Homework 2

1. Consider a simple estimation of a single parameter: $y = a$, where the measurement noise has unit variance.

(a) Determine the accuracy of the parameter estimate as a function of the number of measurements.

(b) Verify the results through a monte-carlo simulation.

Solution:
The variance on any given measurement can be defined with expectation as:

$$\sigma^2 = E\{(y-\bar{y})(y-\bar{y})\} = \sigma^2$$

With more measurements at the same point, the variance decreases as follows:

$$\sigma_M^2 = \frac{\sigma^2}{M} \tag{1}$$

Where M is the number of measurements. Taking the square root of *equation 1* provides the standard deviation which is in the same units as the measurement. To confirm this with a monte-carlo simulation, a simple simulation with on a measurement of $y = 1$ is performed. A total of 1-500 measurements were simulated for each of 1000 monte-carlo runs. Below is a plot of the results:

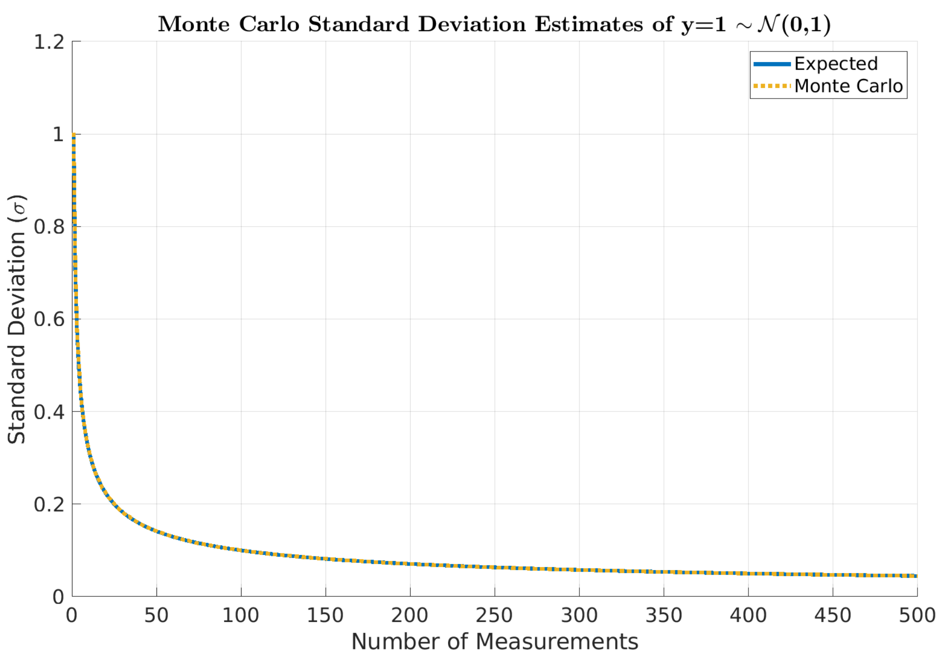


Figure 1: Effect of the Number of Measurements on Accuracy

2. Given the set of data, perform a least-squares fit to solve for the model coefficients and the predicted estimation error (1-σ) for the coefficient *a* assuming the 1-σ measurement noise on y is 0.4.

x	0	1	2	3	4
y	0.181	2.680	3.467	3.101	3.437

(a) $y = a + bx$

(b) $y = a + bx + cx^2$

(c) $y = a + bx + cx^2 + dx^3$

(d) Is the estimate for a consistent? Why or why not? Which is probably the correct prediction of the estimation error on a?

Solution:

Defining the observation matrix for each part (subscript defining part):

$$H_a = \begin{bmatrix} 1 & x \end{bmatrix}$$
$$H_b = \begin{bmatrix} 1 & x & x^2 \end{bmatrix}$$
$$H_c = \begin{bmatrix} 1 & x & x^2 & x^3 \end{bmatrix}$$

With the measurement vector (y) being the defined in the table and estimating the coefficients in each part with the least squares formula below, the estimate of a is as follows:

$$x = (H^T H)^{-1} H^T y$$

$$a_a = 1.186$$
$$a_b = 0.404$$
$$a_c = 0.126$$

This estimate is not consistent. Looking at the standard deviation on each estimate of a:

$$\sigma_{a_a} = 0.310$$
$$\sigma_{a_b} = 0.376$$
$$\sigma_{a_b} = 0.397$$

It appears that the linear estimate of a is the most accurate. This is likely due to the fact that the additional higher order terms dominate the overall curve fit as well as not including measurement noise in the higher order models. However, looking at the plot below, it is clear that a third order approximation provides the best fit.

