

Night 5: Polar LIDAR Express

Quantitative Engineering Analysis

Spring 2019

Solution:

Note: The Matlab Script used to make the solution plots is [here](#).

1 LIDAR in Polar and Cartesian Coordinates

1.1 Laser Scanner

Laser scanners are becoming an increasingly important sensor in modern robotics. A key accelerator of this trend is the development of self-driving cars, which use laser scanners as their principal source of information about the world around them (e.g., check out [this video](#) of a 3D map of a highway in the San Francisco Bay Area).

Your Neato comes with a much more basic laser scanner than the long-range 3D scanners you see in self-driving cars. The Neato's laser scanner provides an estimate of the distance to nearby features in the environment (e.g., obstacles) at a rate of 5 Hz, with an angular resolution of 1 degree, and a maximum usable range of about 3 meters. Under the hood, the sensor works by emitting a laser and then detecting the reflected laser light using a camera. From the known geometry of the camera / laser pair and the position of the reflected laser beam in the camera image, the depth of the obstacle can be recovered (see Figure 1). Further, the entire laser / camera assembly spins, providing an estimate of the distance to features in the environment in a full 360-degree sweep around your robot.

Sometimes it is useful to take the raw points detected by the laser scanner and interpret them as geometrical objects (e.g., lines, curves, polygons). For instance, given the laser scan in Figure 3 we may wish to automatically interpret the laser scan data (i.e. pick out lines, circles, etc.). In the context of the present challenge, the ability to differentiate between lines and circles is crucial to running The Gauntlet™ since walls are obstacles and the circle denotes the goal.

Conveniently, in modules 1 and 2 we've built a lot of the machinery needed to do just this (aren't we sneaky?). Here, you'll have a chance to bring it all together.

1.2 It's like your birthday, but way better! [1 hr]

Today you get to play with a new sensor, and that's awesome! Before you start using the Neato's LIDAR to do really interesting and useful

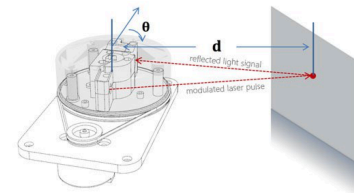


Figure 1: A schematic of the Neato's laser scanner. Due to the relative position of the emitter and detector, the position of the reflected light changes predictably as a function of d . The laser scanner spins with a rate of 5Hz. The angle of the scanner is denoted as θ .

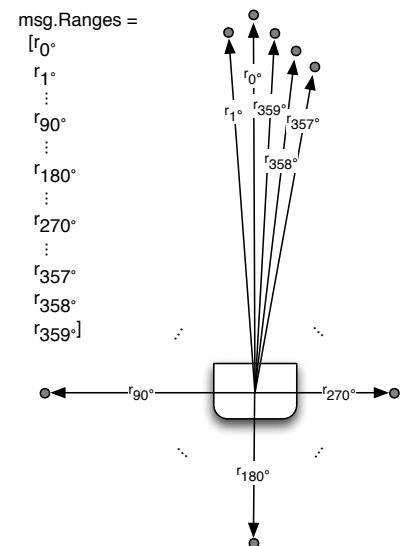


Figure 2: A schematic of the laser scan data. In the figure, r_{θ° indicates the distance in meters to a detected object at a bearing of θ degrees. `msg.Ranges` shows how each of these distances maps to a particular index in the array. The array holds the distances, and the angles are implicitly encoded by the position in which they appear in the array. If no detection is made at a particular bearing, the value 0.0 is used instead. Note: the angles of the measurements around $\theta = 0^\circ$ have been exaggerated for visualization purposes (i.e. the angles shown are larger than a single degree).

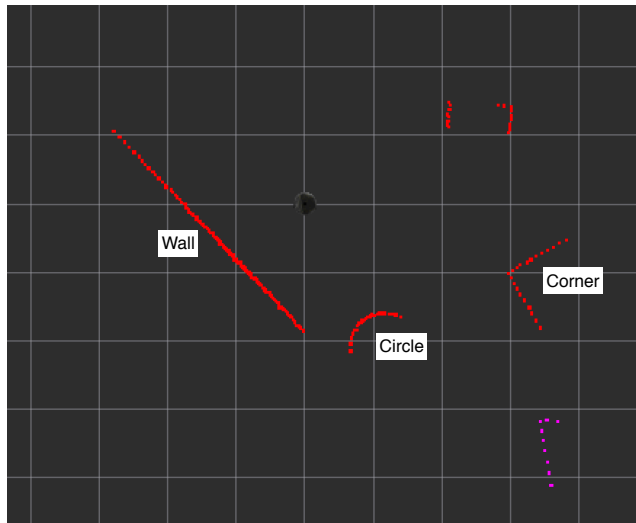


Figure 3: A laser scan with high-level features identified.

