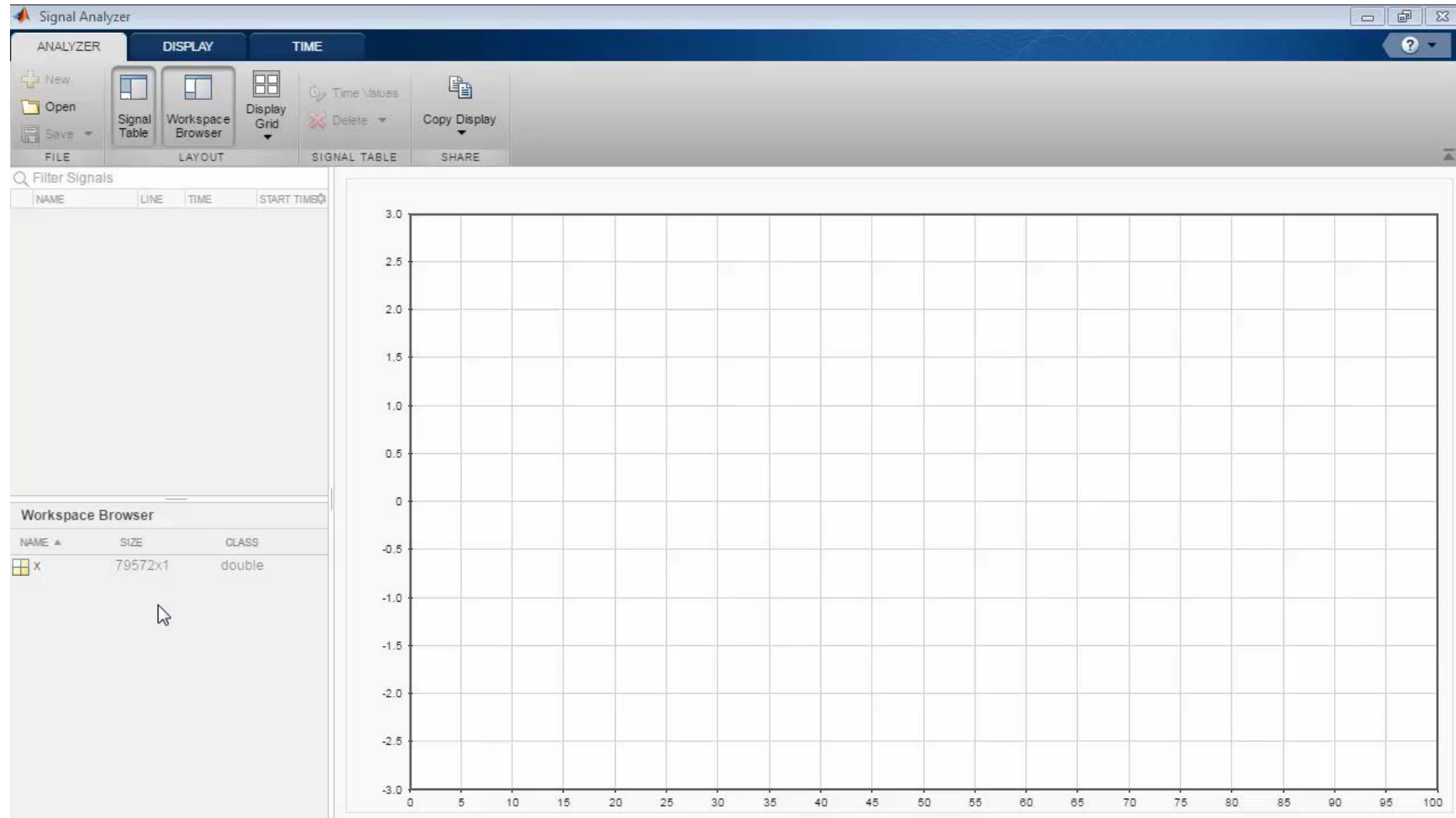


Viewing and Exploring Signals with Signal Analyzer App

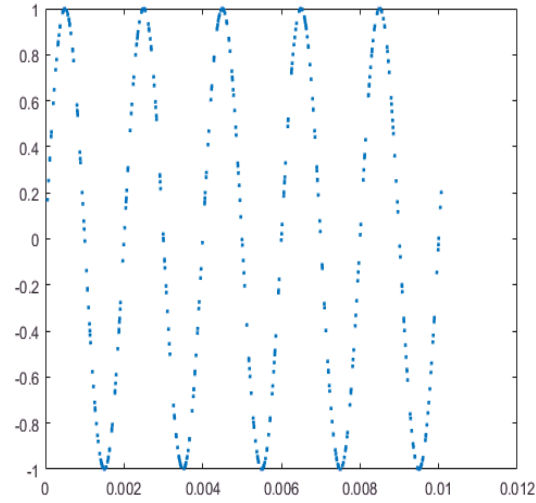
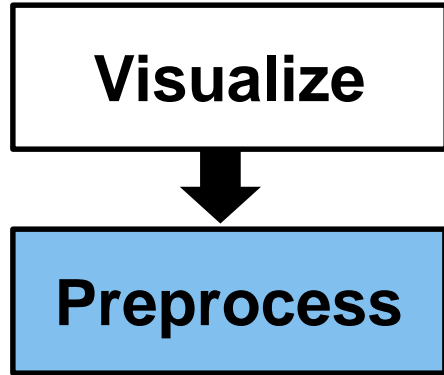
Visualize

Extraction
Features

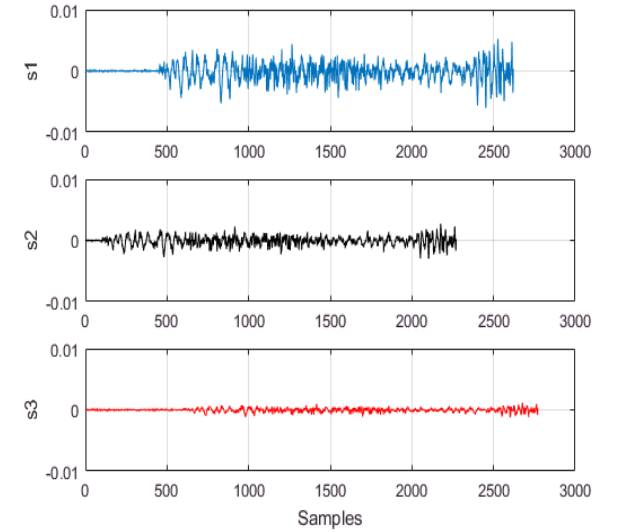
- Time and frequency
- Navigate, pan, & zoom
- Compare multiple signals
- Extract regions of interest



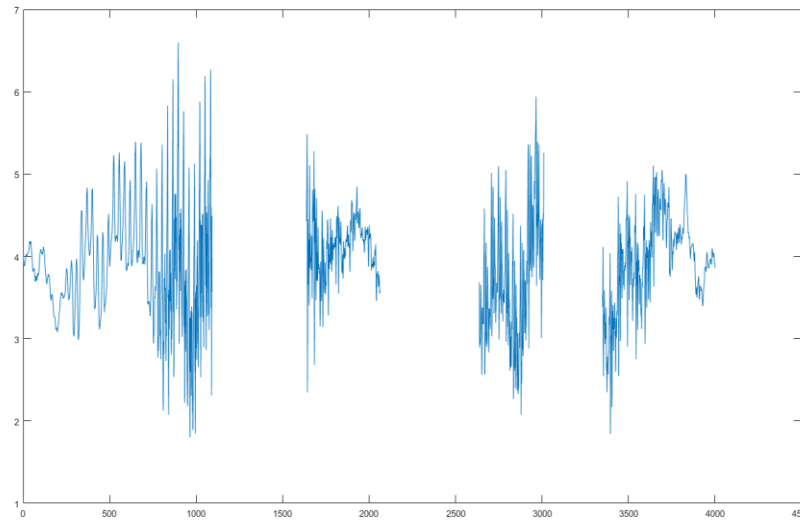
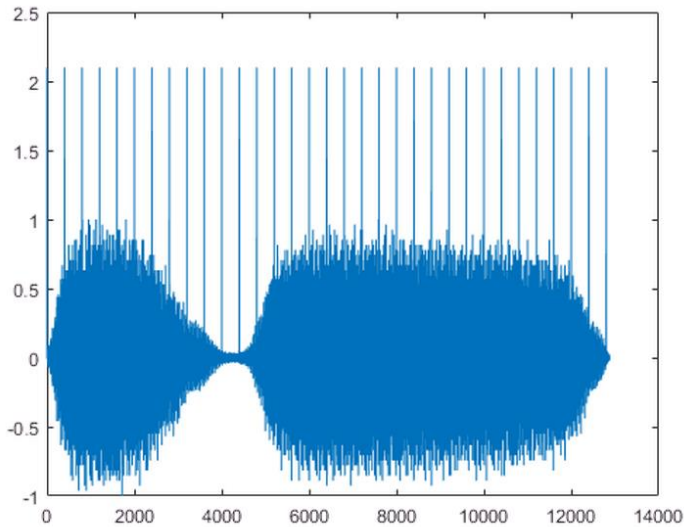
Preprocess Messy Signals



