Thursday Warm Up, Unit 1: Filter Design

Prof. Jordan C. Hanson

March 27, 2025

1 The Moving Average Filer

1. Implement the following code in `octave`, to create a gaussian pulse with a width, σ, and frequency, f:

```
fs = 40e3;
dt =1/fs;
T = 1.0;
t = dt:dt:T;
f = 440.0;
sigma = 0.5;
t0 = 0.5;
T = 2.0;
A = 4.0;
x = A*cos(2*pi*f*t).*exp(-0.5*(t-t0).^2/sigma^2);
```

   Create a player to play this data as audio:

```
player1 = audioplayer(x,fs,16);
play(player1)
```

2. Add random gaussian noise, to create a noisy signal with an SNR of 5. Create a new player to play it as audio:

```
y = x+rand(size(x))*0.2;
player2 = audioplayer(y,fs,16);
play(player2)
```

   How does the audio sound, compared to the noiseless version?

3. The **moving average filter** is the *optimal solution* for reducing random gaussian noise, while preserving step response. Implement the moving average filter by defining the kernel like a normalized step:

```
kern = ones(1,400)/400;
z = conv(kern,y);
player3 = audioplayer(z,fs,16);
play(player3)
```

4. Play `player1` and `player3` repeatedly. Do they sound similar? If `player3` sounds quieter, try amplifying it by increasing the signal amplitude. That is, multiply the samples by a constant, and feed it back to the audioplayer algorithm. At a certain point, do you notice *clipping*, or distortion from overamplifying?

2 Graphing the Results

1. **Graph** the noiseless version of the gaussian pulse, with appropriate units, below:

2. **Graph** the noisy version of the gaussian pulse, with appropriate units, below:

3. **Graph** the filtered version of the gaussian pulse, with appropriate units, below: