Timeseries II

10.11.24

Learning Objectives

- Date-Time Functions
- Moving Windows
- Fourier

Pandas Dates

- Extensive handling of date-times

- **Timestamps:** singular moments
  - Can have any spacing between timestamps

- **Periods:** a duration of time, e.g. first-quarter
  - Periods are **regularly spaced** (same time in each period)

- Choice doesn’t really affect most Pandas functions.

Pandas date initialization with date_range

- Create 10 days (“D”) starting in October 2024:

```
Oct=pd.date_range("2024-10",\
      periods=10,freq="D")
```

```
DatetimeIndex(['2024-10-01', '2024-10-02', '2024-10-03', '2024-10-04',
               '2024-10-05', '2024-10-06', '2024-10-07', '2024-10-08',
               '2024-10-09', '2024-10-10'],
              dtype='datetime64[ns]', freq='D')
```

Pandas date initialization with date_range

- Pass to create some periodic data:

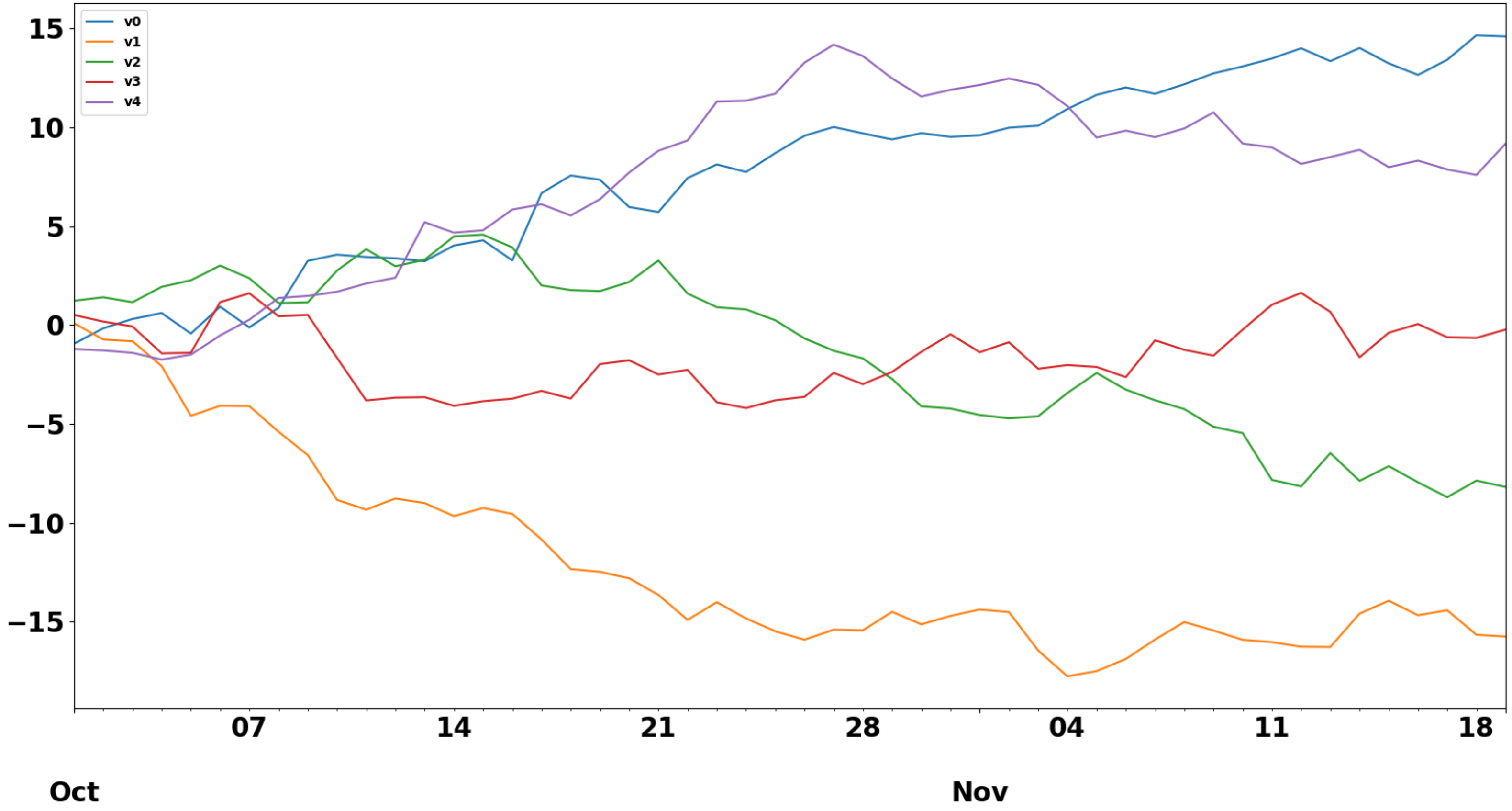
```
Oct=pd.date_range("2024-10",\
    periods=10,freq="D")
TimeData=pd.DataFrame(data,columns=cNames,index=Oct);
```