things, it is important to take a few minutes to understand the sensor's characteristics and limitations. Typically you would approach understanding the capabilities of a new sensor by first reading the relevant data sheets and manuals (I know, not really in the typical engineering playbook), then conducting some simple characterization experiments. For this assignment, let's assume the section above where the LIDAR was introduced and the approximate range and angular resolution of the sensor were given was the data sheet. Congratulations, you read the data sheet!

Exercise (1) Now, think about some simple experiments you could perform to learn more about the LIDAR. You can do these using the “teleopAnd-Visualizer” function to visualize the LIDAR data. Listed below are some suggestions for questions to answer about the LIDAR, but don't feel limited! You don't need to spend much more than an hour on this, we just want you to get a good feel for the LIDAR. You should make note of the question you wanted to answer, the experiment you ran, the results, and what you learned about the LIDAR.

- (a) What is the maximum range of the LIDAR? Does it match the spec given above? Does this range vary with environmental conditions?
- (b) Does the type of surface impact the LIDAR measurement? Matte or glossy? Glass?
- (c) Does the angle of a surface impact the LIDAR measurement?

- (d) Does the LIDAR have a minimum distance or dead-zone around the sensor? What are the implications of this if so?
- (e) How does light impact the LIDAR? Sunlight? Dark?
- (f) What is the minimum object size that can be detected? How does this relate to the angular resolution?

Solution:

Experiments and answers to this question may vary. The goal is to develop an understanding of the performance and limitations of the LIDAR sensor. This is important when creating experimental protocols, identifying bad measurements in a data set, or deciding on a sensor to purchase/use for a given application. Generally LIDAR has trouble with glass and bright sunlight. It should work great in the dark. It seems that the LIDAR on the NEATO has a dead zone around the robot, but it is hard to determine if that is due to the sensor or if measurements within a certain distance are simply rejected in software.

1.3 Polar and Cartesian Coordinates [1 hr]

LIDAR data and indeed data in many other applications are typically presented in polar coordinates. In many other situations however, it is more convenient to represent the data in Cartesian coordinates. In this set of exercises, you will get some practice converting polar to Cartesian coordinates and vice-versa.

In a point in 2-D space can be described as a sum of unit vectors in the \hat{i} and \hat{j} directions which are unit vectors in the directions of the x and y axes respectively.

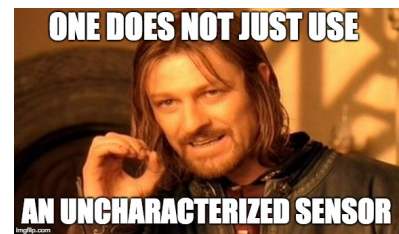
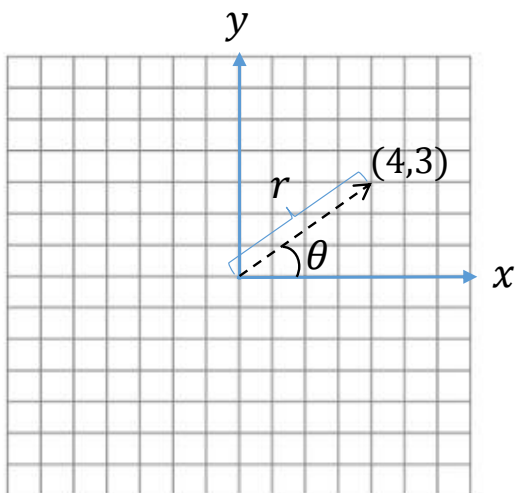


Figure 4: Can't argue with this one.

Figure 5: Illustration of vector in polar and Cartesian coordinates.

