Motion Analysis Methods:

Sampling, Fourier Analysis and Filtering

Engr325 **Lab #7** **Winter, 2019**

Video Analysis and Filtering

Name ________________________________

Partner (s)_______________________________________ **Grade** _______/10

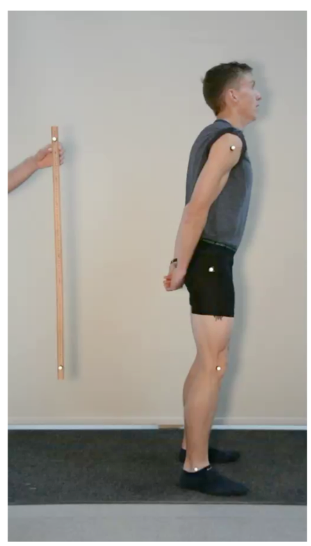
Introduction
The goal of this lab is to obtain a position-time dataset from video analysis and create a Butterworth filter with appropriate smoothing to obtain useful velocity and acceleration graphs.

Objectives
- Understand the basics of video digitization;
- Obtain useful 2D data from a video clip;
- Understand how digital filtering can increase the quality of results.

Equipment Provided
- Computer with appropriate software.

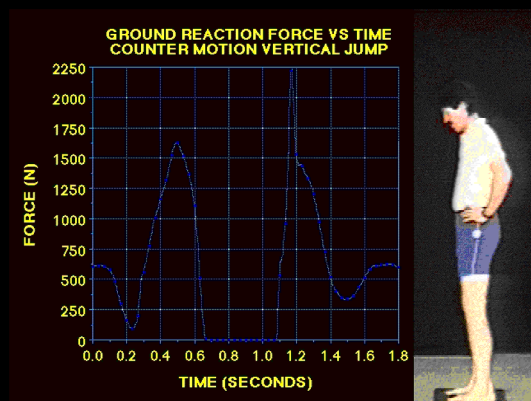
References
- Video digitization and digital filtering textbooks and web sites.

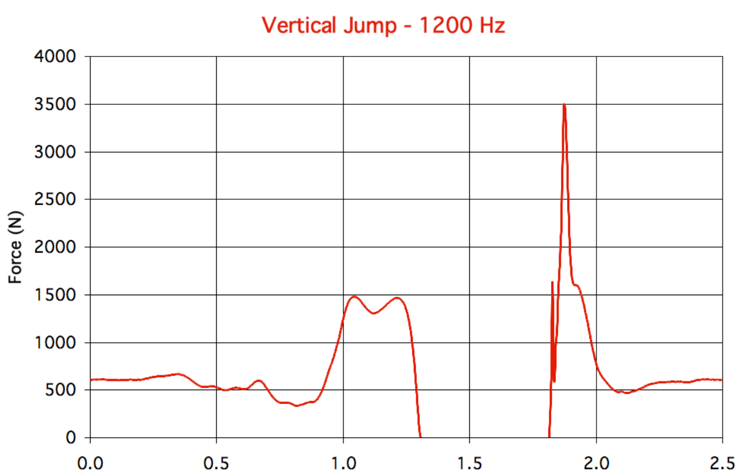
Procedure
1) On the class website under the heading of Lab #7, you will find a video file showing a side view of a human vertical jump with markers indicating various joint centers, including at the hip. Two formats are provided, .mp4 and .mov Download the .mp4 video file and save it to your lab computer (sometimes the .mov format works better, but see how your results are with the.mp4 file).