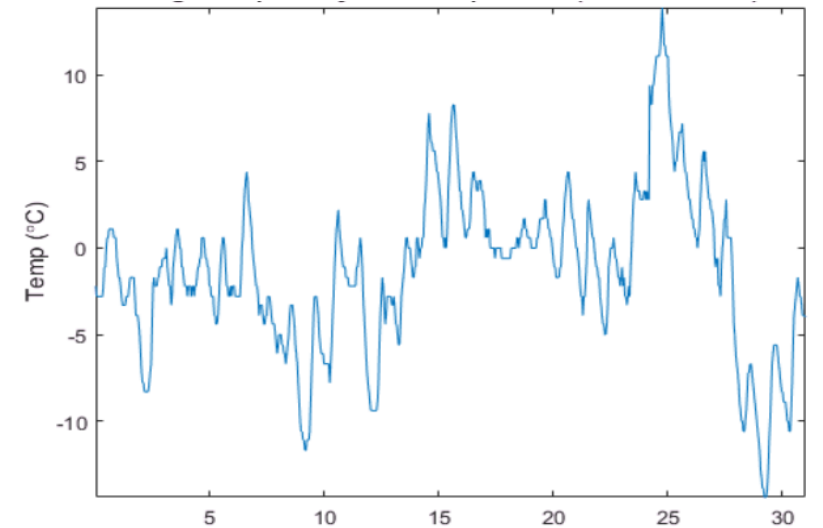
Non-uniformly sampled signals



Misaligned signals

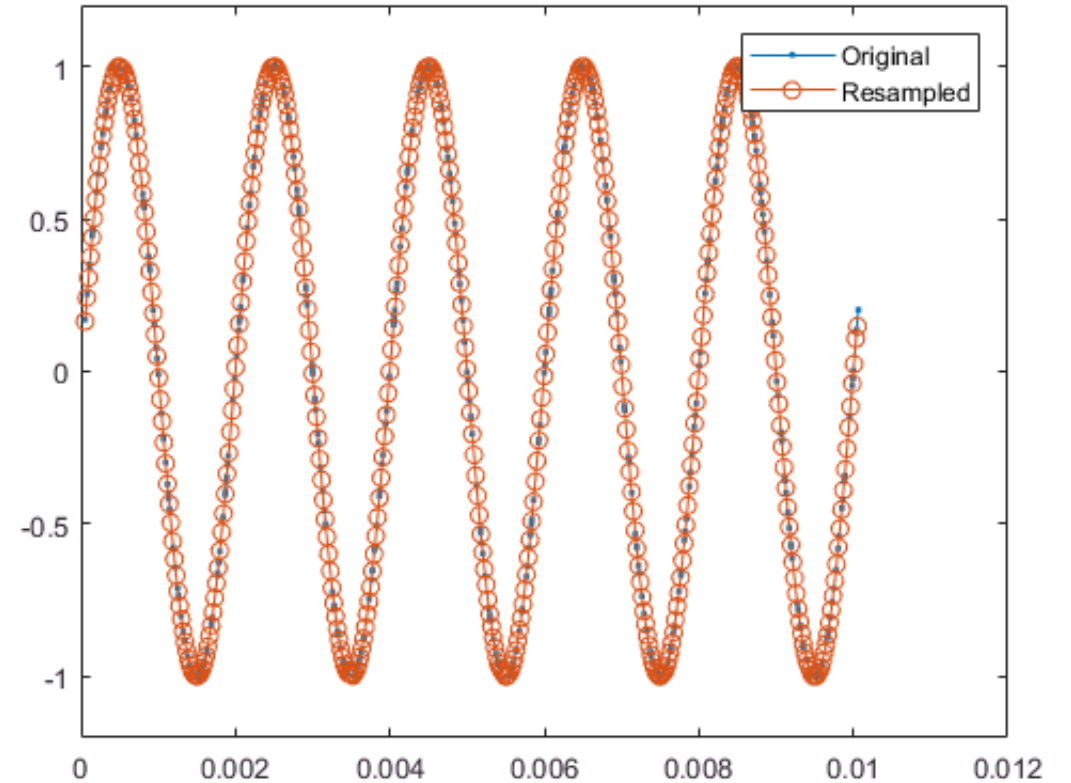
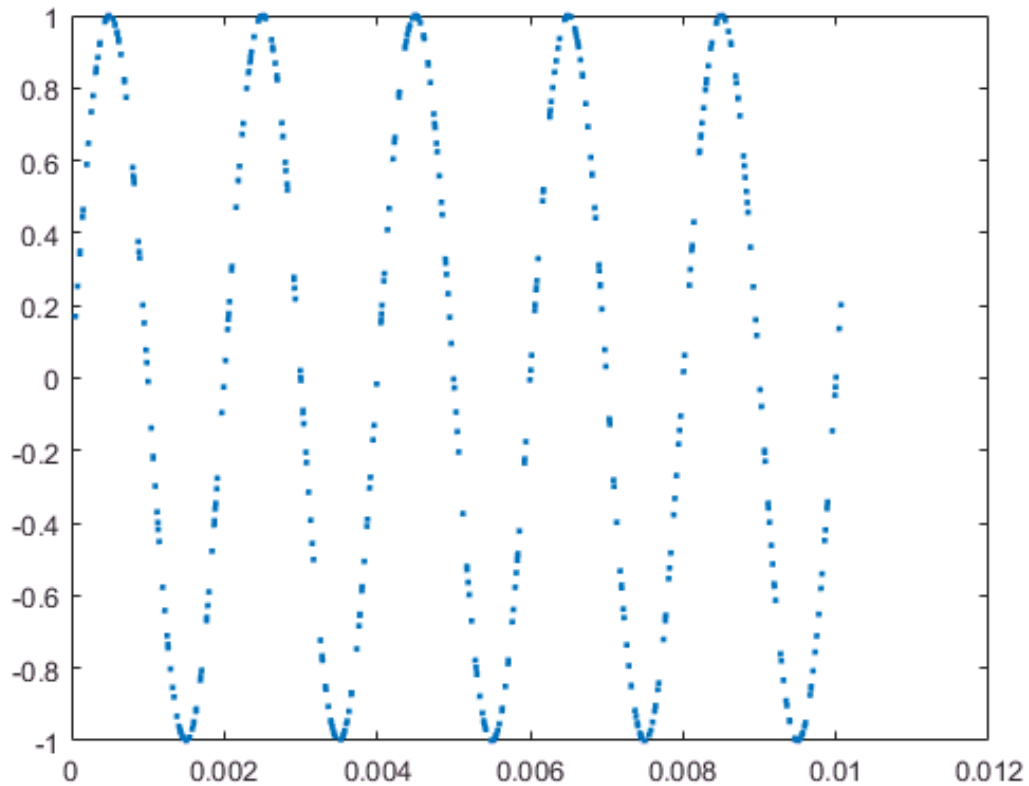