Consider the point $(4,3)$ illustrated in Figure 5. You can write the vector representing this point as a sum of $\hat{\mathbf{i}}$ and $\hat{\mathbf{j}}$ as follows

$$\begin{bmatrix} 4 \\ 3 \end{bmatrix} = 4\hat{\mathbf{i}} + 3\hat{\mathbf{j}} \quad (1)$$

We can also write this point in terms of polar co-ordinates as (r, θ) , where

$$r = \sqrt{4^2 + 3^2} = 5 \quad (2)$$

$$\theta = \tan^{-1} \left(\frac{3}{4} \right) \quad (3)$$

In general a vector (x, y) in Cartesian coordinates can be expressed in polar coordinates as (r, θ) where

$$r = \sqrt{x^2 + y^2} \quad (4)$$

$$\theta = \tan^{-1} \left(\frac{y}{x} \right) \quad (5)$$

To go from polar to cartesian coordinates, we can use the following formulas

$$x = r \cos \theta$$

$$y = r \sin \theta$$

Besides expressing a vector as a linear combination of the $\hat{\mathbf{i}}$ and $\hat{\mathbf{j}}$ vectors, we can also define two new unit vectors $\hat{\mathbf{r}}$ and $\hat{\boldsymbol{\theta}}$ defined as follows

$$\hat{\mathbf{r}} = \begin{bmatrix} \cos \theta \\ \sin \theta \end{bmatrix} \quad (6)$$

$$\hat{\boldsymbol{\theta}} = \begin{bmatrix} -\sin \theta \\ \cos \theta \end{bmatrix} \quad (7)$$

Note that these vectors are not absolute with respect to the origin of the cartesian coordinates. While it does seem strange to represent unit vectors in this manner, this kind of representation will prove useful in the near future when we are dealing with dynamics of rotating bodies (i.e. you will not directly use this until next semester, but we thought we should give you a preview).

Exercise (2) Convert the following from Cartesian to polar coordinates.

(a) $(x, y) = (-8, 6)$

Solution:

$\theta = 2.5$ radians, $R = 10$

(b) $(x, y) = (5, 12)$

Solution:

$$\theta = 1.18 \text{ radians, } R = 13$$

Exercise (3) Convert the following from polar to Cartesian coordinates.

(a) $(r, \theta) = (2, \frac{\pi}{6})$

Solution:

$$x = 1.73, y = 1.00$$

(b) $(r, \theta) = (2, -\frac{\pi}{3})$

Solution:

$$x = 1, y = -1.73$$

Exercise (4) What is $\hat{r}^T \hat{\theta}$?

Solution:

\hat{r} and $\hat{\theta}$ are orthogonal, so $\hat{r}^T \hat{\theta} = 0$.

Exercise (5) In words, describe the \hat{r} and $\hat{\theta}$ vectors.

Solution:

The \hat{r} vector has length 1 and points in the direction of \vec{r} , and $\hat{\theta}$ has length 1 and points in the direction of increasing θ , orthogonal to \hat{r} .

Exercise (6) Write the following points in terms of the \hat{r} and $\hat{\theta}$ vectors

(a) $(4, 3)$.

Solution:

This was a very poorly worded problem and was not graded.

(b) $(3, \frac{\pi}{3})$

Exercise (7) Now let's look at some LIDAR data that was collected from the Chamber of Emptiness™, which is a desolate room with one wall. Load the data in *scan1.mat*, which should contain the variables r for range (distance from the scanner in meters) and θ for the angle from the front of the robot (see Figure 3).

(a) Plot these data using:

```
plot(theta, r, 'ks')
```

Solution:

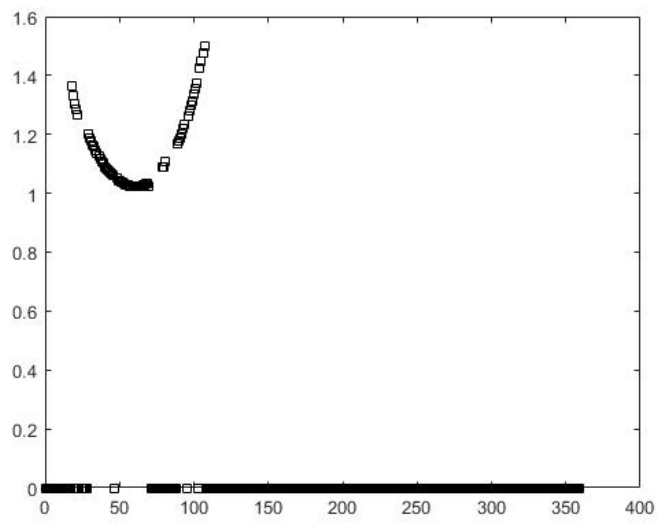


Figure 6: Answer for 7a.

