ENPM673

Perception for Autonomous Robots — PROJECT-1

22 February 2023

Vamshi Kalavagunta

119126332

vamshik@umd.edu

A. James Clark School of Engineering

University of Maryland College Park - 20742

1 Problem 1

In the given video, a red ball is thrown against a wall. Assuming that the trajectory of the ball follows the equation of a parabola:

1. Detect and plot the pixel coordinates of the center point of the ball in the video.
(Hint: Read the video using OpenCV's inbuilt function. For each frame, filter the red channel)

2. Use Standard Least Squares to fit a curve to the extracted coordinates. For the estimated parabola you must,
a. Print the equation of the curve.
b. Plot the data with your best fit curve.

3. Assuming that the origin of the video is at the top-left of the frame as shown below, compute the x-coordinate of the ball's landing spot in pixels, if the y-coordinate of the landing spot is defined as 300 pixels greater than its first detected location.

1.1 Solution

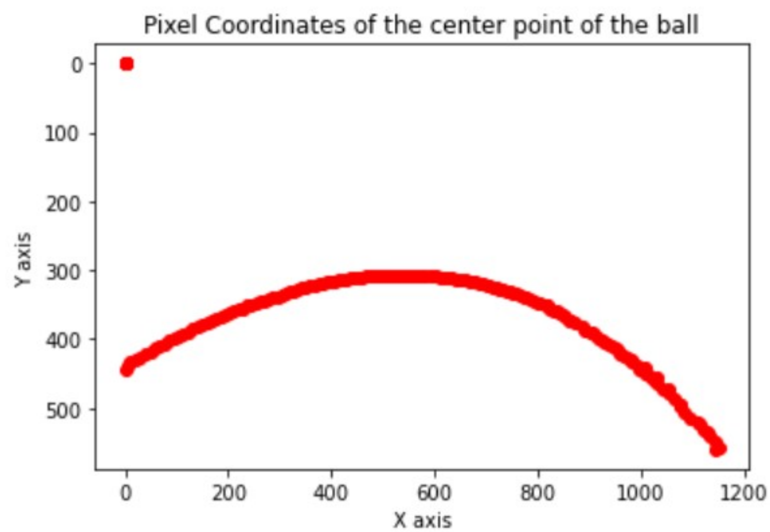
1. Detect and plot the pixel coordinates of the center point of the ball in the video:
   - Use OpenCV to read the video frames and apply a red color filter to each frame by masking to extract the pixels corresponding to the red ball.
   - Store the x and y coordinates of the center point in each frame.
2. Use Standard Least Squares to fit a curve to the extracted coordinates:
   - Use the x and y coordinates of the center point as the input data for the curve fitting algorithm.
   - Fit a parabolic curve to the data using the standard least squares method.
   - Print the equation of the curve in the form $y = ax^2 + bx + c$.
   - Plot the input data and the fitted curve.
3. Compute the x-coordinate of the ball's landing spot in pixels:
   - Use the equation of the fitted parabola to predict the x-coordinate of the landing spot for the given y-coordinate.
   - Plot the predicted landing spot on the same graph as the input data and the fitted curve.

Our objective here is to fit a parametric model that relates the equation of the parabola to a set of points. If we model the line as $y = ax2 + bx + c$, our aim is to estimate the parameters a, b, c so that the value of error gets minimized.
The total of residuals or how far off our estimate is for each point is represented by E. This might be seen as selecting a line in order to minimize the distance between the line and the observed data points.

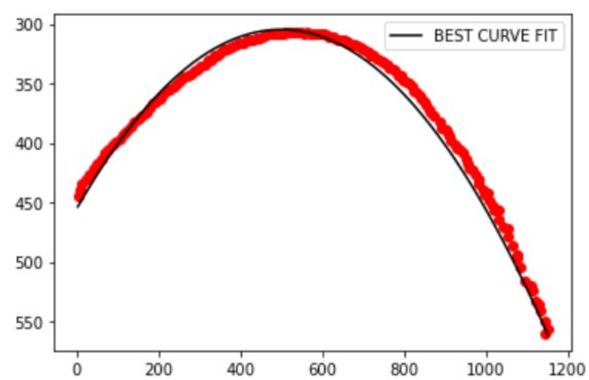
This is simply expressed as a least squares issue. $E = \sum_{i=0}^{n} \left(y_i - ax^2 - bx - c\right)$ denotes the objective function that we are attempting to minimize. This equation may be rewritten in the form $E = \|Y - XB\|^2$, where $\mathrm{B} = [\mathrm{abc}]^{\mathrm{T}}$ and $X = \begin{bmatrix} x^2 x & 1 \end{bmatrix}$ and Y is the vector holding all of our data's y values. Our objective is to discover the a, b, c parameters that minimize E. They may be derived by equating the derivative of E with respect to B to zero.