- Can now select data based upon time-ranges:

```
TimeData['2024-10-01':'2024-10-09']
```

Pandas timeseries plot

- Directly call plot method from timeseries dataframe: TimeData**.plot()**



Pandas resample

- You can downsample timeseries using aggregate rules
- e.g. give min value every 3 days:

```
dnSamp=TimeData.resample('3D').min();
```

- Other aggregators: max, mean, median etc.

Convolution: Reminder

Example 1

f: [A B C D E]

g: [1 3]

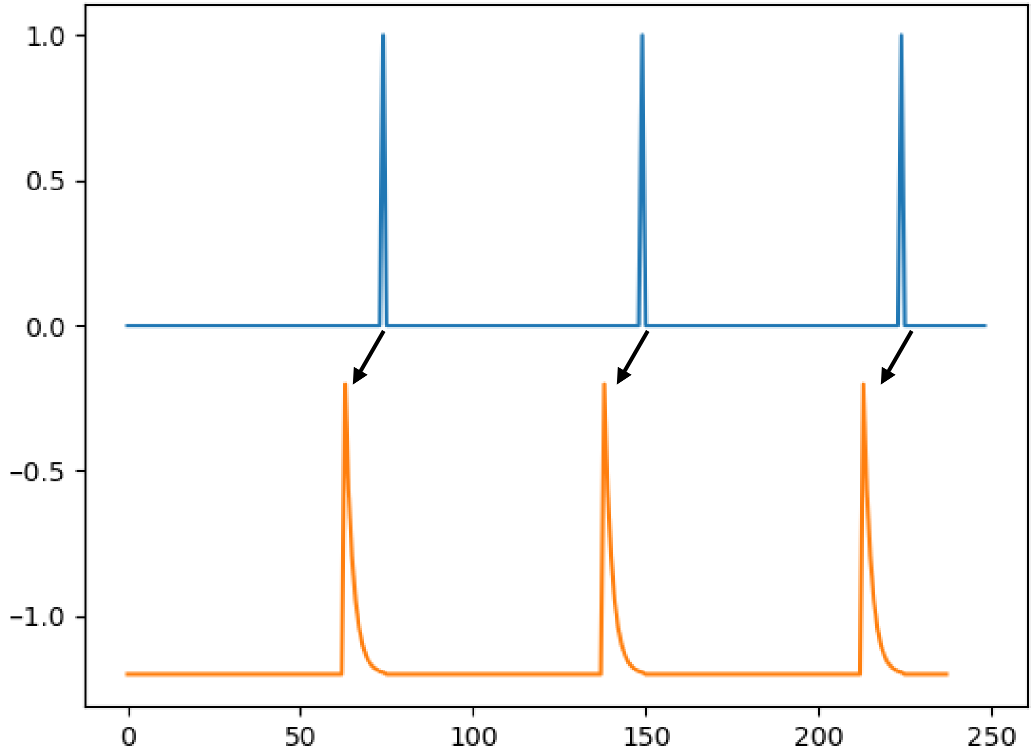
- Step 1: **flip g**
- Step 2: **moving sum**

