

Robust Fitting

Mathematical Models and Methods for Image Processing

Giacomo Boracchi

<https://boracchi.faculty.polimi.it/>

May 24th 2022

Fitting geometric primitives
is ubiquitous in Computer Vision

Example: Line Fitting for Vanishing Point Estimation



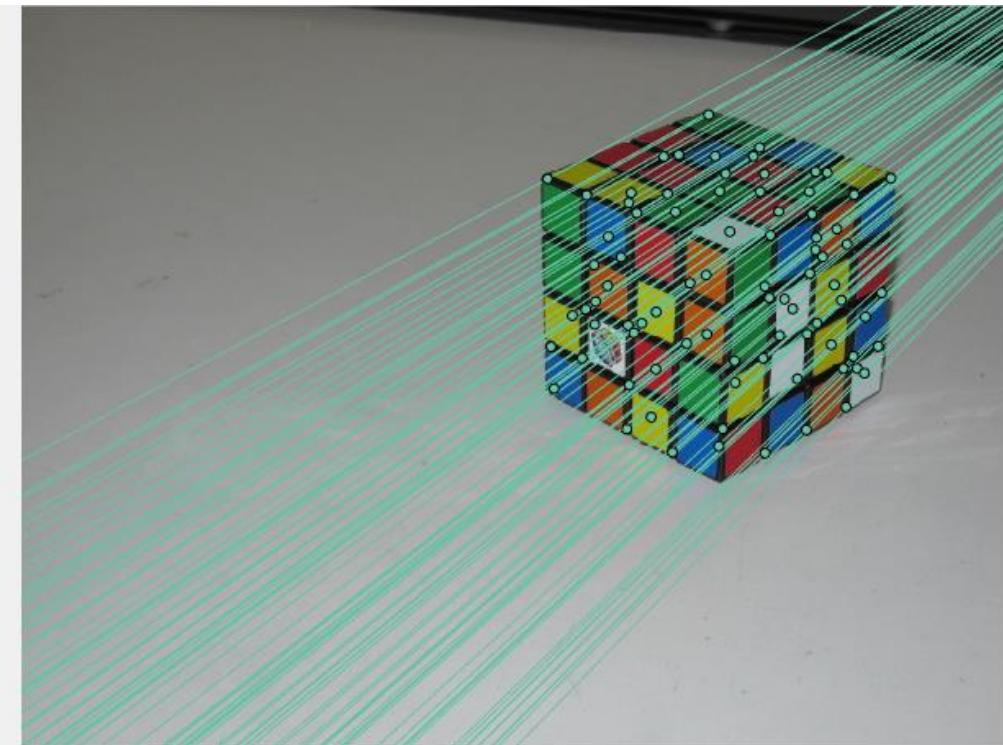
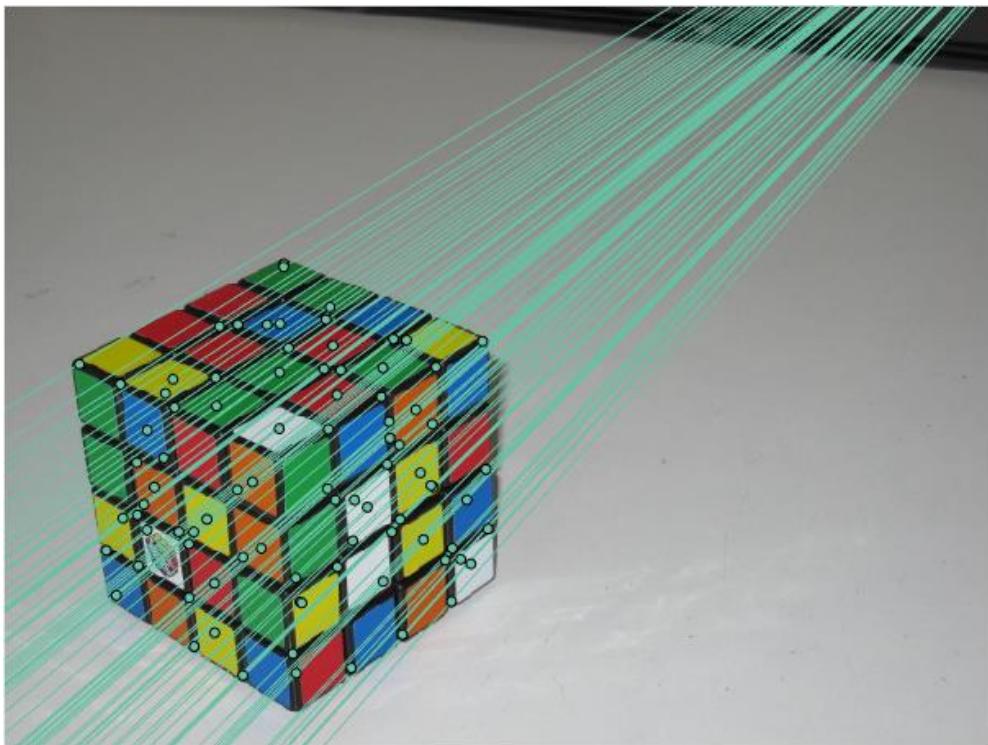
Example: Conic Fitting



Example: Estimating Homographic Transformations



Example: Estimating Fundamental Matrix



In all these cases the problem boils down
to fitting a parametric model to
(presumably) noisy input data

Ordinary Least Square

All these problems boils down to..

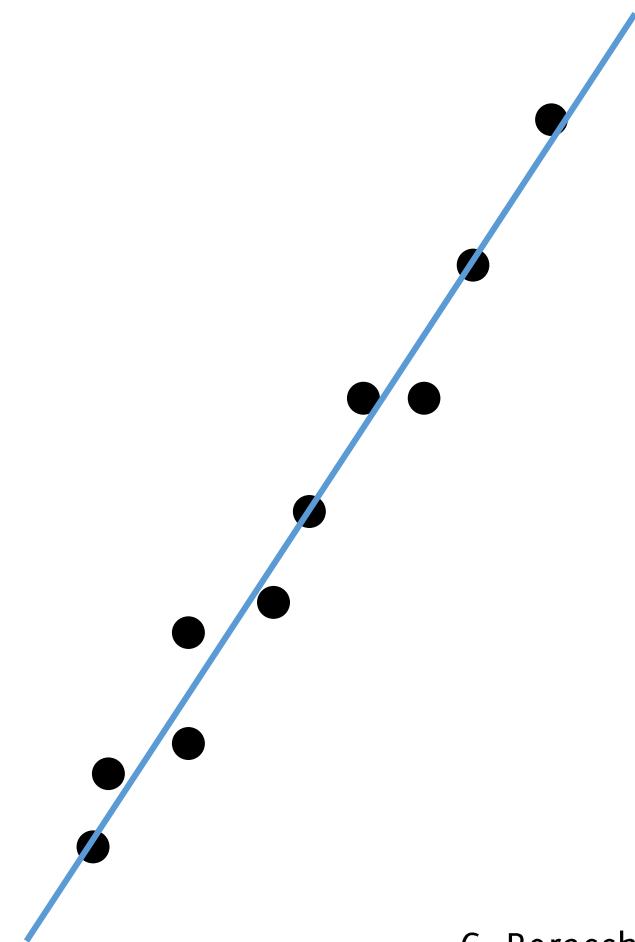
Given a set of N points (or matches..)

$$X = \{(x_1, y_1), \dots, (x_N, y_N)\}$$

Given a parametric model

$$y = mx + q$$

Estimate the parameters m, q yielding the
best fit



Least Square Regression

Given a set of N points (or matches..)

$$X = \{(x_1, y_1), \dots, (x_N, y_N)\}$$

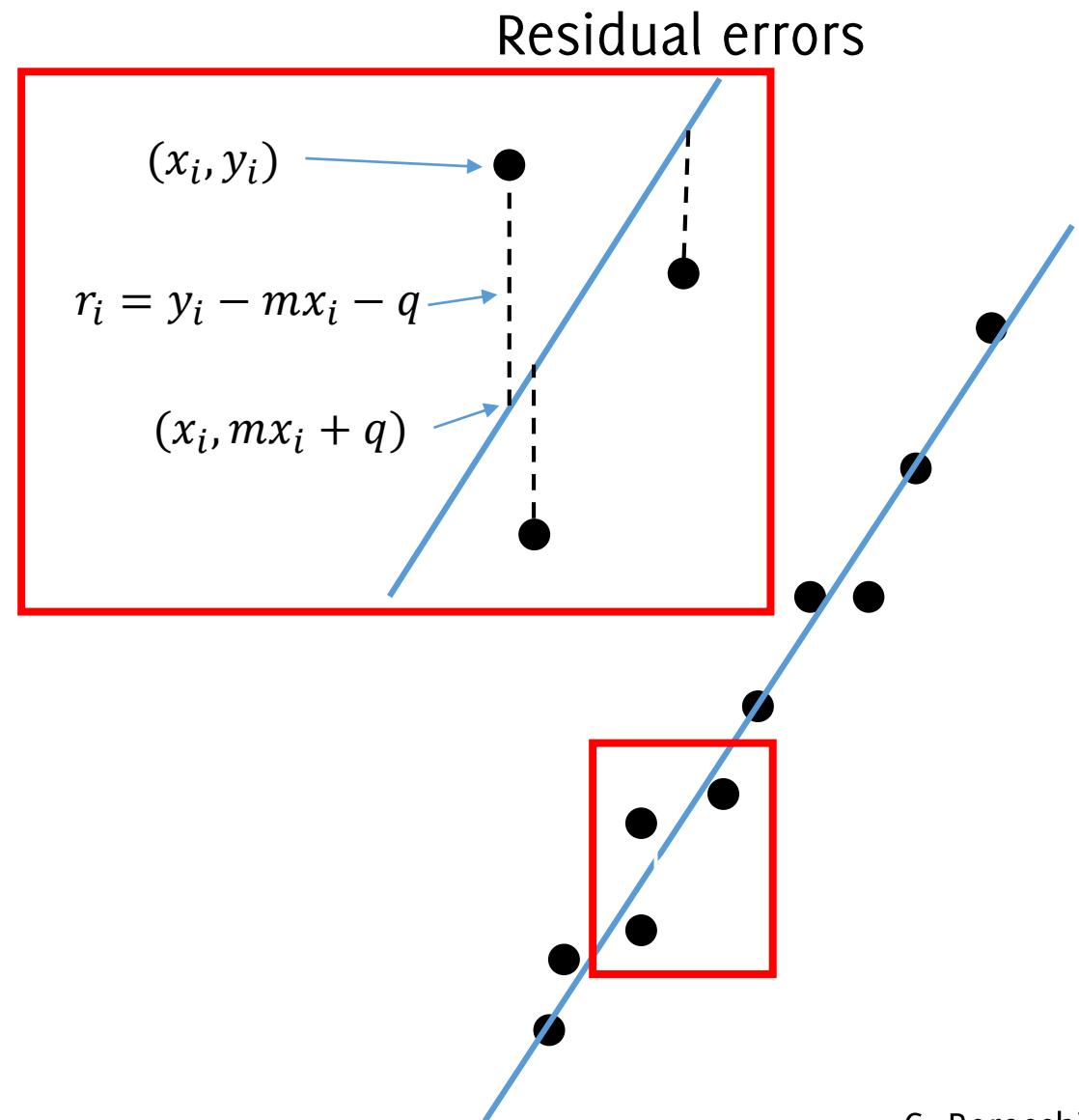
Given a parametric model

$$y = mx + q$$

Estimate the parameters m, q yielding the
best fit

The best fit is the one **minimizing some
residual error over the whole data**

$$r_i = y_i - mx_i - q$$



Ordinary Least Square (OLS) Regression

The loss function is the sum of squared residual errors

$$E = \sum_{i=1}^N (r_i)^2 = \sum_{i=1}^N (y_i - mx_i - q)^2 =$$

Which in matrix form becomes

$$r_i = y_i - [x_i \ 1] \begin{bmatrix} m \\ q \end{bmatrix}, \quad i = 1, \dots, N$$

$$E = \left\| \begin{bmatrix} y_1 \\ \vdots \\ y_N \end{bmatrix} - \begin{bmatrix} x_1 & 1 \\ \vdots & \ddots \\ x_N & 1 \end{bmatrix} \begin{bmatrix} m \\ q \end{bmatrix} \right\|_2^2$$

Then, the solution can be computed via least square regression

$$[\hat{m}, \hat{q}] = \underset{m,q}{\operatorname{argmin}} \left\| Y - X \begin{bmatrix} m \\ q \end{bmatrix} \right\|_2^2$$

Ordinary Least Square (OLS) Regression

OLS consists in solving the following problem

$$[\hat{m}, \hat{q}] = \underset{m,q}{\operatorname{argmin}} \frac{1}{2} \|Y - X \begin{bmatrix} m \\ q \end{bmatrix}\|_2^2$$

by zeroing the derivative of the loss function

$$\frac{\partial}{\partial \theta} \frac{1}{2} \|Y - X\theta\|_2^2 = 0, \quad \theta = \begin{bmatrix} m \\ q \end{bmatrix}$$

This follows from matrix calculus

$$\frac{\partial}{\partial \theta} \frac{1}{2} \|Y - X\theta\|_2^2 = \frac{1}{2} 2X^\top(X\theta - Y)$$

Thus the solution becomes

$$X^\top(X\hat{\theta} - Y) = 0 \rightarrow \hat{\theta} = (X^\top X)^{-1} X^\top Y$$

**This error does not make sense
when the line is vertical**

What about minimizing point-line distance?

Given a set of N points (or matches..)

$$X = \{(x_1, y_1), \dots, (x_N, y_N)\}$$

Given a parametric model

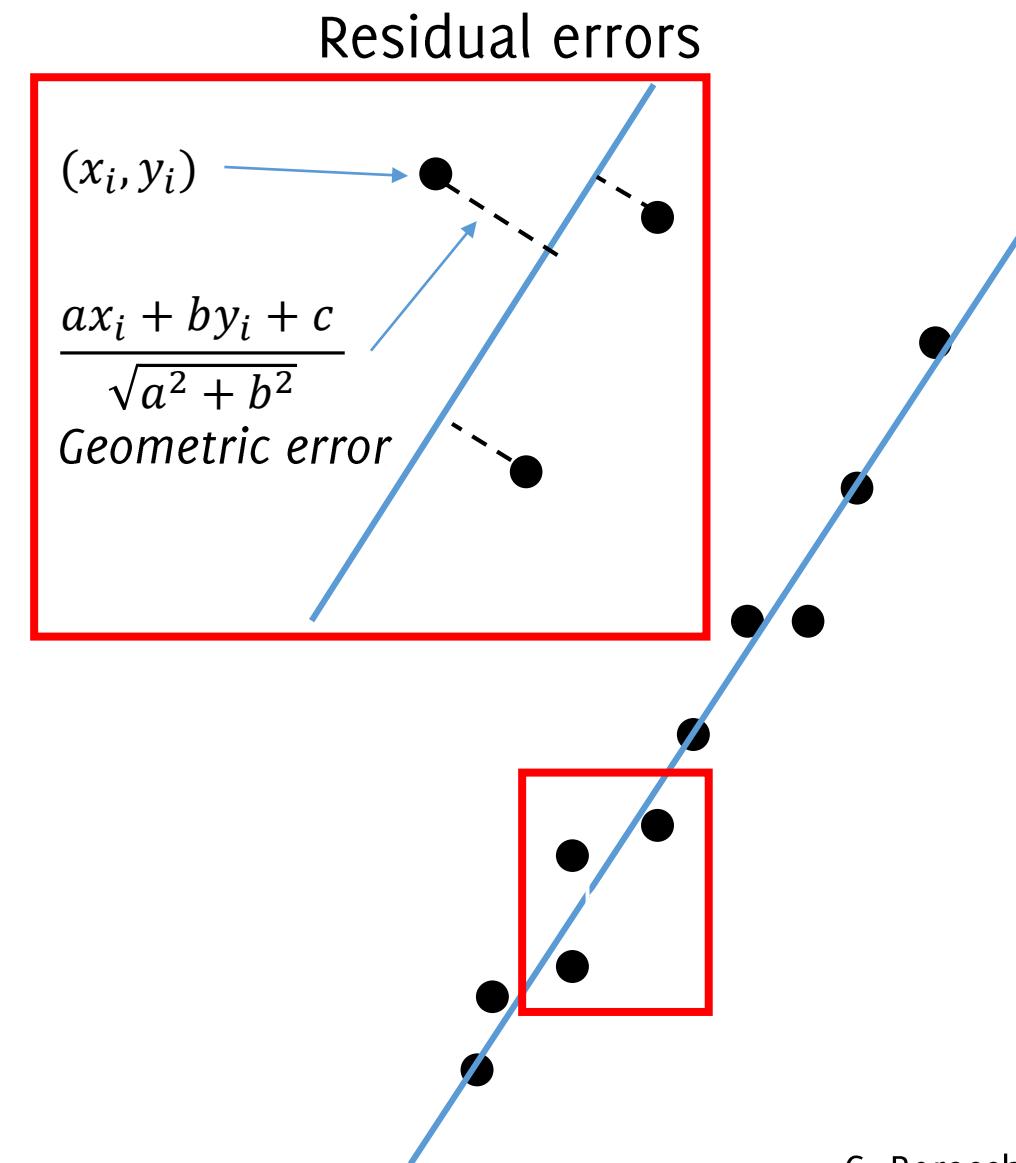
$$ax + by + c = 0$$

Estimate the line parameters a, b, c yielding
the **best fit** minimizing the residual error

$$E = \sum_{i=1}^N (ax_i + by_i + c)^2$$

If we take

$$r_i = ax_i + by_i + c$$



What about minimizing point-line distance?

