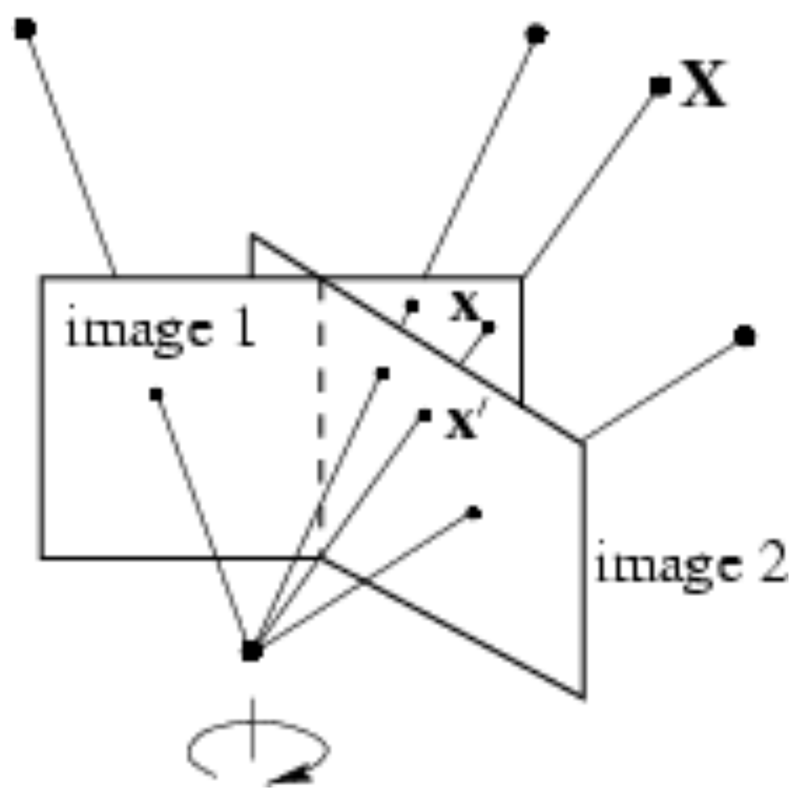
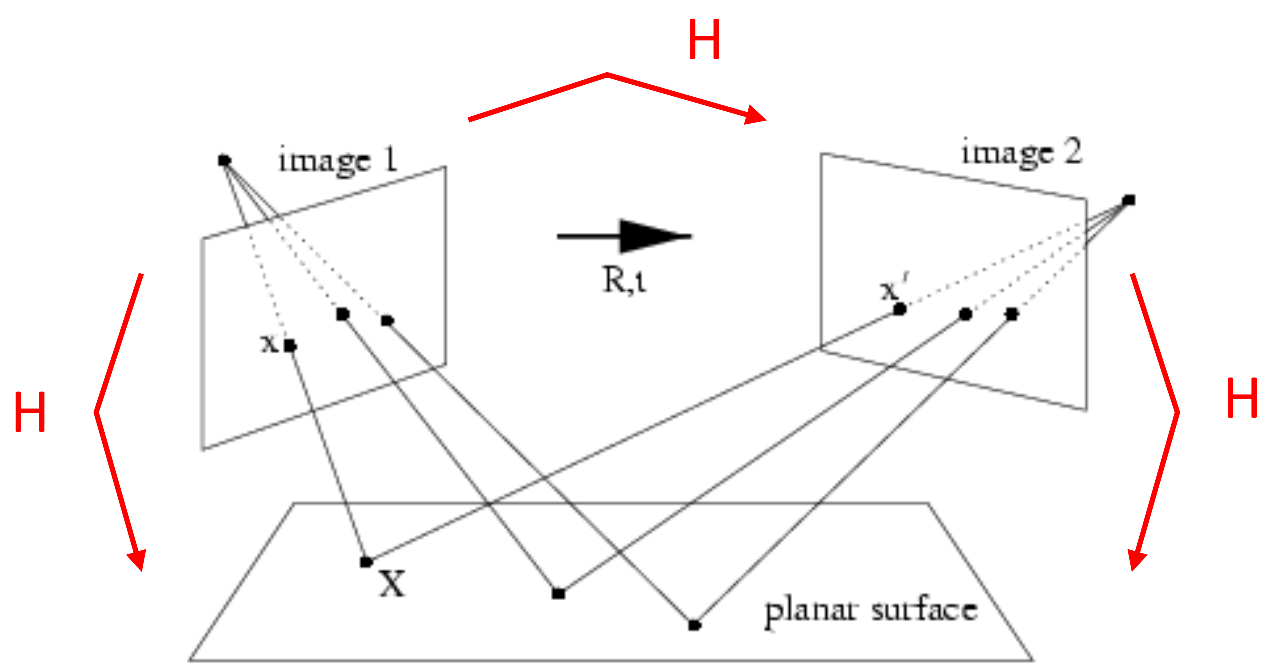
Robust Fitting

Recall homography estimation

When can we use a homography to relate image pairs?



rotating camera, arbitrary world



arbitrary camera motion, planar world
(i.e., as we assume for Duckietown)