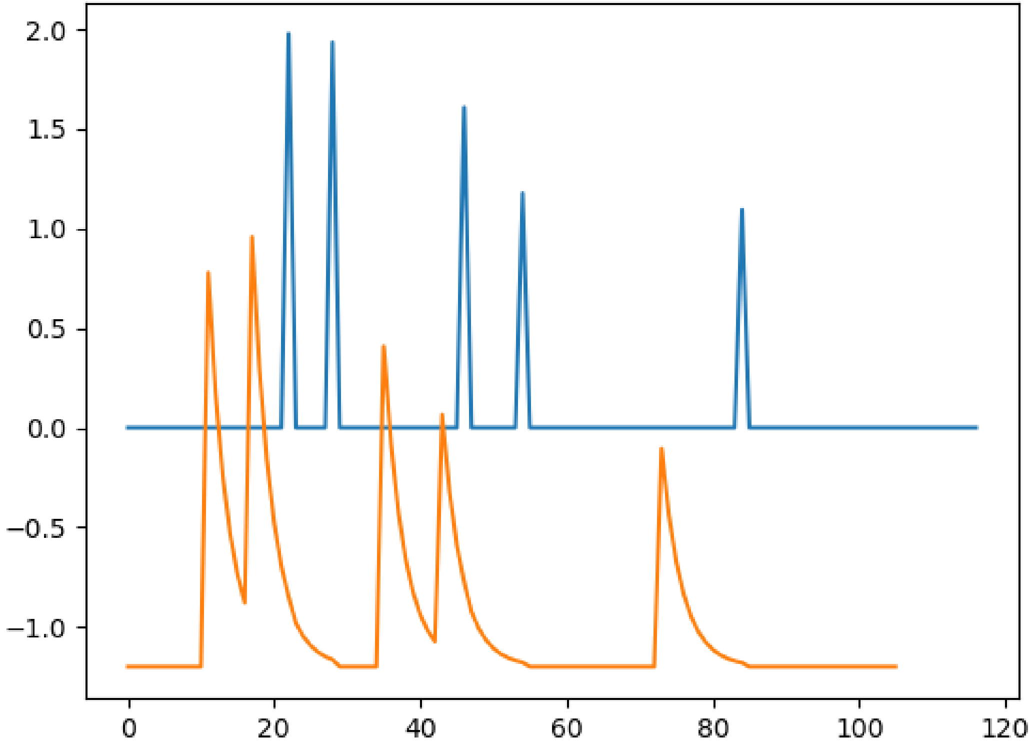
[A	B	C	D	E]
3	1			
[A	B	C	D	E]
	3	1		
[A	B	C	D	E]
		3	1	
[A	B	C	D	E]
			3	1

→

3A+B
3B+C
3C+D
3D+E

Point spread as Spike-Responses





np.convolve(data, kernel, 'valid')

Manual Convolution

- np.**convolve**(data,kern,type)

	Start			End	
type='valid':	[A B 3 1	C	D E] 3 1		
type='same':	[0 A 3 1	B	C	D E] 3 1	
type='full':	[0 A 3 1	B	C	D	E 0] 3 1

Zero-padding

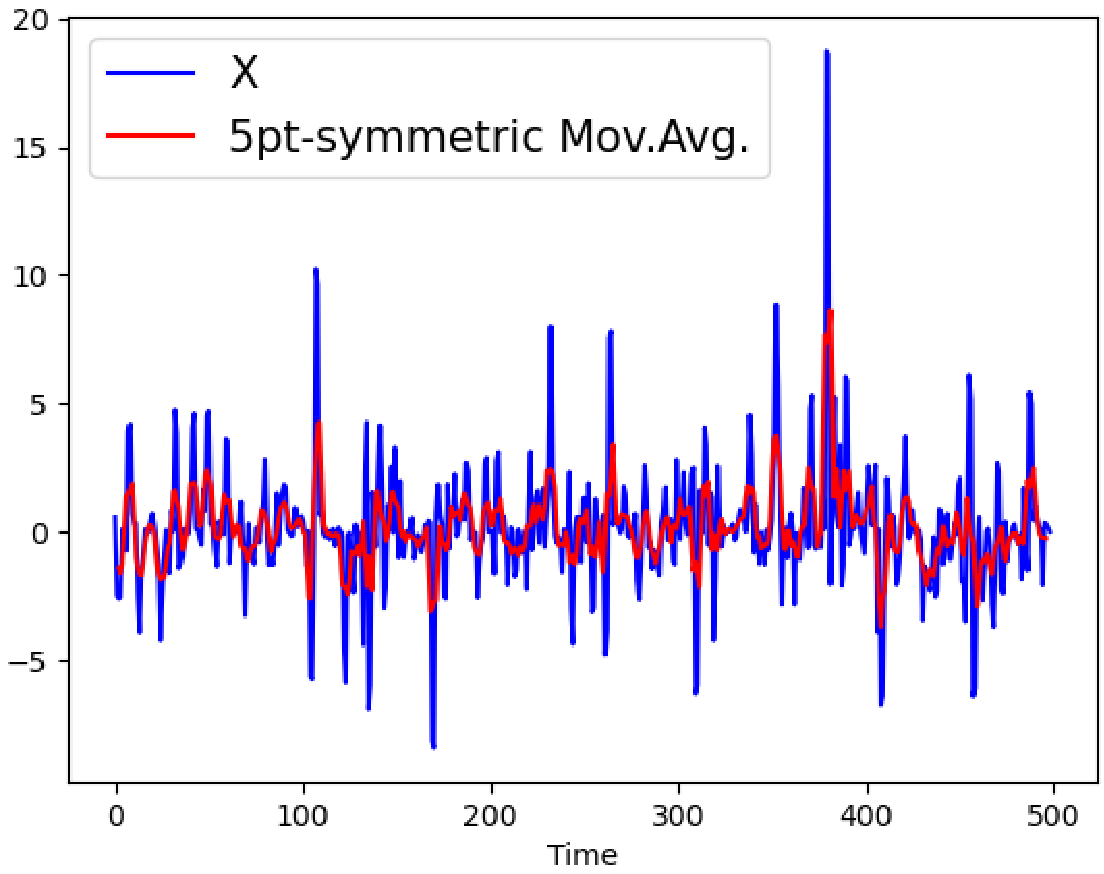
5-point moving average (on-center)