$\frac{dE}{dB} = -2X^TY + 2X^TXB = 0$ Solving this equation yields B, which may be represented as $B = \left(X^TX\right)^{-1} X^TY$. This solution may be understood as the best (in terms of least squares) parameters $a, b.c.$(Formulas are taken from class slides)

1.2 Proof for the solutions




```
Parabola equation: y = 0.0006033537799284626 x^2 + -0.6017913823053869 x + 454.86542586961576
```




```
The x-coordinate of theball landing spot is 1352.86 pixels
```

2 Problem 2

Given are two csv files, pc1.csv and pc2.csv, which contain noisy LIDAR point cloud data in the form of (x, y, z) coordinates of the ground plane.
1. Using pc1.csv: a. Compute the covariance matrix.
b. Assuming that the ground plane is flat, use the covariance matrix to compute the magnitude and direction of the surface normal.
2. In this question, you will be required to implement various estimation algorithms such as Standard Least Squares, Total Least Squares and RANSAC.
a. Using pc1.csv and pc2, fit a surface to the data using the standard least square method and the total least square method. Plot the results (the surface) for each method and explain your interpretation of the results.
b. Additionally, fit a surface to the data using RANSAC. You will need to write RANSAC code from scratch. Briefly explain all the steps of your solution, and the parameters used. Plot the output surface on the same graph as the data. Discuss which graph fitting method would be a better choice of outlier rejection.

2.1 Solution 2.1

- Variance measures the variation of a single random variable , whereas covariance is a measure of how much two random variables vary together . The formula for variance is given by

$$\sigma_x^2 = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})^2$$

  where n is the number of samples and $\bar{x}$ is the mean of the random variable x . The covariance $\sigma(x, y, z)$ of two random variables x, y and z is given by

$$\sigma(x, y, z) = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})(z_i - \bar{z})$$

  with n samples. The variance σ_x^2 of a random variable x can be also expressed as the covariance with itself by $\sigma(x, x)$

  Covariance Matrix: covariance matrix, which is a square matrix given by $C_{i,j} = \sigma(x_i, x_j)$

  The Calculation of the Covariance matrix can be expressed as:

$$C = \frac{1}{n-1}\sum_{i=1}^{n}\left(X_i - \bar{X}\right)\left(X_i - \bar{X}\right)^T$$

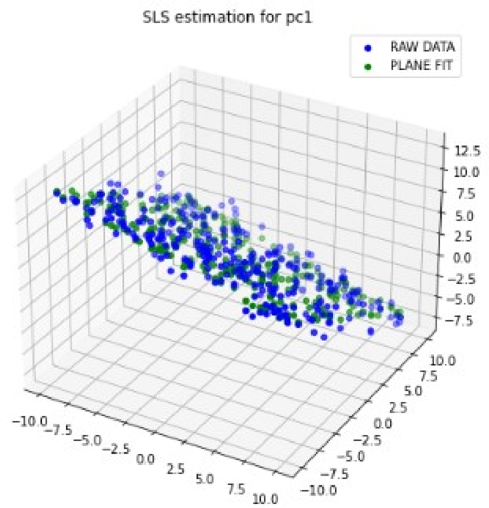
- The covariance matrix to compute the magnitude and direction of the surface normal is calculated by using minimum eigen value and its eigen vector.

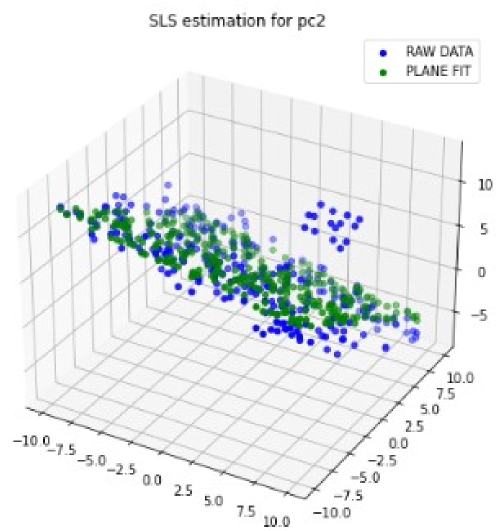
```
Covariance matrix:
[[ 33.6375584   -0.82238647 -11.3563684 ]
 [ -0.82238647  35.07487427 -23.15827057]
 [-11.3563684  -23.15827057  20.5588948 ]]
The eigenvector corresponding to the smallest eigenvalue is the direction of the surface normal
[ 0.28616428  0.90682723 -0.30947435]

Magnitude of surface normal: 0.6672780805108618
```

2.2 Solution 2.a

1. Standard least squares: the procedure of the standard least square is similar to the first question.
2. The outputs for the given data sets is given below





