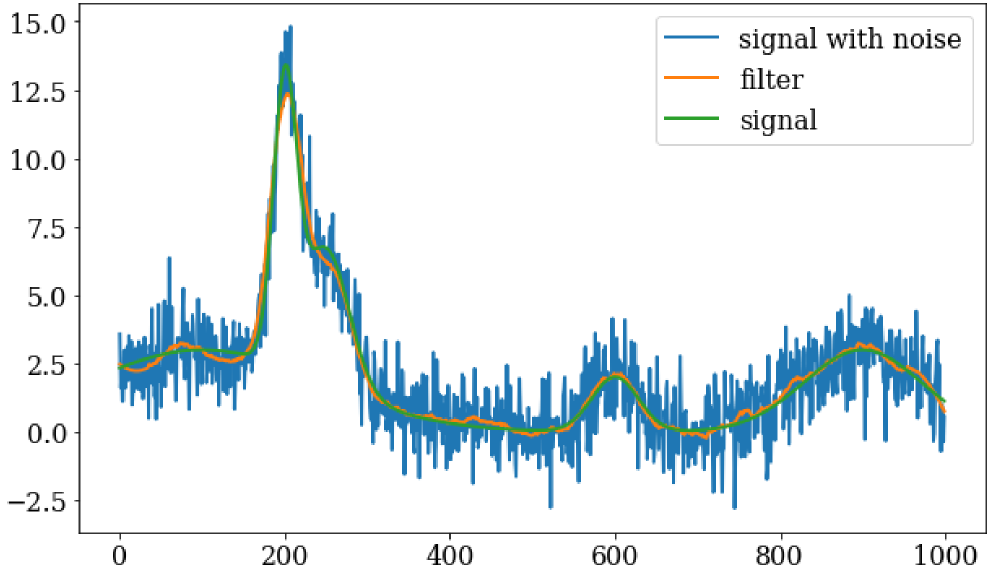
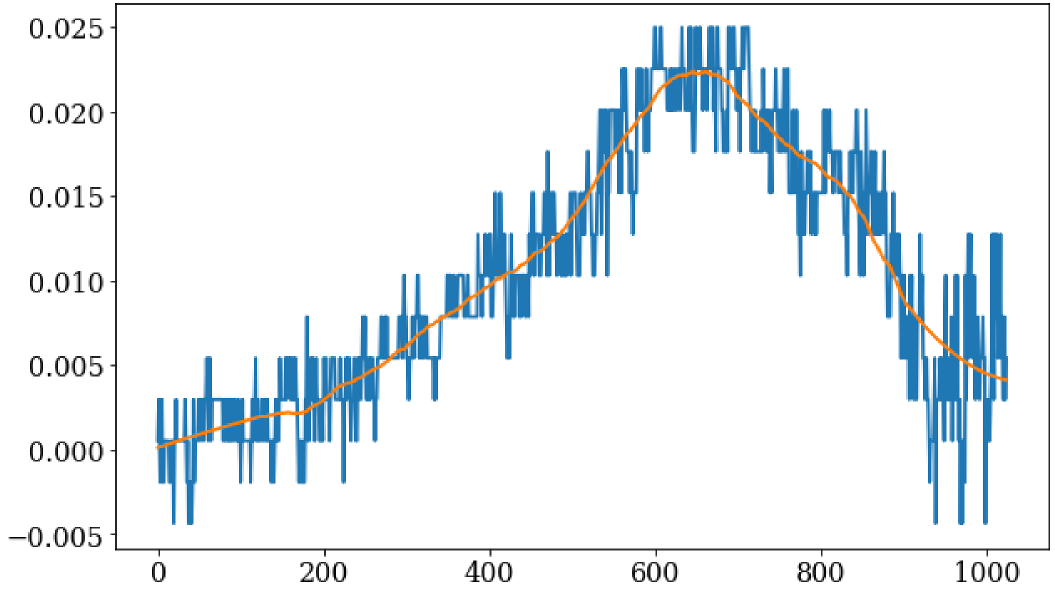
Self-optimizing Savitzky-Golay filter for generalized signal denoising





Experimental Meeting
Online (Zoom)
10/1/21

University of Rochester
Laboratory for Laser Energetics

Noise filtration can be automated through the use of Gram polynomials** used in a Savitzky-Golay algorithm*

- All filter choices in signal processing contain free parameters such as window size that require user input
  - What is the "best" choice for these free parameters?
- Polynomial filter methods like Savitzky-Golay are very effective at noise filtration but cannot be optimized if used with general nth order polynomials
- Discrete Chebyshev or Gram polynomials as basis functions for Savitzky-Golay allows them to be used for optimizing window length
  - Gram polynomials are perpendicular and are iteratively defined making them great to use in computation