- np.**convolve**(data,kern,type)

```
plt.plot(2+np.arange(len(X)-4),\
         np.convolve(X,np.ones(5)/5\
                    ,'valid'),'r')
```

Shifted values of X to account for kernel

$$y_t = \frac{1}{5}\sum_{k=-2}^{2} x_{t+k}$$



Pandas Rolling

- Use aggregate fcts over moving-windows

- Window length: 5
- Start at position 3 (else NaN)

3 column data-frame: data

	0	1	2
0	-0.159214	3.628687	-0.565338
1	10.735521	5.898765	4.445897
2	10.843167	0.319493	-0.233595
3	5.757754	4.330009	3.834718
4	5.824845	1.074398	3.548873
5	7.182152	3.595056	6.071613
6	3.169256	5.210517	8.640147
7	5.400332	4.420175	4.475311

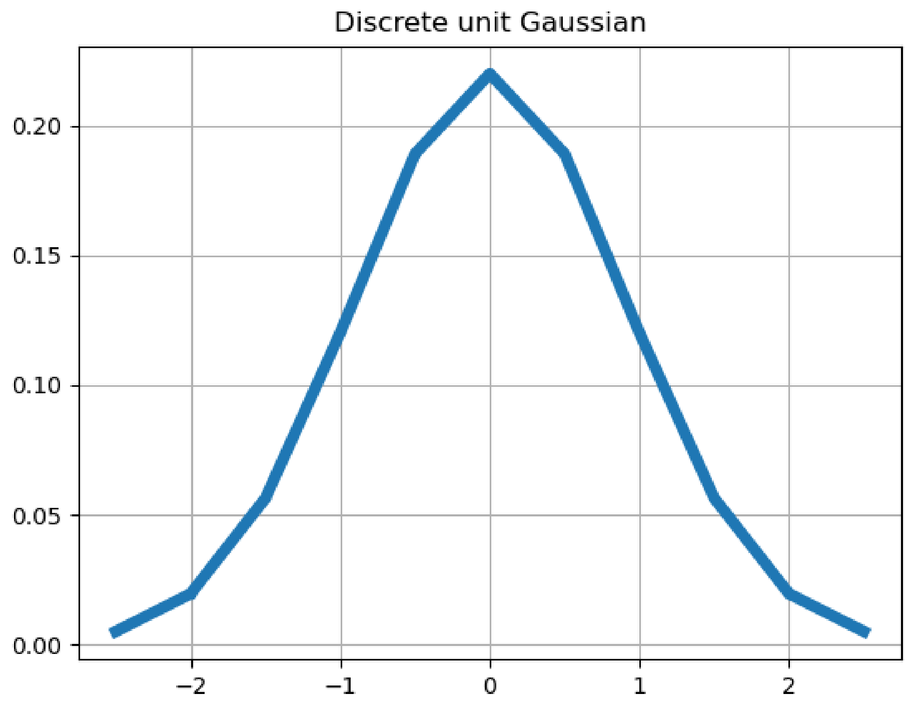
data.**rolling**(5,min_periods=3).**max**()

	0	1	2
0	NaN	NaN	NaN
1	NaN	NaN	NaN
2	10.843167	5.898765	4.445897
3	10.843167	5.898765	4.445897
4	10.843167	5.898765	4.445897
5	10.843167	5.898765	6.071613
6	10.843167	5.210517	8.640147
7	7.182152	5.210517	8.640147