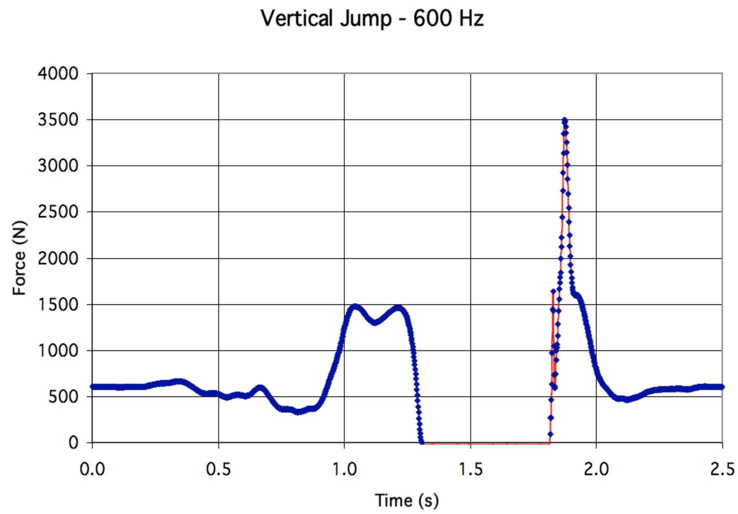


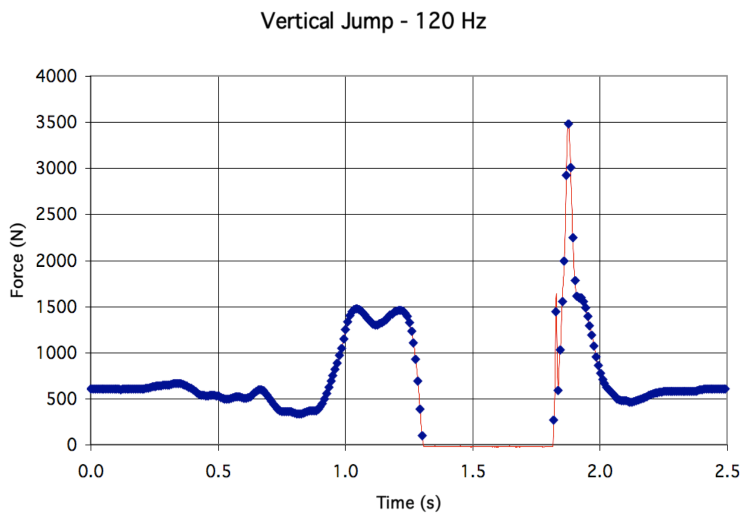
Motion Analysis Methods:

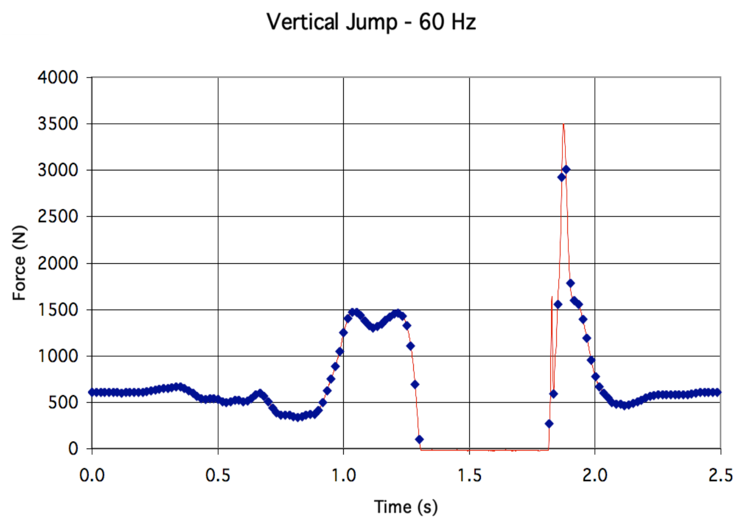
Sampling, Fourier Analysis and Filtering






Vertical Jump - 600 Hz
Force (N)
Time (s)
4000
3500
3000
2500
2000
1500
1000
500
0
0.0
0.5
1.0
1.5
2.0
2.5