Self-optimizing Savitzky-Golay filtration allows for the automation of signal denoising

*Sadeghi, et al. Arxiv 1808.10489
** P. A. Gorry, Anal. Chem. 1990, 62, 570-573

Perpendicular and differentiable polynomials allow for the solution of an optimal window length for the Savitzky-Golay algorithm

Given an input function $x(t) = f(t) + \sigma(t)$ of length L the optimal window length is

$$N_{opt} = \sqrt[2n+5]{\frac{2(n+2)\left((2n+3)!\right)^2}{\left((n+1)!\right)^2}\frac{\sigma^2}{v_n}}$$

Where n is the order of polynomial used and $v_n = \frac{1}{L}\sum_t \left(f^{(n+2)}(t)\right)^2$ is the square average of the denoised input function. Note that power on the function f here denotes the order of differentiation

(Not quite correct in terms of implementation, but this is the solution as presented in the paper)

Order 2 polynomials give the best denoising per computation time

A simple algorithm is implemented to calculate σ, v_n, and N_{opt}

The paper

$$N_{opt} = 3; N_1 = 1;$$

$$\textit{while } N_1 \neq N_{opt}$$

$$N_1 = 2\lfloor N_{opt}/2 \rfloor$$

$$y = SavGol(x, N_1, n)$$

$$dy = SavGol(diff(y, 1), N_1, n)$$

$$Y = diff(dy, 3)$$

$$c_1 = mean(Y^2)$$

$$N_{opt} = \sqrt[2n+5]{\frac{2(n+2)\left((2n+3)!\right)^2}{\left((n+1)!\right)^2}\frac{\sigma^2}{c_1}}$$

Reality

(Optional): $\sigma = x(t) - SavGol(x(t), 130) - Savgol(x(t), 30)$

$$N_{opt} = 3; N_1 = 1$$

$$\textit{while } N_1 - N_{opt} > 1$$

$$N_1 = 2\lfloor N_{opt}/2 \rfloor$$

$$y = SavGol(x, N_1, 2)$$

$$dy^{(3)} = SavGol(diff\left(SavGol\left(diff\left(SavGol(diff(y), N_1, n)\right)\right)\right))$$

$$c_1 = Mean({dy^{(3)}}^2)$$

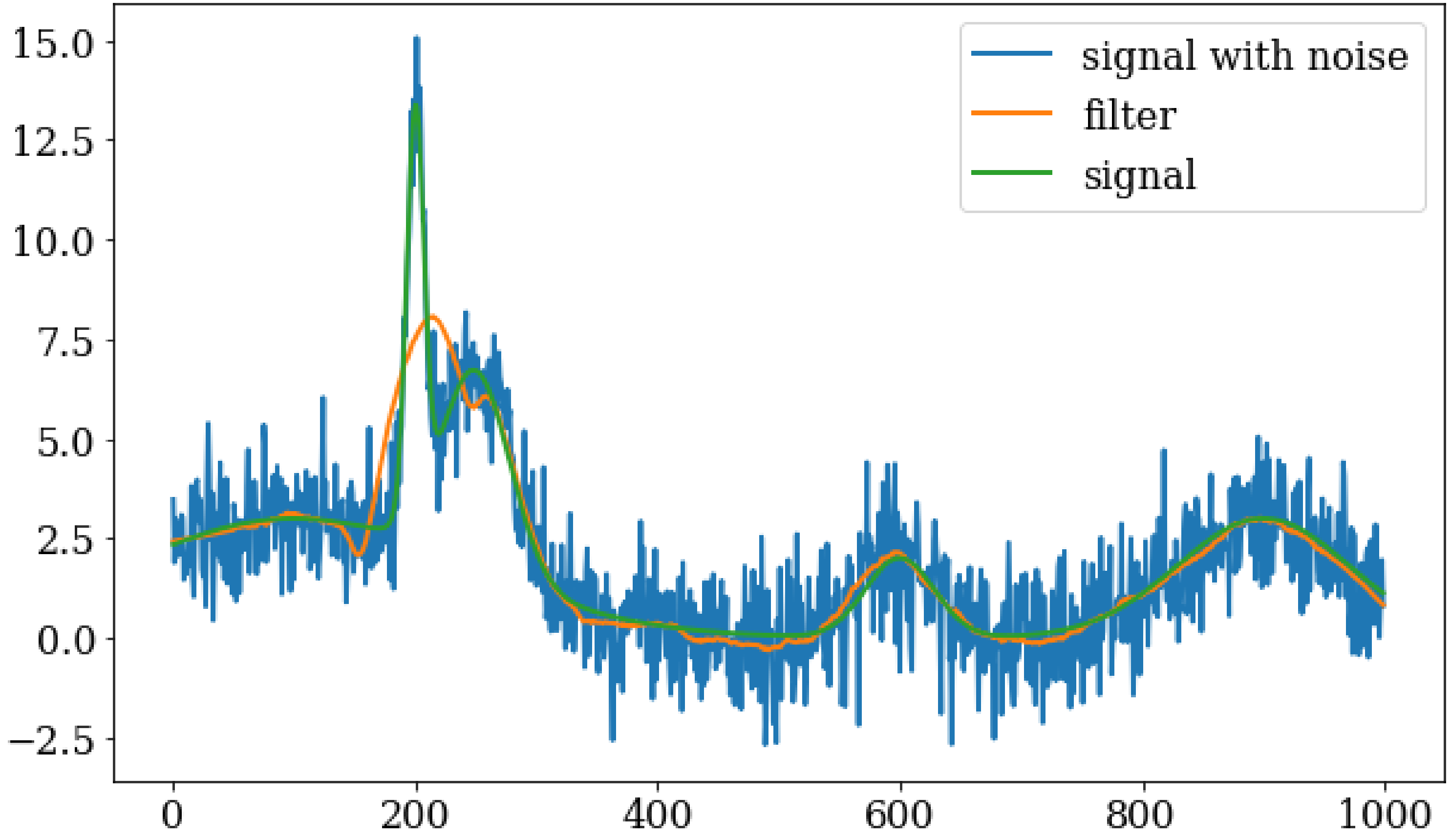
$$N_{opt} = \sqrt[2n+5]{\frac{2(n+2)\left((2n+3)!\right)^2}{\left((n+1)!\right)^2}\frac{\sigma^2}{c_1}}$$