Pandas Rolling: Gaussian filter

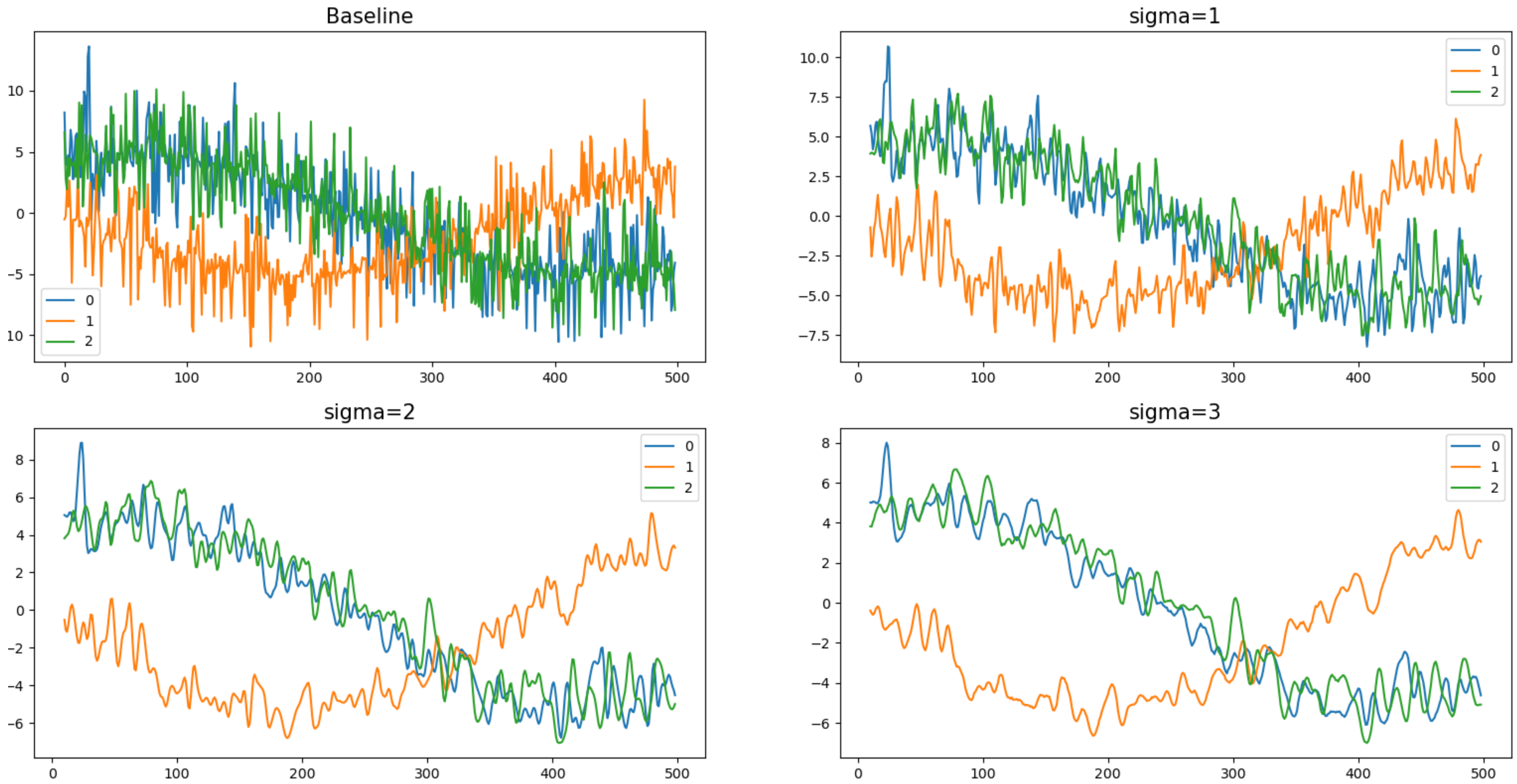
On-center Gaussian with variable SD

```
def GaussFilt(X,sigma):
    ## Setup some parameters
    var2=2*(sigma**2);
    sz=X.shape[0];
    ## Shift x-values on-center, variable length
    xVals=np.linspace(-sz/2,sz/2,sz);
    ## Discrete Gaussian
    gKern=np.exp(-(xVals**2)/var2);
    gKern=gKern/np.sum(gKern);
    ## Inner product
    return gKern.reshape([1,sz])@X,gKern,xVals
```