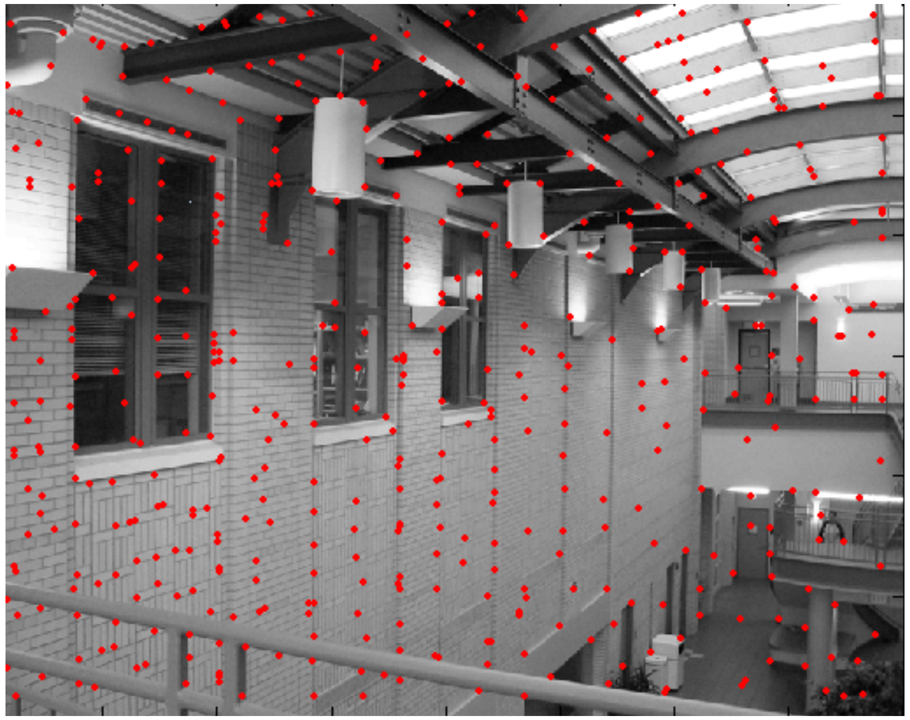
Recall homography estimation

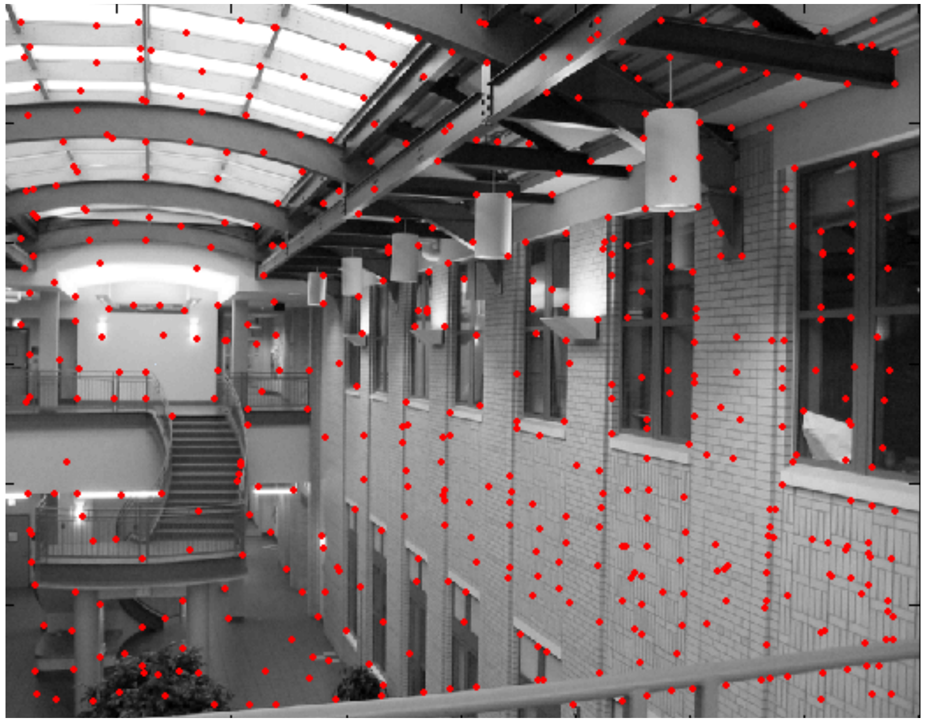
Set of n 3D <-> 2D point correspondences
(2 equations for each point) $\longrightarrow$ $A\mathbf{h} = \mathbf{0}$

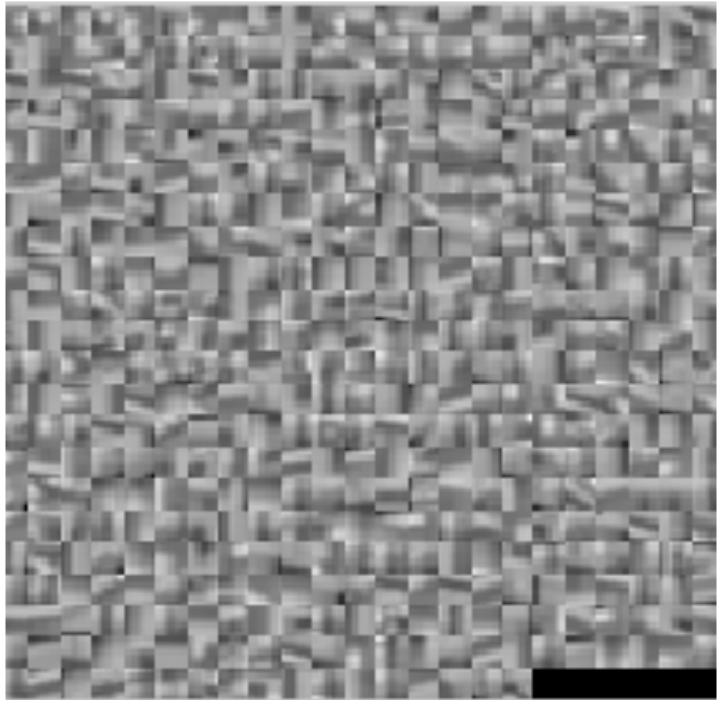
vector form of homography matrix H

- Over-determined case:
  - Find the “best fit” solution
  - $argmin_h \|Ah\| s.t. \|h\| = 1$
  - Minimized by unit vector corresponding to lowest eigenvalue of $A^\top A$
- Alternatively, we may want to minimize reprojection error (geometric distance) via nonlinear optimization

