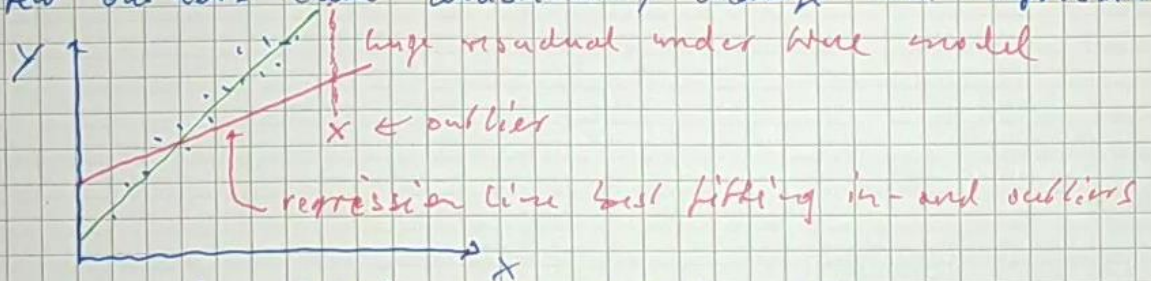
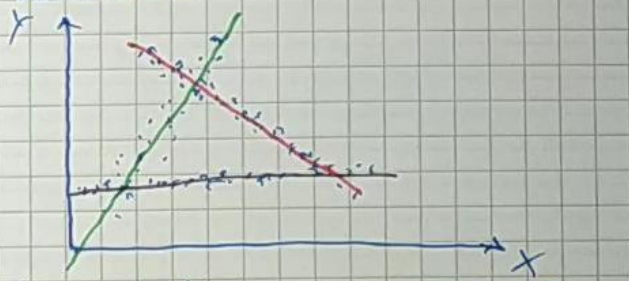
Robust Regression: regression in the presence of outliers

outliers: data instances where the model assumptions are violated

a few outliers can drastically change the regression solution



incorrect or highly unusual data points



data come from a superposition of several models ⇒ inliers of one model are outliers for others

caveat: robust regression must be used with care, because often it is not clear which points are outliers, for example:

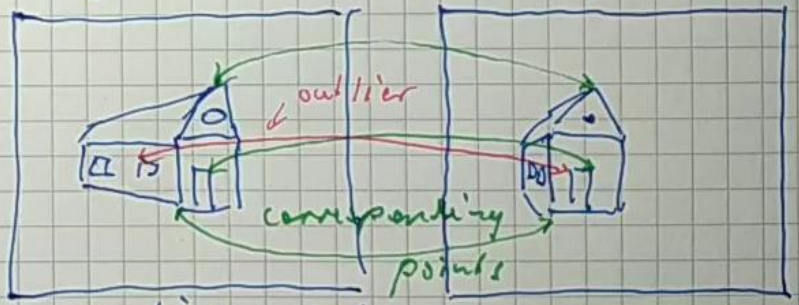
- discovery of the ozone hole over antarctica: 1985 using ground-based observations. Q: why did the satellite NIMBUS 7 not discover it earlier? A: analysis SW considered the low ozone values as outliers
- discovery of the positron: a german professor made first photographs of it in a cloud chamber (1930), but considered it an outlier
  ⇒ Nobel prize went to Anderson

⇒ don't throw-away your surprising Nobel prize discovery data as outliers!

two approaches: (1) detect and eliminate outliers, (2) down-weight them in the robust loss functions
⇒ proceed with ordinary regression of inliers ⇒ regression with a robust loss

RANSAC algorithm ("random sample consensus", [Fischer & Bolles, 1981])

- detect inliers, applicable when
  - inlier fraction is pretty small (< 50%, as low as 10%)
  - the data contain multiple model instances
- require: true model has relatively few degrees of freedom ($D \leq 10$)
  - if we have the minimum number of data points to fit the model at all, we need an efficient alg. to compute the solution, e.g.
    - fit a 2-D line: at least 2 points
    - fit a hyper-plane in D-dimensional space: at least D points
    - fit a circle in 2D: 3 points (=> circum circle of triangle)
    - stereo imaging (2 images, measure disparity between corresponding points and calculate depth)
      need to know how the viewing angle differs between the two images ("Fundamental matrix" between camera positions) => 8 pairs of corresponding points



simple alg. for correspondence detection have high outlier fraction

[variant: DSAC = differentiable RANSAC: train a neural network ~~that~~ to detect correspondences with ~~high~~ low outlier fraction]

require: a distance function between data points and a candidate model

$$d(Y_i, f_\theta(X_i)) = \begin{cases} \text{small, when } Y_i \text{ compatible with model (inlier.)} \\ \text{large, otherwise} \end{cases}$$

algorithm: ⓪ define a threshold κ for inliers: $d(Y_i, f_\theta(X_i)) \leq \kappa$

(for example: $\kappa \sim 5$ * standard deviation of inlier distribution $\kappa = 2\sigma$)

hyper-parameter

① for $t = 1, \ldots, T$:

ⓐ randomly select a subset $S^{(t)} \in$ TS such that size $|S|$ equals the minimum number of points for model fitting

ⓑ $\theta^{(t)} = \underset{\Theta}{\operatorname{argmin}}\ \text{loss}(\Theta, S)$ model fit to $S^{(t)}$

ⓒ compute residuals for entire TS and count how many points are closer to model than κ ("inlier count")

② return the model with largest inlier number

$$\hat{\theta} = \underset{t}{\operatorname{argmax}}\ \#(\text{inliers} \mid \theta^{(t)})$$

How many iterations do we need to get a good solution with high probability?

Assumption: - a good subset $S^{(t)}$ happens to contain only inliers $\Rightarrow$ good model

- a bad subset contains at least one outlier $\Rightarrow$ bad model

$\Rightarrow$ T must be big enough to get at least one good $S^{(t)}$ with probability $\alpha \approx 1$

hyper-parameter

- let γ be the ~~outlier~~ inlier rate (you know roughly, how many outliers, but which they are)

$$\gamma = \frac{N_{in}}{N}$$

k: minimum number of points needed

$p(S^{(t)} \text{ is good}) = p(\text{choose } k \text{ inliers}) = \gamma^k$

$p(S^{(t)} \text{ is bad}) = 1 - \gamma^k$

$p(\text{all ~~models~~ subsets } S^{(1)} \dots S^{(T)} \text{ are bad}) = (1-\gamma^k)^T$

$p(\text{at least one } S^{(t)} \text{ in } S^{(1)} \dots S^{(T)} \text{ is good}) = 1 - (1-\gamma^k)^T \doteq \alpha$

$\Rightarrow \quad (1-\gamma^k)^T = 1-\alpha$, solve for T:

$$T \geq \frac{\log(1-\alpha)}{\log(1-\gamma^k)}$$

examples

$\alpha = 99\%$

	k	$\gamma = 90\%$	$\gamma = 50\%$	$\gamma = 30\%$
line in 2D	2	$T=3$	17	49
circle in 2D	3	4	35	169
Fundamental matrix	8	9	1172	70188

This is still fast ~~so~~ when the "minimal fitting" alg. and inlier counting (distance computation) are fast

- if the data contains several model instance:
  - run RANSAC once => get model with most inliers => remove inliers from data set
  - repeat until all models are found (need to know beforehand, how many models are to be found)
- points should be chosen <u>without</u> replacement (success prob. γ^k is valid when $k \ll N$)