- (b) Plot the data in polar coordinates using:

```
polarplot(deg2rad(theta),r,'ks',...  
          'MarkerSize',6,'MarkerFaceColor','m')
```

Solution:

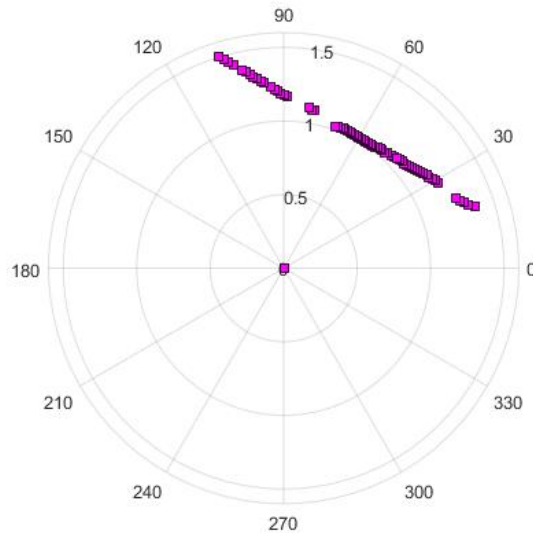


Figure 7: Answer for 7b.

- (c) You should see two subsets of data (a line and a parabola in the plot from part a). Where are these two subsets represented in the polarplot?

Solution:

The samples making up the line in part a are all at zero distance, and appear as a cluster at the center of the polar plot. The parabola from part a becomes the line in part b.

- (d) When the LIDAR does not detect any object at a given angle, it outputs a 0 for the value of r at that angle (this is a common problem in the Chamber of Emptiness). Create new variables θ_{clean} and r_{clean} which represent the angles and values of r when r is not zero.

Solution:

one implementation:

```
index=find(r~=0);  
r_clean=r(index);  
 $\theta_{clean}=\theta(index);$ 
```

- (e) Now convert r_{clean} and θ_{clean} to Cartesian coordinates and plot them using the `plot()` function.

Solution:

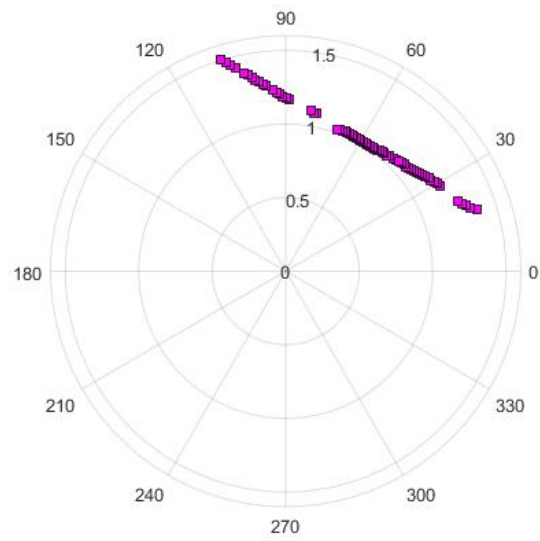


Figure 8: Answer for 7d.

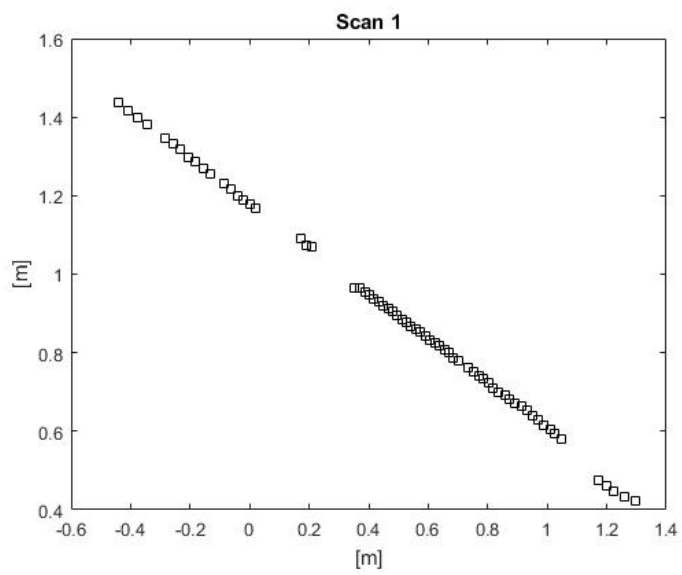


Figure 9: Answer for 7e.