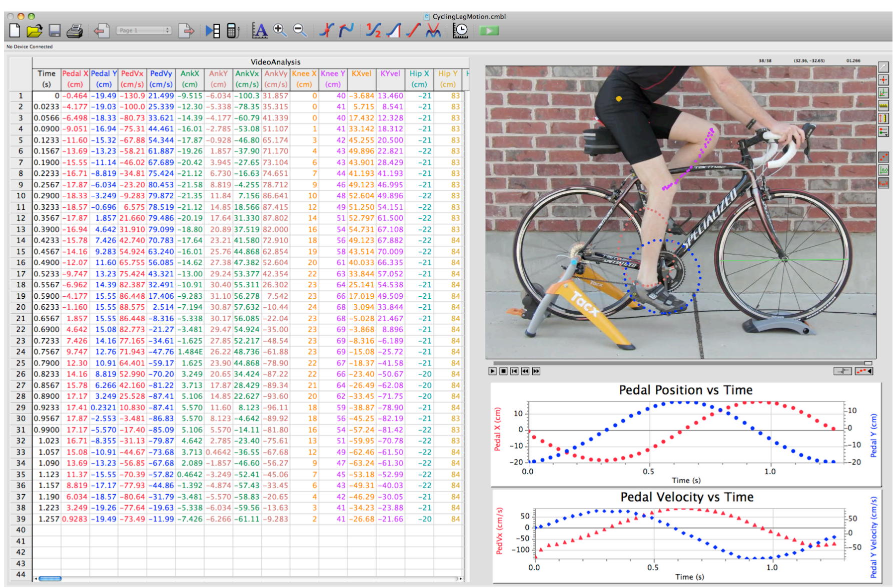

Vertical Jump - 120 Hz
Force (N)
Time (s)
4000
3500
3000
2500
2000
1500
1000
500
0
0.0
0.5
1.0
1.5
2.0
2.5



Proper Sampling:

"If you can exactly *reconstruct*
the analog signal from the samples,
you must have done the sampling properly."

(from *Digital Signal Processing* by Steven W. Smith, p. 39)