Given a set of N points (or matches..)

$$X = \{(x_1, y_1), \dots, (x_N, y_N)\}$$

Given a parametric model

$$ax + by + c = 0$$

Estimate the line parameters a, b, c yielding the best fit minimizing the residual error

$$E = \sum_{i=1}^N (ax_i + by_i + c)^2$$

What about minimizing point-line distance?

$$E = \left\| \begin{bmatrix} x_1 & y_1 & 1 \\ \dots \\ x_N & y_N & 1 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \end{bmatrix} \right\|_2^2$$
$$\hat{\theta} = \operatorname{argmin}_{\theta} \|A\theta\|_2^2,$$

Being the parameter vector $\theta = [a; b; c]$ and constraining $\|\theta\|_2 = 1$ due to the equivalence relation between θ and lines

$$\hat{\theta} = \operatorname{argmin}_{\theta} \|A\theta\|_2^2, \text{ subject to } \|\theta\|_2 = 1$$

The DLT solves this problem by minimizing the algebraic error!

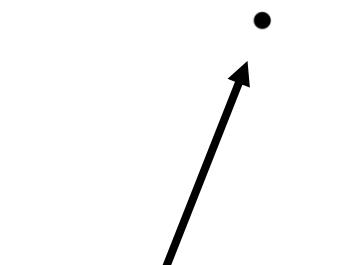
$$\theta = V(:, \text{end}), \text{ being } A = UDV^\top$$

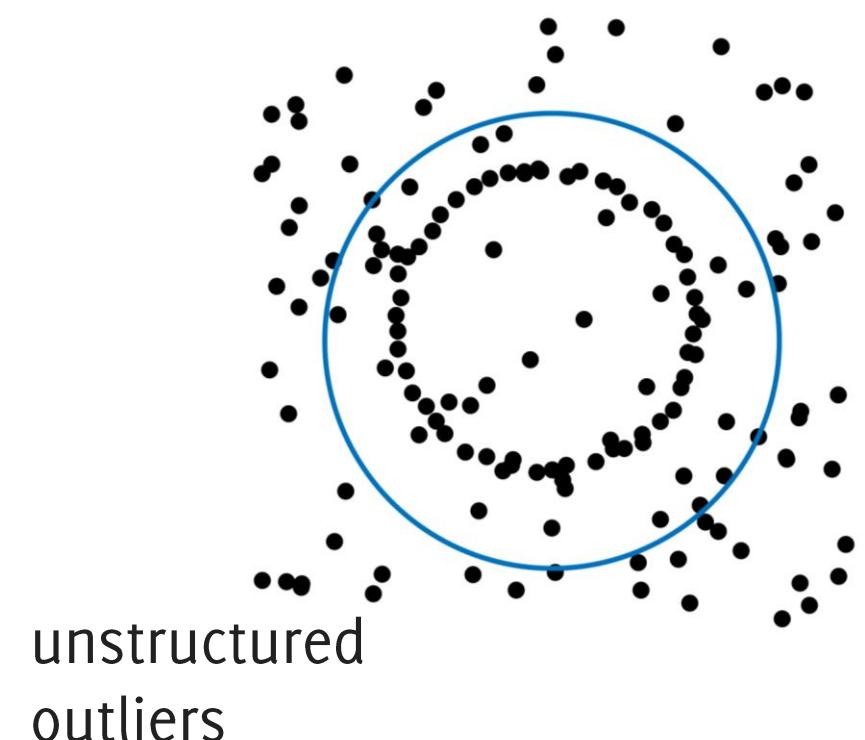
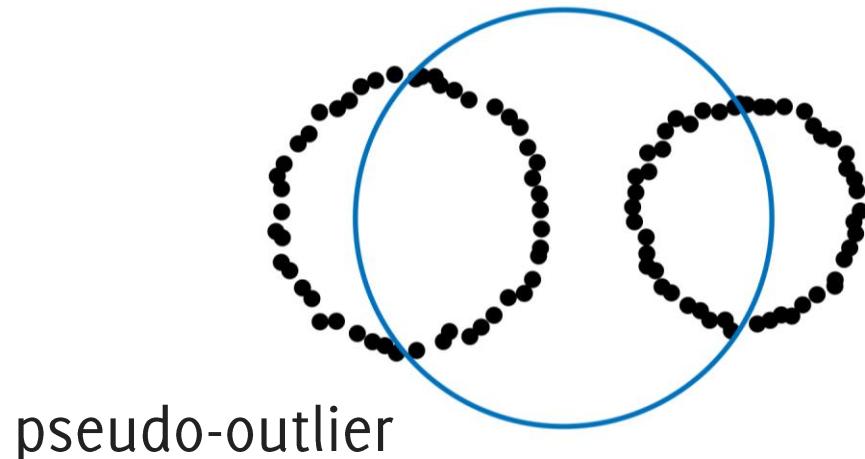
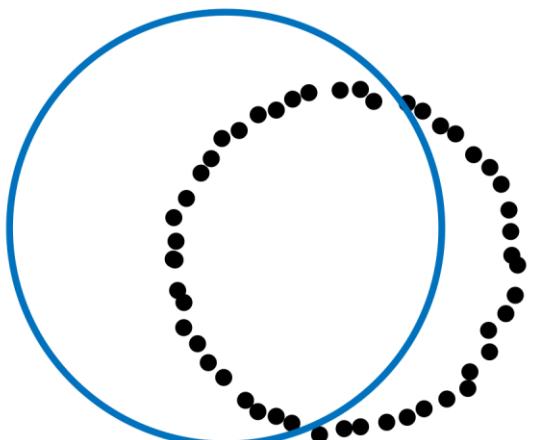
Robustness to Outliers

Least squares breaks down

Break down point: the proportion of incorrect observations that can be handled before giving an arbitrarily large uncorrect result.

Least squares has 0% breakdown point (the outlier might be arbitrarily large, i.e., $\rightarrow \infty$)


single outlier

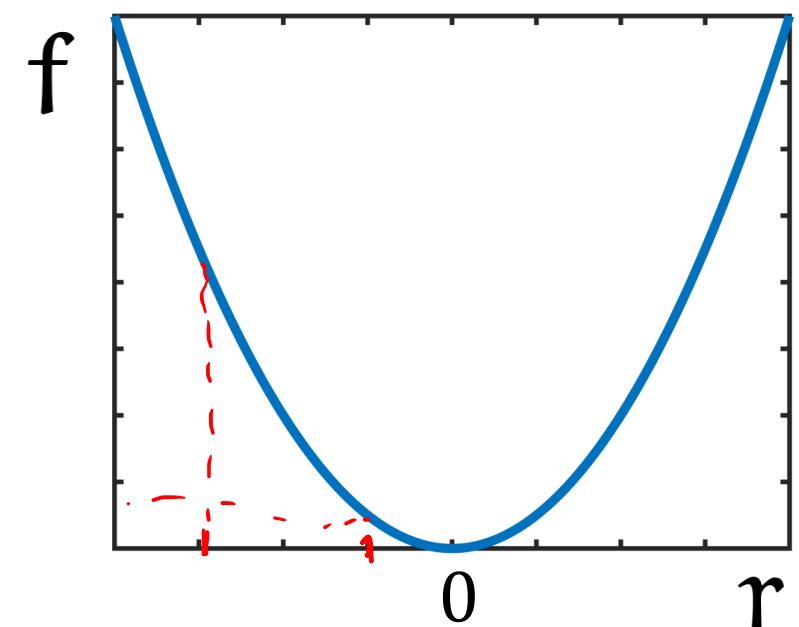


The loss function in OLS

The loss so far is the sum of a function of all the residuals

$$E = \sum_{i=1}^N f(r_i), \text{ where } f(r_i) = r_i^2$$

However, other options for f are viable, giving rise to different loss functions, and different results

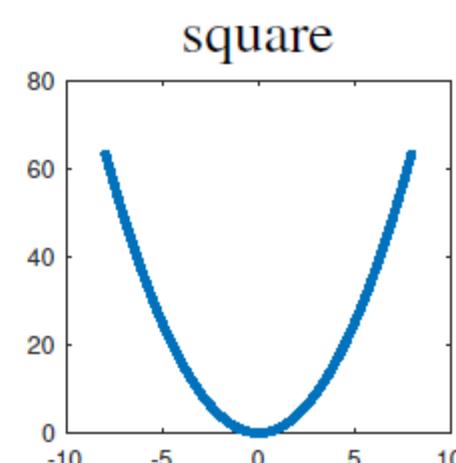
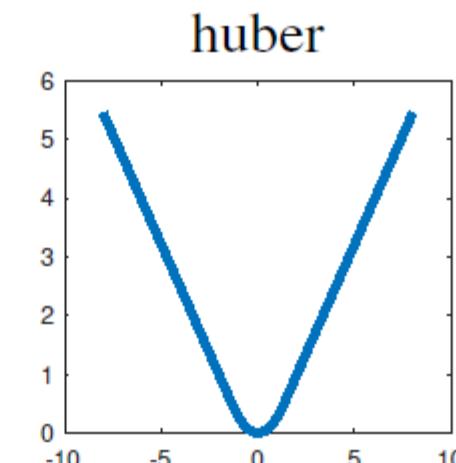
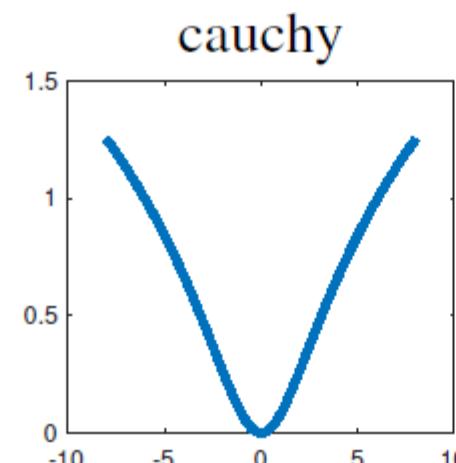
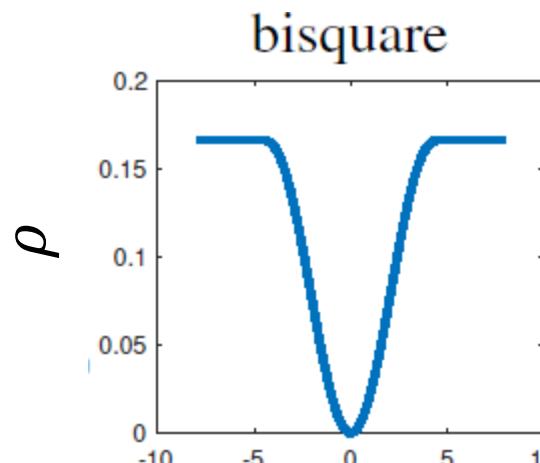


M-Estimator

Replaces the squared loss in the OLS with a different loss function which penalizes less large residual values (deemed to correspond to outliers)

$$\hat{\theta} = \operatorname{argmin}_{\theta} \sum_{i=1}^N \rho(r_i(\theta))$$

Where ρ a symmetric, positive-definite function having a unique minimum in zero



M-Estimator

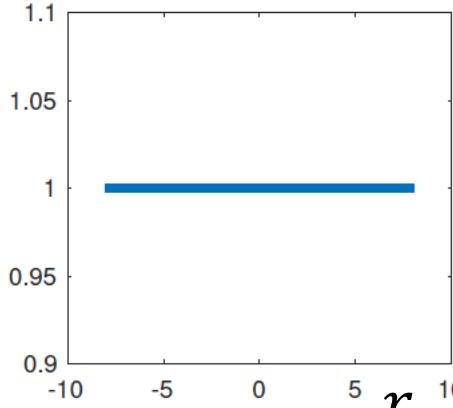
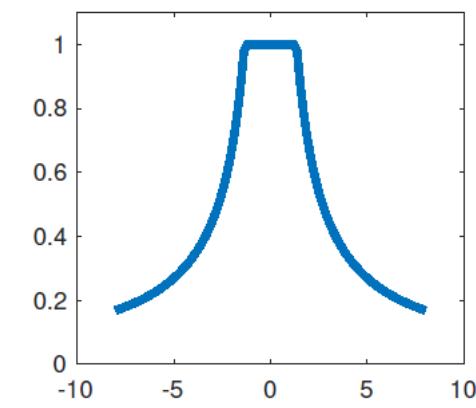
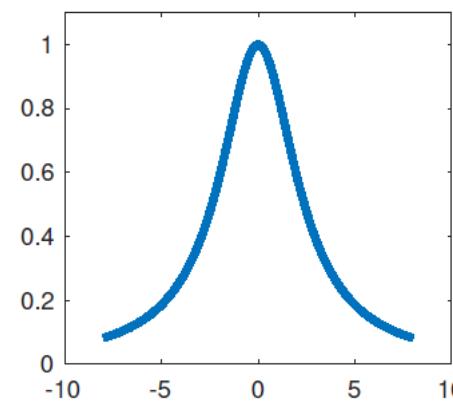
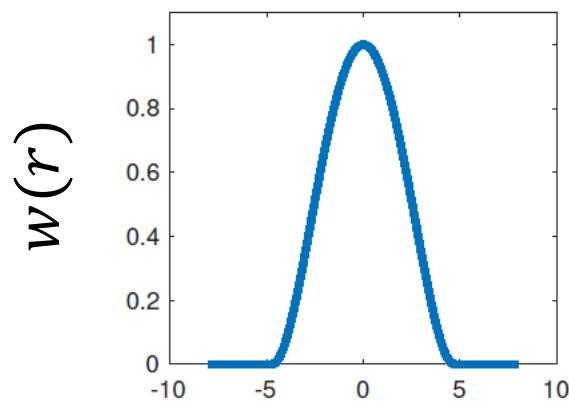
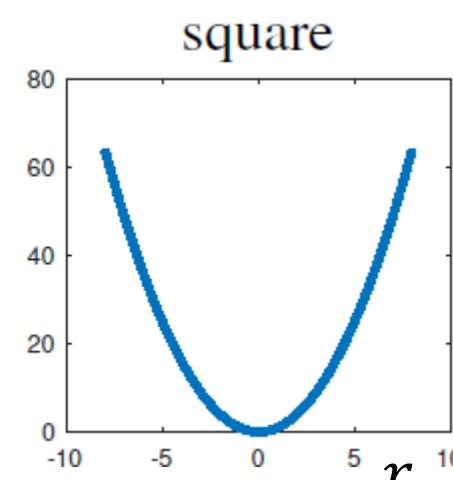
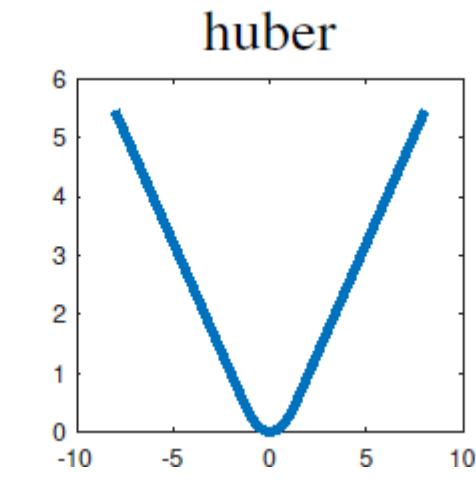
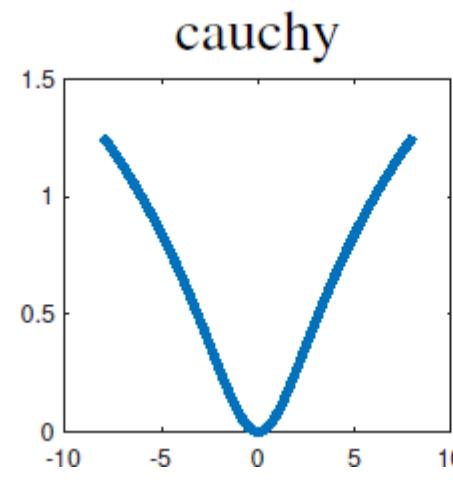
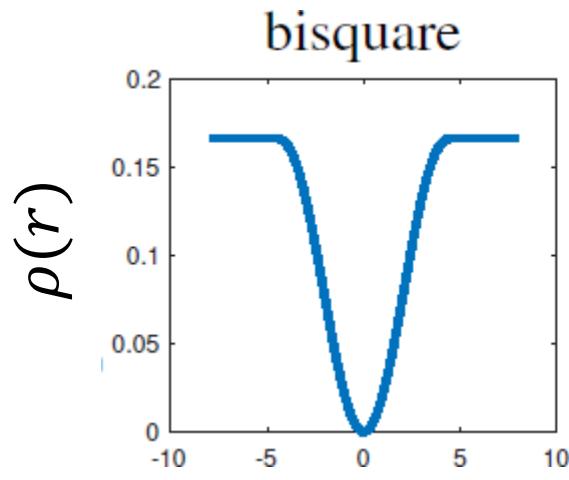
To solve this minimization problem we need to zero the derivative for each dimension of θ