Outliers & data gaps

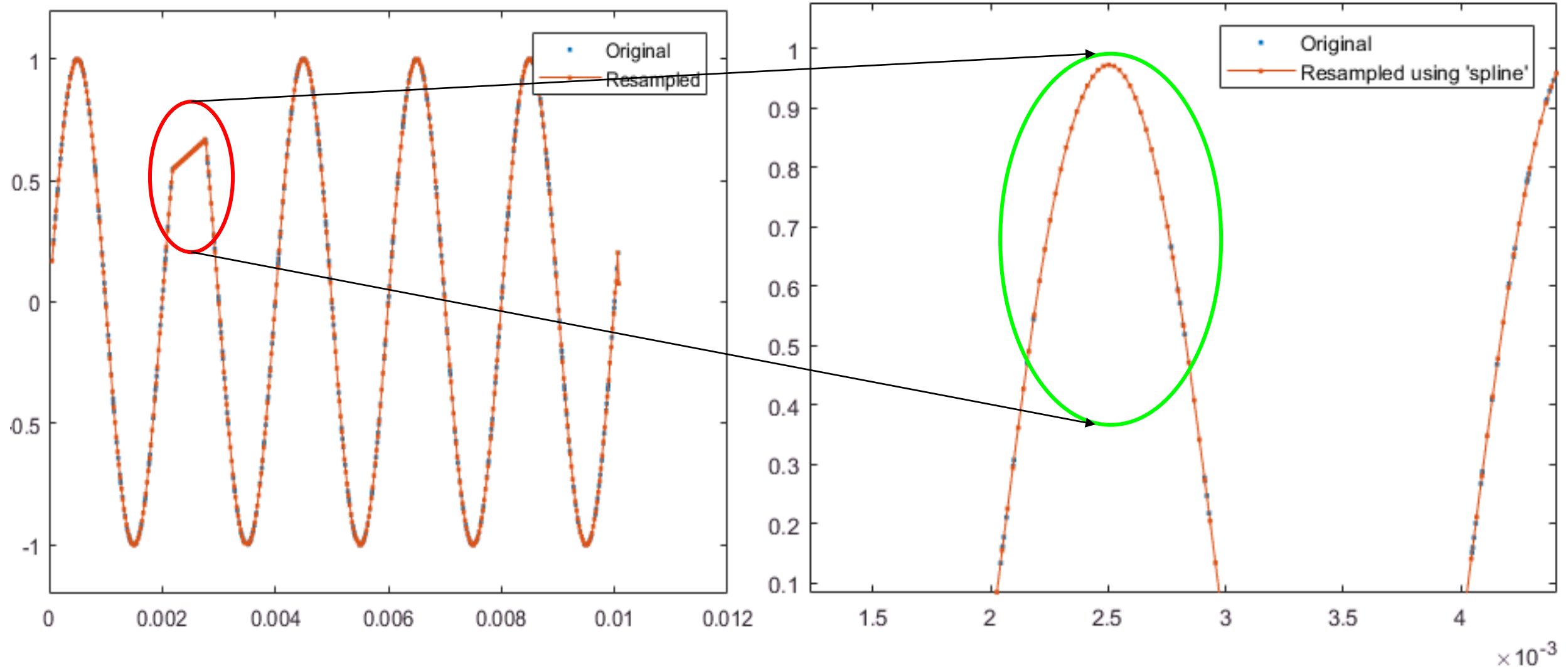


Noise or unwanted frequency content

Resample Non-uniformly Sampled Signals

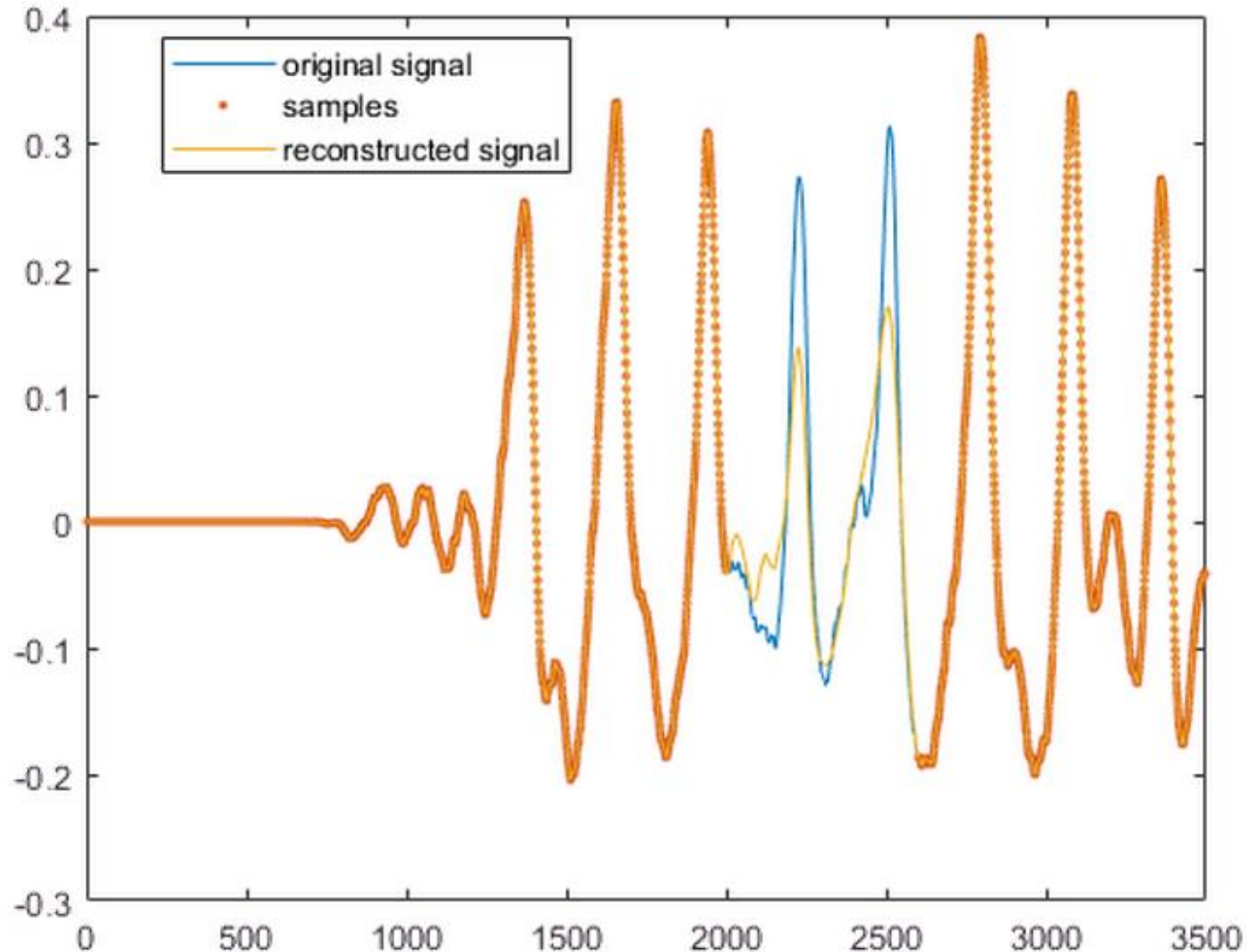


What if Data is Missing?



```
>> [y, Ty] = resample(x, irregTx, desiredFs, 'spline');
```