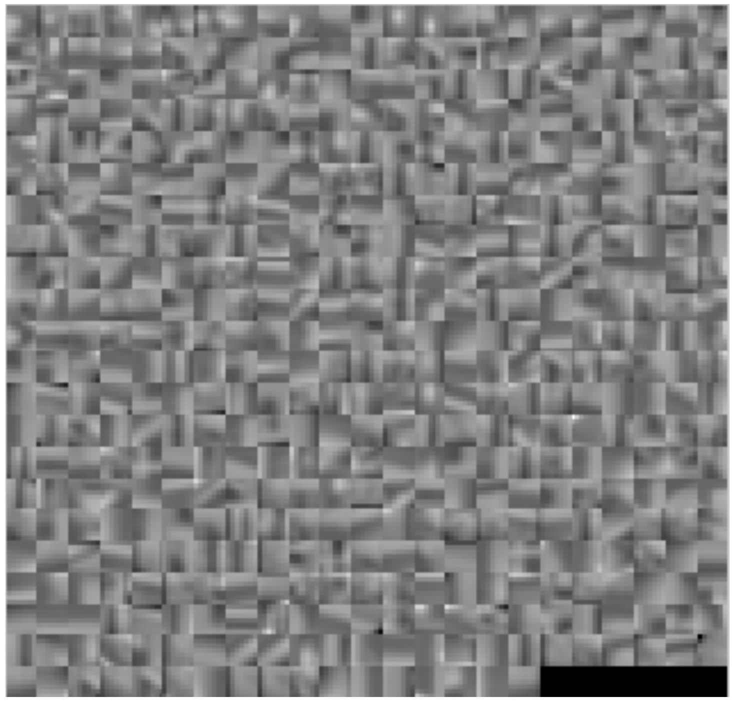
Recall homography estimation





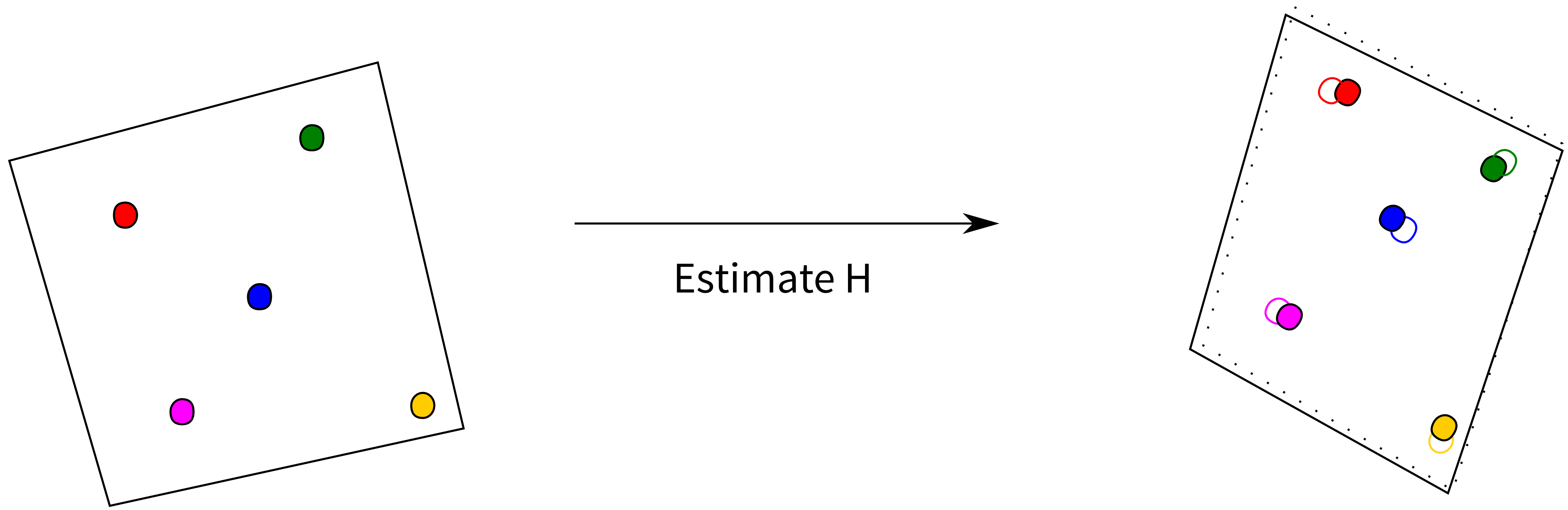


descriptors from first image



descriptors from second image

Recall homography estimation



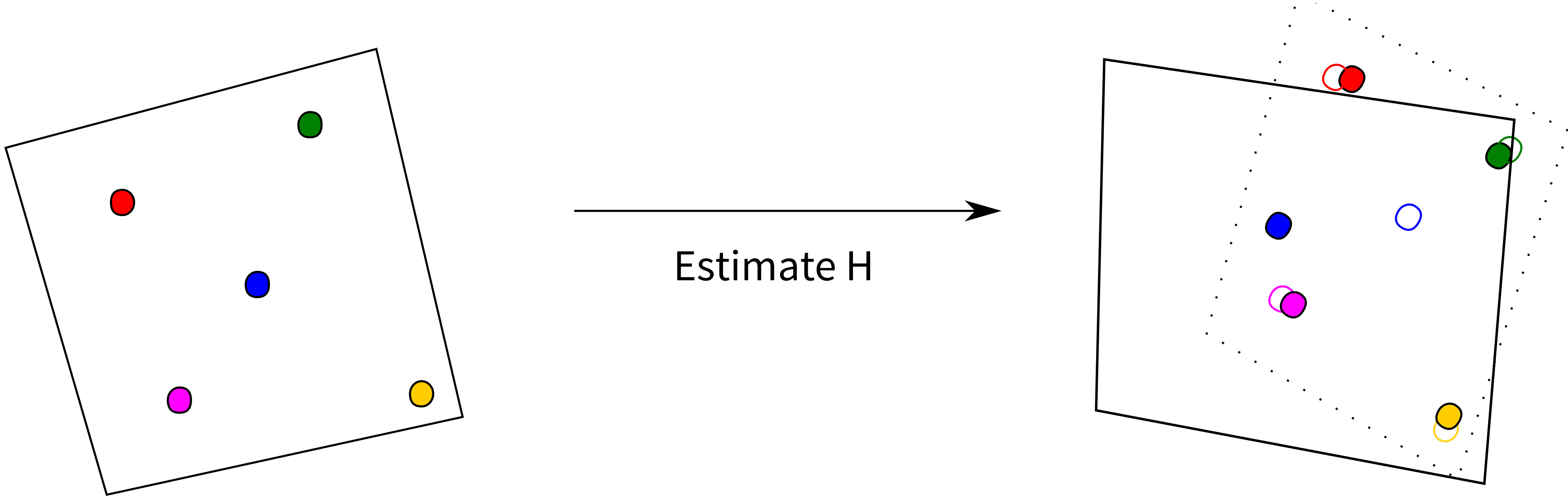
In practice, correspondences will be noisy

- (feature) detection isn't perfect
- descriptors may be aliased

Small perturbations in correspondences lead to small changes in tomography estimate

Recall homography estimation

In practice, there will be errors
(one correspondence is wrong)



In practice, correspondences will be noisy

- (feature) detection isn't perfect
- descriptors may be aliased

Minimizing the reprojection error across all correspondences significantly skews the estimate

Robust estimation

- **Goal**: Estimate parameters of a function (***model-fitting***) from a given set of points
  - Determine homography from point-pairs
  - Fit lines (or segments) to a set of detected edge points
- Without any noise/errors, there are closed-form solutions for some problems and straightforward solutions for others
- But …. there will always be noise and point correspondences may be wrong
- We would like a solution that is robust to these effects

Robust estimation

- For a set of samples C, minimize error (e.g., reprojection error)

$$h = \arg\min_h \sum_{i \in C} E_i(h)$$

- Robust version:

$$h = \arg\min_h \sum_{i \in C} \min(E_i(h), \epsilon)$$

- Minimizes the influence of outliers
- If we knew C, we could have $E_i > \epsilon$ for the correct h, and solve as normal
- But, we don't know C

Robust estimation: Iterative method

$$h = \arg\min_h \sum_{i \in C} \min(E_i(h), \epsilon)$$

1. Find best h for all samples in full set C
2. Given the current estimate of h, compute the new inlier set $C' = i : E_i(h) < \epsilon$
3. Update estimate of h by minimizing over only the inlier set C'
4. Return to Step 2, stopping when error no longer decreases (perhaps oscillates)

- Guaranteed to converge, but possibly to a local optima
- Only way to solve exactly involves looking at a combinatorial number of sets

RANSAC (RANdom Sampling and Consensus)