VideoAnalysis

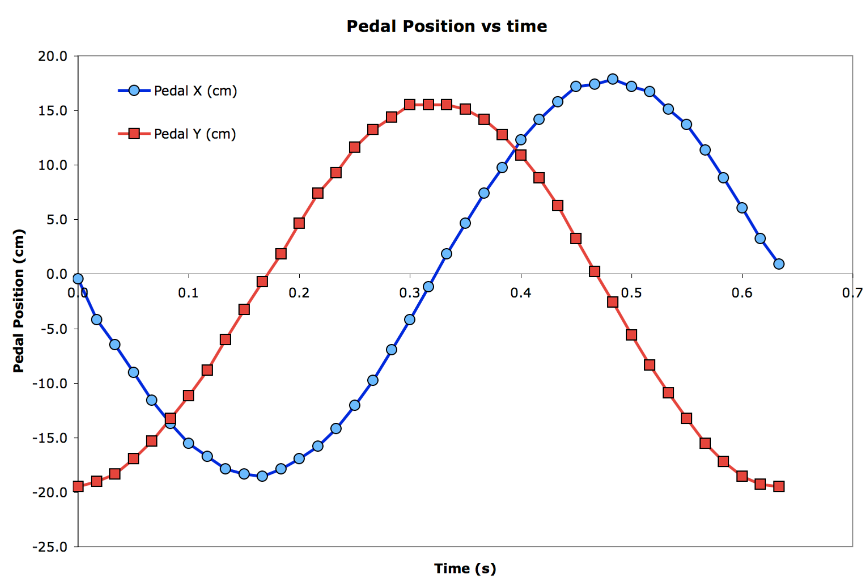
	Time (s)	Pedal X (cm)	Pedal Y (cm)	PedVx (cm/s)	PedVy (cm/s)	AnkX (cm)	AnkY (cm)	AnkVx (cm/s)	AnkVy (cm/s)	Knee X (cm)	Knee Y (cm)	KXvel	KYvel	Hip X (cm)	Hip Y (cm)
1	0	−0.464	−19.49	−130.9	21.499	−9.515	−6.034	−100.3	31.857	0	40	−3.684	13.460	−21	83
2	0.0233	−4.177	−19.03	−100.0	25.339	−12.30	−5.338	−78.35	35.315	0	41	5.715	8.541	−21	83
3	0.0566	−6.498	−18.33	−80.73	33.621	−14.39	−4.177	−60.79	41.339	0	40	17.432	12.328	−21	83
4	0.0900	−9.051	−16.94	−75.31	44.461	−16.01	−2.785	−53.08	51.107	1	41	33.142	18.312	−21	83
5	0.1233	−11.60	−15.32	−67.88	54.344	−17.87	−0.928	−46.80	65.174	3	42	45.255	20.500	−21	83
6	0.1567	−13.69	−13.23	−58.21	61.887	−19.26	1.857	−37.90	71.170	4	43	49.896	22.821	−22	83
7	0.1900	−15.55	−11.14	−46.02	67.689	−20.42	3.945	−27.65	73.104	6	43	43.901	28.429	−21	83
8	0.2233	−16.71	−8.819	−34.81	75.424	−21.12	6.730	−16.63	74.651	7	44	41.193	41.193	−21	83
9	0.2567	−17.87	−6.034	−23.20	80.453	−21.58	8.819	−4.255	78.712	9	46	49.123	46.995	−21	83
10	0.2900	−18.33	−3.249	−9.283	79.872	−21.35	11.84	7.156	86.641	10	48	52.604	49.896	−22	83
11	0.3233	−18.57	−0.696	6.575	78.519	−21.12	14.85	18.566	87.415	12	49	51.250	54.151	−22	83
12	0.3567	−17.87	1.857	21.660	79.486	−20.19	17.64	31.330	87.802	14	51	52.797	61.500	−22	83
13	0.3900	−16.94	4.642	31.910	79.099	−18.80	20.89	37.519	82.000	16	54	54.731	67.108	−22	83
14	0.4233	−15.78	7.426	42.740	70.783	−17.64	23.21	41.580	72.910	18	56	49.123	67.882	−22	84
15	0.4567	−14.16	9.283	54.924	63.240	−16.01	25.76	44.868	62.854	19	58	43.514	70.009	−22	84
16	0.4900	−12.07	11.60	65.755	56.085	−14.62	27.38	47.382	52.604	20	61	40.033	66.335	−21	84
17	0.5233	−9.747	13.23	75.424	43.321	−13.00	29.24	53.377	42.354	22	63	33.844	57.052	−21	84
18	0.5567	−6.962	14.39	82.387	32.491	−10.91	30.40	55.311	26.302	23	64	25.141	54.538	−21	84
19	0.5900	−4.177	15.55	86.448	17.406	−9.283	31.10	56.278	7.542	23	66	17.019	49.509	−21	84
20	0.6233	−1.160	15.55	88.575	2.514	−7.194	30.87	57.632	−10.44	24	68	3.094	33.844	−21	84
21	0.6567	1.857	15.55	86.448	−8.316	−5.338	30.17	56.085	−22.04	23	68	−5.028	21.467	−21	84
22	0.6900	4.642	15.08	82.773	−21.27	−3.481	29.47	54.924	−35.00	23	69	−3.868	8.896	−21	84
23	0.7233	7.426	14.16	77.165	−34.61	−1.625	27.85	52.217	−48.54	23	69	−8.316	−6.189	−21	84
24	0.7567	9.747	12.76	71.943	−47.76	1.484E	26.22	48.736	−61.88	23	69	−15.08	−25.72	−21	84
25	0.7900	12.30	10.91	64.401	−59.17	1.625	23.90	44.868	−78.90	22	67	−18.37	−41.58	−21	84
26	0.8233	14.16	8.819	52.990	−70.20	3.249	20.65	34.424	−87.22	22	66	−23.40	−50.67	−20	84
27	0.8567	15.78	6.266	42.160	−81.22	3.713	17.87	28.429	−89.34	21	64	−26.49	−62.08	−20	84
28	0.8900	17.17	3.249	25.528	−87.41	5.106	14.85	22.627	−93.60	20	62	−33.45	−71.75	−20	84
29	0.9233	17.41	0.2321	10.830	−87.41	5.570	11.60	8.123	−96.11	18	59	−38.87	−78.90	−21	84
30	0.9567	17.87	−2.553	−3.481	−86.83	5.570	8.123	−4.642	−89.92	18	56	−45.25	−82.19	−21	83
31	0.9900	17.17	−5.570	−17.40	−85.09	5.106	5.570	−14.11	−81.80	16	54	−57.24	−81.42	−22	83
32	1.023	16.71	−8.355	−31.13	−79.87	4.642	2.785	−23.40	−75.61	13	51	−59.95	−70.78	−22	83
33	1.057	15.08	−10.91	−44.67	−73.68	3.713	0.4642	−36.55	−67.68	12	49	−62.46	−61.50	−22	84
34	1.090	13.69	−13.23	−56.85	−67.68	2.089	−1.857	−46.60	−56.27	9	47	−63.24	−61.30	−22	84
35	1.123	11.37	−15.55	−70.39	−57.82	0.4642	−3.249	−52.41	−45.06	7	45	−53.18	−52.99	−22	84
36	1.157	8.819	−17.17	−77.93	−44.86	−1.392	−4.874	−57.43	−33.45	6	43	−49.31	−40.03	−22	84
37	1.190	6.034	−18.57	−80.64	−31.79	−3.481	−5.570	−58.83	−20.65	4	42	−46.29	−30.05	−21	84
38	1.223	3.249	−19.26	−77.64	−19.63	−5.338	−6.034	−59.56	−13.63	3	41	−34.23	−23.88	−21	84
39	1.257	0.9283	−19.49	−73.49	−11.99	−7.426	−6.266	−61.11	−9.283	2	41	−26.68	−21.66	−20	84
40															
41															
42															
43															
44															