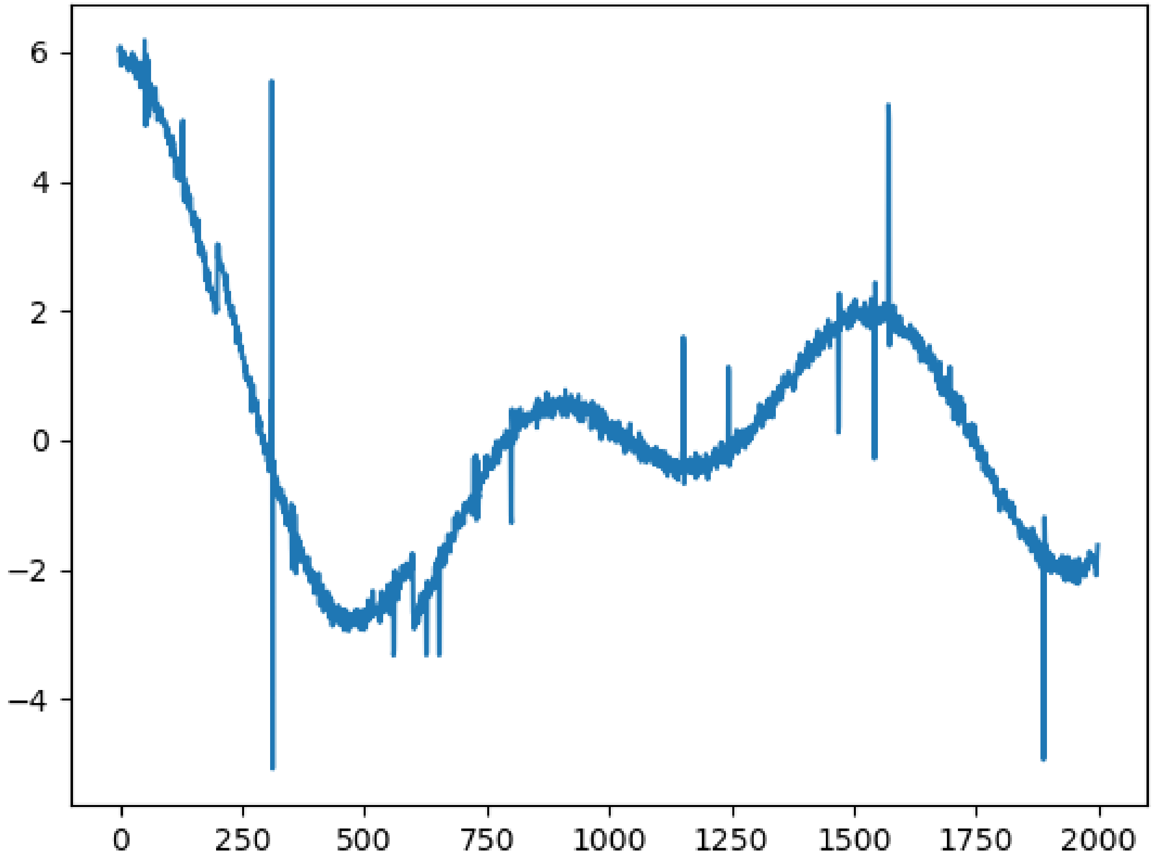
Pandas Rolling: Gaussian Filter

```
mvGauss=data.rolling(11).apply\
    (lambda x:GaussFilt(x,sigma=kk+1)[0])
```