$$\sum_{i=1}^N \rho'(r_i) \frac{\partial r_i}{\partial \theta_j} = 0, \quad j = 1, \dots, M$$

This problem is solved by an iterative reweighted least square

M-Estimator

Weights associated to the previous losses are



RanSaC

Robust single model fitting: consensus maximisation

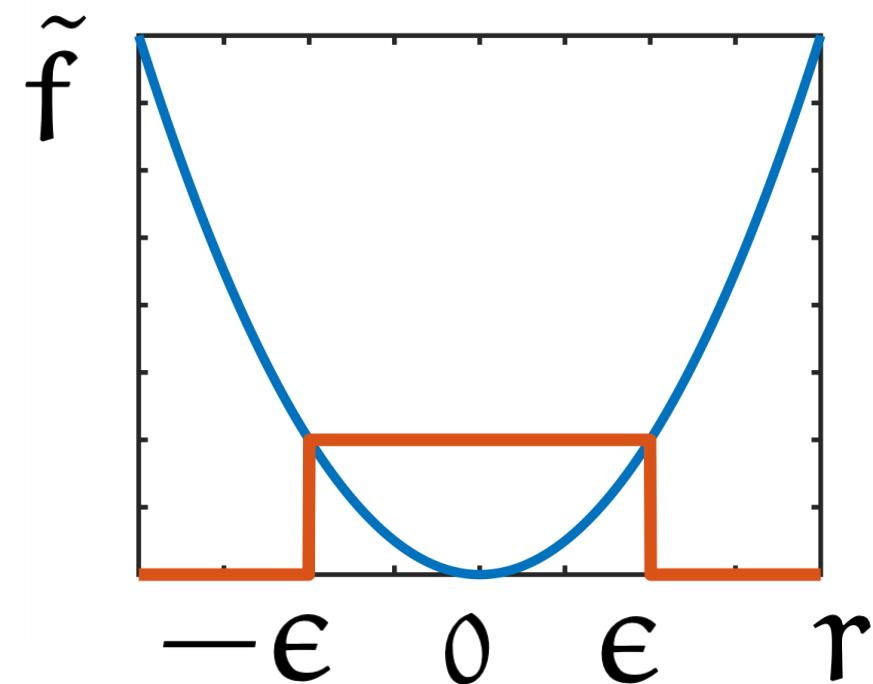
Instead of minimizing the residuals, **maximize the consensus**. Define:

- an inlier threshold $\epsilon > 0$
- a consensus function \tilde{f} which is

$$\tilde{f}(r_i) = \begin{cases} 1, & r_i \leq \epsilon \\ 0, & r_i > \epsilon \end{cases}$$

Identify $\hat{\theta}$ that maximizes the consensus

$$\hat{\theta} = \operatorname{argmax}_{\theta} \sum_{i=1}^N \tilde{f}(r_i(x_i, \theta))$$



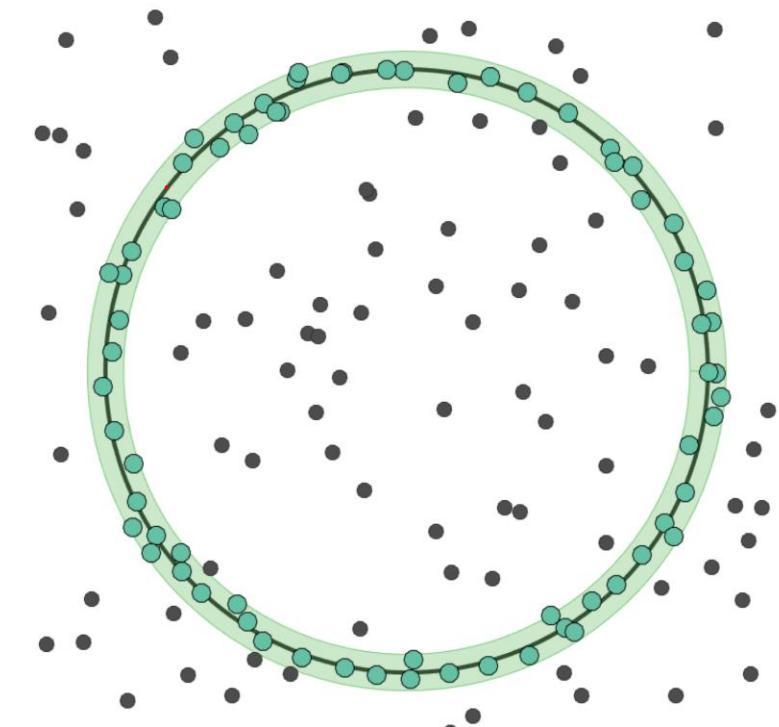
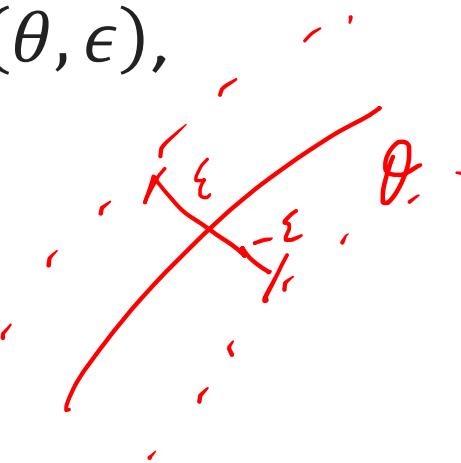
The Consensus Set

Consensus set

$$CS(\theta, \epsilon) = \{x_i \mid r_i \leq \epsilon\}$$

Being $r_i = r(x_i, \theta)$, the residual of the model θ at a point x_i

The larger the consensus set $CS(\theta, \epsilon)$,
the better the model θ



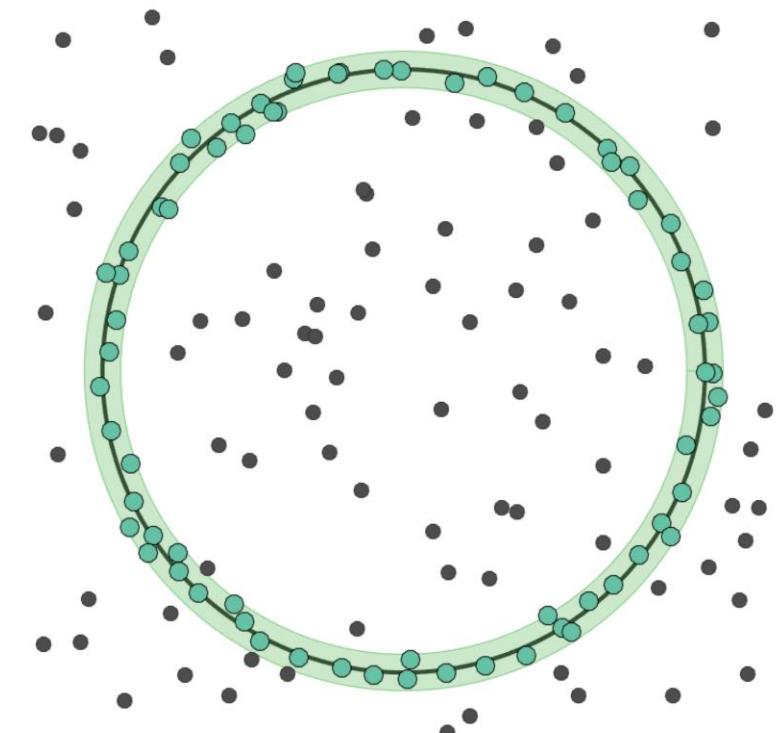
The Consensus Set

Consensus set

$$\text{CS}(\theta, \epsilon) = \{x_i \mid r_i \leq \epsilon\}$$

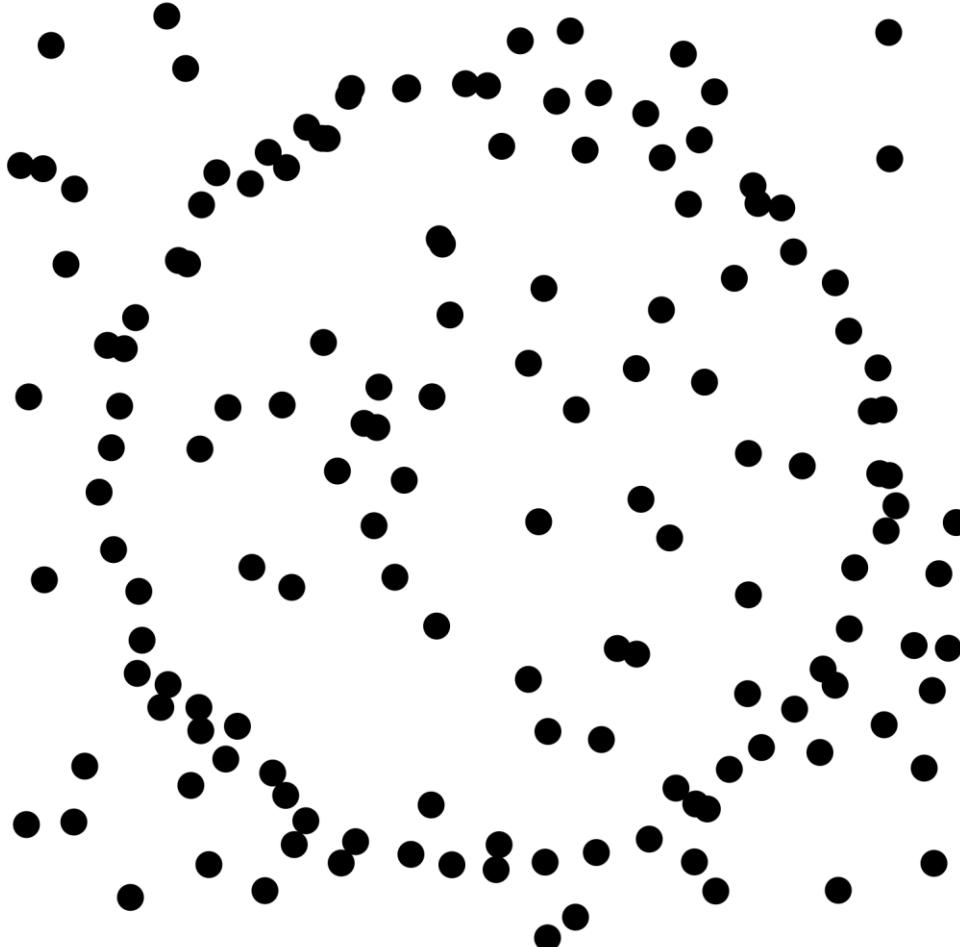
Being $r_i = r(x_i, \theta)$, the residual of the model θ at a point x_i

The larger the consensus set $\text{CS}(\theta, \epsilon)$, the better the model θ



We have been fitting lines so far, but
everything holds for any parametric model
(e.g. circle, conics, homographies,
fundamental matrices)

Randomized Sample Consensus [Fischler and Bolles 1981]



Input: X data, ϵ inlier threshold, k_{\max} max iteration

Output: θ^* model estimate

$J^* = -\infty, k = 0;$

repeat

Select randomly a minimal sample set $S \subset X$;

Estimate parameters θ on S ;

Evaluate $J(\theta) = \sum_{x \in S} \hat{f}_\epsilon(r(x, \theta))$;

if $J(\theta) > J^*$ **then**

$\theta^* = \theta$;

$J^* = J(\theta)$;

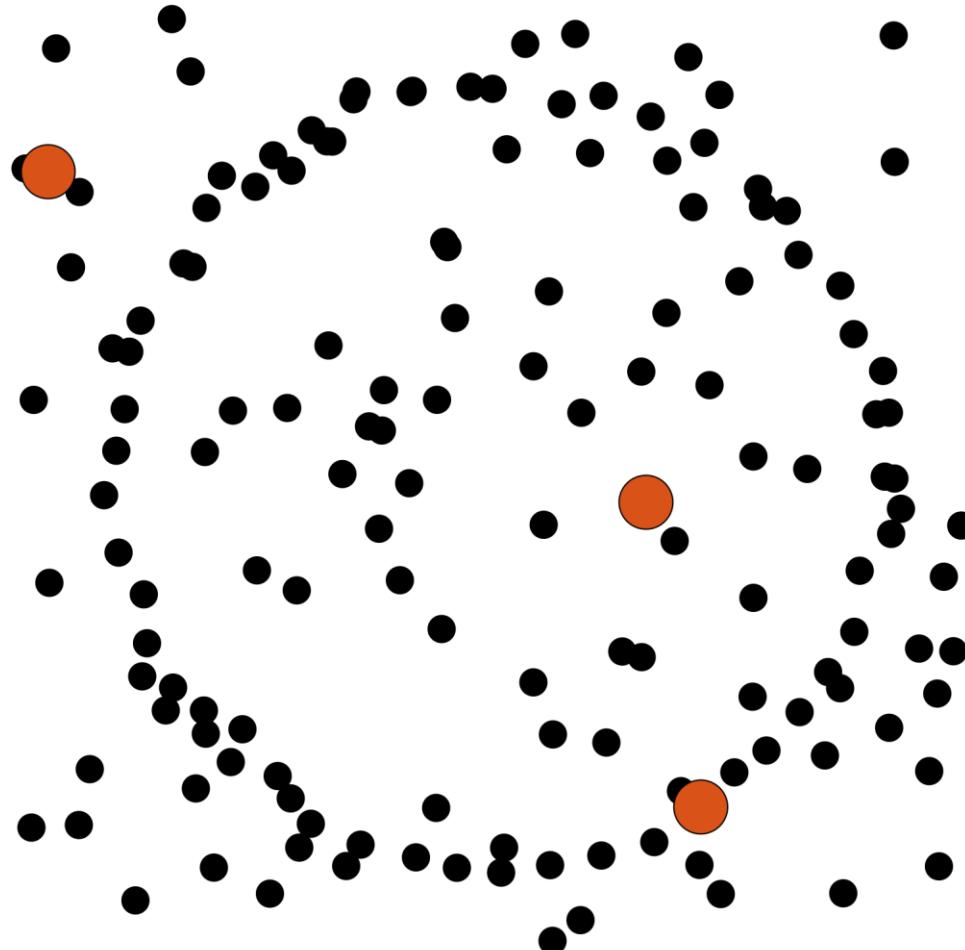
end

$k = k + 1$;

until $k > k_{\max}$;

Optimize θ^* on its inliers.