2 Fitting Models to Laser Scan Data [3 hours]

2.1 Least Squares for Line Fitting

The first structure we would like to detect in the laser scan data is a line. We've seen the notion of a line of best fit a couple of times in Module 2. The first time we saw it was when we studied linear regression. For instance, during Day 5 in module 2 you used the temperatures of Boston, Providence, and Washington D.C. to predict the temperature in New York. This led to the following model for predicting the temperature in New York on day i : $\hat{T}_N(i) = \beta_3 T_W(i) + \beta_2 T_B(i) + \beta_1 T_P(i) + \beta_0$. This model gave rise to the following equation for MSE.

$$\begin{aligned} \text{MSE}(\beta_0, \beta_1, \beta_2, \beta_3) &= \frac{1}{n} \sum_{i=1}^n (T_N(i) - \hat{T}_N(i))^2 \\ \text{MSE}(\beta_0, \beta_1, \beta_2, \beta_3) &= \frac{1}{n} \sum_{i=1}^n (T_N(i) - (\beta_3 T_W(i) + \beta_2 T_B(i) + \beta_1 T_P(i) + \beta_0))^2 \end{aligned} \quad (8)$$

In the previous section, we converted each detection in our laser scan data into Cartesian coordinates. Let's call this set of points $(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)$. Further, let's assume that the relationship between x and y is given by $y = \beta_1 x + \beta_0 + \epsilon$ (where ϵ is an error term that we'd like to minimize).

Desired learning outcomes:

- Increase proficiency in coding solutions to linear regression problems.
- Draw connections to previous work in this class.
- Determine conditions under which an algorithm is ill-suited for application.

Exercise (7) Find the linear regression line of best fit (values of β_0 and β_1) for the Cartesian representation of the laser scan from *scan1.mat*. Plot the line of best fit. Save your values for the line of best fit; you will recreate this again later to compare against another method. It may be helpful to revisit your work from Module 2, Day 4 and Night 4; Module 2, Day 5; and/or your smile detection code.

Solution:

Exercise (8) Repeat Exercises 6d-e and 7 for *scan2.mat*, *scan3.mat*, and *scan4.mat*.

Solution:

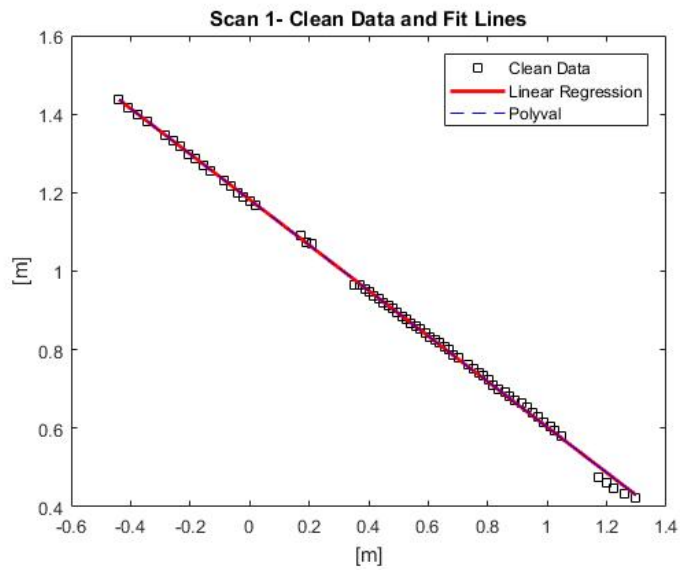


Figure 10: Clean data and best fit lines using linear regression and the Matlab Polyfit function for Scan 1.