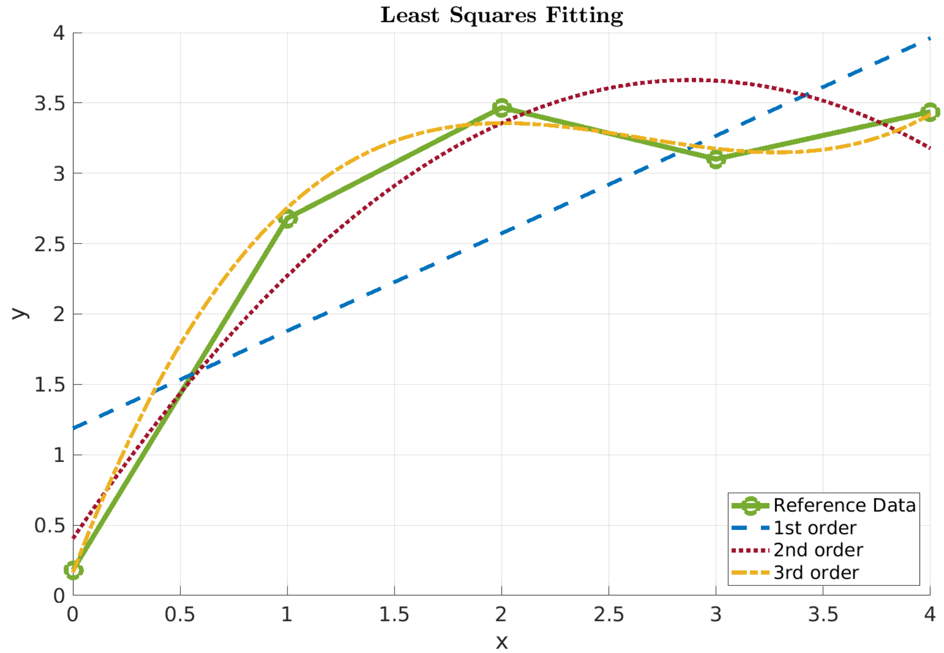


Figure 2: Curve Fit Coefficient Matching

3. Given the following range equation $r^2 = (x-a)^2 + (y-b)^2$ with a range error of 0.5 meters (1-σ) and the table below:

a	0	10	0	10
b	0	0	10	10
r^2	25	65	45	85

(a) Find the Jacobian Matrix.

(b) What is the expected solution uncertainty?

(c) What is the solution for x and y?

(d) Perform a monte-carlo simulation and verify *part b*.

Solution:

Given this equation is linear for r^2, the Jacobian can be easily made by taking the partial derivatives of r^2 with respect to x and y.

$$\frac{\partial r^2}{\partial x} = 2(x-a)$$
$$\frac{\partial r^2}{\partial y} = 2(y-b)$$

Applying a taylor series expansion:

$$r^2 = (x_0-a)^2 + (y_0-b)^2 + 2(x_0-a)\Delta x + 2(y_0-b)\Delta y$$

Creating the Jacobian by plugging in a and b:

$$H = \begin{bmatrix} 2(x_0 - 0) & 2(y_0 - 0) \\ 2(x_0 - 10) & 2(y_0 - 0) \\ 2(x_0 - 0) & 2(y_0 - 10) \\ 2(x_0 - 10) & 2(y_0 - 10) \end{bmatrix}$$