Randomized Sample Consensus [Fischler and Bolles 1981]



Input: X data, ϵ inlier threshold, k_{\max} max iteration

Output: θ^* model estimate

$J^* = -\infty, k = 0;$

repeat

Select randomly a minimal sample set $S \subset X$;

Estimate parameters θ on S ;

Evaluate $J(\theta) = \sum_{x \in S} \hat{f}_\epsilon(r(x, \theta))$;

if $J(\theta) > J^*$ **then**

$\theta^* = \theta$;

$J^* = J(\theta)$;

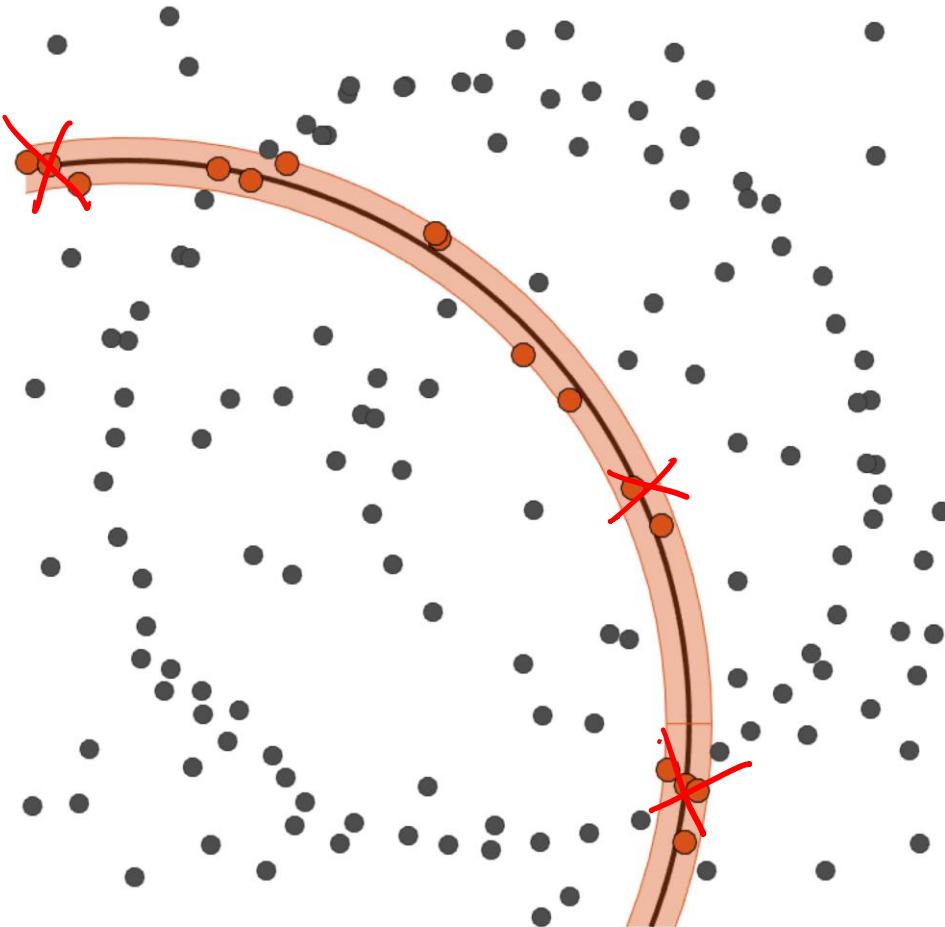
end

$k = k + 1$;

until $k > k_{\max}$;

Optimize θ^* on its inliers.

Randomized Sample Consensus [Fischler and Bolles 1981]



Input: X data, ϵ inlier threshold, k_{\max} max iteration

Output: θ^* model estimate

$J^* = -\infty, k = 0;$

repeat

Select randomly a minimal sample set $S \subset X$;

Estimate parameters θ on S ;

Evaluate $J(\theta) = \sum_{x \in S} \hat{f}_\epsilon(r(x, \theta))$;

if $J(\theta) > J^*$ **then**

$\theta^* = \theta$;

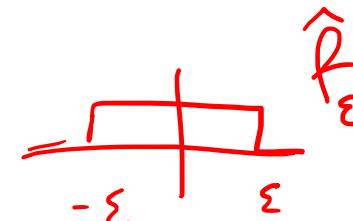
$J^* = J(\theta)$;

end

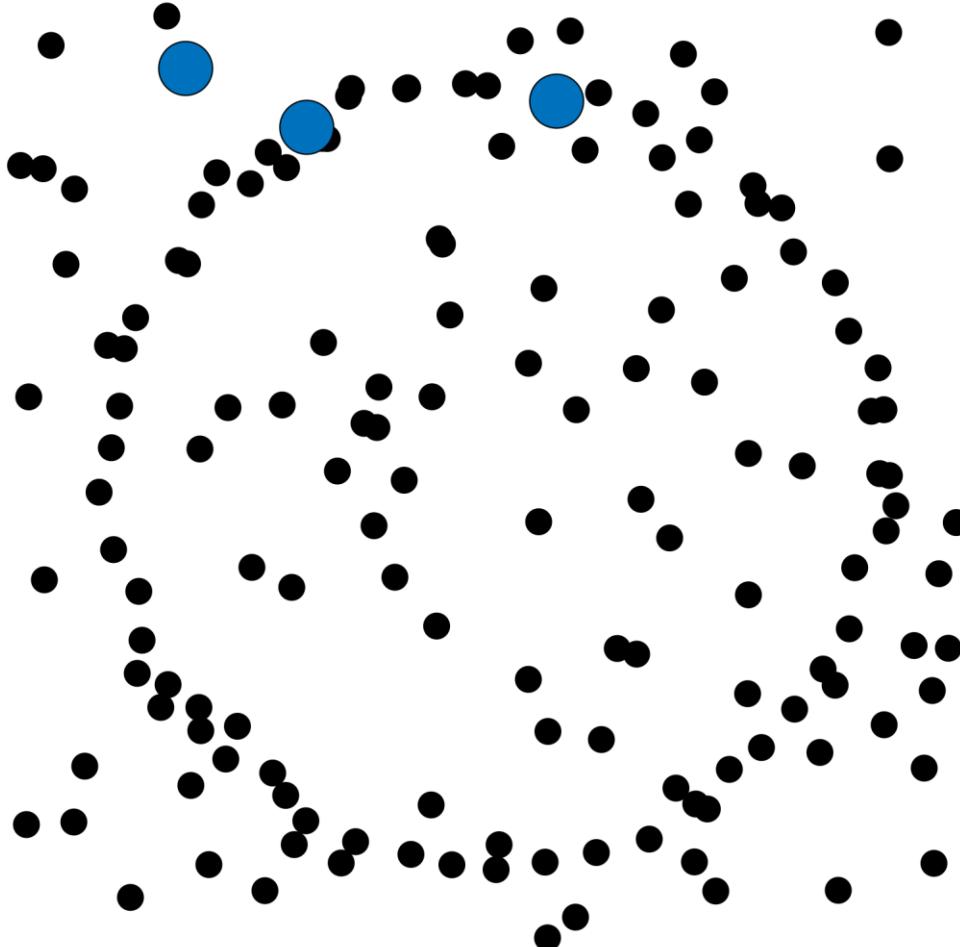
$k = k + 1$;

until $k > k_{\max}$;

Optimize θ^* on its inliers.



Randomized Sample Consensus [Fischler and Bolles 1981]



Input: X data, ϵ inlier threshold, k_{\max} max iteration

Output: θ^* model estimate

$J^* = -\infty, k = 0;$

repeat

Select randomly a minimal sample set $S \subset X$;

Estimate parameters θ on S ;

Evaluate $J(\theta) = \sum_{x \in S} \hat{f}_\epsilon(r(x, \theta))$;

if $J(\theta) > J^*$ **then**

$\theta^* = \theta$;

$J^* = J(\theta)$;

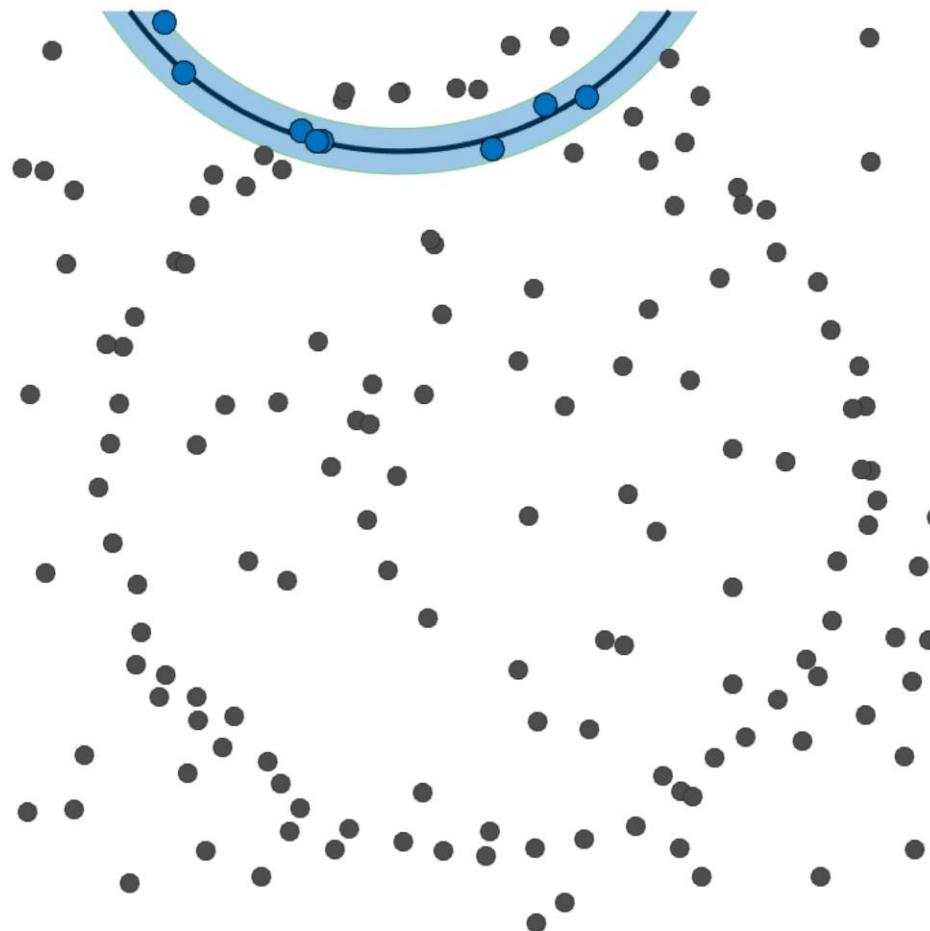
end

$k = k + 1$;

until $k > k_{\max}$;

Optimize θ^* on its inliers.

Randomized Sample Consensus [Fischler and Bolles 1981]



Input: X data, ϵ inlier threshold, k_{\max} max iteration

Output: θ^* model estimate

$J^* = -\infty, k = 0;$

repeat

Select randomly a minimal sample set $S \subset X$;

Estimate parameters θ on S ;

Evaluate $J(\theta) = \sum_{x \in X} \hat{f}_\epsilon(r(x, \theta))$;

if $J(\theta) > J^*$ **then**

$\theta^* = \theta$;

$J^* = J(\theta)$;

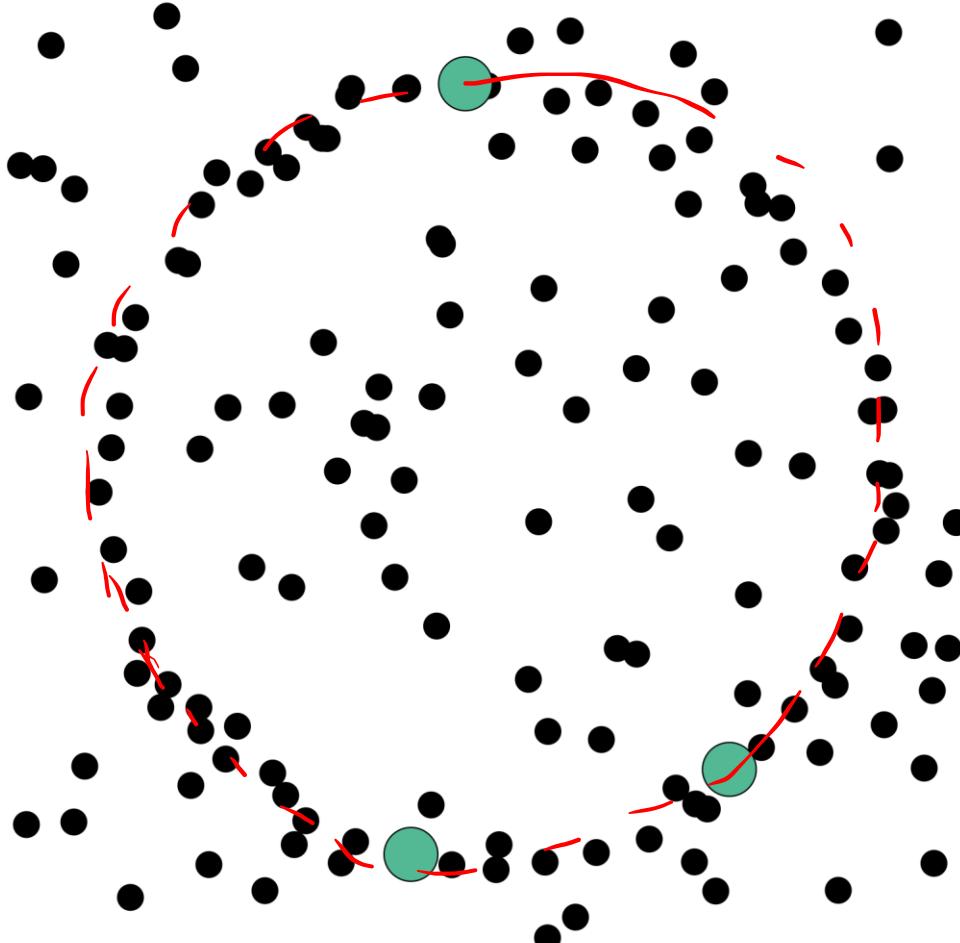
end

$k = k + 1$;

until $k > k_{\max}$;

Optimize θ^* on its inliers.