UR LLE

UNIVERSITY of ROCHESTER

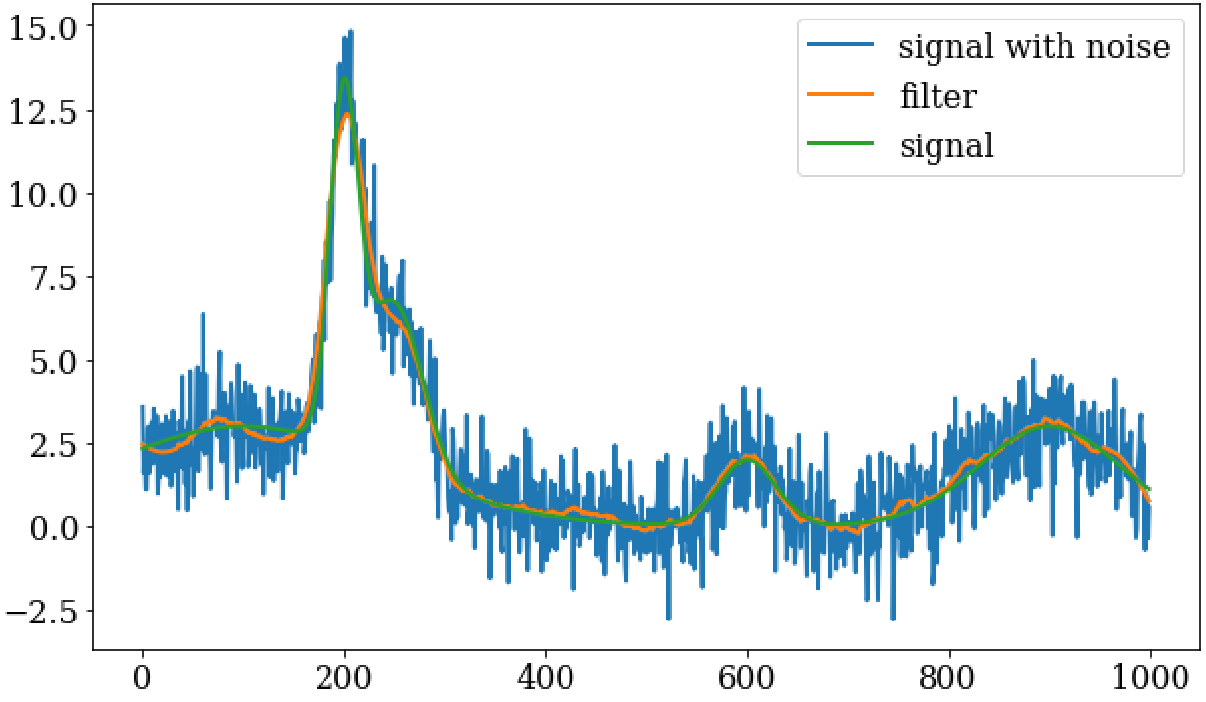
Optimal window length filtration and limitations