Multiple Ways to Reconstruct Missing Data

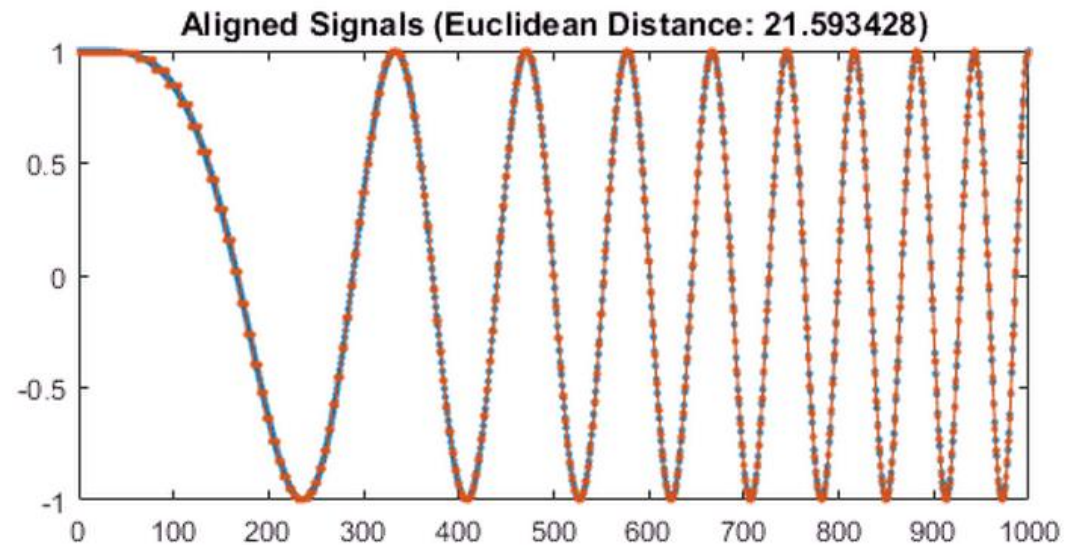
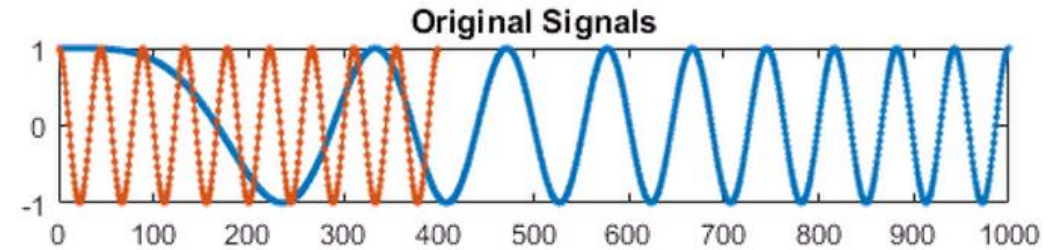
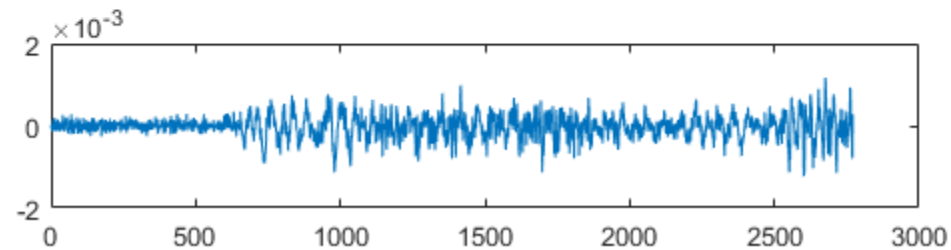
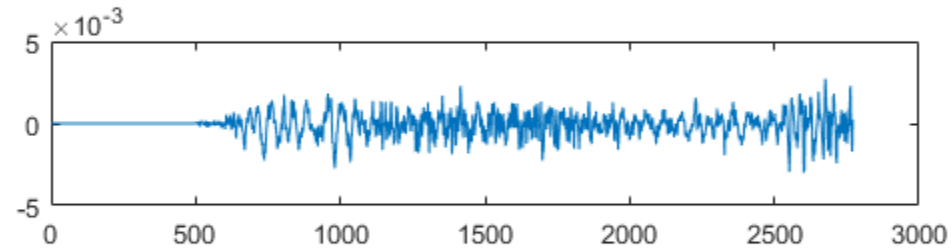
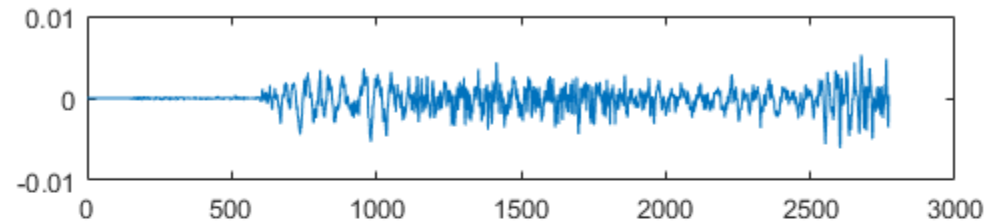


- Resampling often best for low frequency components
- For large gaps in wideband signals, autoregressive modeling is more effective

```
>> x = y(1:3500);  
>> x(2000:2600) = NaN;  
>> y2 = fillgaps(x);
```


Synchronizing Signals from Multiple Sensors

Data collected asynchronously by multiple sensors may require alignment



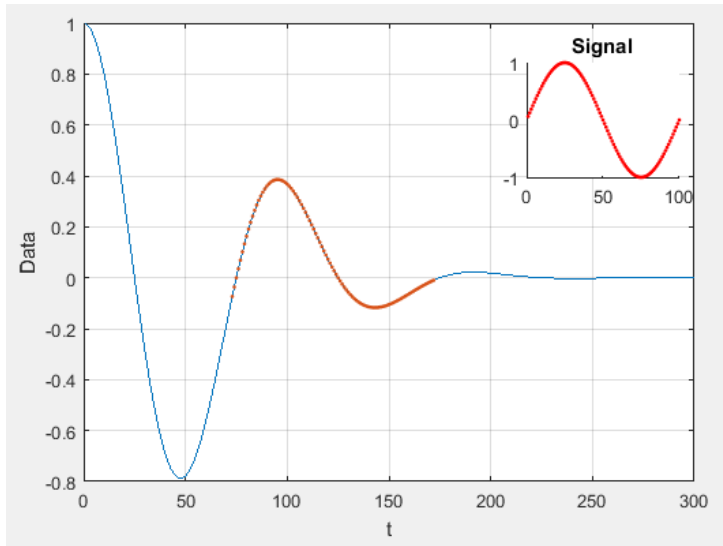
```
» [x1, x3] = alignsignals(s1, s3)
```

```
» x2 = alignsignals(s2, s3);
```