And the following system:

$$y = Hx$$

$$\begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} = \begin{bmatrix} 2(x_0 - 0) & 2(y_0 - 0) \\ 2(x_0 - 10) & 2(y_0 - 0) \\ 2(x_0 - 0) & 2(y_0 - 10) \\ 2(x_0 - 10) & 2(y_0 - 10) \end{bmatrix} \begin{bmatrix} 25 - ((x_0 - 0)^2 + (y_0 - 0)^2) \\ 65 - ((x_0 - 10)^2 + (y_0 - 0)^2) \\ 45 - ((x_0 - 0)^2 + (y_0 - 10)^2) \\ 85 - ((x_0 - 10)^2 + (y_0 - 10)^2) \end{bmatrix}$$

Initializing at $\begin{bmatrix} 0 & 0 \end{bmatrix}^T$ and using Newton-Raphson to iteratively solve the problem, the solution and expected uncertainty are:

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 3 \\ 4 \end{bmatrix} \text{ m}$$

$$\sigma_x = 0.0232 \text{ m}$$

$$\sigma_y = 0.0246 \text{ m}$$

Performing a monte-carlo of 1000 iterations results in a simulation standard deviation of:

$$\sigma_x = 0.0230 \text{ m}$$

$$\sigma_y = 0.0250 \text{ m}$$

Which matches very closely to the expected value.

4. Using the provided data file containing GPS and SOOP satellites along with their simulated psuedoranges and a 1-σ accuracy of 0.5 meters:

(a) Calculate the position and expected horizontal and vertical error using the first 4 GPS satellites, initialized at the center of the earth. How many iterations does it take?

(b) Repeat *part a* with all 9 satellites.

(c) Repeat *part a* assuming a perfect clock.

(d) Calculate the position using 2 GPS and SOOP satellites. What happens and why?

(e) Repeat *part c* with an initial condition of $[423000, -5362000, 3417000]$.

Solution:

Using the psuedorange geometry matrix defined in class:

$$G = \begin{bmatrix} \frac{-u_x}{r} & \frac{-u_y}{r} & \frac{-u_z}{r} & 1 \end{bmatrix}$$

Table 1: Parts a-e solution

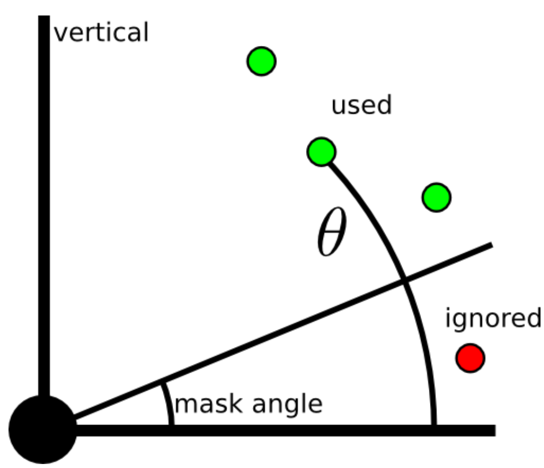
	# SV	# SOOP	Iterations	HDOP	VDOP
a	4	0	5	7.403	8.307
b	9	0	5	0.977	1.086
c	4	0	5	7.403	8.307
d	2	2	N/A	N/A	N/A
e	9	0	3	0.977	1.086

The data when utilizing the SOOP satellites in combination with the GPS satellites is unavailable because the SOOP satellites cause the geometry matrix to be poorly conditioned. This causes the user position to be unobservable since the number of usable measurements is fewer than 4.

5. The ionosphere (50-1000 km above the Earth) is one source of error on GPS psuedoranges. Ionospheric delay is a function of how much atmosphere the signal passes through, which is a function of the elevation angle, θ. A simple model is as follows:

$$I(\theta) = A_1(1 + 16(0.53 - \frac{\theta}{180})^3)$$

Where $A_1 = 5(10^{-9})$ seconds and θ is in degrees. Elevation masks, a constant angle below which all signals are ignored, combat the significant delays at low elevation angles.



Using the 9 GPS satellite positions in the data file and the Ionospheric delay model, generate a plot of position accuracy versus mask angle. Comment on this plot.