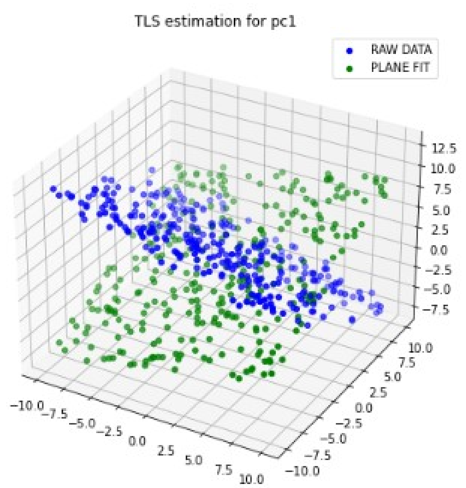
1. Total least squares: Total least squares is similar to sLS but instead of using vertical residuals , it uses diagonal residuals . Here, our goal is to parameterize a model that describes the equation of the line to a series of points. If we model the line as $ax + by = d$, we happen to avoid the case of infinite slope. Here our goal is to estimate the parameters a, b, d to minimize the value of E (error). E is the sum of residuals or how far off our estimate is for each point. It can be shown that minimizing E with respect to a, b, d is equivalent to solving the homogeneous system $Ah = 0$, where $h = \begin{bmatrix} a & b & d \end{bmatrix}^T$ and the matrix A collects all the coordinates of the data points. We can solve $Ah = 0$ using Singular Value Decomposition (SVD).

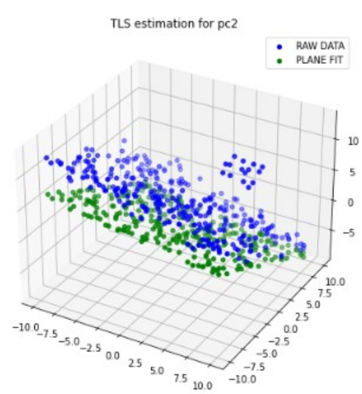
   When we take the SVD of a matrix, we are separating the matrix into three parts, U, Σ , and V.

$$A = U\Sigma V^T$$

   Here $U = AA^T, V = A^T A$ and $\Sigma = diag\left(\sigma_1, \sigma_2 \ldots\right)$. Where $\sigma = \sqrt{\lambda}$ and λ is the eigen value of AA^T or $A^T A$ (both yield the same result).

2. The outputs for the given data sets is given below





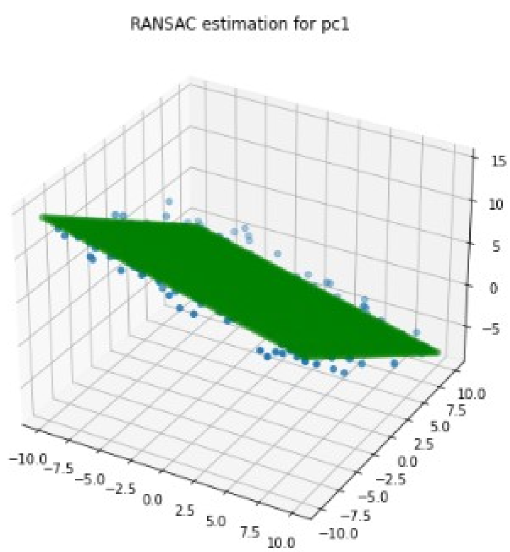
1. RANSAC:Random sample consensus, or RANSAC, is an iterative method for estimating a mathematical model from a data set that contains outliers. The RANSAC algorithm works by identifying the outliers in a data set and estimating the desired model using data that does not contain outliers.
   The number of iterations, N, is chosen high enough to ensure that the probability p (usually set to 0.99) that at least one of the sets of random samples does not include an outlier. Let u represent the probability that any selected data point is an inlier and $v = 1 - u$ is the probability of observing an outlier. Now N iterations of the minimum number of points (m) are required. Where,

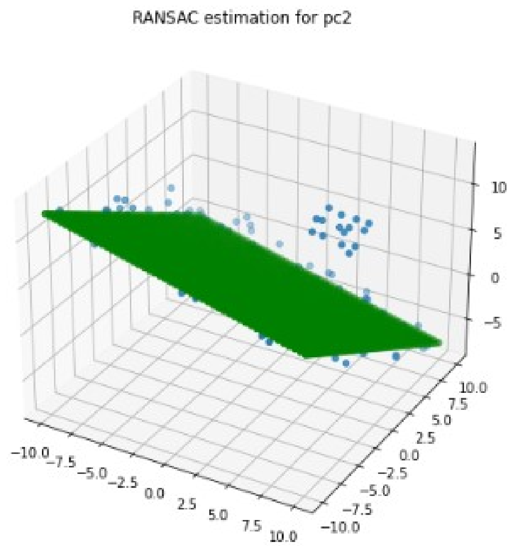
$$1 - p = \left((1 - u)^m\right)^N$$

   Now after doing some manipulation, we can write the equation in the manner of getting N

$$N = \frac{\log(1 - p)}{\log\left(1 - (1 - v)^m\right)}$$

2. The outputs for the given data sets is given below





2.3 Solution 2.b

1. Based on the distribution of data in point clouds, the Standard Least Square Fitting worked well with both noisy data (outliers) and noiseless data. the disadvantages are SLS fails for vertical lines,
2. The Total Least Square Fitting made little difference for noiseless data, but it created a better plane for noisy data when dealing with outliers.
3. RANSAC could not perform as well as TLS and SLS for noiseless data because the surface plane fit could not fit across data points as in TLS and SLS. Yet, given noisy data, RANSAC gave a better plane fit by removing outliers. The biggest issue with RANSAC is that it requires additional tests and more adjustment because of random samples.