1. Randomly select k points (e.g., correspondences) as inlier set ($k \geq 4$ for homographies)

2. Fit model (h) to this inlier set

3. Check whether (subset of) other points agree with the model estimate

4. Store model and cost (e.g., number of outliers)

- Repeat this step to get N different models and associated costs

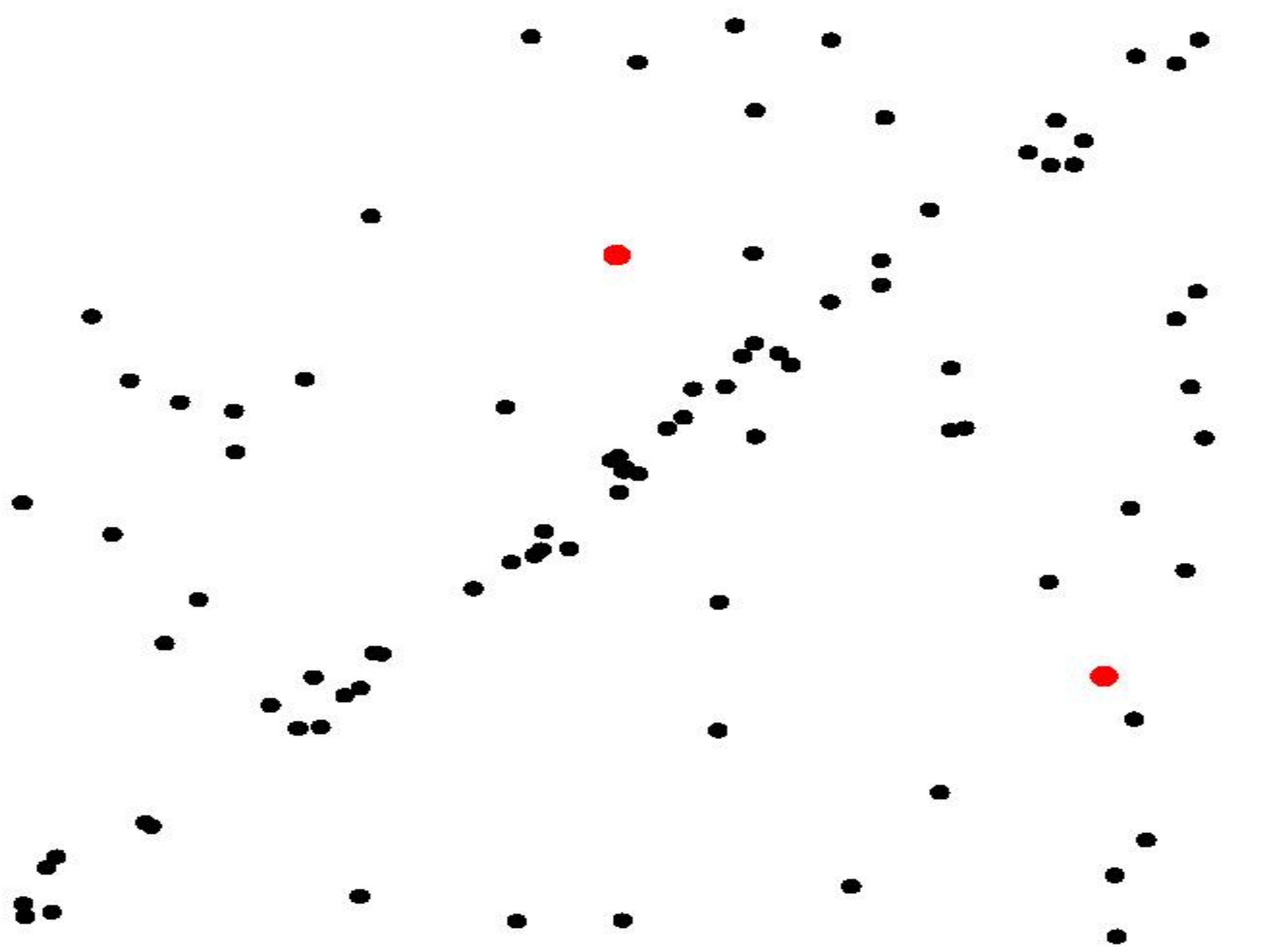
- Choose model with the lowest cost and then refine model using iterative algorithm or least squares using new inliers