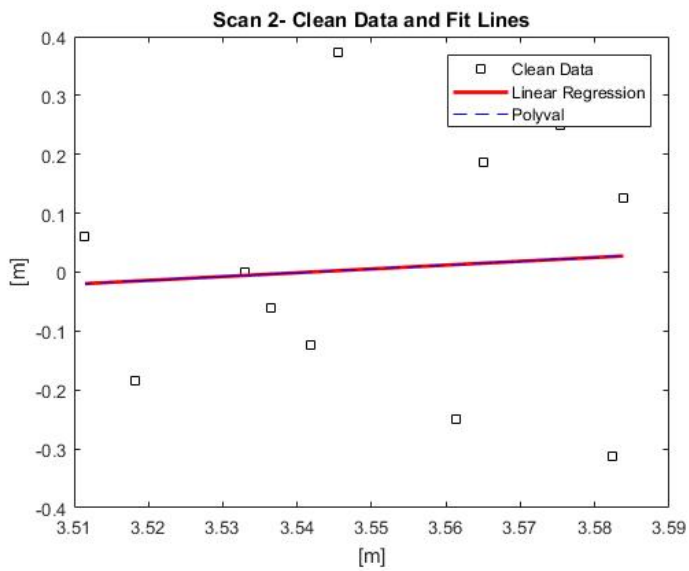


Figure 11: Clean data and best fit lines using linear regression and the Matlab Polyfit function for Scan 2.

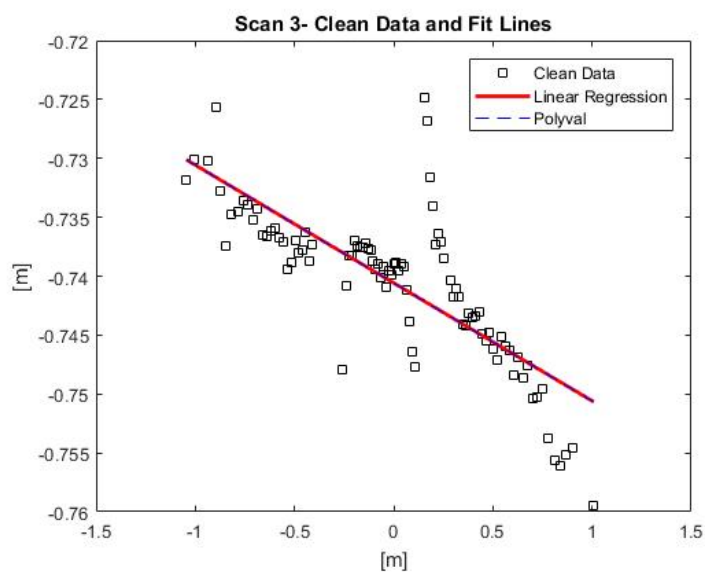


Figure 12: Clean data and best fit lines using linear regression and the Matlab Polyfit function for Scan 3.

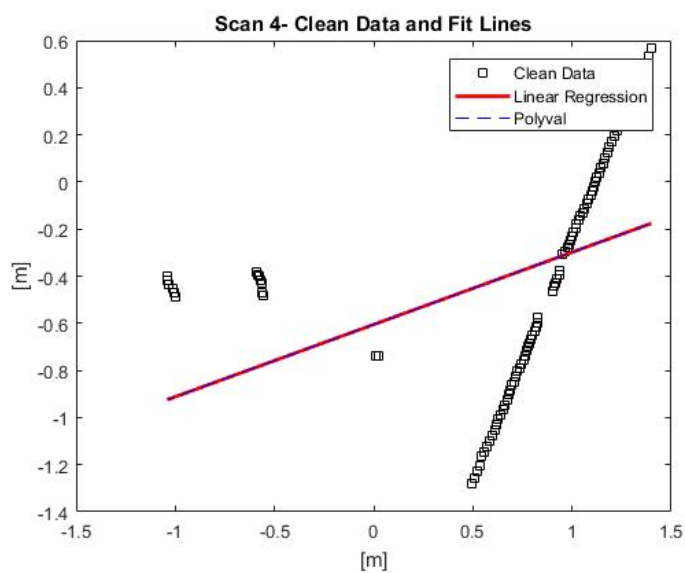


Figure 13: Clean data and best fit lines using linear regression and the Matlab Polyfit function for Scan 4.

Exercise (9) Note what you observe, and comment on any deficiencies with this approach to line fitting.

Solution:

Most apparent deficiency is that linear regression is skewed by the presence of outliers in the dataset.