Randomized Sample Consensus [Fischler and Bolles 1981]



Input: X data, ϵ inlier threshold, k_{\max} max iteration

Output: θ^* model estimate

$J^* = -\infty, k = 0;$

repeat

Select randomly a minimal sample set $S \subset X$;

Estimate parameters θ on S ;

Evaluate $J(\theta) = \sum_{x \in S} \hat{f}_\epsilon(r(x, \theta))$;

if $J(\theta) > J^*$ **then**

$\theta^* = \theta$;

$J^* = J(\theta)$;

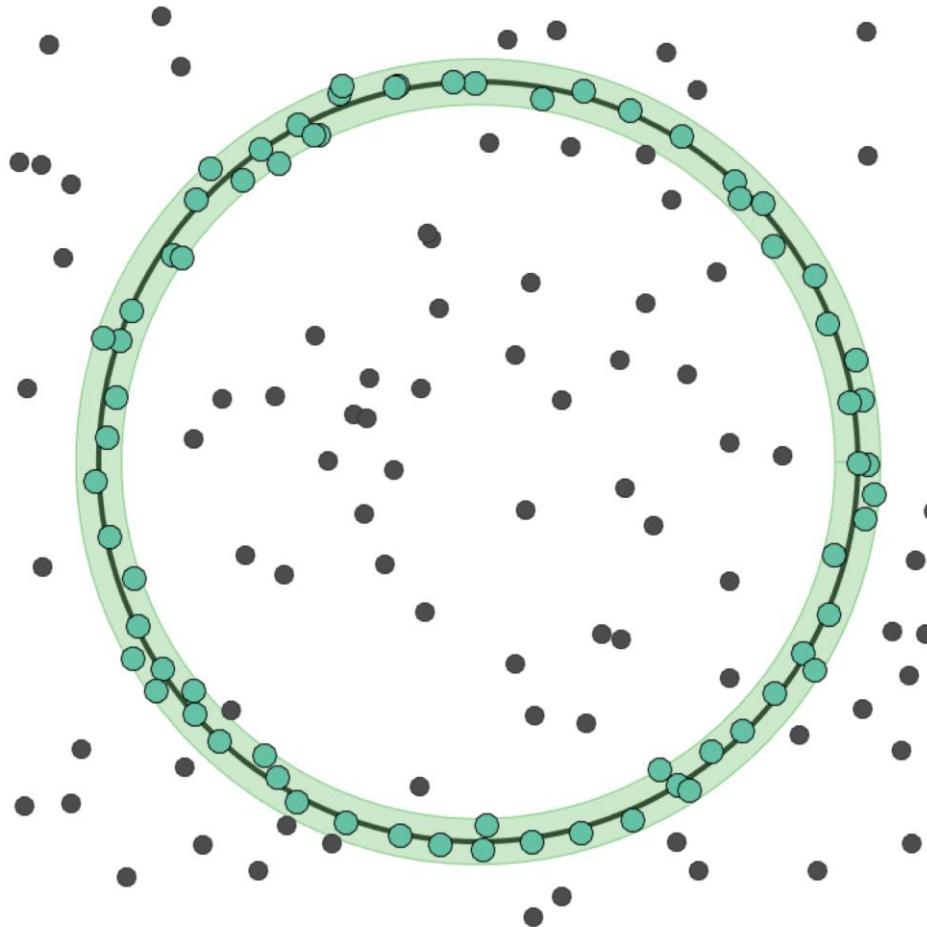
end

$k = k + 1$;

until $k > k_{\max}$;

Optimize θ^* on its inliers.

Randomized Sample Consensus [Fischler and Bolles 1981]



Input: X data, ϵ inlier threshold, k_{\max} max iteration

Output: θ^* model estimate

$J^* = -\infty, k = 0;$

repeat

Select randomly a minimal sample set $S \subset X$;

Estimate parameters θ on S ;

Evaluate $J(\theta) = \sum_{x \in X} \hat{f}_\epsilon(r(x, \theta))$;

if $J(\theta) > J^*$ **then**

$\theta^* = \theta$;

$J^* = J(\theta)$;

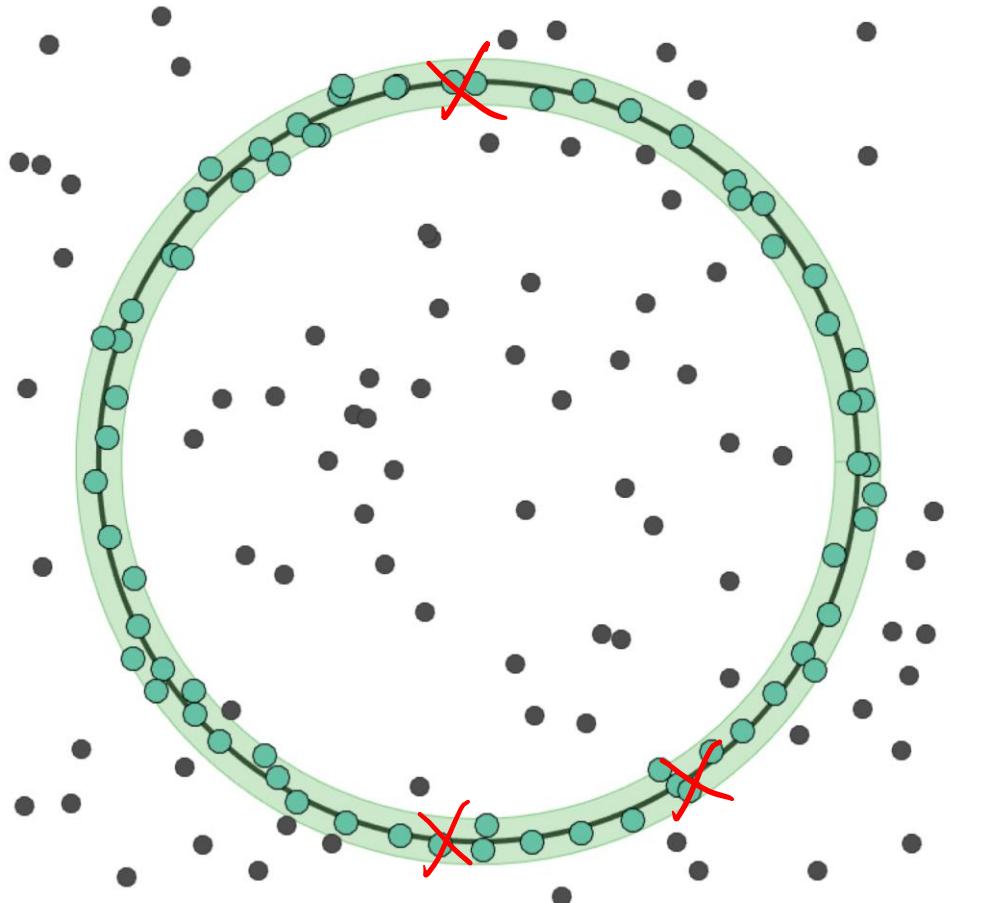
end

$k = k + 1$;

until $k > k_{\max}$;

Optimize θ^* on its inliers.

Randomized Sample Consensus [Fischler and Bolles 1981]



Using OLS,
Increase stability of

Input: X data, ϵ inlier threshold, k_{\max} max iteration

Output: θ^* model estimate

$J^* = -\infty, k = 0;$

repeat

Select randomly a minimal sample set $S \subset X$;

Estimate parameters θ on S ;

Evaluate $J(\theta) = \sum_{x \in S} \hat{f}_\epsilon(r(x, \theta))$;

if $J(\theta) > J^*$ **then**

$\theta^* = \theta$;

$J^* = J(\theta)$;

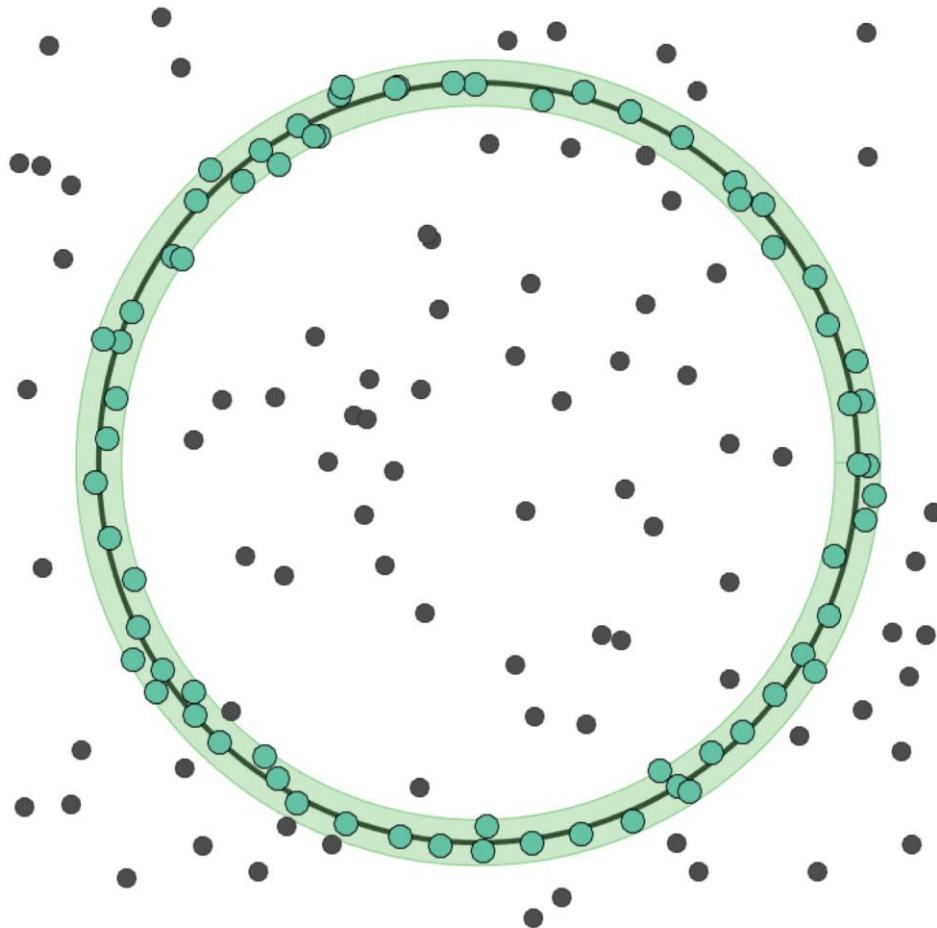
end

$k = k + 1$;

until $k > k_{\max}$;

Optimize θ^* on its inliers.

Randomized Sample Consensus

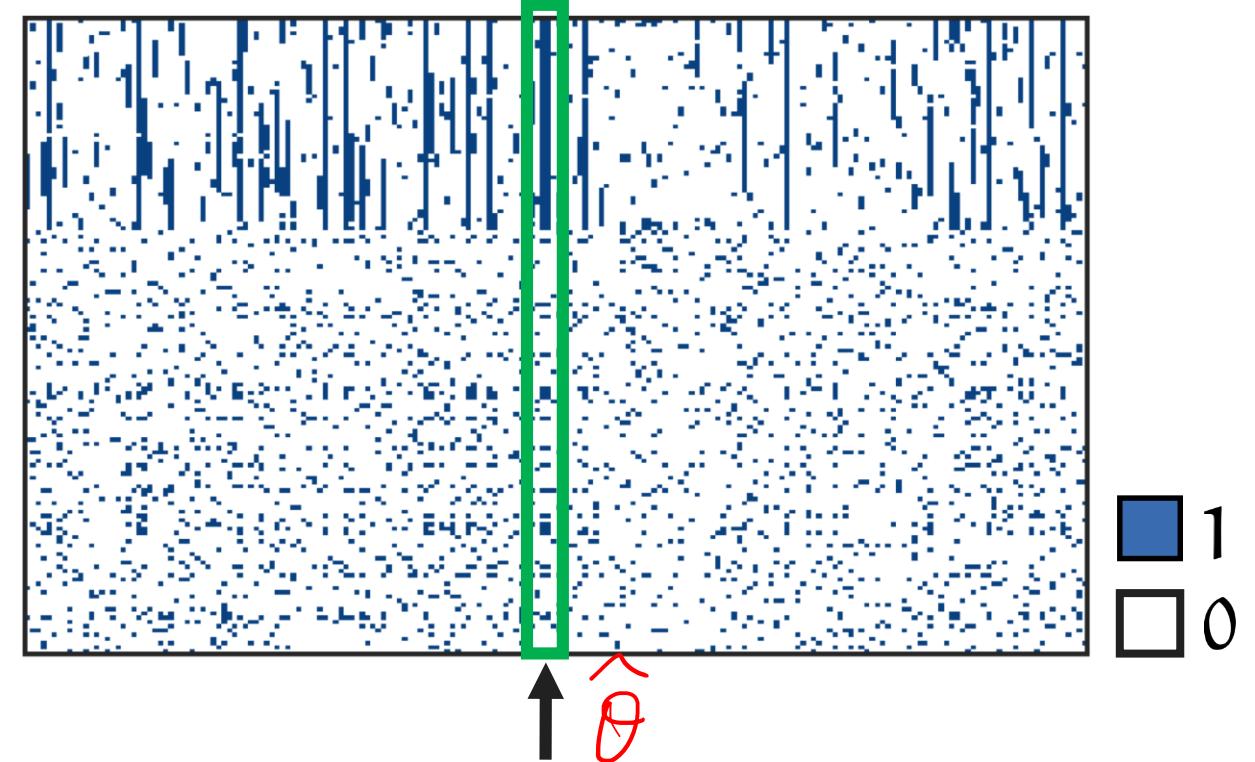


#X

points

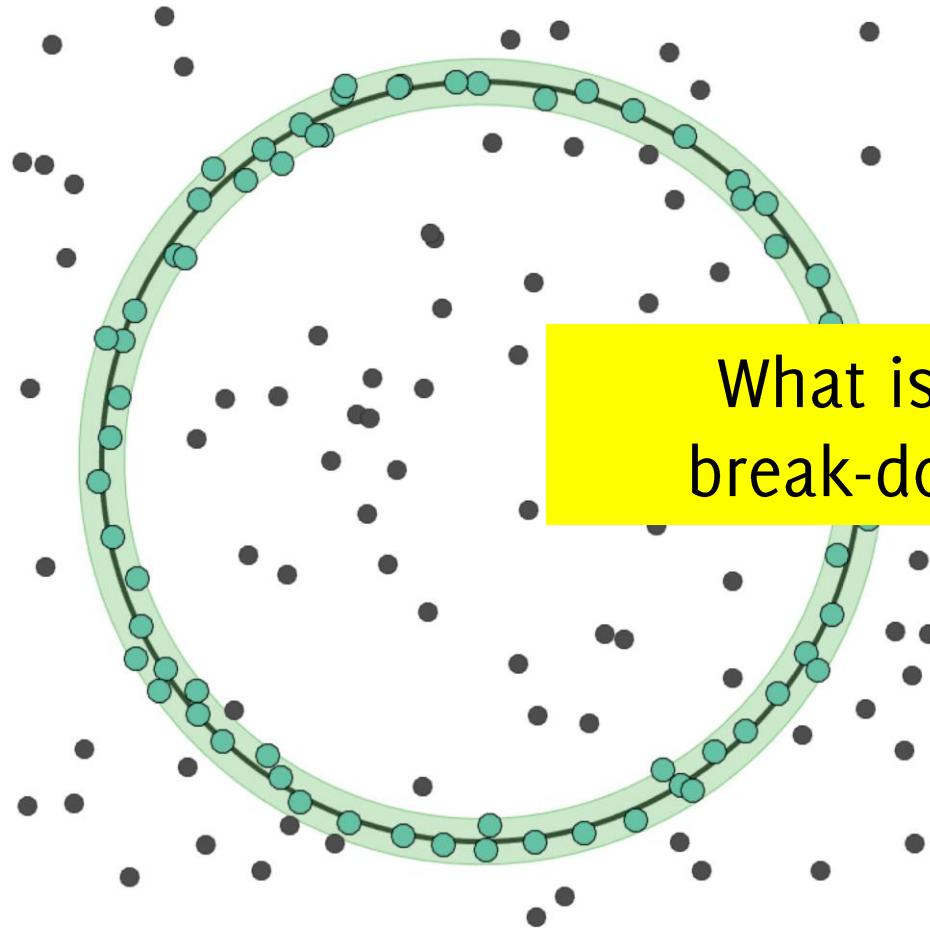
Data driven search of model space
 $H = \{\theta_1, \theta_2, \dots, \theta_m\} \approx \Theta$