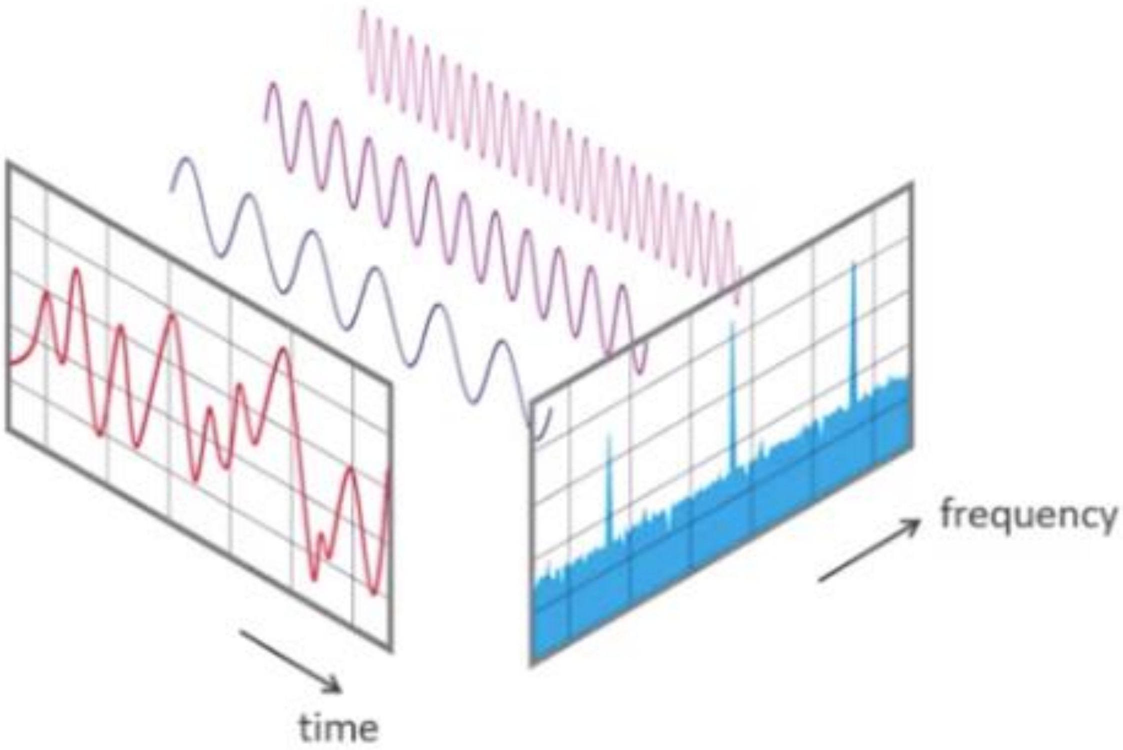
Practice Together: Median Filter

- I have a timeseries with both drift and spikes:
- I need to detrend this signal