2.2 Principal Component Analysis [3 hr]

Linear regression defines the line of best fit in terms of the sum of squared distances, measured vertically, between the data points and the line (see Figure 18). However, when determining the best fitting line for a laser scan, there's nothing special about the vertical direction. In fact, it would make as much sense to minimize the sum of the squared horizontal distances. In this context, a much more natural way to think about the line of best fit is to minimize the perpendicular distances between the line and the data points (see Figure 18).

Desired learning outcomes:

- Generalize knowledge of an algorithm you've already learned by applying it in a new context.
- Plot lines that are defined parametrically.
- See the connection between different objective functions and the resultant optimal solutions.

Exercise (10) Think about how to use principal component analysis to determine the line of best fit that minimizes the sum of squared perpendicular distances as shown by the blue dashed lines in Figure 18.

- (a) Describe conceptually why you think this algorithm will achieve the desired goal.

Solution:

We can recall from Module 2 Night 7 that PCA identifies the direction of greatest variation in a data set (this direction is given by the eigenvector associated with the largest eigenvalue). Conceptually, the direction of *greatest* variation should be orthogonal to the direction of *least* variation. So, the direction perpendicular, or orthogonal, to the line found using PCA should minimize variation, and therefore minimize the sum of squared perpendicular distances.

- (b) Write pseudocode to describe the steps you will need to take. You should consider what an eigenvector represents. (Hint: if you translate your data, be sure to translate it back.)

Solution:

- i. Convert data from polar to cartesian coordinates.
- ii. Calculate the x and y means for the dataset.
- iii. Use the x and y means to center the data.
- iv. Find the singular value decomposition (SVD).
- v. Identify the direction of greatest variation in the dataset as the eigenvector associated with the largest eigenvalue.
- vi. Translate the data back to its original positions.

Exercise (11) In MATLAB, do the following for the Cartesian data from each of the four scans *scan1.mat*, *scan2.mat*, *scan3.mat* and *scan4.mat*.

- (a) Compute the line of best fit using PCA.
- (b) Plot the data points, line of best fit from PCA, and line of best fit from linear regression. Be sure to note which line is which.

Solution:

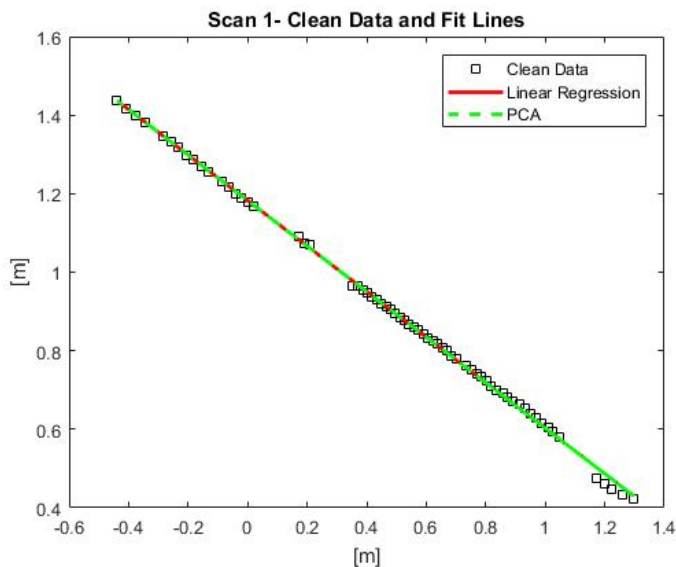


Figure 14: Clean data and best fit lines using linear regression and PCA for Scan 1.

- (c) Comment on the differences between the fits for each data set.

