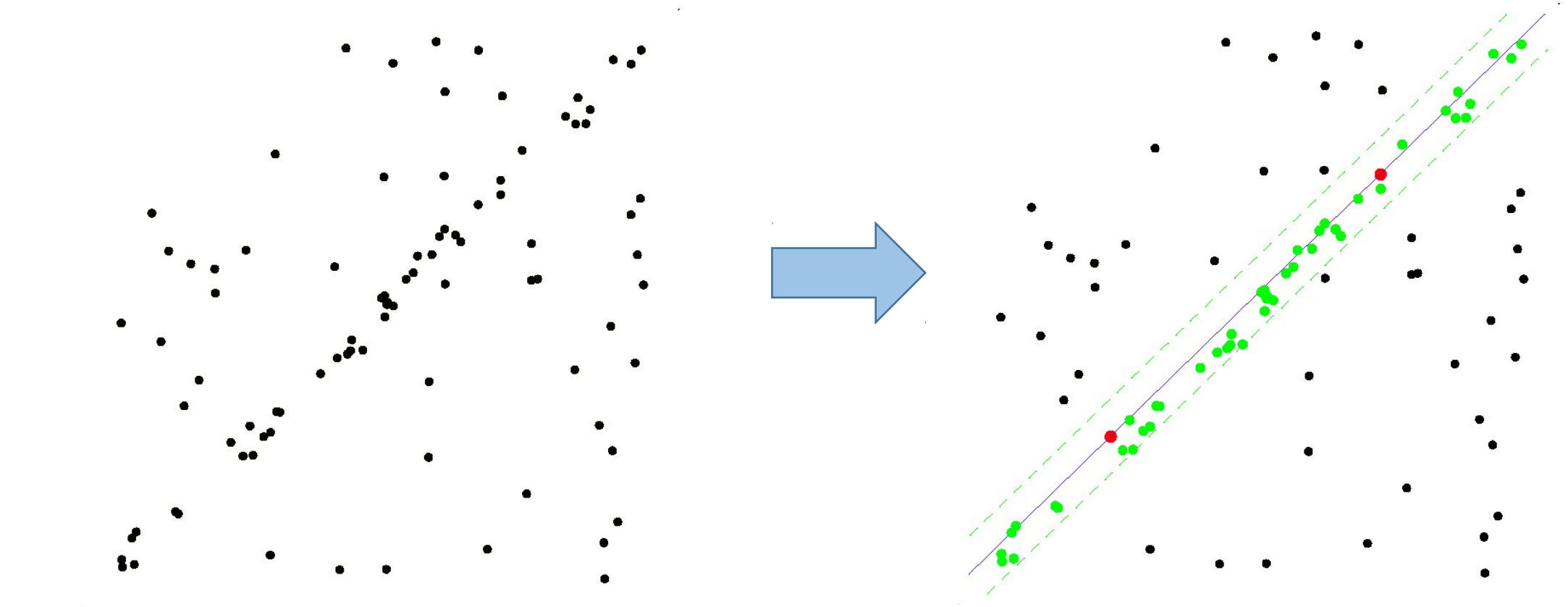
Home assignment #4(a)

Home assignment 4(a): Generate a noisy dataset with a line and detet the line using RANSAC algorithm

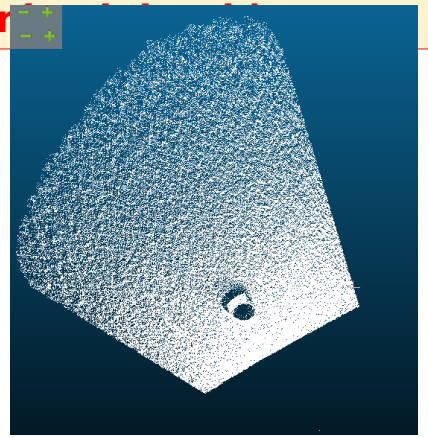
- **Generate a 2D dataset with a noisy line (using e.g. Gauss noise or a noise of another distribution)**
- **Create a code in Python (or use an existing library) that detects the line parameters (e.g., a and b coefficients of**
  y = ax + b line model) using RANSAC algorithm
- **Visualize the RANSAC-based line detection**



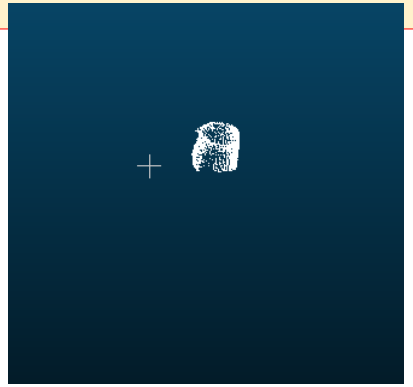
Home assignment #4(b)

Home assignment 4(b): Detect & visualize an object from 3D point cloud by eliminating the environment, and estimate the distance to the object (the minimal depth measurement)

- **Record (Download or Generate) indoor 3D sensor dataset in a room environment with an object (tips: you can measure a real 3D Kinect/3D LIDAR dataset at the robotics lab, or find a 3D Kinect/3D LIDAR dataset in the Internet, or Simulate your own dataset in a Robotic Simulator, e.g. Gazebo/ROS, Webots, CoppeliaSim, or just model the noisy 3D point cloud indoor environment with an object in Python)**
- **Detect & Remove the redundant 3D point clouds that belong to environment (e.g. by filtering the depth with setting the reasonable thresholds on minimal and maximal range)**
- **Detect & Remove the redundant 3D point clouds that belong to a floor (using a plane detector based on RANSAC)**
- **Mark the object in the 3D point cloud (changing the object's color), and estimate the distance to the object (the mi**



3D Point Cloud received from Kinect 2 sensor



The object (a cylinder) extracted from 3D point cloud