- Slowly varying and quickly varying signals combined do not recover the input signal
  - No real way of dealing with this, but it is a minute case

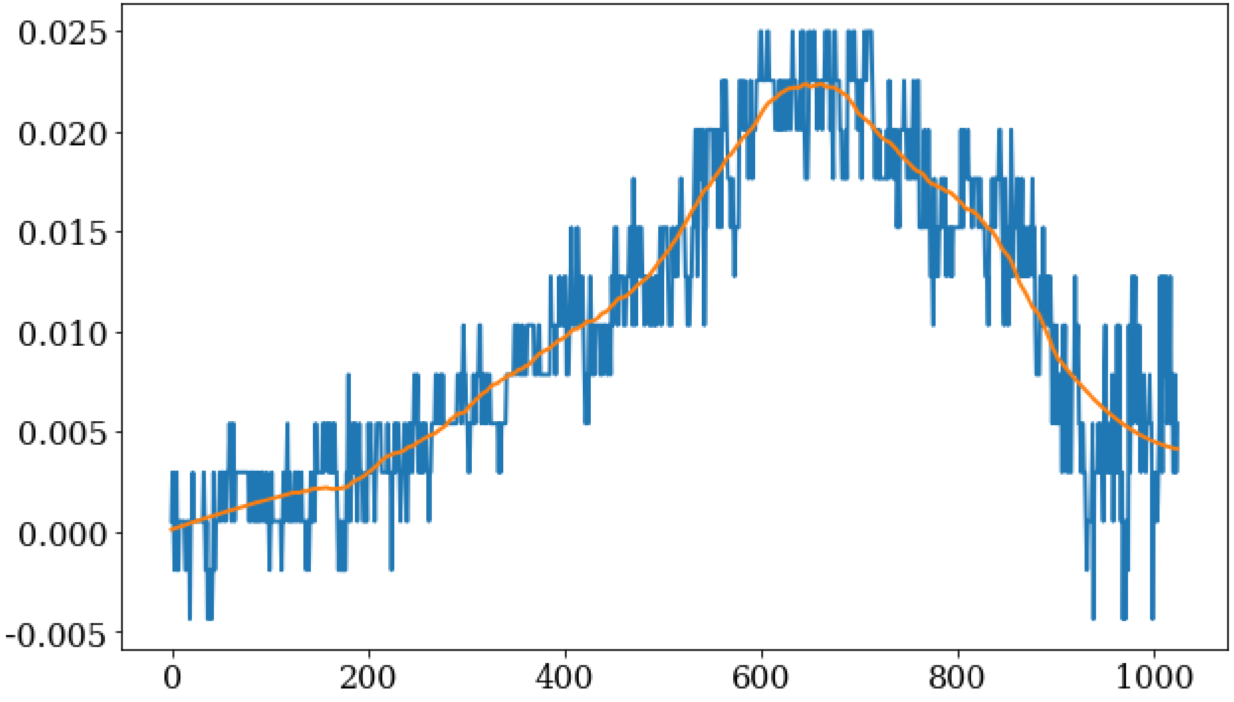


But it works well otherwise!

Almost identical to last signal with no quickly varying features



Actual oscilloscope trace. No really.



How can I use this marvelous piece of code?!

- **Import the sgfilter module and use it like so:**

```python
# -*- coding: utf-8 -*-
"""
Created on Mon Sep 27 22:14:53 2021

@author: Daniel Barnak
"""
import sgfilter

#%% nOpt test
signal = "user input"

nOpt = sgfilter.n_opt(signal[0:300])
filterOpt = sgfilter.sg_filter_gram(signal, nOpt, 2)

plt.figure(10)
plt.plot(signal, label = 'signal with noise')
plt.plot(filterOpt, label = 'filter')
```

Both the optimized SG filter module and the example workspace are available now at: https://github.com/Plasmacaster/Optimized_SavGol

Noise filtration can be automated through the use of Gram polynomials** used in a Savitzky-Golay algorithm*

- All filter choices in signal processing contain free parameters such as window size that require user input
  - What is the “best” choice for these free parameters?
- Polynomial filter methods like Savitzky-Golay are very effective at noise filtration but cannot be optimized if used with general nth order polynomials
- Discrete Chebyshev or Gram polynomials as basis functions for Savitzky-Golay allows them to be used for optimizing window length
  - Gram polynomials are perpendicular and are iteratively defined making them great to use in computation

Self-optimizing Savitzky-Golay filtration allows for the automation of signal denoising

*Sadeghi, et al. Arxiv 1808.10489
** P. A. Gorry, Anal. Chem. 1990, 62, 570-573