tentative models



pick the column with the maximum sum

Randomized Sample Consensus

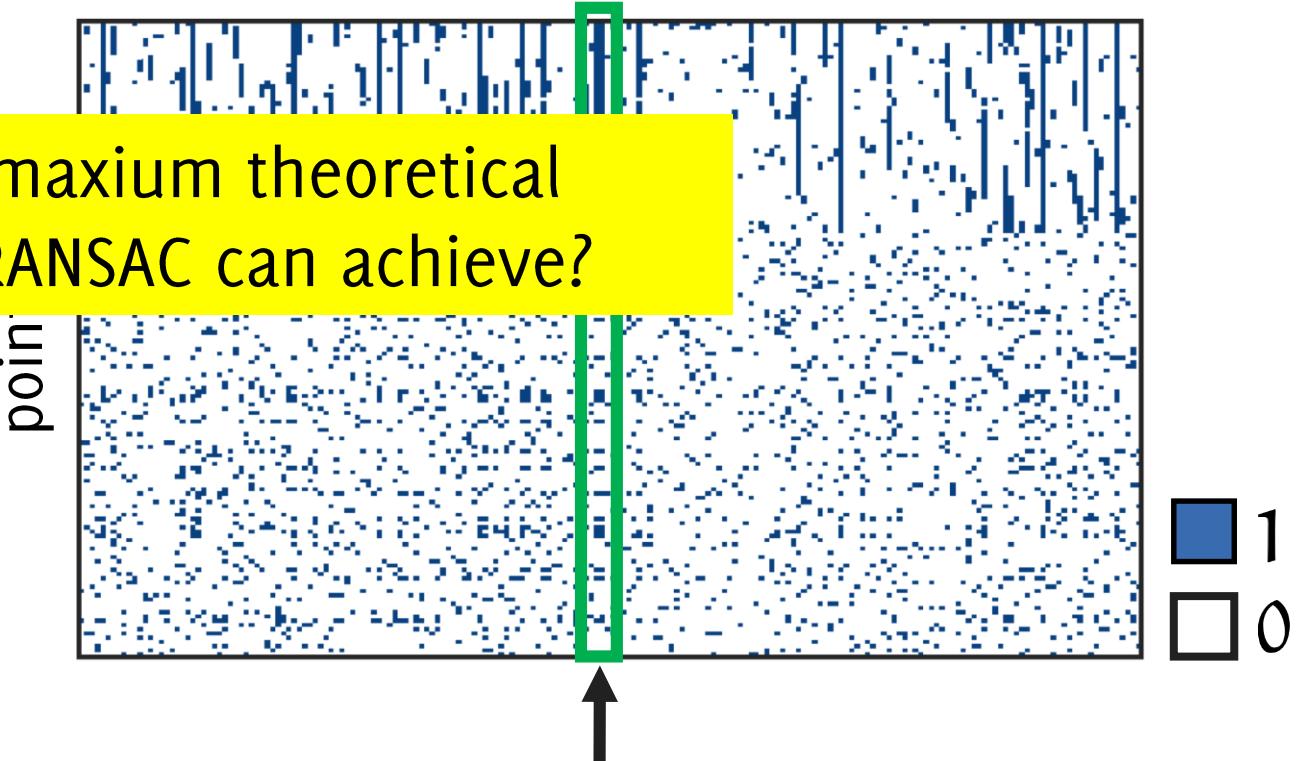


What is the maximum theoretical break-down RANSAC can achieve?

Data driven search of model space

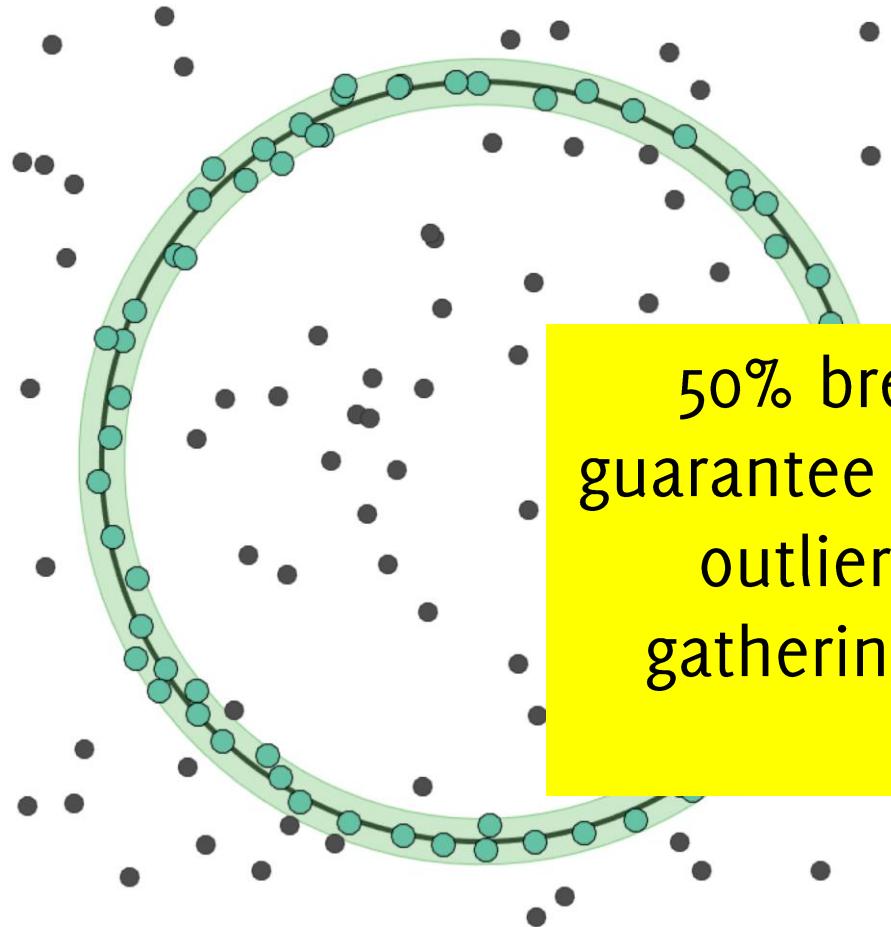
$$H = \{\theta_1, \theta_2, \dots, \theta_m\} \approx \Theta$$

tentative models



pick the column with the maximum sum

Randomized Sample Consensus

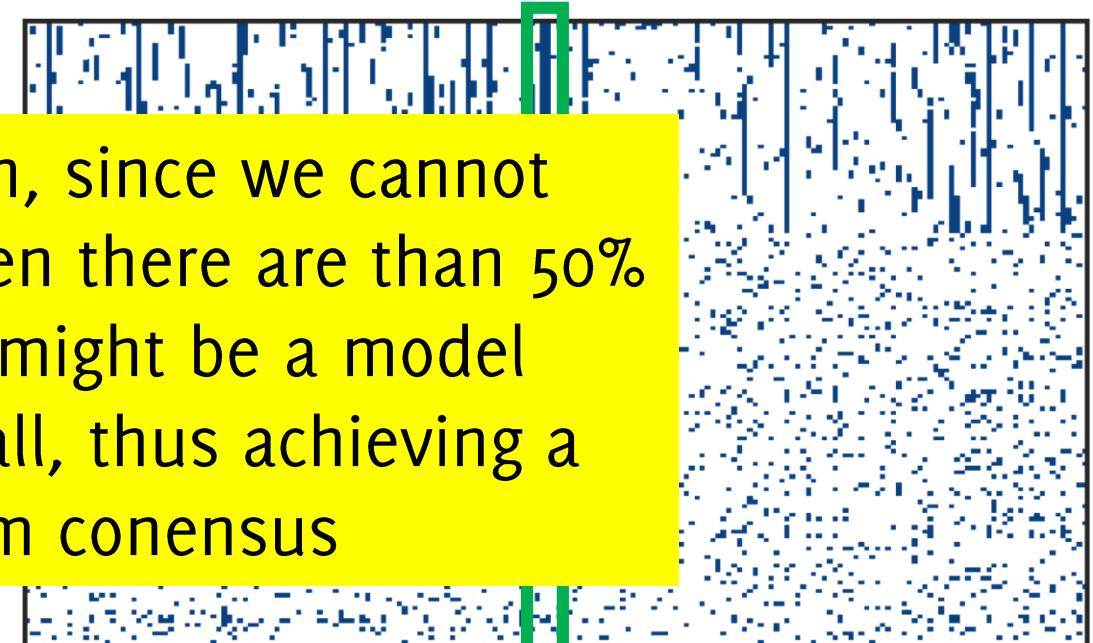


50% break-down, since we cannot guarantee that when there are than 50% outliers, there might be a model gathering them all, thus achieving a maximum consensus

Data driven search of model space

$$H = \{\theta_1, \theta_2, \dots, \theta_m\} \approx \Theta$$

tentative models



1

0

pick the column with the maximum sum

Ransac: practical issues

The size of the minimum sample S : this is the bare minimum number of points to fit the parametric model at hand

The inlier threshold ϵ : this can be estimated from the noise in the data

The number of iterations n (or k_{MAX}): the criteria for selecting the number of samples n : “Choose n so that, with probability p (e.g. $p = 0.99$), at least one random sample is without outliers”

The maximum number of iterations n

Let e the probability of a sample to be an outlier and $1 - e$ the probability of an inlier (can be estimated / provided by a-priori information)

The probability that all s points are inliers: $(1 - e)^s$

The probability that at least one point in S is an outlier: $1 - (1 - e)^s$
(this is the probability for a sample S to yield the right model)

The probability that all the n selected set contain outliers

$$(1 - (1 - e)^s)^n$$

The probability that at least one the n set is without outliers:

$$1 - (1 - (1 - e)^s)^n$$

Set n to have the above probability below a parameter p

$$p = (1 - (1 - (1 - e)^s)^n) \rightarrow n = \log(1 - p) / \log(1 - (1 - e)^s)$$

Ransac: practical issues

Choose n so that, with probability p , at least one random sample is free from outliers (e.g. $p = 0.99$) (outlier ratio: e)

$$n = \log(1 - p) / \log(1 - (1 - e)^s)$$

s	proportion of outliers e							
	5%	10%	20%	25%	30%	40%	50%	
2	2	3	5	6	7	11	17	
3	3	4	7	9	11	19	35	
4	3	5	9	13	17	34	72	
5	4	6	12	17	26	57	146	
6	4	7	16	24	37	97	293	
7	4	8	20	33	54	163	588	
8	5	9	26	44	78	272	1177	

Ransac: details

Repeat n times:

- Draw s points uniformly at random
- Fit line to these s points
- Find inliers to this line among the remaining points (i.e., points whose distance from the line is less than t)
- Update n

$$\bullet e = 1 - \frac{\text{number of inliers}}{\text{number of points}}$$

$$\bullet n = \log(1 - p) / \log(1 - (1 - e)^s)$$

Choose the best model

Re-estimate the line with the inliers only through ordinary least square

Ransac

Pros:

- very popular (>22900 citations in Google Scholar)
- many improvements have been proposed
- very versatile
- agnostic on outlier percentage
- mild assumption: know the scale noise to set the inlier threshold ϵ

Cons:

- can take longer than expected

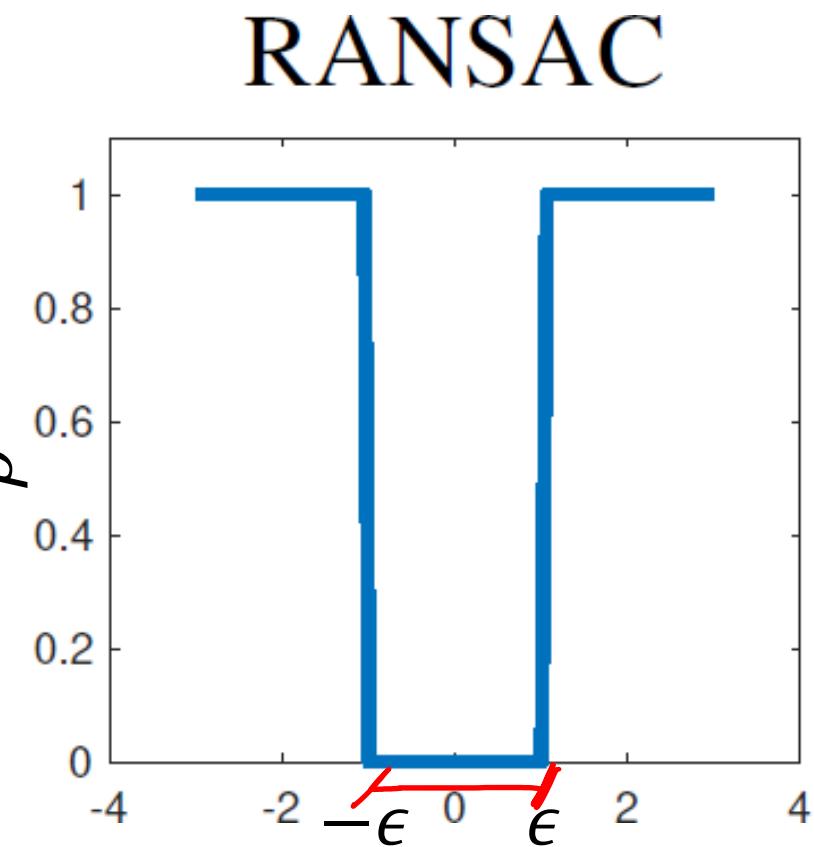
Ransac as M-estimator

(Steward 1999) RanSaC can be seen as a particular M-estimator since the **loss it minimizes** is the number of points having residual above the inlier threshold ϵ

$$f(r_i) = \begin{cases} 1, & r_i > \epsilon \\ 0, & r_i \leq \epsilon \end{cases}$$

Of course selecting inlier threshold ϵ is very critical

Ransac achieves a theoretical breakdown of 50% of outliers, but in practice, provided a good selection of ϵ , this can be even higher

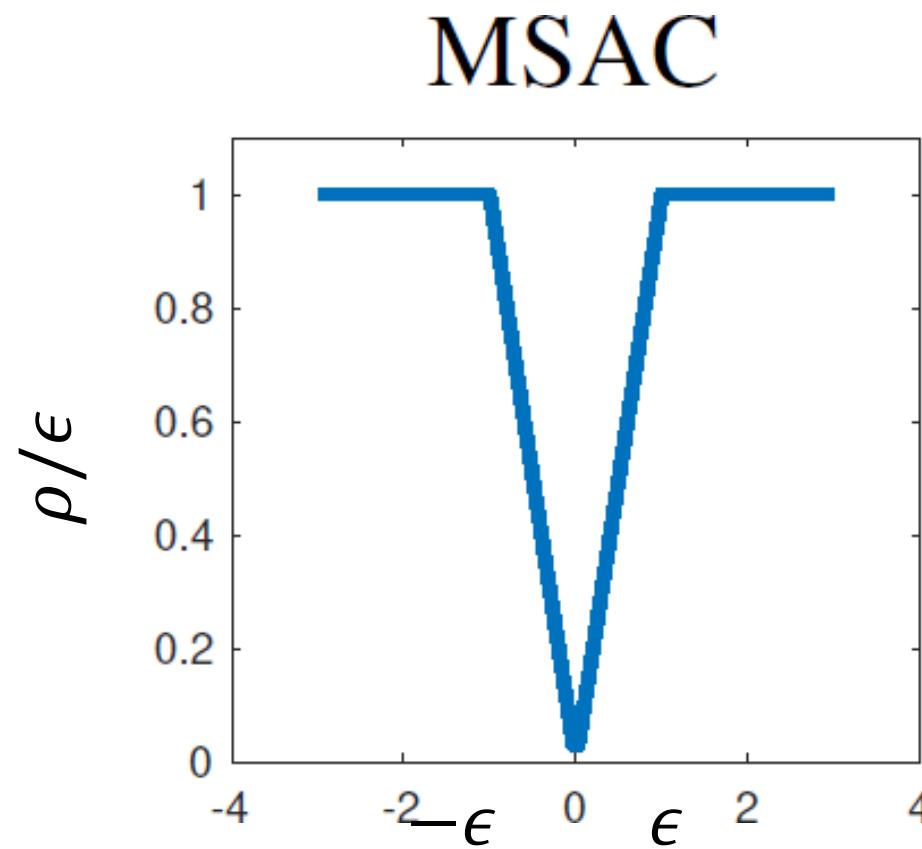


MSAC

(Torr and Zisserman 2000) a different loss function to be minimized within the RanSaC framework

$$f(r_i) = \begin{cases} \epsilon, & |r_i| > \epsilon \\ |r_i|, & |r_i| \leq \epsilon \end{cases}$$

This turns to be more effective and should be preferred to RanSaC



Ransac vs MSaC

Input: X data, ϵ inlier threshold, k_{\max} max iteration

Output: θ^* model estimate

$J^* = -\infty, k = 0;$

repeat

Select randomly a minimal sample set $S \subset X$;

Estimate parameters θ on S ;

Evaluate $J(\theta) = \sum_{x \in S} \hat{f}_\epsilon(r(x, \theta))$;

if $J(\theta) > J^*$ **then**

$\theta^* = \theta$;

$J^* = J(\theta)$;

end

$k = k + 1$;

until $k > k_{\max}$;

Optimize θ^* on its inliers.