Fourier Analysis: From time to frequency

- NumPy **fft** submodule
- Many functions in the scipy **signals** submodule

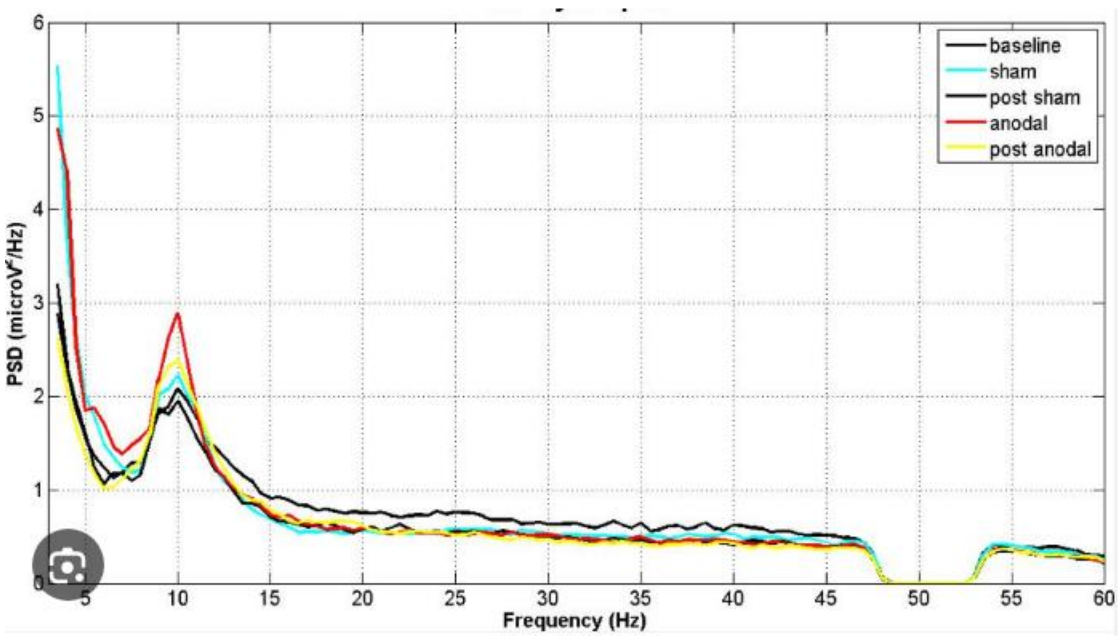


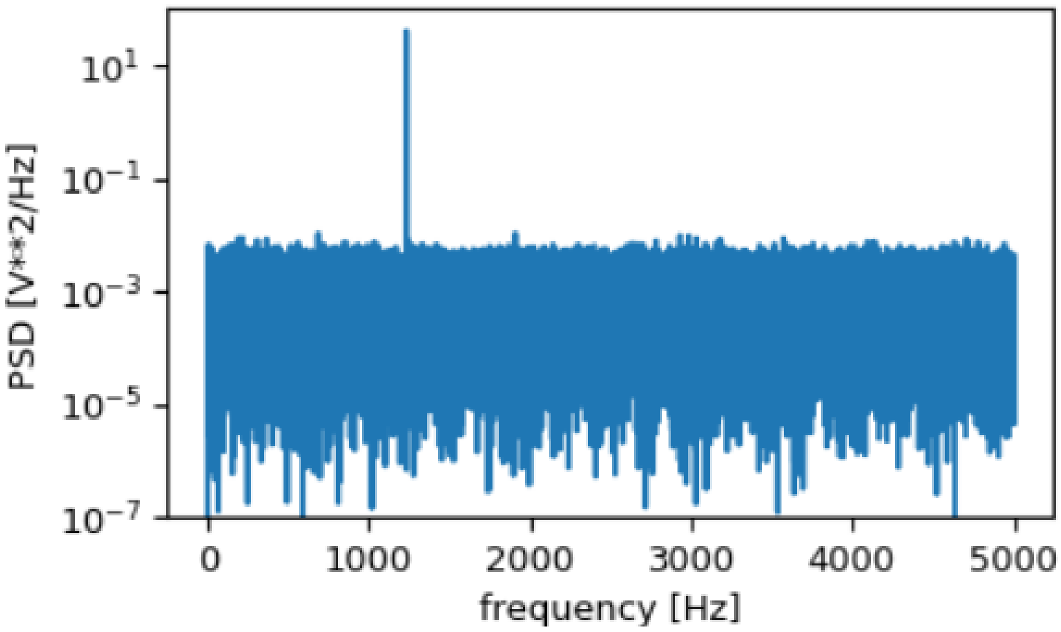
Fourier transform

$$\widehat{f}(\xi) = \int_{-\infty}^{\infty} f(x)\, e^{-i2\pi\xi x}\, dx.$$

Power-Spectral Density: **signal.periodogram**(x,Fs)

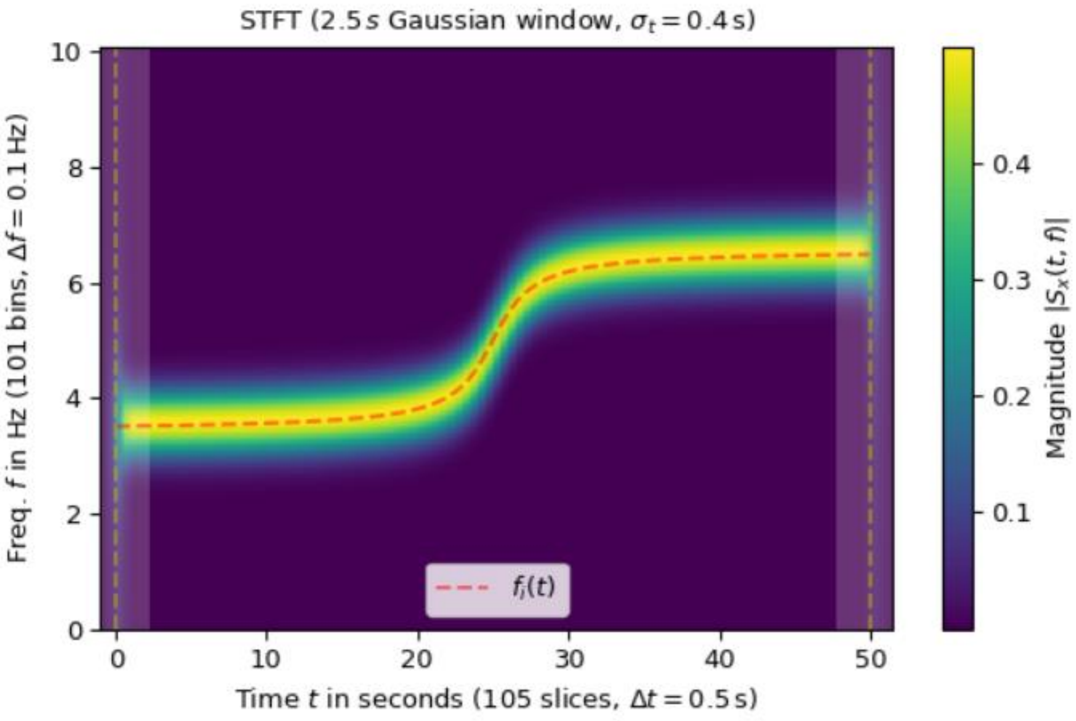
- How much is there of each signal?
- Plotted as power (squared complex-mod) of FT





Spectrograms

- Short-time frequency content
- SciPy spectrogram of an oscillator shifting from 3.5 to 6.5 Hz



Learning Objectives

- Date-Time Functions
- Moving Windows
- Fourier

Fin