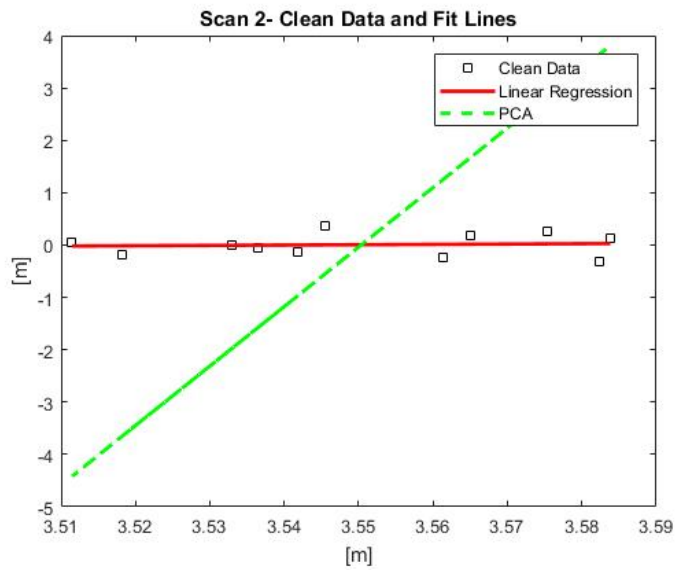


Figure 15: Clean data and best fit lines using linear regression and PCA for Scan 2.

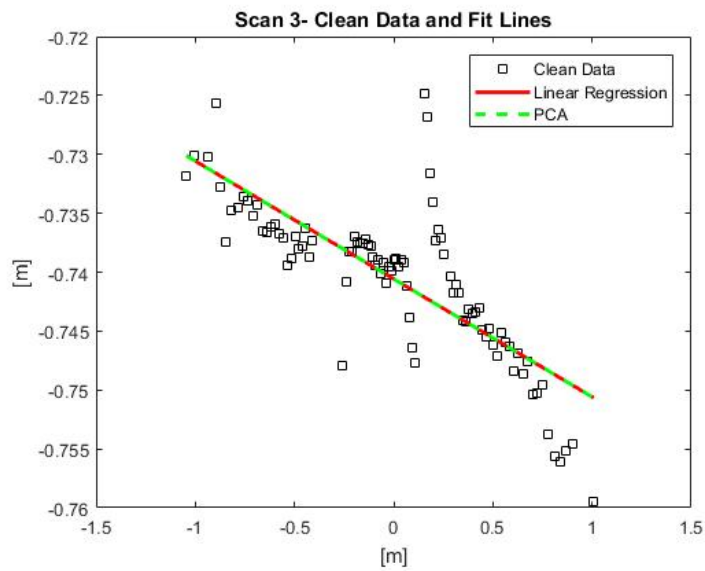


Figure 16: Clean data and best fit lines using linear regression and PCA for Scan 3.

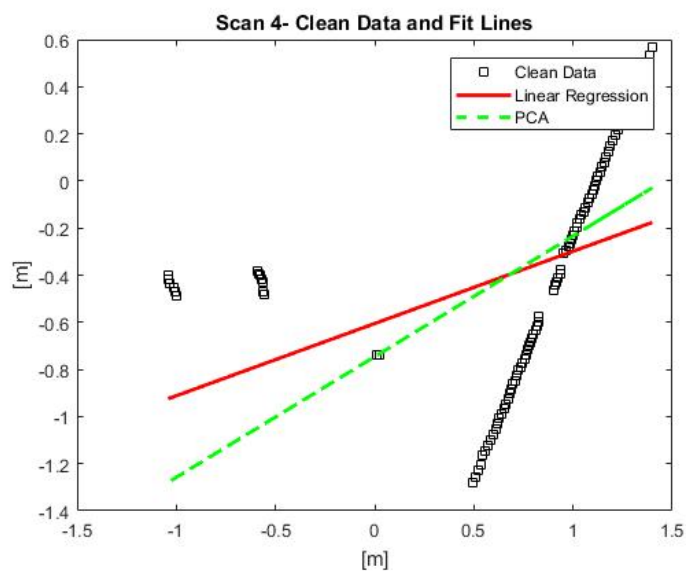


Figure 17: Clean data and best fit lines using linear regression and PCA for Scan 4.

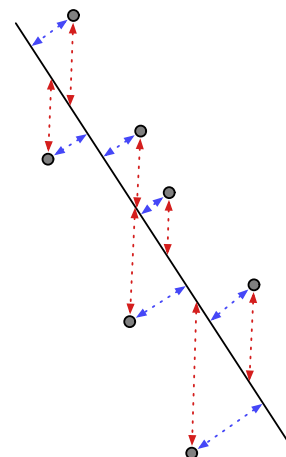


Figure 18: Laser scan points (gray dots), a proposed fitted line, and two different notions of error. The red dashed lines provide the vertical distance (which is used in linear regression). The blue dashed lines provide the perpendicular distance which is a more natural way to define line of best fit in the context of fitting lines to laser scan data.