Input: X data, ϵ inlier threshold, k_{\max} max iteration

Output: θ^* model estimate

$J^* = +\infty, k = 0;$

repeat

Select randomly a minimal sample set $S \subset X$;

Estimate parameters θ on S ;

Estimate inlier set $I = \{x \in X : r(x, \theta)^2 < \epsilon^2\}$;

Evaluate $J(\theta) = \sum_{x \in I} r(x, \theta) + (|X| - |I|)\epsilon$;

if $J(\theta) < J^*$ **then**

$\theta^* = \theta$;

$J^* = J(\theta)$;

end

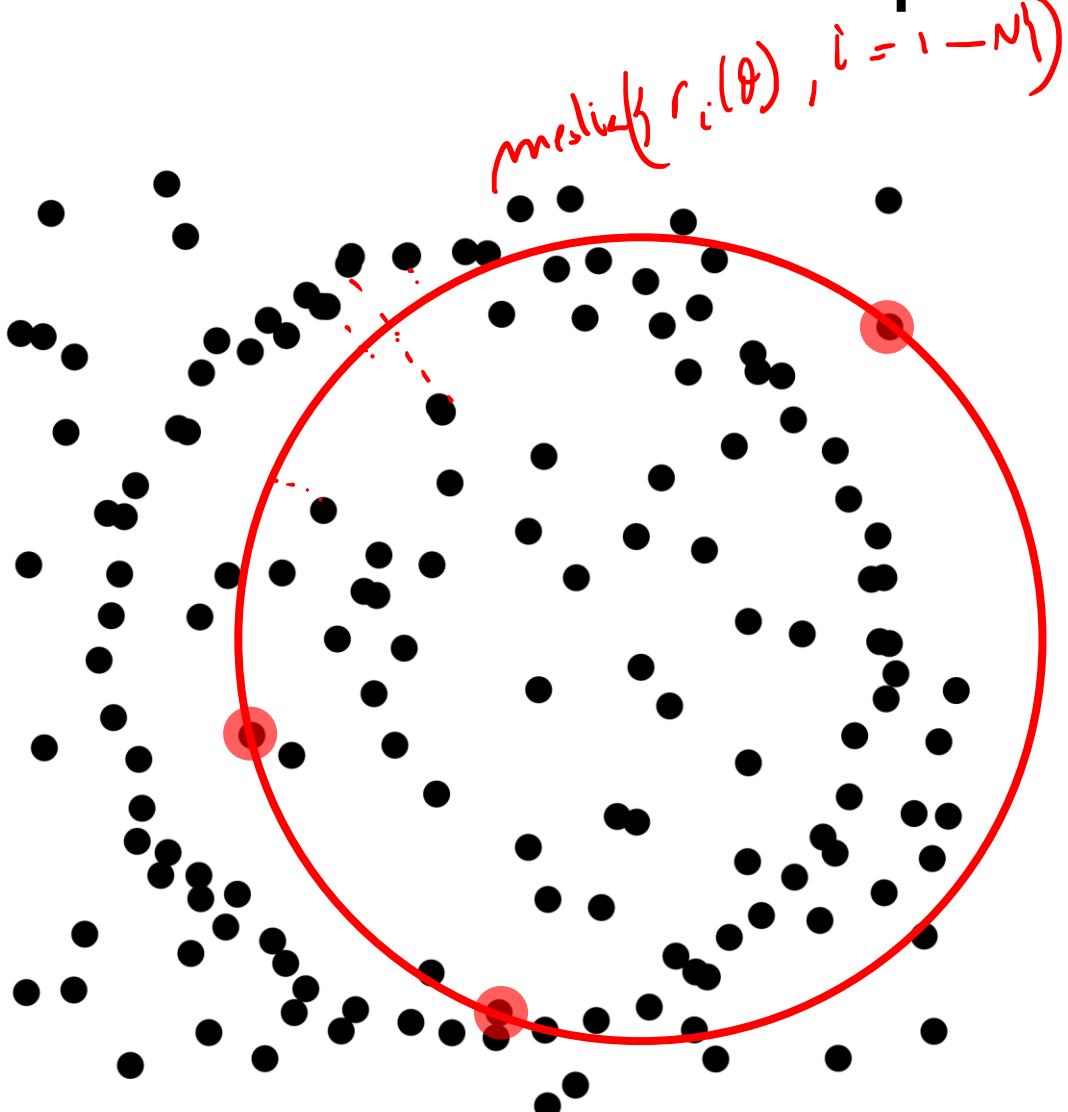
$k = k + 1$;

until $k > k_{\max}$;

Optimize θ^* on its inliers.

Least Median of Squares

L-meds: Least Median of Squares, Rousseeuw e Leroy (1987)



Input: X data, k_{\max} max iteration

Output: θ^* model estimate

$J^* = +\infty, k = 0;$

repeat

Select randomly a minimal sample set $S \subset X$;
Estimate parameters θ on S ;

Evaluate $J(\theta) = \text{median}_{x \in X}(r(x, \theta))$;

if $J(\theta) < J^*$ **then**

$\theta^* = \theta$;

$J^* = J(\theta)$;

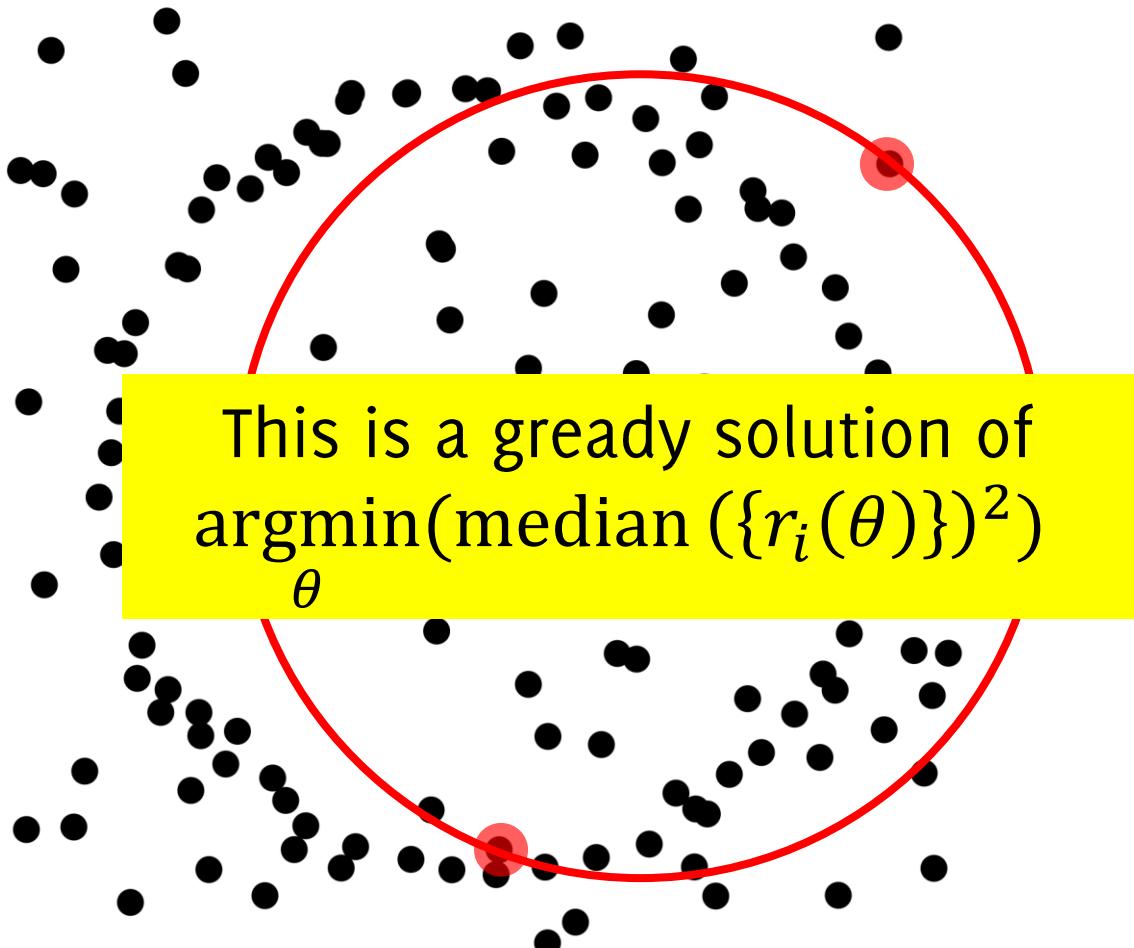
end

$k = k + 1$;

until $k > k_{\max}$;

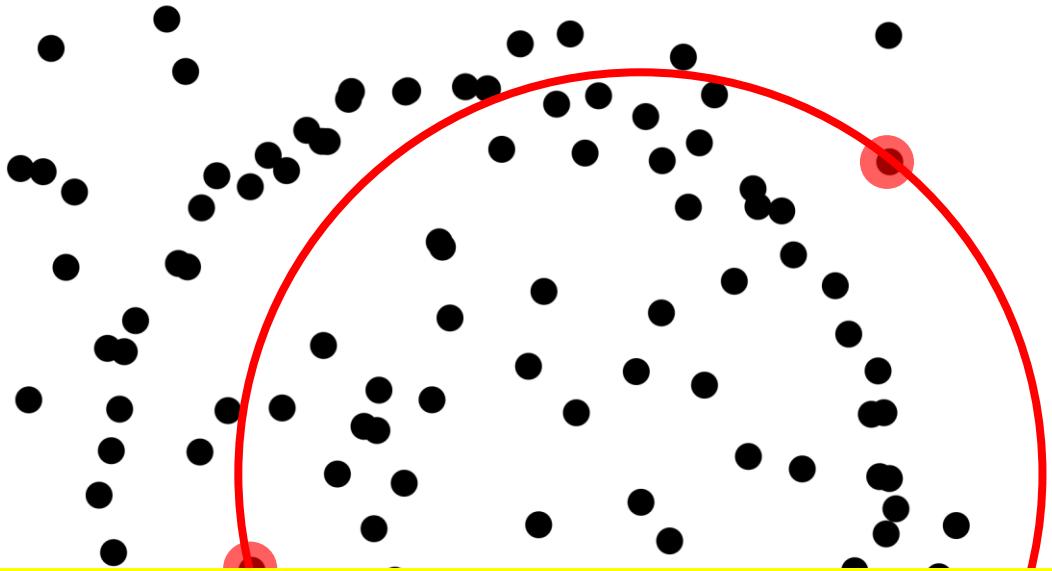
Optimize θ^* on its inliers.

L-meds: Least Median of Squares, Rousseeuw e Leroy (1987)



Input: X data, k_{\max} max iteration
Output: θ^* model estimate
 $J^* = +\infty, k = 0;$
repeat
 Select randomly a minimal sample set $S \subset X$;
 Estimate parameters θ on S ;
 Evaluate $J(\theta) = \text{median}_{x \in S} (r(x, \theta))$;
 if $J(\theta) < J^*$ **then**
 $\theta^* = \theta$;
 $J^* = J(\theta)$;
 end
 $k = k + 1$;
until $k > k_{\max}$;
Optimize θ^* on its inliers.

L-meds: Least Median of Squares, Rousseeuw e Leroy (1987)



Since there is no explicit definition of inliers here, inliers can be identified as points having residuals (w.r.t. to the final model) that are smaller than 2.5σ

Input: X data, k_{\max} max iteration

Output: θ^* model estimate

$J^* = +\infty, k = 0;$

repeat

Select randomly a minimal sample set $S \subset X$;
Estimate parameters θ on S ;

Evaluate $J(\theta) = \text{median}_{x \in X}(r(x, \theta))$;

if $J(\theta) < J^*$ **then**

$\theta^* = \theta$;

$J^* = J(\theta)$;

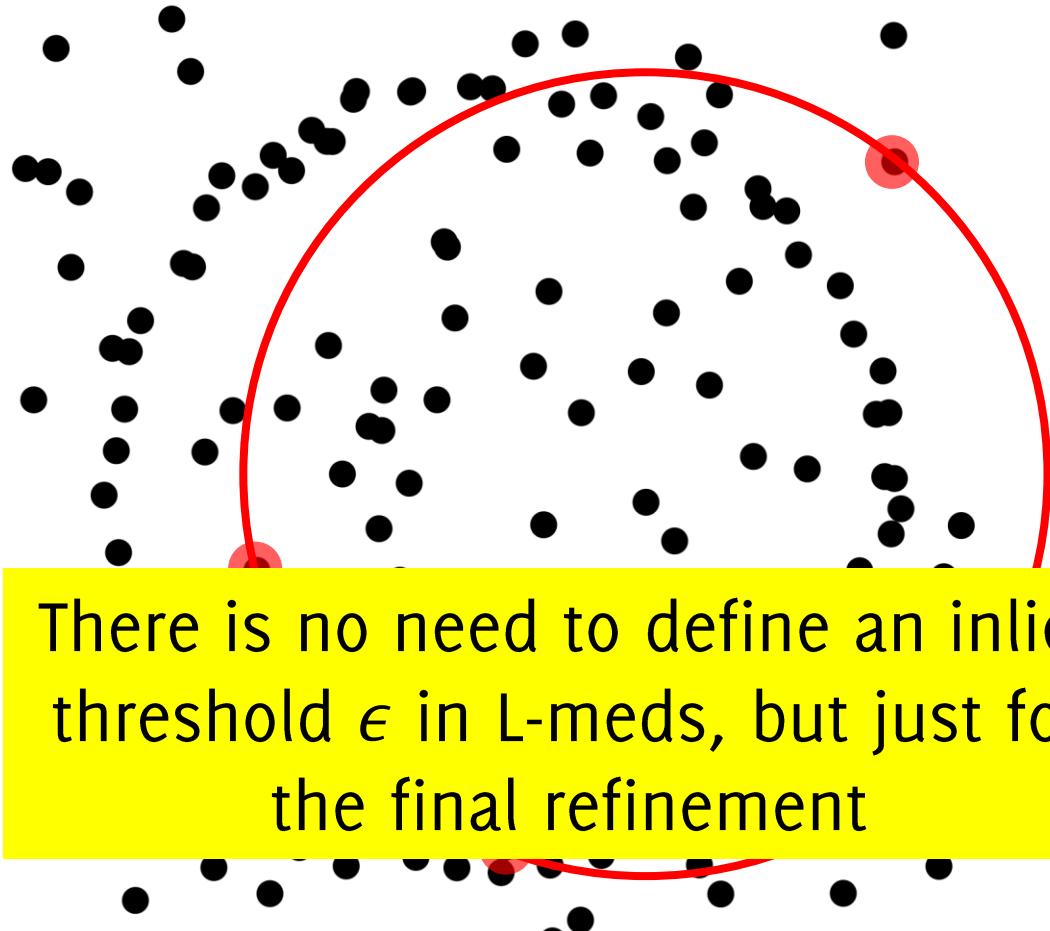
end

$k = k + 1$;

until $k > k_{\max}$;

Optimize θ^* on its inliers.

L-meds: Least Median of Squares, Rousseeuw e Leroy (1987)



Input: X data, k_{\max} max iteration
Output: θ^* model estimate
 $J^* = +\infty, k = 0;$
repeat
 Select randomly a minimal sample set $S \subset X$;
 Estimate parameters θ on S ;
 Evaluate $J(\theta) = \text{median}_{x \in X}(r(x, \theta))$;
 if $J(\theta) < J^*$ **then**
 $\theta^* = \theta$;
 $J^* = J(\theta)$;
 end
 $k = k + 1$;
until $k > k_{\max}$;
Optimize θ^* on its inliers.

