Fits

Jan 17, 2017

Author: Jacob Cluff

Abstract:
In both the dynamic spring and Gaussian exercises, the given data is imported to and worked in Python. In the dynamic spring measurement exercise, the spring constant k is found to be 1.46 ± 0.111 N/m (7.60%); m_0 is found to be 0.0912 ± 0.0694 kg (76.1%), which agrees with the "theory" value of 0.100 kg. The peak separation for the Gaussian fitting exercise is found to be 0.915 mm

Dynamic Spring Constant:

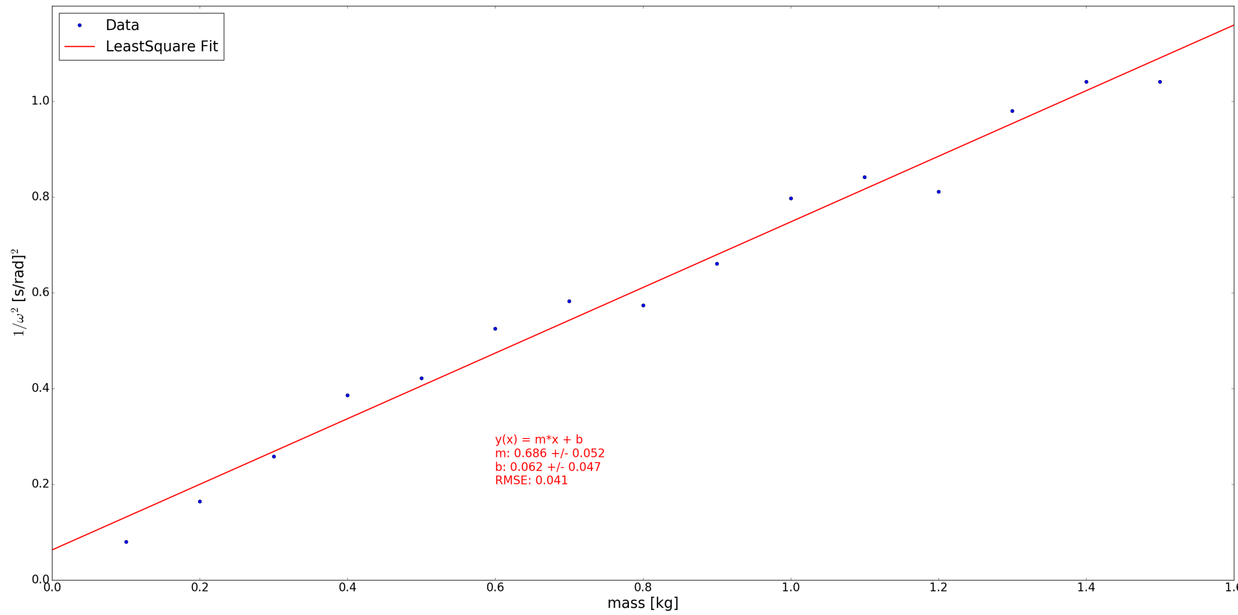

Figure 1: All the essential parameters for this linear model is found in Table 1 below. An R^2 value of 0.982 indicates that the model is a good match.

Table 1: Parameters for Dynamic Spring

Variable	**Value**	**Uncertainty**	%
A **[m/N]**	0.686	0.0521	7.60
B **[kg m/N]**	0.0625	0.0473	75.8
k **[N/m]**	1.46	0.111	7.60
m_0 **[kg]**	0.0912	0.0694	76.1
SS_E	0.0247		
$RMSE$	0.0406		
R^2	0.982		

In the dynamic spring exercise, the angular frequency is given as

$$\omega^2(m) = \frac{k}{m + m_0} \tag{1}$$

,

where k is the spring constant, m is the mass attached to the oscillating spring, and m_0 is the effective mass of the spring.

k and m_0 are found by rearranging equation (1) into linear form, $y = Ax + B$.

$$\begin{aligned}
\frac{1}{\omega^2(m)} &= \frac{m + m_0}{k} \\
&= \frac{1}{k}m + \frac{m_0}{k} \\
\Rightarrow A &= \frac{1}{k},\ B = \frac{m_0}{k}
\end{aligned} \tag{2}$$

Next, A, ΔA, B, ΔB, the residual Sum of Squares (SS_E), and the coefficient of of determination (R^2) are found using equations (1-20) from "UncertaintySlopeInterceptOfLeastSquaresFit.pdf". The Root Mean Sum of Squares ($RMSE$) is found by,

$$RMSE = \sqrt{\frac{SS_E}{n}}, \tag{3}$$

where n is the degrees of freedom, or number of data points.

The propagation of uncertainty is given as

$$\Delta x = \sqrt{\sum_{i=1}^{N} \left(\frac{\partial x}{\partial q_i} \Delta q_i \right)^2}, \tag{4}$$

where q_i is all variables with intrinsic or experimental uncertainty that are being accounted for in x. The uncertainty in k can then be found by

$$\begin{aligned}
A &= \frac{1}{k} \\
\Rightarrow k &= \frac{1}{A} \\
\Rightarrow \Delta k &= \sqrt{\left(\frac{\partial k}{\partial A} \Delta A \right)^2} \\
&= \sqrt{\left(-\frac{\Delta A}{A^2} \right)^2} \\
&= \frac{\Delta A}{A^2}.
\end{aligned} \tag{5}$$