- We can model this probabilistically to determine number of models N

Example: Line fitting

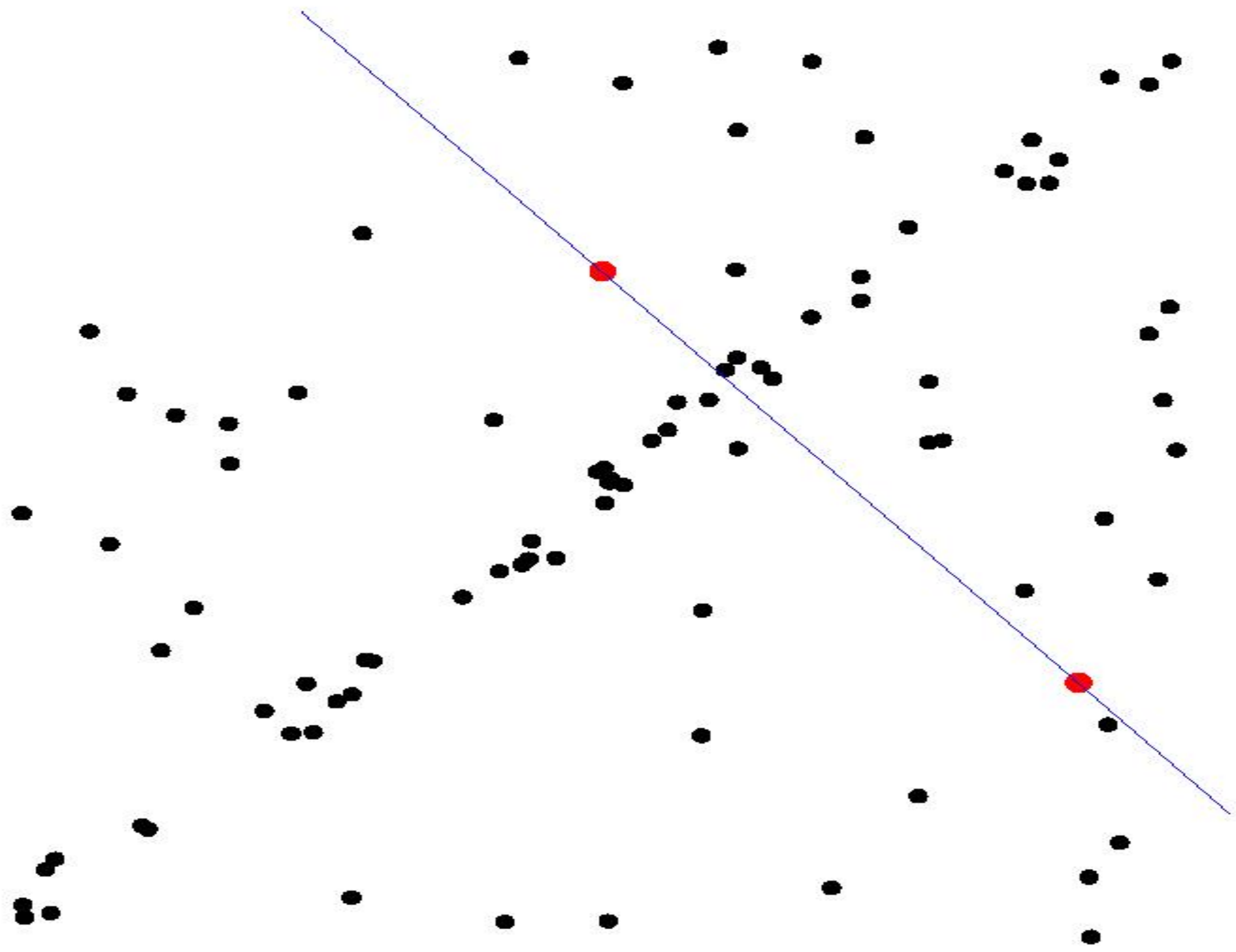
Example: Line fitting

- Select 2 points at random as the inlier set



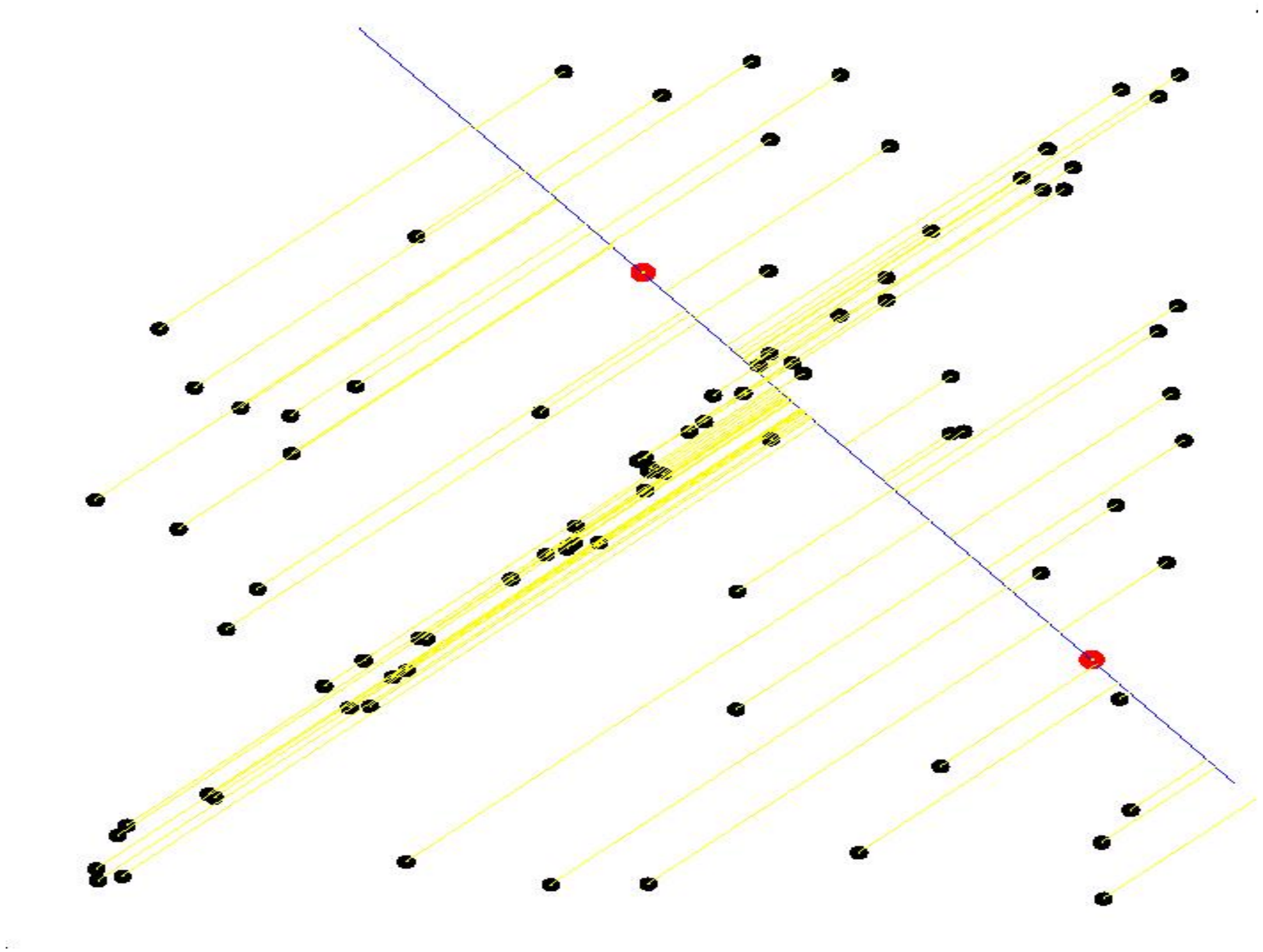
Example: Line fitting

- Select 2 points at random as the inlier set
- Estimate the model parameters using the inlier set