Assigments

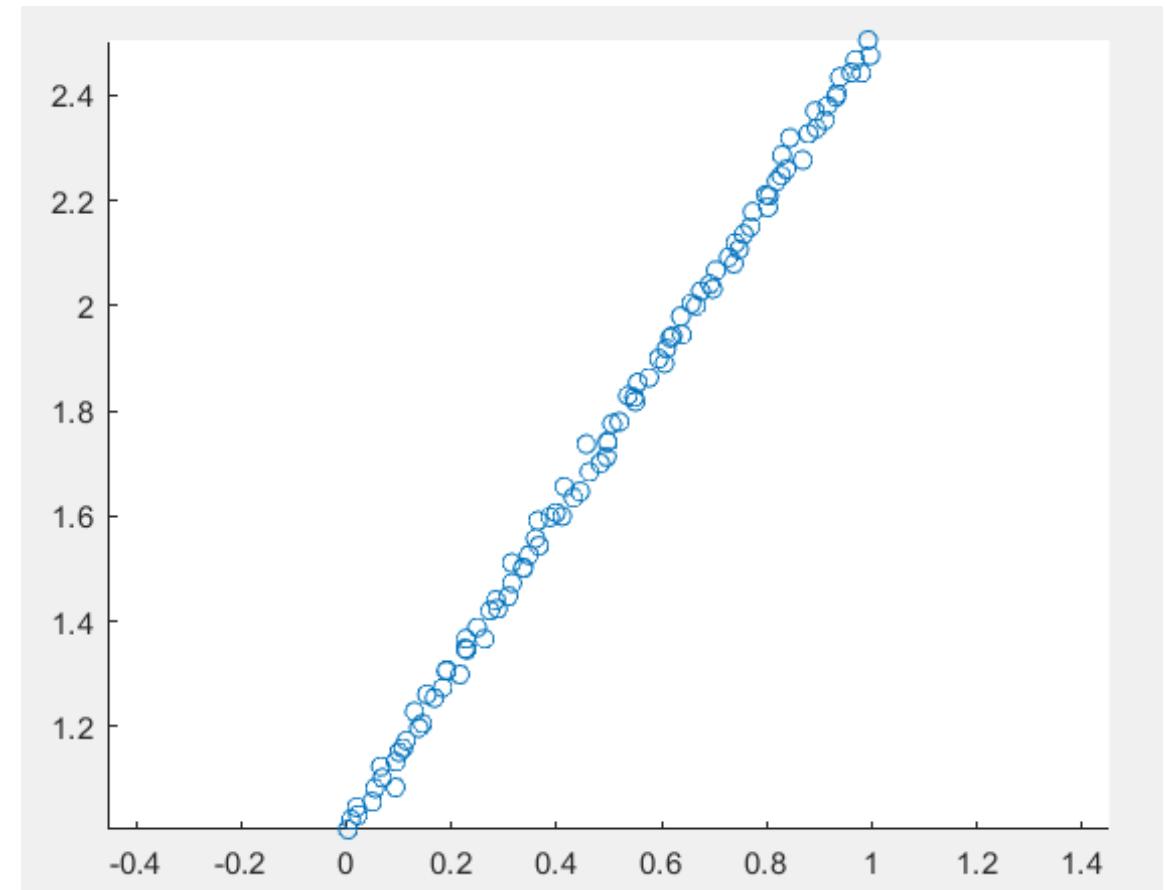
Fitting Functions

Implement

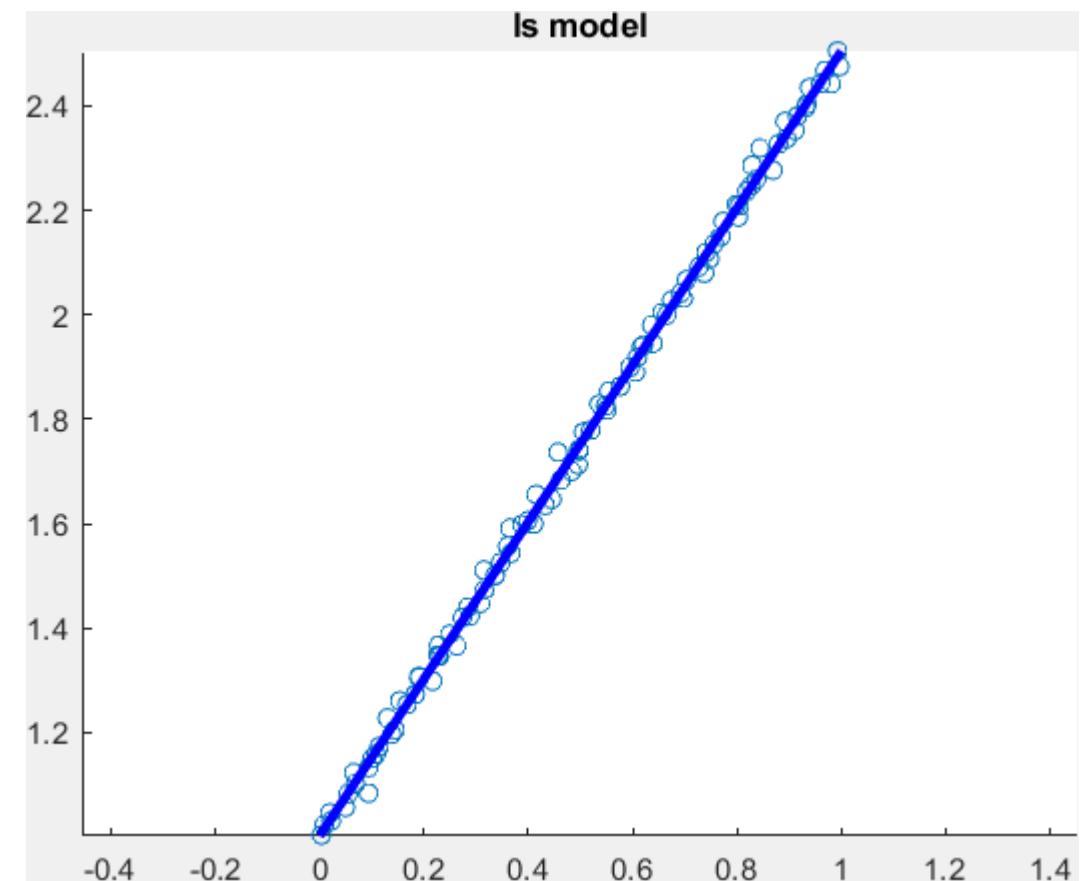
- **fit_line_ols** to perform line fitting using Ordinary Least Square (minimization of algebraic error, corresponding to vertical distances)
- **fit_line_DLT** to perform line fitting minimizing the geometric distance

Invoke these from **demo_robustmf_TODO** and **test robustness to outliers**

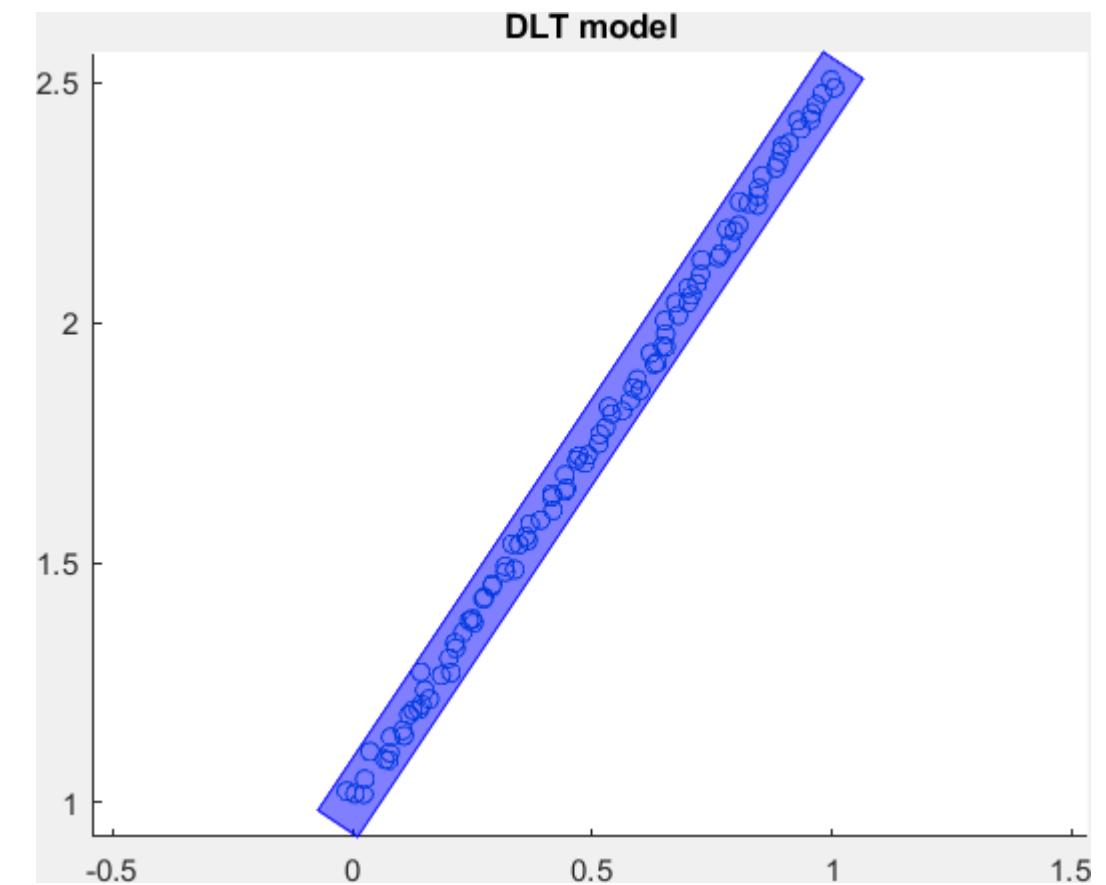
Here are the data



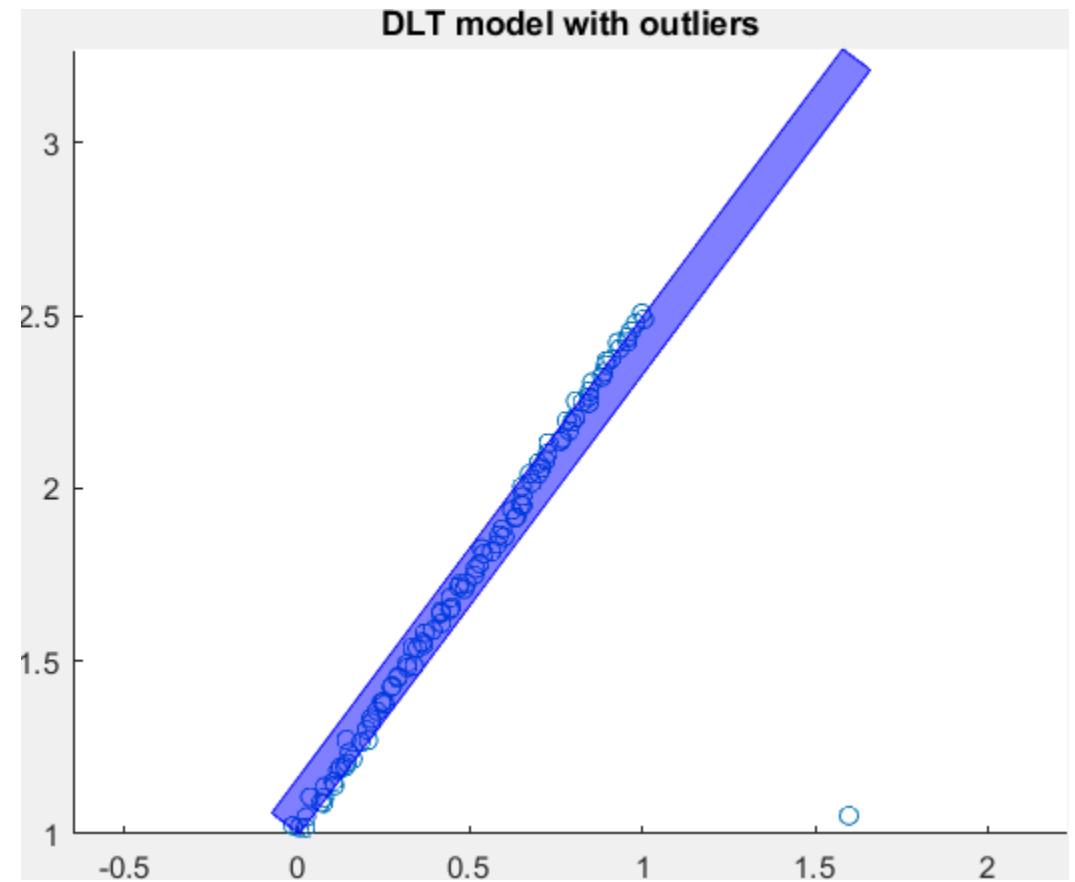
Least square fit



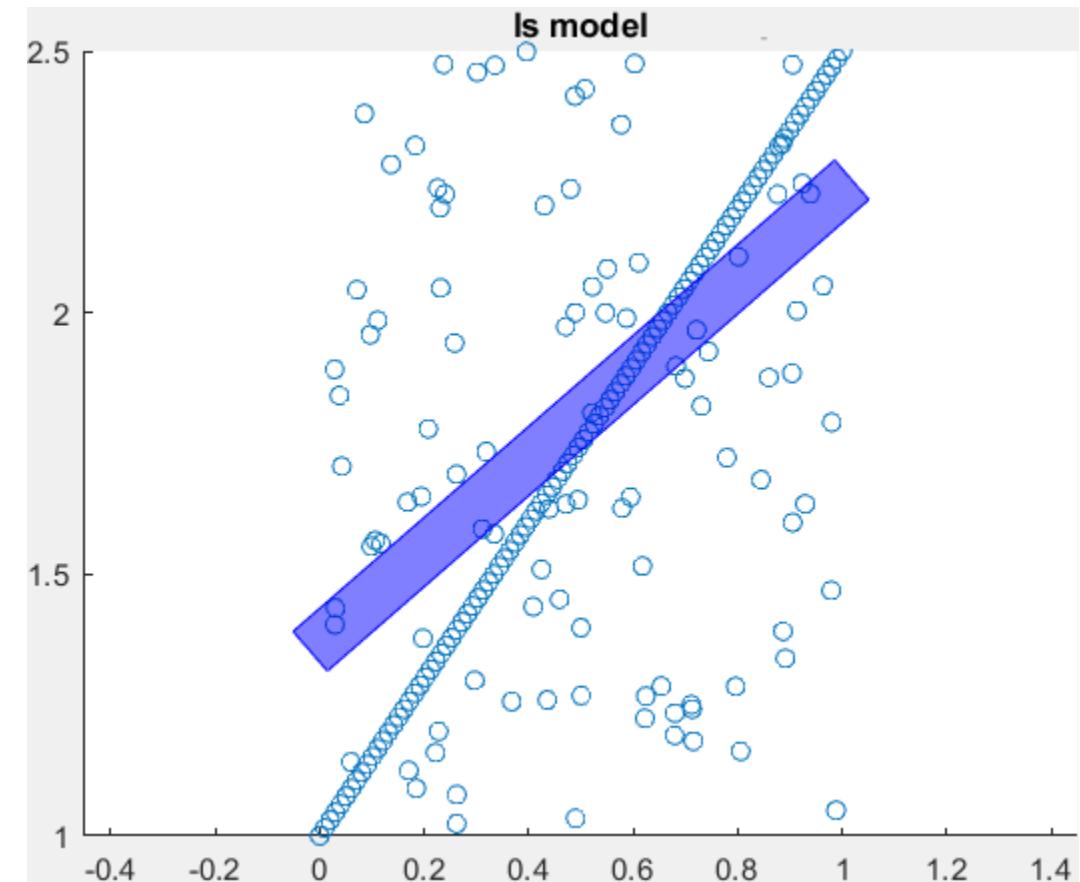
Inner Bands



A single outlier can impact the fitting



50% outliers, 50% inliers



RANSAC,