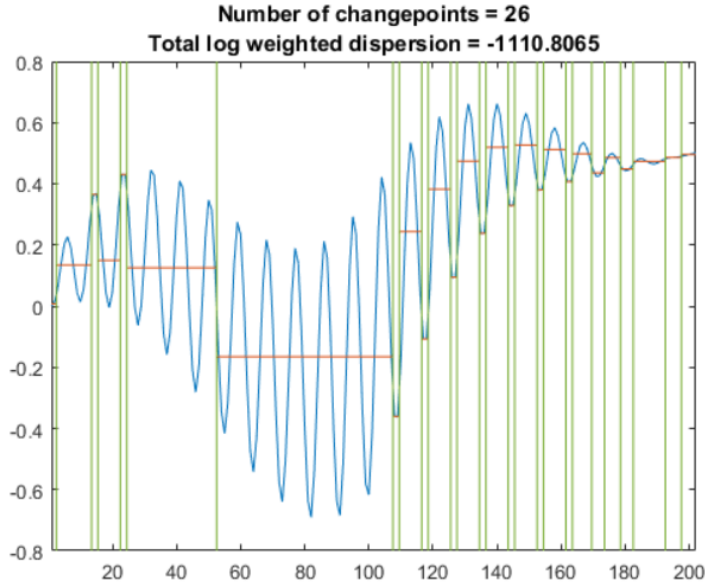
```
» dtw(s1, s2)
```

Extract Features

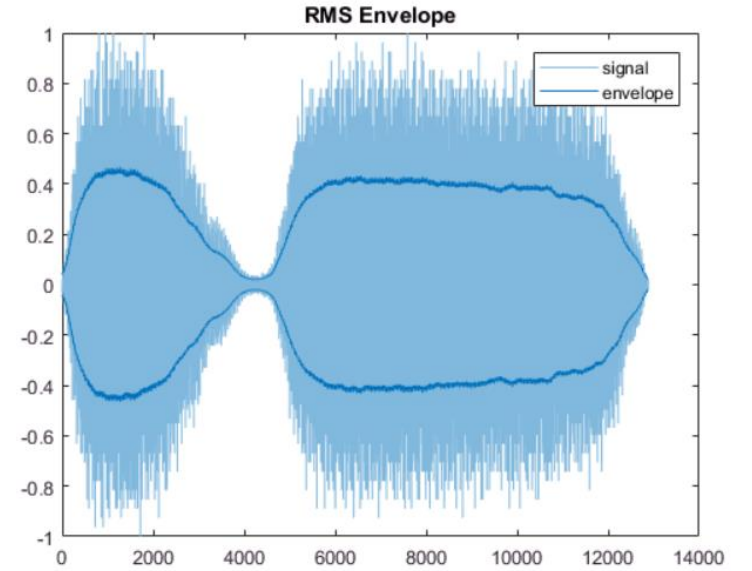

Extract Features



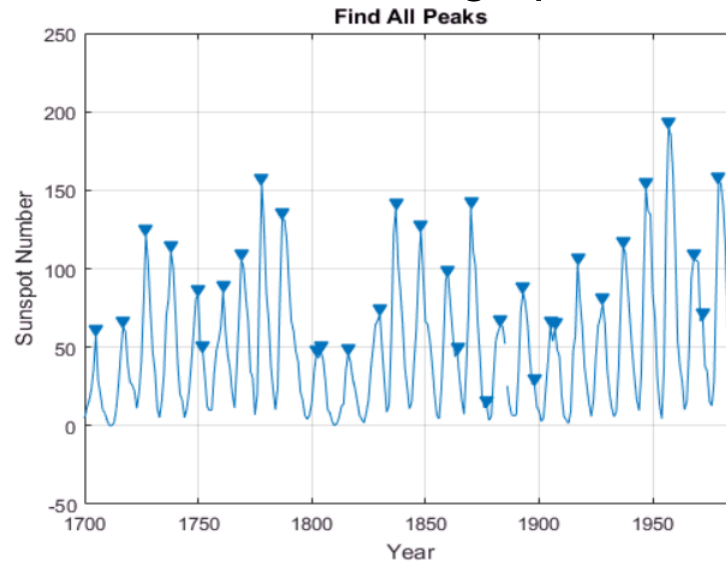
Find desired signal from patterns



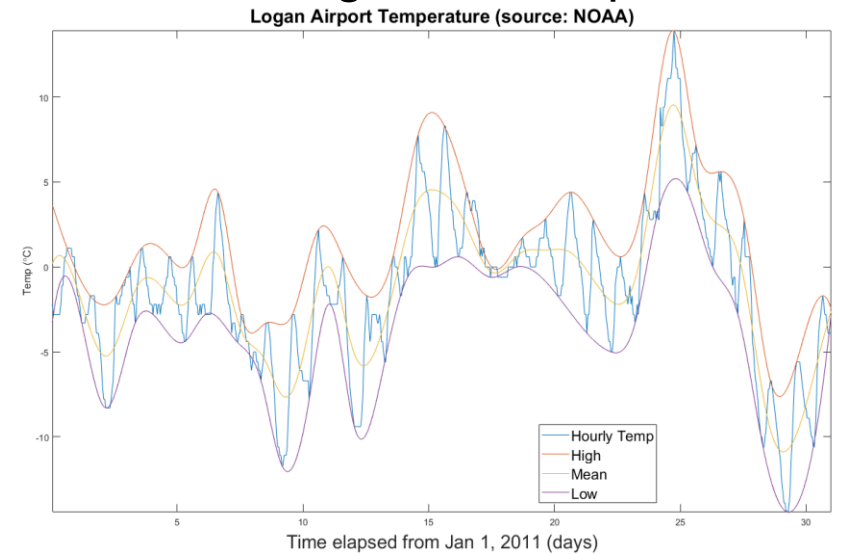
Detect change points



Find signal envelope



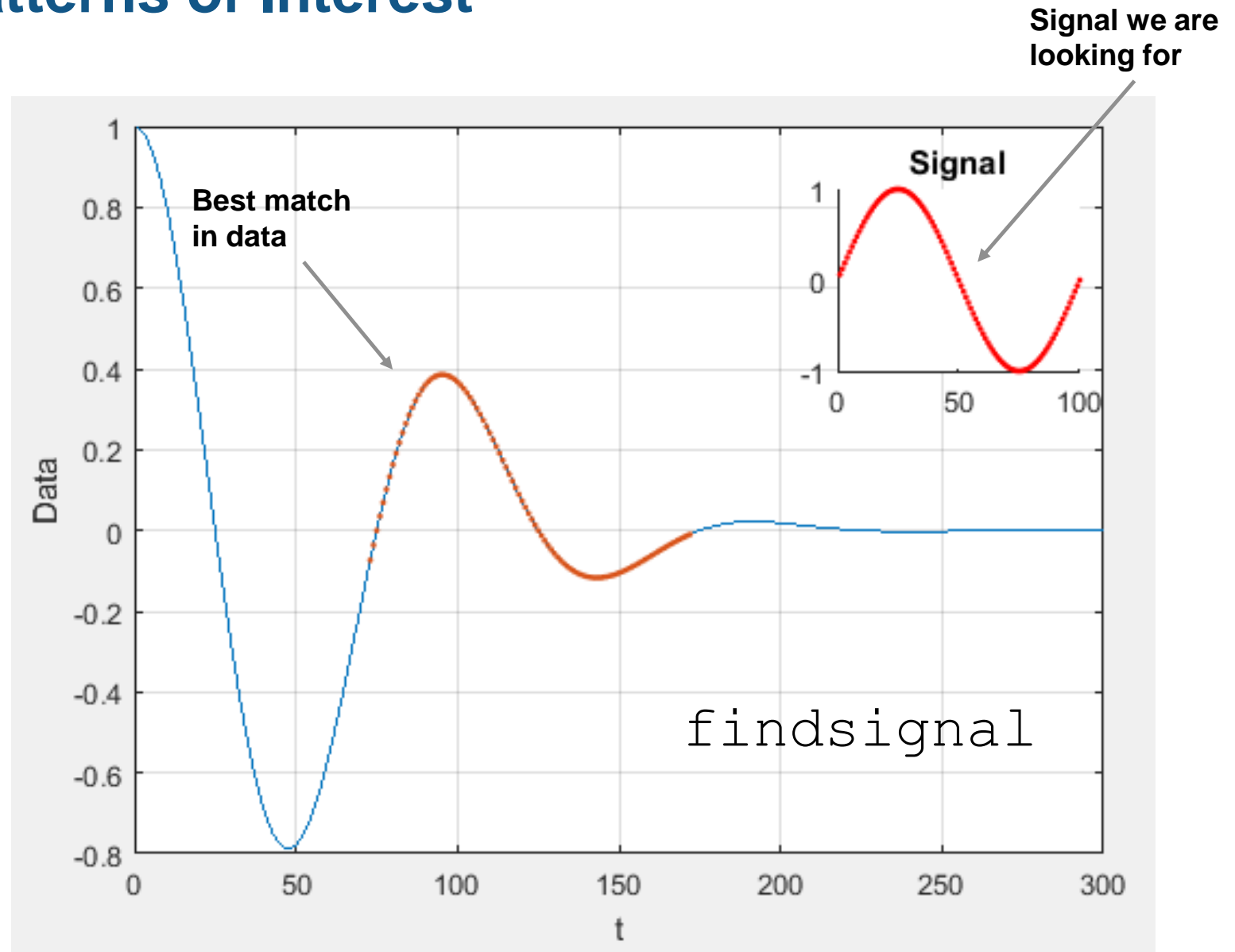