Likewise, for the uncertainty in m_0,

$$\begin{aligned}
B &= \frac{m_0}{k} \\
\Rightarrow m_0 &= Bk \\
\Rightarrow \Delta m_0 &= \sqrt{\left(\frac{\partial m_0}{\partial B} \Delta B \right)^2 + \left(\frac{\partial m_0}{\partial k} \Delta k \right)^2} \\
&= \sqrt{(k\Delta B)^2 + (B\Delta k)^2}
\end{aligned} \tag{6}$$

Gaussian Fits:

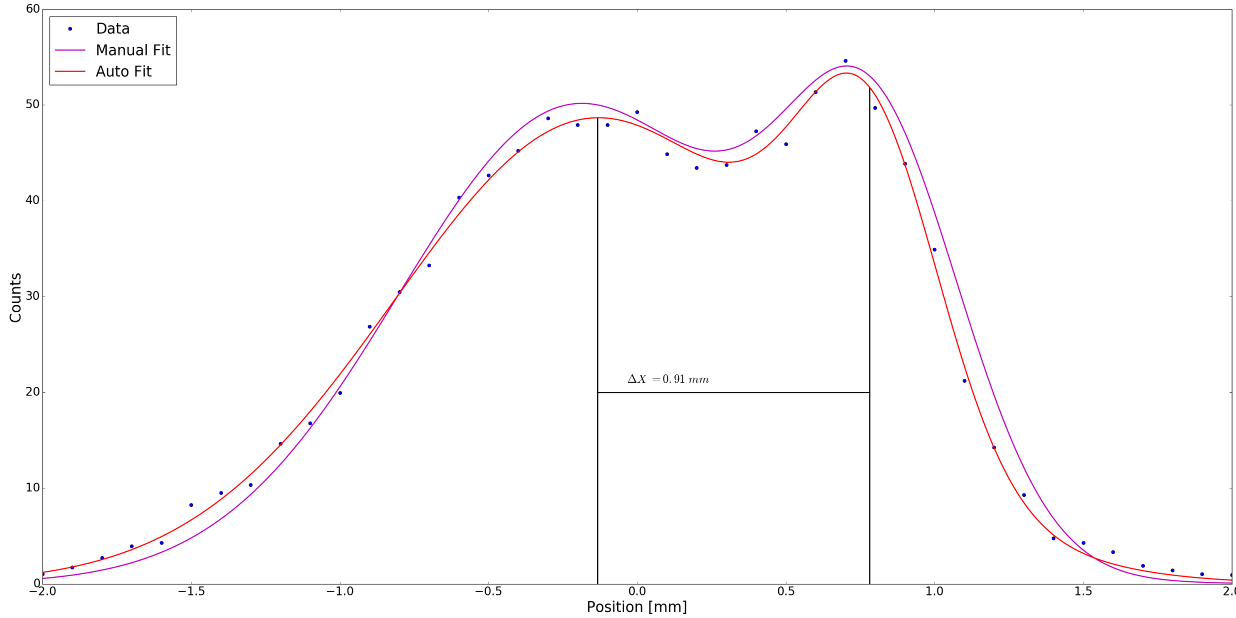

Figure 2: The parameters for both Gaussian distributions are shown in Tables 1 and 2 below. The RMSE (Equation 3) found is .997, which implies a good fit to the data. The uncertainty in the parameters is determined by manually adjusting each of the six parameters individually and recording the smallest variation that results in a curve that does not visually fit the data. a_1 occurs pretty much where it appears visually because σ_2 is relatively small and has a negligible contribution in the region a_1; however, since σ_1 is relatively large, it has a significant contribution in in the region of the second distribution, increasing the apparent peak height of the second distribution and shifting the visual peak to the left of the real peak.

Table 2: Gaussian 1 Parameters

	Peak Height [counts]	**Average [mm]**	**STD [mm]**
Manual Fit	50	-0.2	0.6
Auto Fit	48.7 ± 2	$-0.133 \pm .02$	$0.685 \pm .03$

Table 3: Gaussian 2 Parameters

	Peak Height [counts]	**Average [mm]**	**STD [mm]**
Manual Fit	40	0.8	0.3
Auto Fit	31.9 ± 1	$0.782 \pm .01$	$0.239 \pm .02$

The equation for two Gaussian peaks is given by Equation (7),

$$Counts(x) = C_1 \exp\left(-\frac{(x-a_1)^2}{2\sigma_1^2}\right) + C_2 \exp\left(-\frac{(x-a_2)^2}{2\sigma_2^2}\right) \tag{7}$$

,

where C_i is the max peak height, a_i is the peak average, and σ_i is the standard deviation of the peak from the average.

Appendix:

The structured data array for the dynamic spring exercise is:

d1 = 'kg' : [.1,.2,.3,.4,.5,.6,.7,.8,.9,1,1.1,1.2,1.3,1.4,1.5] , 'rad/s' : [3.55,2.47,1.97,1.61,1.54,1.38,1.31,1.32,1.23,1.12, 1.09,1.11,1.01,.98,.98]

The structured array used for the Gaussian distributions is:

d2 = 'mm' : np.arange(-2,2.1,.1), 'cnt' : [1.01,1.74,2.71,3.93,4.30,8.25,9.52,10.35,14.62,16.76,19.97,26.87,30.49, 33.27,40.34,42.65,45.25,48.64,47.93,47.93,49.28,44.90,43.45,43.73,47.29, 45.93,51.36,54.62,49.72,43.87,34.91,21.20,14.23,9.30,4.77,4.30,3.31,1.89, 1.41,1.03,0.94]