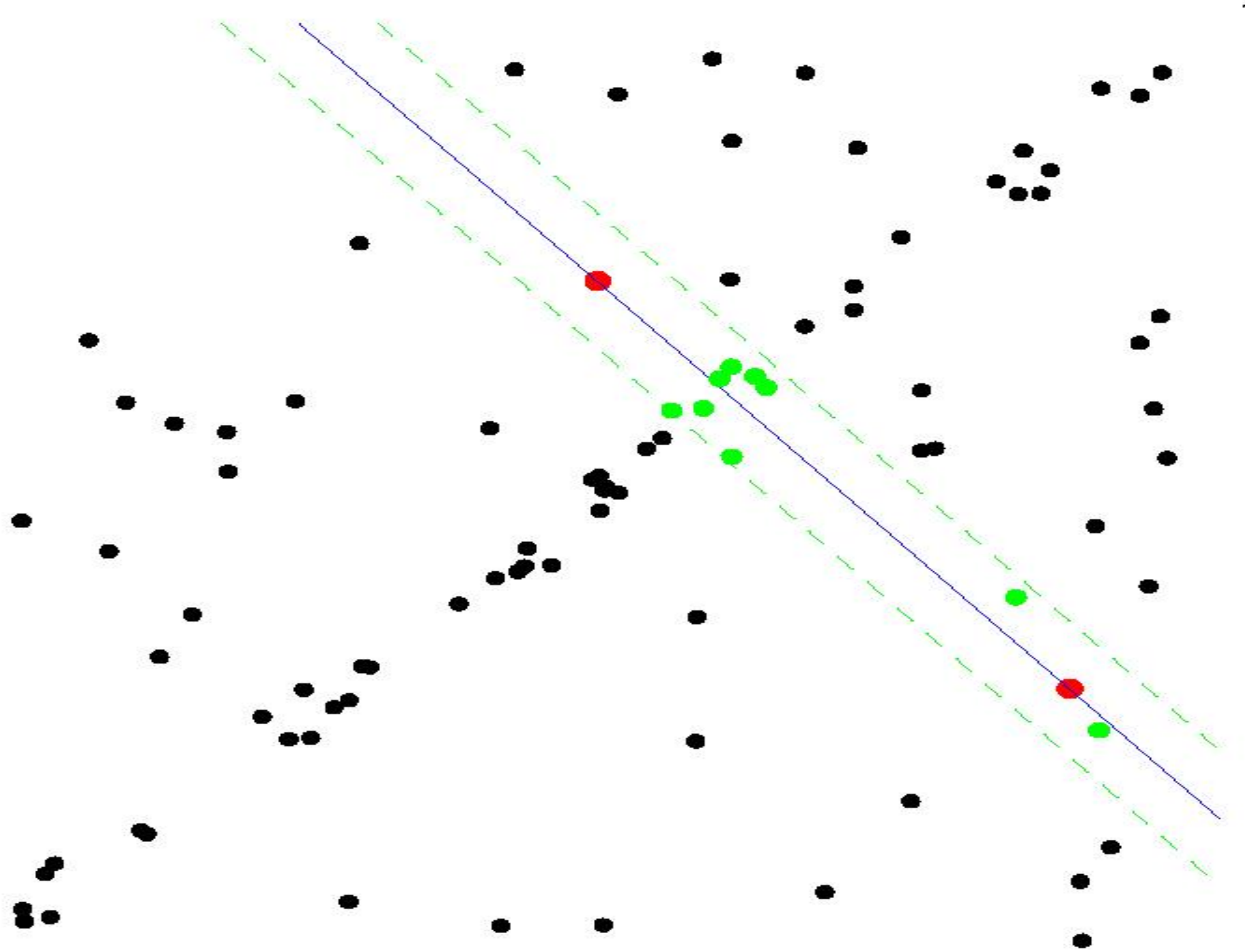
Example: Line fitting

- Select 2 points at random as the inlier set
- Estimate the model parameters using the inlier set
- Compute model error for (a subset of) remaining points