Find peaks



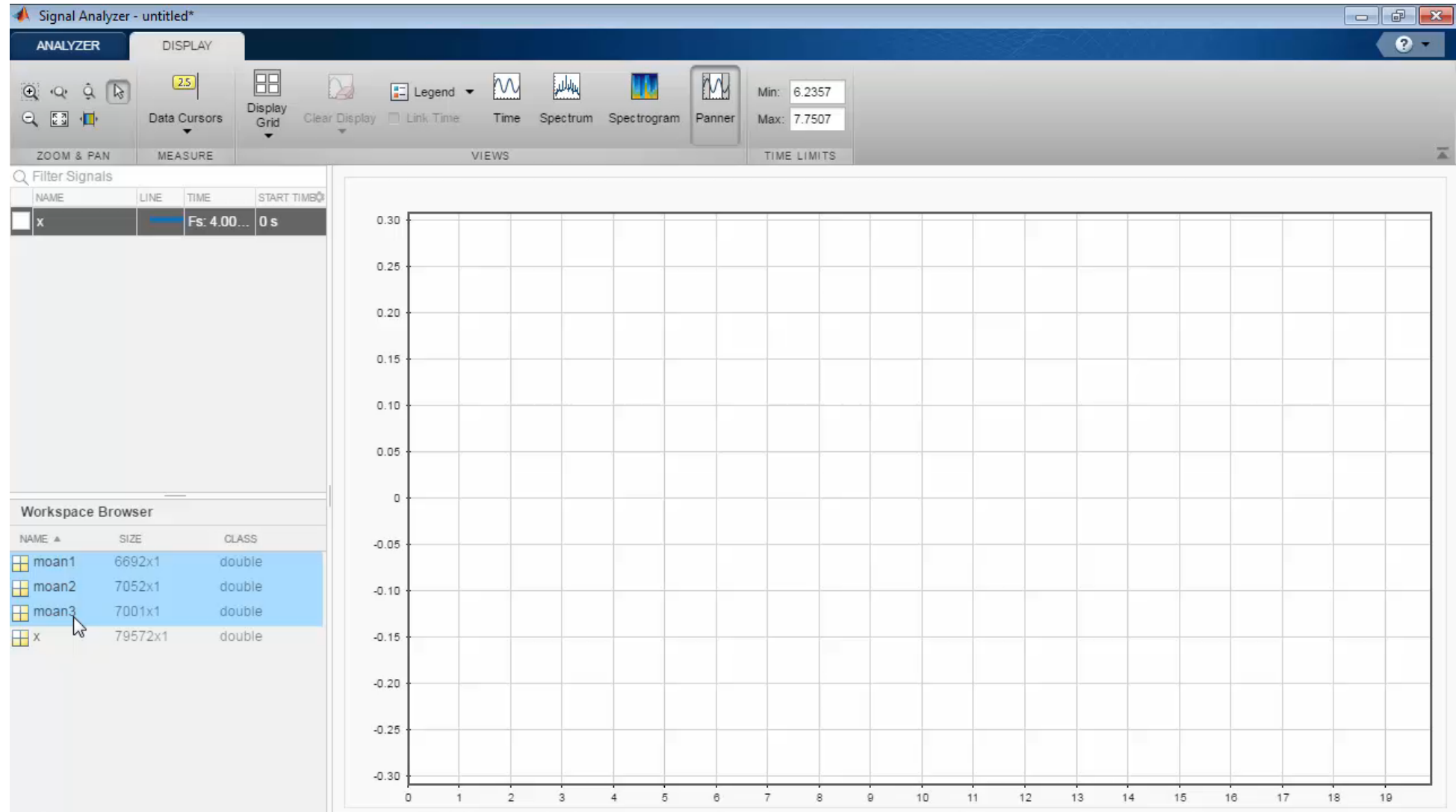
Determine signal statistics

Finding Signals and Patterns of Interest

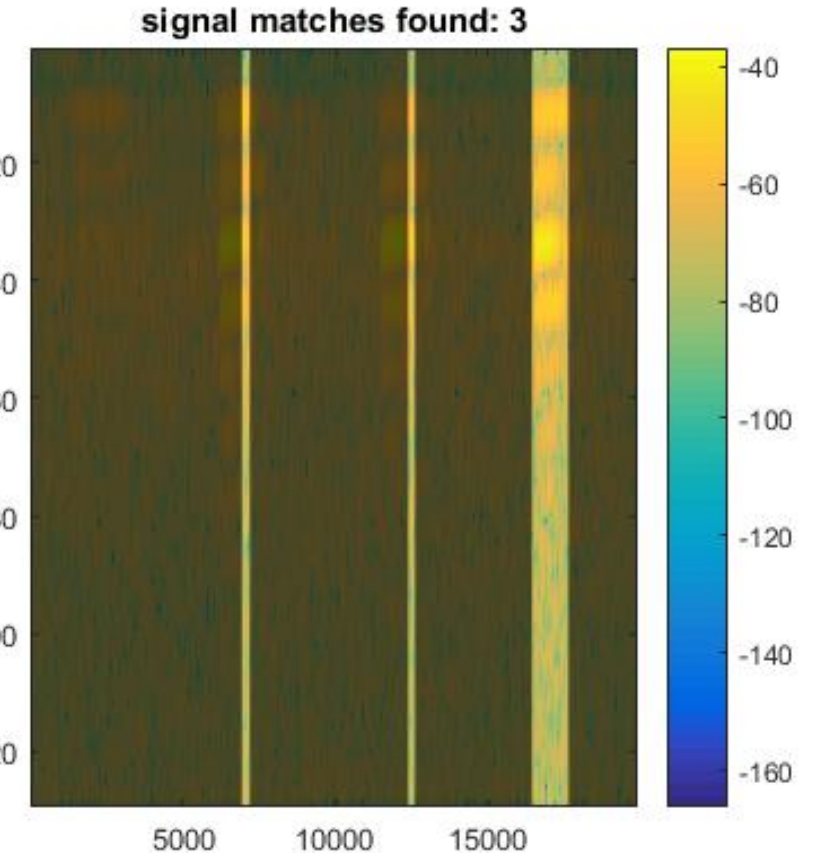
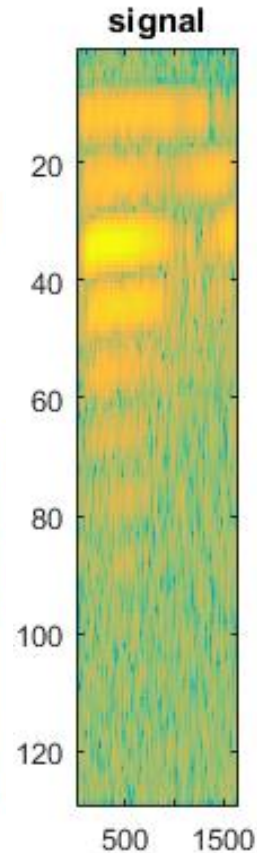
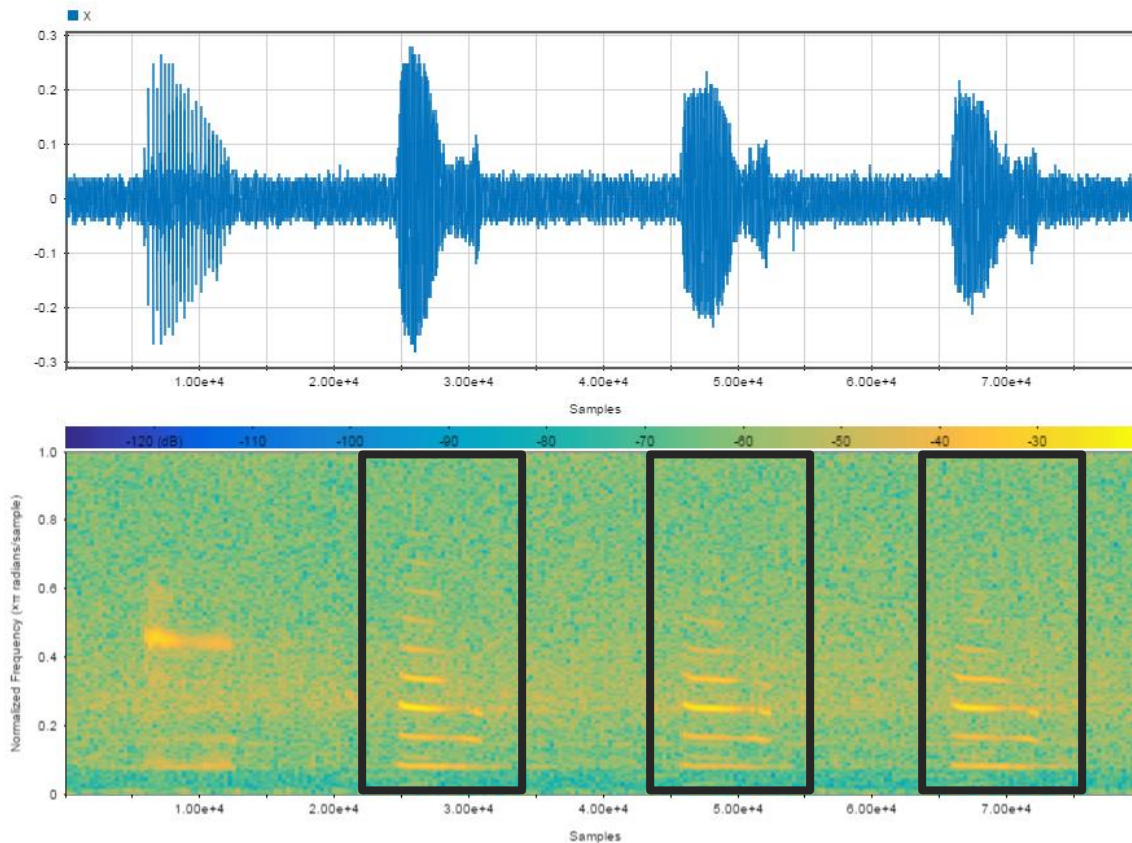
- Similarity search for finding repeat occurrences
- `findsignal` can be used with time or frequency data