Solution:

Iterating through elevation angles starting at 0 degrees and ending when there is fewer than 4 satellites in view creates the following plot:

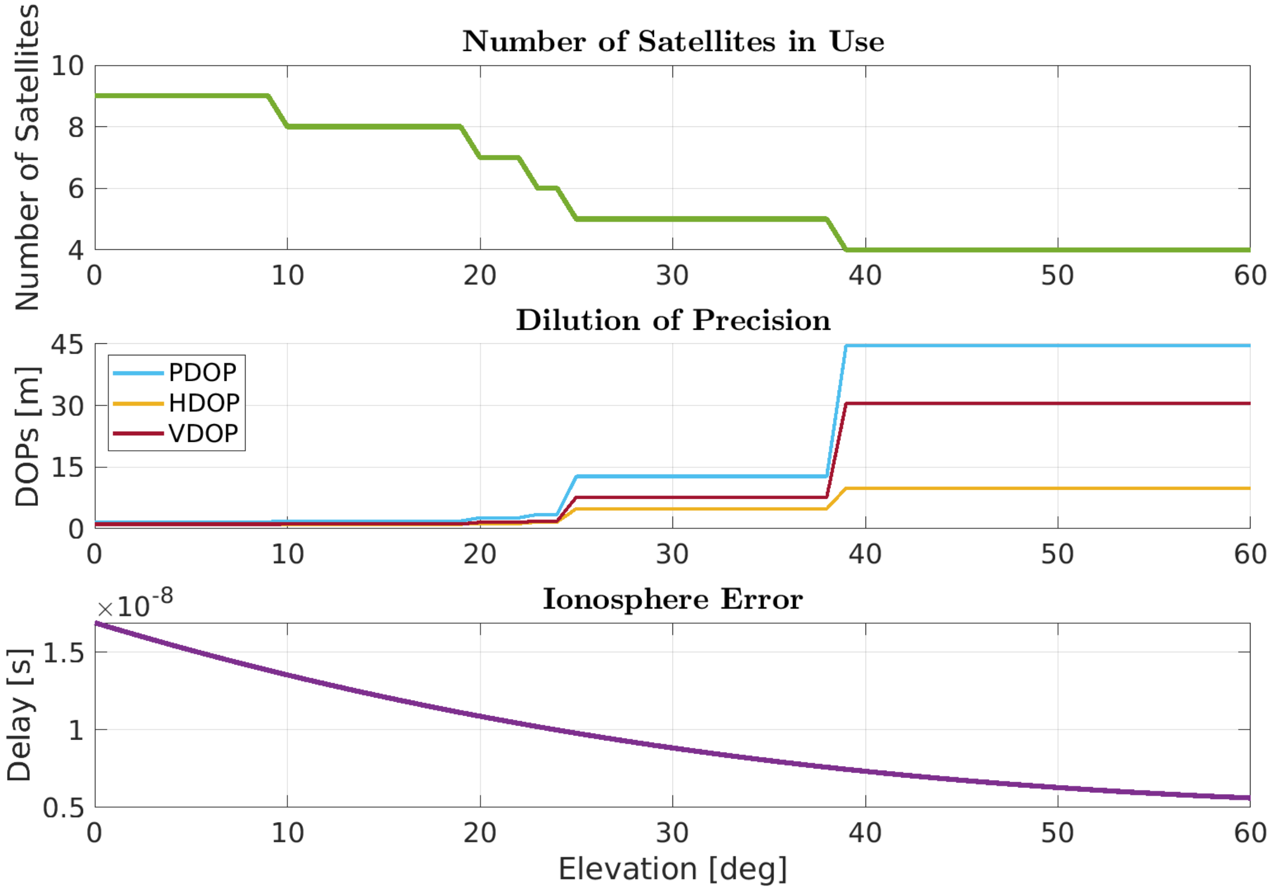


Figure 3: Errors when Increasing the Elevation Mask Angle

As shown, the positioning error increases as the elevation mask is increased, particularly in the vertical direction. This is due to the reduced observability as the satellites at lower elevations (closer to tangent with the higher elevation satellites) get removed. The ionosphere error also decreases as the elevation mask increases because less of the ionosphere is traversed by the signal.