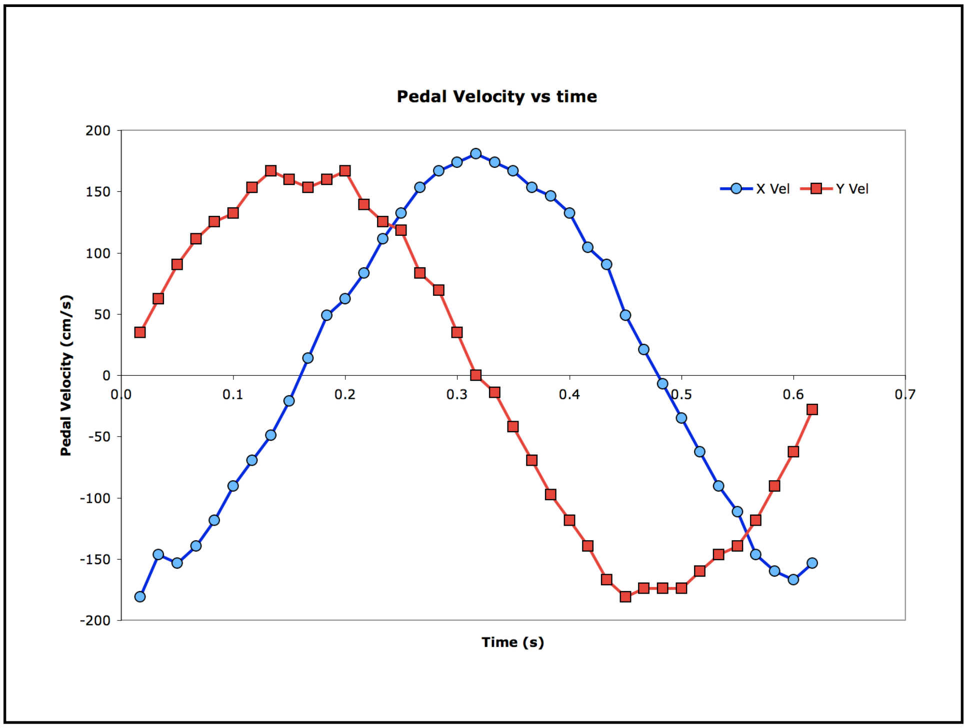
Analysis Methods for Position-Time Data:

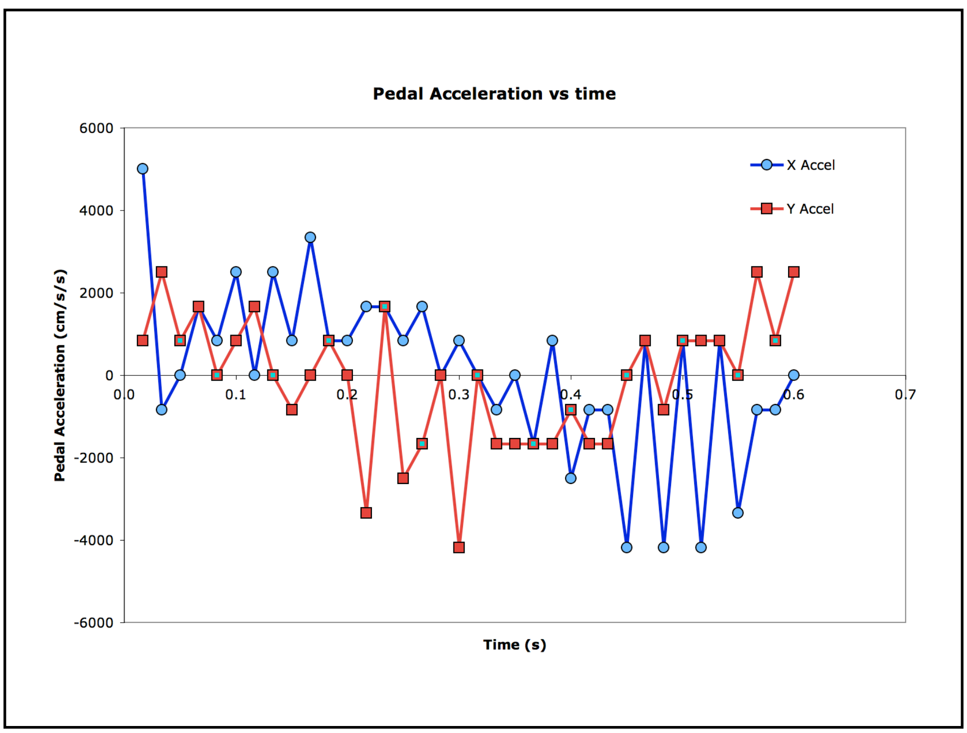
First Central Difference formulas:

Velocity: $$V_i = \frac{X_{i+1} - X_{i-1}}{t_{i+1} - t_{i-1}}$$

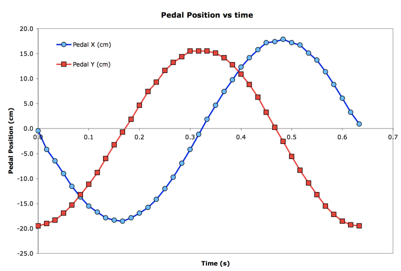
Acceleration: $$a_i = \frac{X_{i+1} - 2X_i + X_{i-1}}{(t_{i+1} - t_i)^2}$$

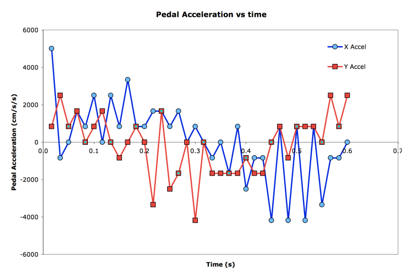



Pedal Velocity vs time
Pedal Velocity (cm/s)
200
150
100
50
0
-50
-100
-150
-200
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
X Vel
Y Vel
Time (s)


Pedal Acceleration vs time
Pedal Acceleration (cm/s/s)
6000
4000
2000
0
-2000
-4000
-6000
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
X Accel
Y Accel
Time (s)

Why does taking the derivative of position-time data seem to amplify the noise in a signal?





Why does taking the derivative of position-time data seem to amplify the noise in a signal?