Searching the Spectral Content



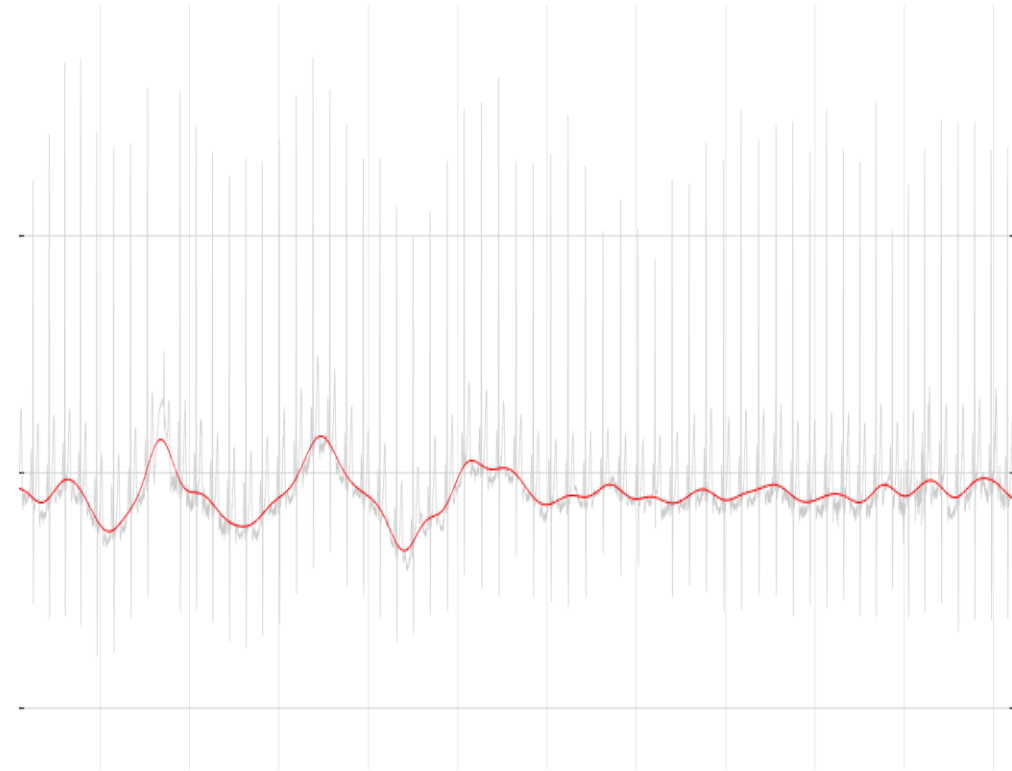
Finding a Signal of Interest



```
>>findsignal(PxxSignal,PxxMoan,'Normalization','power','TimeAlignment','dtw',...
'Metric','symmkl','MaxNumSegments',3);
```

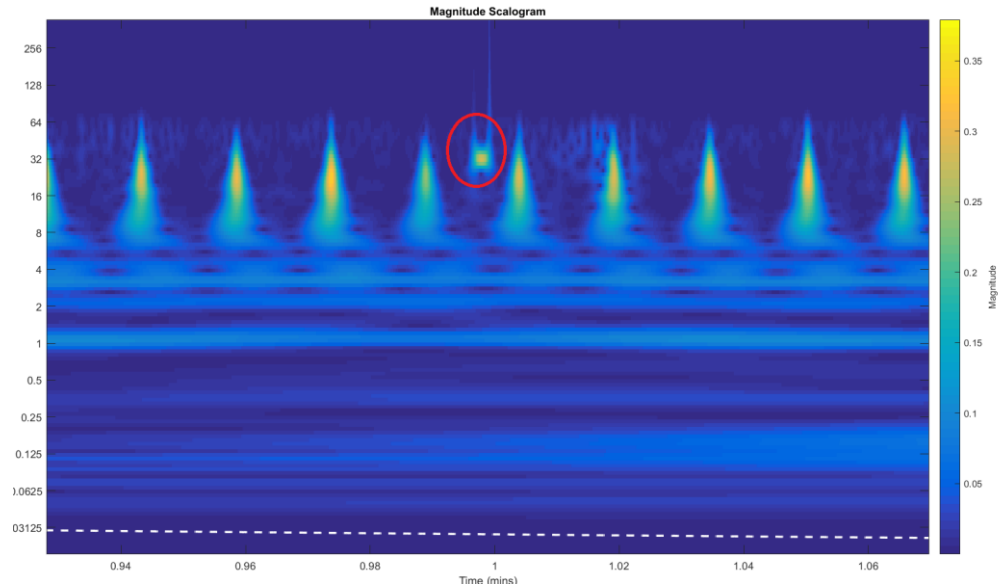
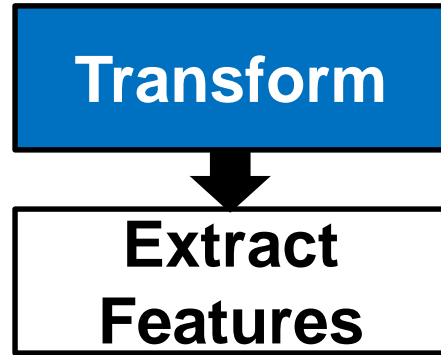

Challenges of Time-Frequency Analysis

- Fixed spectral windows can limit time-frequency resolution
- Features occurring at different scales may be missed
- Sinusoids may not be well localized in frequency