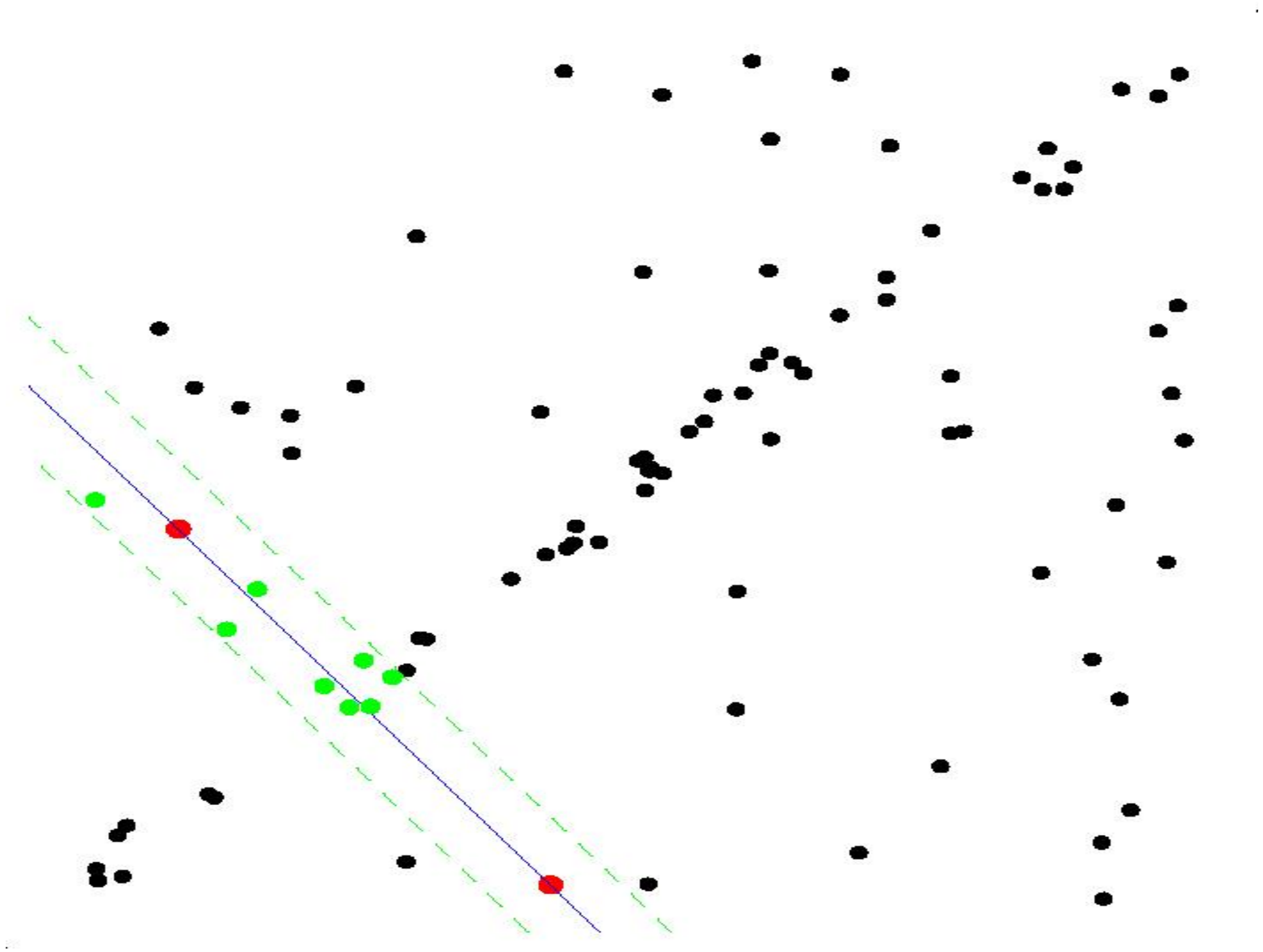
Example: Line fitting

- Select 2 points at random as the inlier set
- Estimate the model parameters using the inlier set
- Compute model error for (a subset of) remaining points
- Select points that support model estimate



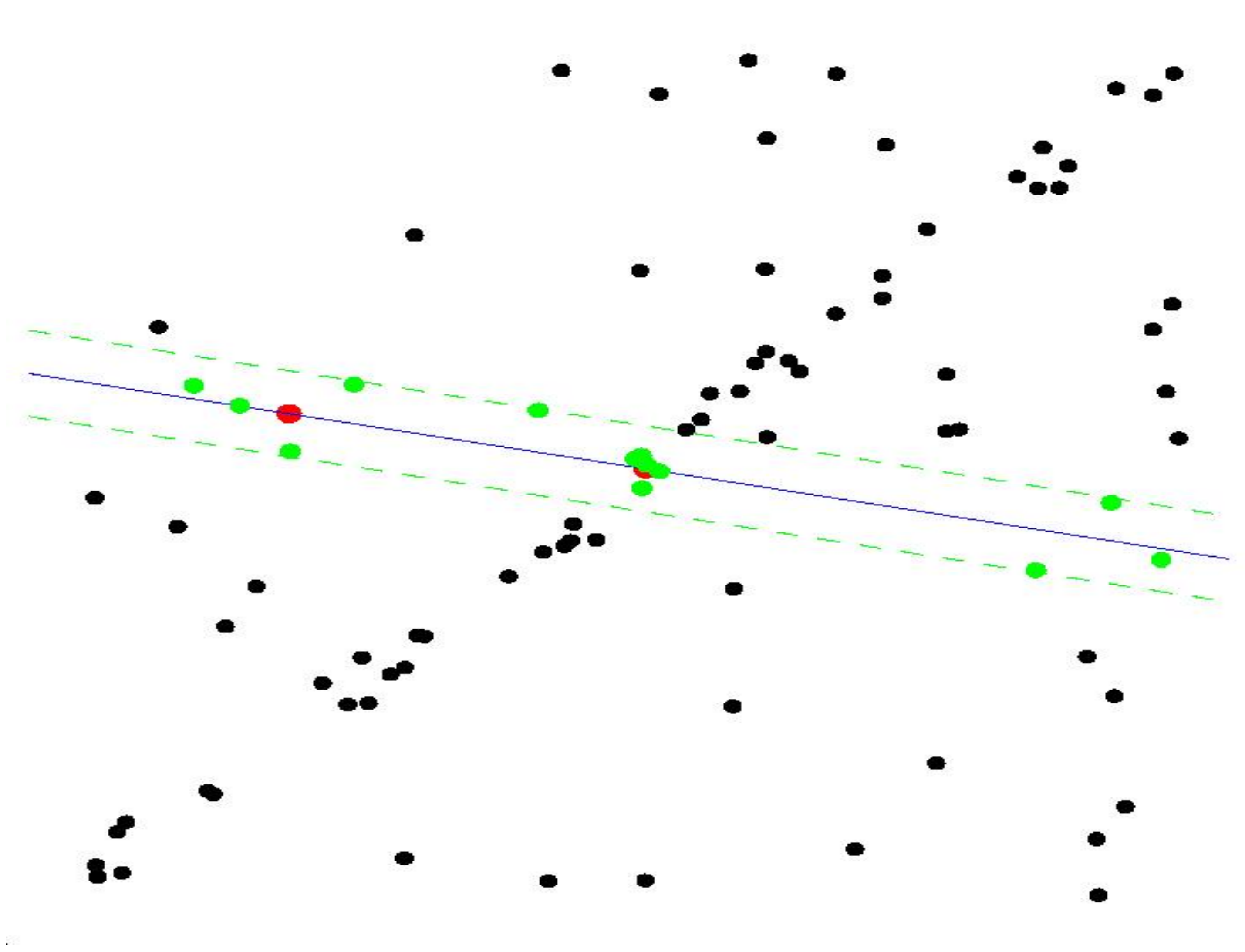
Example: Line fitting

- Select 2 points at random as the inlier set
- Estimate the model parameters using the inlier set
- Compute model error for (a subset of) remaining points
- Select points that support model estimate
- Repeat for a new sample set



Example: Line fitting

- Select 2 points at random as the inlier set
- Estimate the model parameters using the inlier set
- Compute model error for (a subset of) remaining points
- Select points that support model estimate
- Repeat for a new sample set



Example: Line fitting

- Select model with highest score (maximum number of inliers)
- Robustly estimate model using new inlier set (least squares)