Low Frequency Signal (f = 1 Hz)

$f(t) = 100 \sin(2 \pi f t)$

$f'(t) = 200 \pi f \cos(2 \pi f t)$

$f''(t) = -400 \pi^2 f^2 \sin(2 \pi f t)$

High Frequency Noise (f = 10 Hz)

$f_n(t) = 1 \sin(2 \pi f t)$

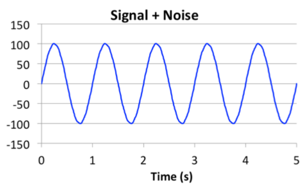
$f'_n(t) = 2 \pi f \cos(2 \pi f t)$

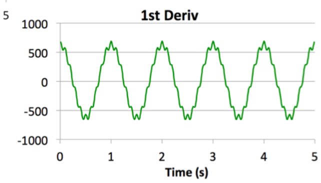
$f''_n(t) = -4 \pi^2 f^2 \sin(2 \pi f t)$

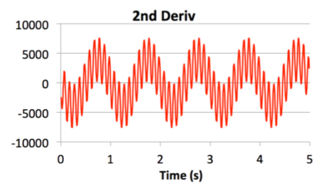
Signal to Noise Ratio:

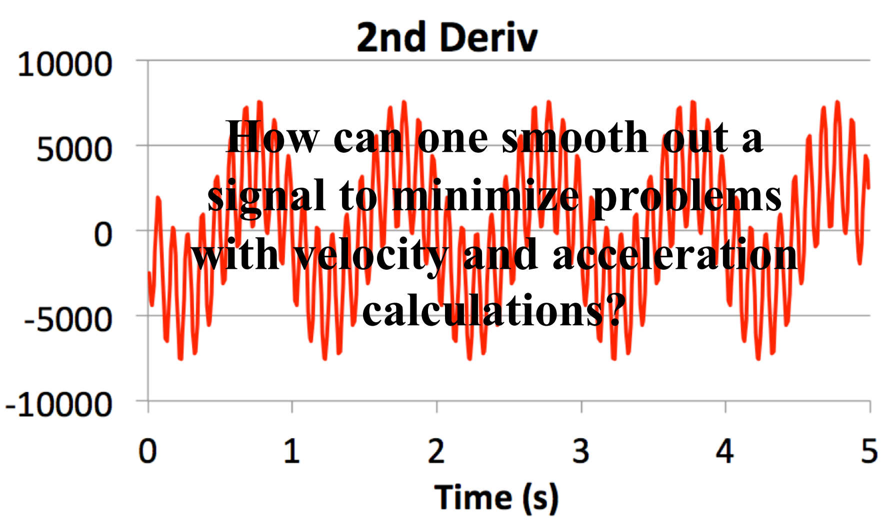
100 : 1
10 : 1
1 : 1

Why does taking the derivative of position-time data seem to amplify the noise in a signal?









How can one smooth out a signal to minimize problems with velocity and acceleration calculations?

Butterworth filtering:

$$X'_i = a_0 X_i + a_1 X_{i-1} + a_2 X_{i-2} + b_1 X'_{i-1} + b_2 X'_{i-2}$$

X = raw data points
X′ = previously filtered data points (recursion)

$$\omega_c = \tan\left(\frac{\pi f_c}{f_s}\right)$$

$K_1 = \sqrt{2}\omega_c$ for a Butterworth filter,
or, $2\omega_c$ for a critically damped filter

$$K_2 = \omega_c^2, \qquad a_0 = \frac{K_2}{(1 + K_1 + K_2)} \qquad a_1 = 2a_0, \qquad a_2 = a_0$$

$$K_3 = \frac{2a_0}{K_2}, \qquad b_1 = -2a_0 + K_3$$

$$b_2 = 1 - 2a_0 - K_3, \qquad \text{or}, \qquad b_2 = 1 - a_0 - a_1 - a_2 - b_1$$