May need a different class of functions to analyze real world signals

Time-Frequency Analysis



Signal Processing Toolbox

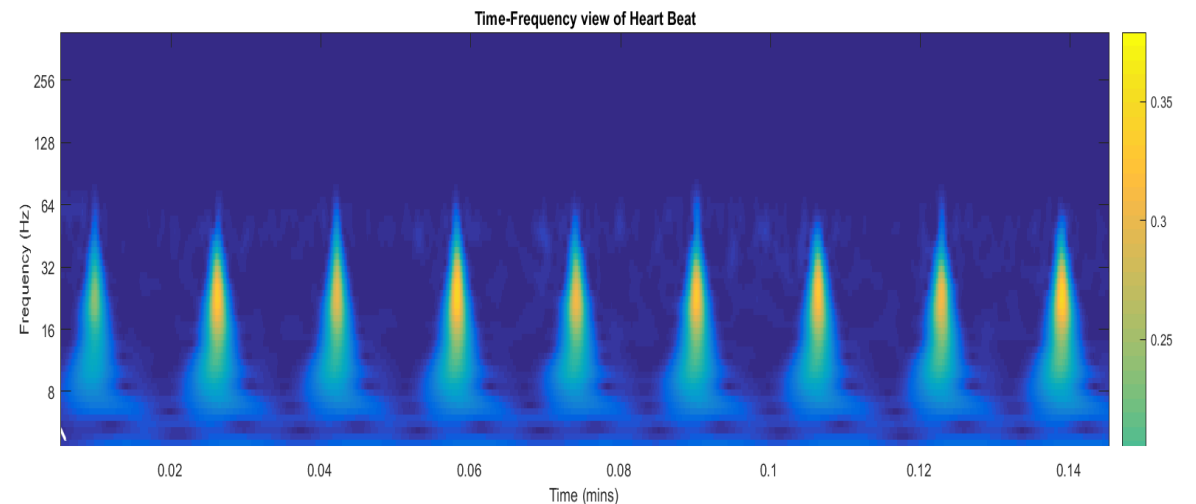
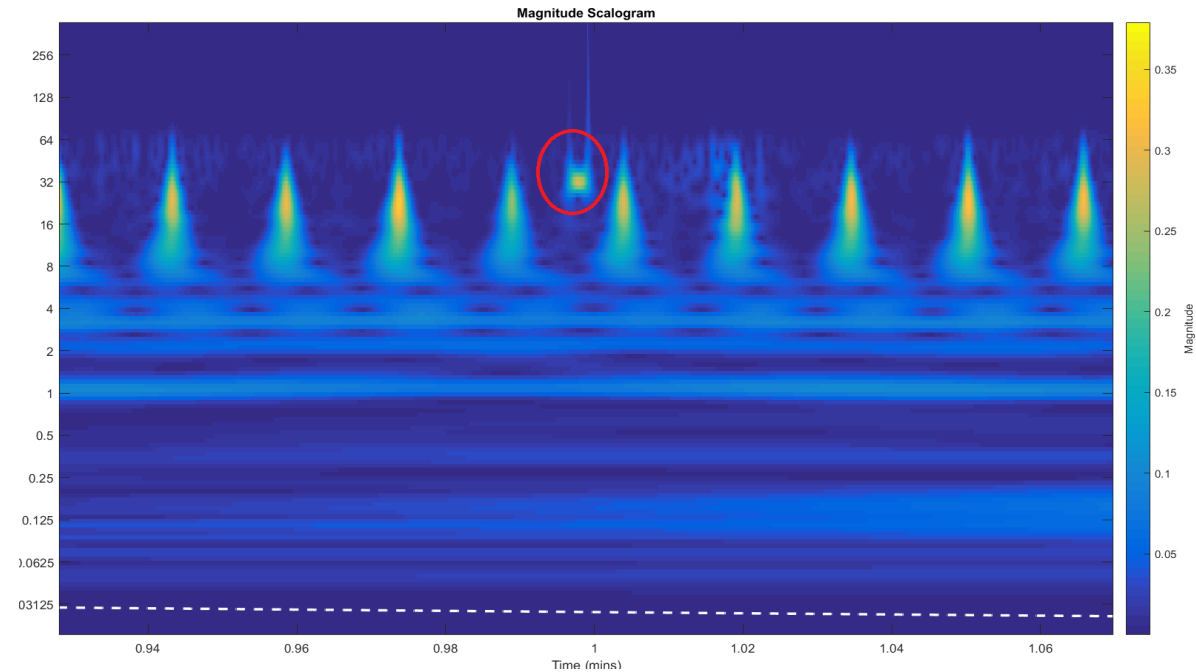
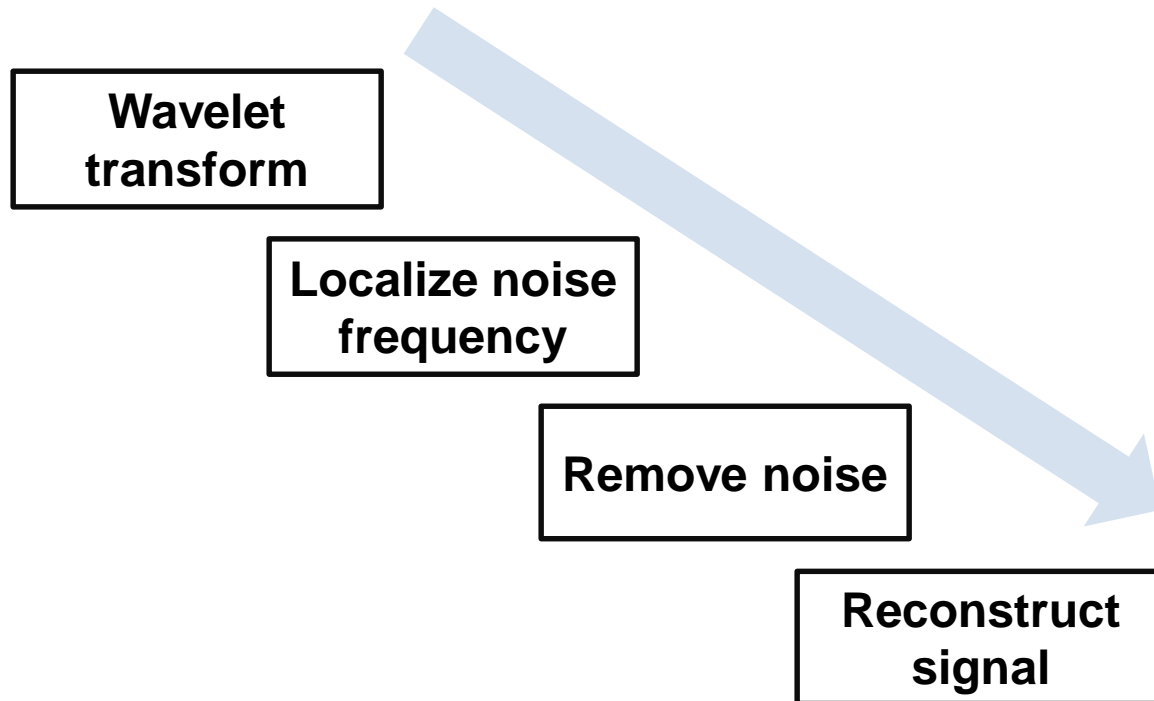
- Spectrogram
- Fourier Synchronsqueezed Transform

Wavelet Toolbox

- Continuous Wavelet Analysis
- Discrete Wavelet Analysis
- Denosing and Compression
- Filter Banks

Localizing Unwanted Frequency Components

- Wavelets used to localize & remove unwanted spectral components



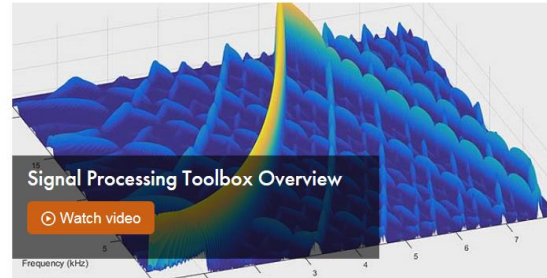
Summary

- Real world signals are challenging
 - MathWorks tools make preprocessing and feature extraction easy
- MathWorks website includes many examples to get started with
 - Data Analytics, Industrial, Automotive, Medical, Noise and Vibration, and many others
- Thank you for attending

More Resources

Perform signal processing and analysis

Signal Processing Toolbox™ provides functions and apps to generate, measure, transform, filter, and visualize signals. The toolbox includes algorithms for resampling, smoothing, and synchronizing signals, designing and analyzing filters, estimating power spectra, and measuring peaks, bandwidth, and distortion. The toolbox also includes parametric and linear predictive modeling algorithms. You can use Signal Processing Toolbox to analyze and compare signals in time, frequency, and time-frequency domains, identify patterns and trends, extract features, and develop and validate custom algorithms to gain insight into your data.

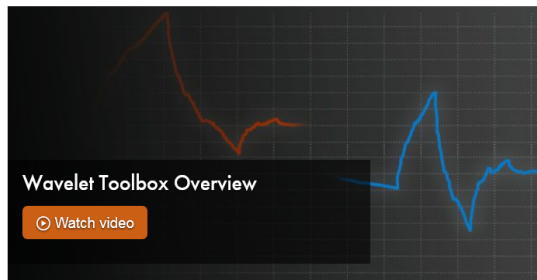


R2017a
 » Explore the latest features

Analyze and synthesize signals and images using wavelets

Wavelet Toolbox™ provides functions and apps for analyzing and synthesizing signals, images, and data that exhibit regular behavior punctuated with abrupt changes. The toolbox includes algorithms for the continuous wavelet transform (CWT), scalograms, and wavelet coherence. It also provides algorithms and visualizations for discrete wavelet analysis, including decimated, nondecimated, dual-tree, and wavelet packet transforms. In addition, you can extend the toolbox algorithms with custom wavelets.

The toolbox lets you analyze how the frequency content of signals changes over time and reveals time-varying patterns common in multiple signals. You can perform multiresolution analysis to extract fine-scale or large-scale features, identify discontinuities, and detect change points or events that are not visible in the raw data. You can also use Wavelet Toolbox to efficiently compress data while maintaining perceptual quality and to denoise signals and images while retaining features that are often smoothed out by other techniques.



R2017a
 » Explore the latest features

<https://www.mathworks.com/products/signal.html>

<https://www.mathworks.com/products/wavelet.html>

Wavelet Tech Talks

- Series of 4 short videos on wavelet concepts including MATLAB-based examples

MATLAB EXPO 2017