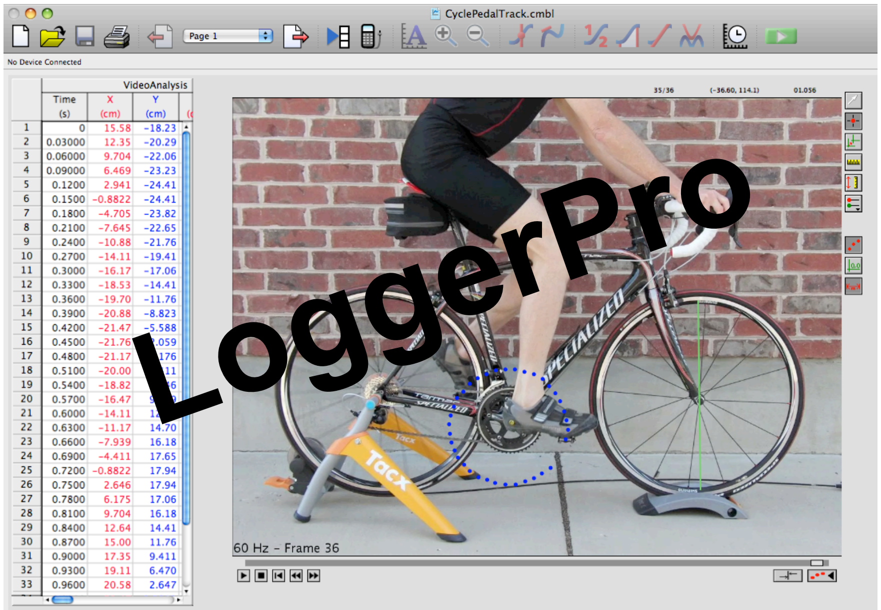
Table 2.2 Coefficients for Butterworth Low-Pass Filter

f_s/f_c	a_0	a_1	a_2	b_1	b_2
4	0.29289	0.58579	0.29289	0.00000	−0.17157
5	0.20657	0.41314	0.20657	0.36953	−0.19582
6	0.15505	0.31010	0.15505	0.62021	−0.24041
7	0.12123	0.24247	0.12123	0.80303	−0.28796
8	0.09763	0.19526	0.09763	0.94281	−0.33333
9	0.08042	0.16085	0.08042	1.05333	−0.37502
10	0.06746	0.13491	0.06746	1.14298	−0.41280
11	0.05742	0.11484	0.05742	1.21719	−0.44687
12	0.04949	0.09898	0.04949	1.27963	−0.47759
13	0.04311	0.08621	0.04311	1.33291	−0.50533
14	0.03789	0.07578	0.03789	1.37889	−0.53045
15	0.03357	0.06714	0.03357	1.41898	−0.55327
16	0.02995	0.05991	0.02995	1.45424	−0.57406
17	0.02689	0.05379	0.02689	1.48550	−0.59307
18	0.02428	0.04856	0.02428	1.51338	−0.61051
19	0.02203	0.04407	0.02203	1.53842	−0.62655
20	0.02008	0.04017	0.02008	1.56102	−0.64135
21	0.01838	0.03676	0.01838	1.58152	−0.65505
22	0.01689	0.03378	0.01689	1.60020	−0.66776
23	0.01557	0.03114	0.01557	1.61730	−0.67958
24	0.01440	0.02880	0.01440	1.63299	−0.69060
25	0.01336	0.02672	0.01336	1.64746	−0.70090

Butterworth Low-Pass Filter is easy to implement in Excel:

	A	B	C	D	E	F	G
1	**Butterworth Filter Coefficients: Vertical Jump Filtering**						
2						**A0**	0.06746
3				**Wc**	0.32492	**A1**	0.13491
4	**SAMPLING FREQUENCY:**		**60**	**K1**	0.45951	**A2**	0.06746
5	**CUTOFF FREQUENCY:**		**6**	**K2**	0.10557	**B1**	1.14298
6				**K3**	1.27789	**B2**	-0.41280

How to do simple 2D motion analysis?



LoggerPro Software can be used to analyze video images: